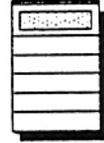


IPSO FACTO



The Newsletter of the International Psion Organiser User Group

Volume IV . No. 1 . March 1990

Editorial Office: 130 Stapleford Lane . Toton . Beeston . Nottingham . NG9 6GB . UK

Phone: (0602) 735482 . Microlink Mailbox MAG11839

IPSOMEET 90

I have had a good response from members wishing to attend IPSOMEET 90. Just to recap, this years meeting will be held in the Nottingham area on SATURDAY 5 AND SUNDAY 6 MAY.

Many of you have already let me know of your interest and day preference. Will you all please now send £8 for each person attending, to cover the cost of morning and afternoon refreshments and the mid-day buffet.

To those who haven't yet booked, may I say that there is still room (on either day) for you. You may bring your wife/husband, and/or a guest, if you wish, provided that you pay their £8!

On receipt of your payment, I will send you full details about the meeting, transport, parking, and overnight accomodation (if required)

Thanks!

As we start our 4th year (yes it really is!) I would like to thank all those who contributed to IPSO FACTO during the past year:

Les Ball, Mike Beckett, A.J. Burns, Chris Burton, Frank Cook, Philip Cook, Neil Draycott, Neil Favager, Graham Foster, John Gardner, Robert Gautier, Alain Geenrits, R.I. Gilchrist, David Green, Jonathan Hurwitt, Rodger Manson, Brod Mason, Colin McGlashan, John Morris, Mike Nash, Denis O'Regan, Richard Parfitt, Adrian Pegg, Glyn Pollington, John Sikora, A.R.P. Spencer, A.C Spry, John Walker, Reg Watson, Mark Wilding

and, in addition, all those who have sent me material which has not yet been published.

Progs & Procs on Disk or Datapak

All procedures published in IPSO FACTO this volume, plus certain other programs which are judged to be too long to publish as listings, will be available on either 5.25" IBM format disks or copied onto your own data or RAM pak.

There will be a small charge for this service of £2.00 for each disk plus £1.00 for each program supplied (multiple procedures comprising a single program will only have one charge).

Data or RAM pak copies will be £2.00 each pak (you provide the pak), I will format the pak if requested and return it by RECORDED DELIVERY mail.

I am currently copying ALL programs published so far onto disk and I will inform you when this process is complete, when these programs will also be available as above. I will also state the amount of memory required for both the full listing and object code only.

Contents

Editorial.....	1
Hardware Project..	2
Letters.....	3
Progs & Procs.....	4
Product News.....	6
PC Comms.....	6
Beginner's Page....	7
Small Ads etc.....	8

A Hardware Project

Collecting data from the Outside World

by Mike Davies

It seems that the majority of members use their Organiser as electronic Filo-faxes, or sophisticated programmable calculators. In fact the Organiser is a very versatile computer which can be used in process control environments - it can capture data from sensors and act upon that information, or simply record the fact that an event has occurred. This is the first of two articles showing how this can be done.

Firstly, although it is possible to build hardware to fit the top-slot - I have done it to interface to the Maplin Electronics anemometer & wind-vane; this is probably beyond the capabilities of the majority of Ipso members. If you stick wires or whatever between the various holes exposed in the top-slot, there is a very high chance of destroying the Organiser. For this reason we shall use the Comms Link adaptor as the hardware interface, since by its' very nature, it incorporates, buffering to the Organiser bus, voltage protection, and is commonly available.

One of the simplest sensors available is the reed-switch. When a magnet is brought near the reed, magnetism is induced into both halves of the reed in same direction. One becomes a north pole and the other a south pole, and the attraction of the poles causes the switch to close. When the magnet is removed, the springiness of the reed enables the switch to open again.

So how many reed-switches can we fit to the 25-way D connector on Comms Link? Two or three. Two definitely, and if handshaking in set to NONE either via the LSET: procedure or Comms Link Set-up menu then three. Where do we fit them and how? Well, it is best not to poke wires directly into the holes on the D connector, since this can expand the holes to much which will result in a loose connection when using Comms Link. Buy a proper plug and use that. Solder some leads to the reed switches, and using pin 5 (CTS) as a common pin to all your reed switches, you can now have the other ends of the reeds between pin 4 (RTS), and pin 6 (DSR & DTR), and if you have handshaking set to NONE pin 2 (RX).

SEE DIAGRAM ON NEXT PAGE

All we need now, is a little program to read the switch status. It would be impossible to give a detailed program because of the wide variety of applications that this sort of setup can be used for, but a simple program is shown below.

PROG1:

```
LOCAL p%(5),pk%(3),ck%(10),x%  
LOCAL mask%,last%,timeout%
```

```
ck%(1)=$8D06 : ck%(2)=$0558 : ck%(3)=$4645 :  
ck%(4)=$4F46 ck%(5)=$383F : ck%(6)=$19CE :  
ck%(7)=$FFFF : ck%(8)=$2503 ck%(9)=$CE00 :  
ck%(10)=$0039
```

```
IF USR(ADDR(ck%()),0)  
VIEW(1,"CommsLink NOT loaded !") : STOP  
ENDIF
```

```
p%(1)=$BD21 : p%(2)=$7400 : p%(3)=$2501  
p%(4)=$5F4F : p%(5)=$1839  
pk%(1)=$4FD6 : pk%(2)=$0318 : pk%(3)=$3900  
x%=USR(ADDR(p%()),0)
```

```
IF x%
```

```
RAISE(x%)
```

```
ENDIF
```

```
timeout%=PEEK($7C) : POKEB $7C,0
```

```
last%=USR(ADDR(pk%()),0)
```

```
DO
```

```
x%=USR(ADDR(pk%()),0)
```

```
mask%=128 : AT 1,1
```

```
WHILE mask%
```

```
IF (x% AND mask%) <> (last% AND mask%)
```

```
PRINT HOUR,":",MINUTE,mask%,
```

```
IF x% AND mask%
```

```
PRINT "Opened"
```

```
ELSE
```

```
PRINT "Closed"
```

```
ENDIF
```

```
ENDIF
```

```
mask%=mask%/2
```

```
ENDWH : last%=x%
```

```
UNTIL KEY
```

```
p%(2)=$7401 : USR(ADDR(p%()),0)
```

```
POKEB $7C,timeout%
```

Because we use Comms Link, and call various internal routines to power-up it up, it is important to make sure that Comms Link is loaded. The machine code program in array ck% does this, and returns TRUE if Comms Link has not been booted.

(continued on next page)

A Hardware Project - 2

It is in theory possible to do this check in OPL using the ONERR command, and then try to call LSET; but I have never had much success with this, whereas the routine in ck% always works.

Even though the software has been booted, you can still unplug the device, so the next machine code routine called, (in array p%) which is used to power-up the device, can report an error. For this reason we have the RAISE error.

Since the program can be left running for a long time, we don't want the Organiser to timeout, so we poke a value into \$7c to stop this happening, and restore the original value again at the end. The main body of the program is just a loop peeking at location 3 - which is strictly the Organiser data bus, but we have set this up to be the Comms Link status port by calling the machine code in p% earlier. We cannot use the PEEK instruction because it doesn't work on locations as low as 3, it always returns zero. What you actually do in the loop is up to you; here we just print the times the switches open and close together with a rough indication of which switch it was.

For a simple application like monitoring the opening and closing of a greenhouse vent - something that might only happen once a day, if at all; and you are not too worried about strict accuracy about recording the exact instant the event occurs, then LZ users can take advantage of the OFF command that species the amount of time to be off. This will conserve battery power enormously. In this case the calls to the machine code in p% will have to be moved inside the DO loop, to power-up the Comms Link again after each switch off.

That is basically all there is to it. Next time we'll look at timing loops, event counting, and how to turn the Organiser into a bicycle speedometer/trip-meter.

Editor's Note: This two-part article is in response to numerous requests for hardware projects. So as not to upset the less advanced members, I am delaying the start of the new Machine Code Series until Issue 3 (May '90) Ed.

Letters

Dear Mike,
My Organiser XP recently suffered a mains spike or some such calamity whilst at work, rendering the Comms Link, Spreadsheet and a 32k datapak (with all my programs for work) unusable.

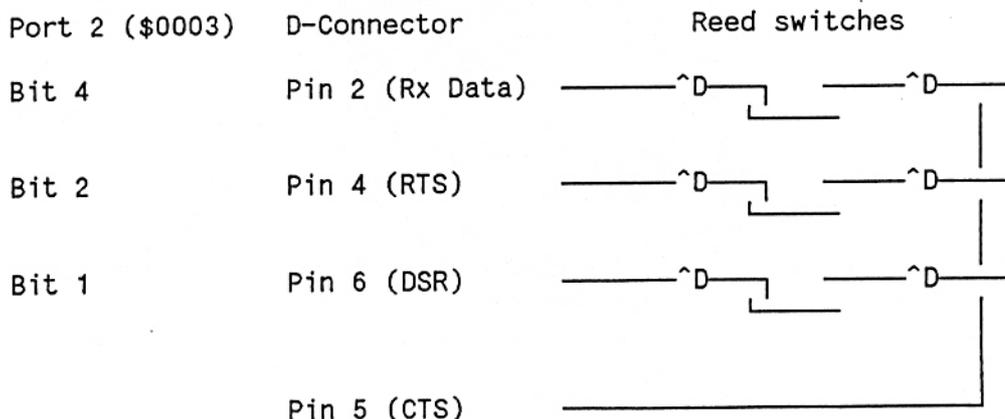
I returned the above items to Psion, together with the power supply, requesting that they advise me of the cost of repairs and that they return it before I next had to go offshore.

I was very pleasantly surprised to hear from them advising me that the Organiser and peripherals would be repaired under warranty: they were also returned to me in time to take to work. Hats off to Psion for their superb support in this matter: I would be grateful if you could publish this as I feel they deserve a pat on the back.

P.A. Carroll, Derby

Editor's Note: It's nice to hear something GOOD about Psion for a change. Have any other members had good/bad/indifferent service from any firms they have dealt with. If so, please let me know as this kind of information if a great help to prospective buyers.

Diagram for Reed Switch Article



Progs & Procs - 1

(Note: square brackets [] show start and end of single program lines and should not be keyed in)

Biorhythms by Adrian Pegg

LZ
only

The theory of BIORHYTHMS is fairly well known. Briefly it states that everyone starts on the three Biorhythm cycles on their date of birth and that these cycles repeat throughout life with the following frequencies:

PHYSICAL - 23 days
EMOTIONAL - 28 days
INTELLECTUAL - 33 days

Whether you believe in this theory is up to you. Some well-known authorities place great store on them!

When the program is first run, you will be asked to enter a date-of-birth - this should be YOUR date-of-birth as it will be saved in a file and become the DEFAULT each time you access the program. If you wish to change the default at a later date, it will be necessary to DELETE the file A:BIODATE via the XFILES or UTILITY menu and run the program again.

The next screen shows the biorhythms for TODAY (taken from the system clock) for that D.O.B The date-of-birth for which calculations are currently being made is displayed on the left of the 4th line. Next comes the current DAY, followed by the number of days since date of birth. Keys available at this time are LEFT and RIGHT cursor to add or subtract one day at a time. The MODE key allows entry of a (one time) alternative D.O.B. Any other key exits the program.

The first three lines show a cycle then a number indicating the number of days into the cycle, followed by a graphical representation based on the %age through each half of the cycle. A + sign indicates the first (positive) half of the cycle and a - sign the second (or negative) half

```
bio:
LOCAL dy,dy1,dy2,n%,p,e,i,c$(9)
LOCAL j%,g%,b$(1),r,pr,x$(1)
ONERR er::
c$="CRITICAL*"
b$=CHR$(255)
CLS
PRINT"WORKING..."
IF NOT EXIST("A:biodate")
  CREATE"A:biodate",a,d%,m%,y%
ELSE OPEN "A:biodate",a,d%,m%,y%
ENDIF
FIRST
a:
IF a.d%=0
  AT 1,1
```

```
PRINT"Enter d.o.b."
PRINT"Date :"
PRINT"Month:"
PRINT"Year :"
AT 8,2
TRAP INPUT a.d%
IF ERR=206
  RETURN
ENDIF
AT 8,3
TRAP INPUT a.m%
IF ERR=206
  RETURN
ENDIF
AT 8,4
TRAP INPUT a.y%
IF ERR=206
  RETURN
ELSEIF a.y%<100
  a.y%=a.y%+1900
ENDIF
IF a.y%<1900
  RAISE 247
ENDIF
IF j%=0
  APPEND
ENDIF
ENDIF
dy1=DAYS(a.d%,a.m%,a.y%)
dy2=DAYS(DAY,MONTH,YEAR)
n%=DOW(DAY,MONTH,YEAR)
CLS
PRINT"Phys:"
PRINT"Emo : "
PRINT"Int : "
b::
dy=dy2-dy1
AT 1,4
[PRINTa.d%/"";a.m%/"";a.y%-1900,
DAYNAME$(n%),dy]
p=dy-((INT(dy/23))*23)
e=dy-((INT(dy/28))*28)
i=dy-((INT(dy/33))*33)
AT 7,1
PRINT p,
IF p=0 OR p=11
  AT 9,1
  PRINT"*";c$;CHR$(26)
ELSE
  pr=p
  DO
    IF pr>5.75
      pr=pr-5.75
    ENDIF
  UNTIL pr<=5.75
  r=(pr*100/5.75)/10
  IF p>5.75 AND p<11.5
```

Progs & Procs - 2

```

r=10-r
ELSEIF p>17.25 AND p<23
  r=10-r
ENDIF
IF p>11.75
  x$="-"
ELSE x$="+"
ENDIF
IF r<1 :r=1 :ENDIF
AT 9,1
PRINT x$;REPT$(b$,r);CHR$(26)
ENDIF
AT 7,2
PRINT e,
IF e=0 OR e=14
  AT 9,2
  PRINT"*";c$;CHR$(26)
ELSE
  pr=e
  DO
    IF pr>7
      pr=pr-7
    ENDIF
  UNTIL pr<=7
  r=(pr*100/7)/10
  IF e>7 AND e<14
    r=10-r
  ELSEIF e>21 AND e<28
    r=10-r
  ENDIF
  IF e>14
    x$="-"
  ELSE x$="+"
  ENDIF
  IF r<1 :r=1 :ENDIF
  AT 9,2
  PRINT x$;REPT$(b$,r);CHR$(26)
ENDIF
AT 7,3
PRINT i,
IF i=0 OR i=17
  AT 9,3
  PRINT"*";c$;CHR$(26)
ELSE
  pr=i
  DO
    IF pr>8.25
      pr=pr-8.25
    ENDIF
  UNTIL pr<=8.25
  r=(pr*100/8.25)/10
  IF i>8.25 AND i<16.5
    r=10-r
  ELSEIF i>24.75 AND i<33
    r=10-r
  ENDIF
  IF i>17

```

```

  x$="-"
  ELSE x$="+"
  ENDIF
  IF r<1 :r=1 :ENDIF
  AT 9,3
  PRINT x$;REPT$(b$,r);CHR$(26)
ENDIF
g%=GET
IF g%=2
  a.d%=0
  j%=1
  CLS
  GOTO a::
ELSEIF g%=6
  dy2=dy2+1
  n%=n%+1
  IF n%=8
    n%=1
  ENDIF
  GOTO b::
ELSEIF g%=5
  dy2=dy2-1
  n%=n%-1
  IF n%=0
    n%=7
  ENDIF
  GOTO b::
ELSE
  RETURN
ENDIF
er::
ONERR OFF
CLS
BEEP 50,200
PRINT"ERROR",ERR
IF ERR=247
  PRINT"Invalid date -"
  PRINT"Please press a key and try again..."
  GET
  CLS
  IF j%=0
    ERASE
  ENDIF
  FIRST
  a.d%=0
  GOTO a::
ELSE
  AT 3,1
  PRINT ERR$(ERR)
  GET
  RETURN
ENDIF

```

Editor's Note: This, and all other programs published in Volume IV, is available on 5.25" disk or your own data/RAM pak. (See Editorial)

Product News

by Mike O'Regan

CUBSOFT Introductory Offer

As I mentioned in the last issue, Cubsoft have completed updated versions of all their software. As a special offer to IPSO members Mike Leigh is reducing all his products by £10 until 15 April. This offer is for this limited period only and applies to products bought direct from Cubsoft.

CUBSOFT are on 061 792 2871

Language Training on the Organiser

Mackay Language Software is selling language training paks, called PRAKPAKS. These work in different languages, such as French, Spanish, and even English (for foreigners). The paks are aimed at "brushing up" for people who already have a reasonable knowledge of the language concerned and wish to extend their vocabulary and test themselves before and after. A novel feature of these paks is that they can display the time in the relevant language. I had a go at the Spanish pak and found it quite helpful with my Spanish (which is a bit rusty after 35 years since I left Gibraltar). The Lsanguage PRAKPAKS cost £39.95 each (on 32k datapak). More information on these and other products is available from:

Mackay Language Software

41 Kensington High Street

Lond W8 5ED

Tel: 01 937 2077 Fax: 01 376 2368

Easy-Writing from Monad International Ltd

is a simple but very useful utility pak. It is based on the idea that most people, when writing down things quickly, often use a kind of shorthand abbreviation which they can understand later. The idea is to store as many "symbols" (as the shorthand bits are called) along with their full equivalents. Then when you type in the symbol on the Organiser, the program automatically fills it out to the full entry. This facility is especially handy in situations where it is inconvenient to type full text (I would have welcomed this program when I designed on-bus data collection systems back in the old days). Easy-Writer can also handle symbols for numeric combinations. It come on a datapak which may be removed when the program has been autobooted. The pak is available from:

Monad International Ltd

42 Laburnum Road

Chertsey

Surrey KT16 8BY

Phone (0932) 565035

COMMUNICATION WITH A PC

by Mike O'Regan

Making the Organiser "talk" to an IBM Personal Computer (or clone) is easier than you think. If you have a Comms Link and the disk which is supplied with it then the job is half done for you.

Before we look at some of the aspects of this communication it is perhaps as well to know why it is worth the effort. Well, handy as the Organiser undoubtedly is, it does have a rather small and somewhat inconvenient keyboard. Wouldn't it be great to have a full size standard keyboard to key in those long procedures from IPSO FACTO? Equally, wouldn't it be rather nice to have a really cheap mass storage for those procedures which are not used every day but which you would nevertheless like to keep ready for use - and datapaks and RAMPaks are a bit on the dear side are they not? Quite another thing is that you may have a large database on your PC which you would like to have (perhaps just a section of the whole file) on your Organiser so that you have access to it anywhere, at any time. All these things are possible if you can successfully connect the Organiser and PC.

The first thing to check is that the two machines are "talking the same language", so to speak by checking the various setup values on each machine. Doing this on the Organiser is a piece of cake - just check the SETUP values from the COMMS menu. On the PC, if you have never reset the Comms values, then you can do this by keying in (from the system prompt e.g. C>):

```
MODE COM1,9600,N,8,2
```

Now check (and change if necessary) the SETUP values on your Organiser. One thing which will probably have to be changed is the PROTOCOL, which should be set to PSION.

The next stage on the PC is to prepare a DIRECTORY for your Organiser programs and data (I am assuming that you have a HARD DISK DRIVE on the PC). Just key in MD PSION <CR>. Now get into that directory by keying CD PSION <CR>.

You can now put the COMMS LINK floppy diskette into the PC's Drive A: and type: COPY a:*. * which will put a copy of each of the COMMS LINK programs in your new directory.

(continued next month)

Beginner's Section

by Mike O'Regan

Last month the demonstration program PAYIN: introduced the use of both a LOOP and a series of CONDITIONAL JUMPS.

Whenever we use a MENU in an OPL program we automatically use two or more conditional jumps (assuming there are two or more options on the menu). In PAYIN: the menu contained FIVE options.

You will notice that there are four lines in the procedure which begin either with IF or ELSEIF. These deal with the first four options on the menu. The fifth option is automatically selected if T (for Total) is entered.

However, what is slightly unusual is that this last option is outside the loop. Let me explain this point. The third line of the procedure contains the command DO, which means that all the following instructions are repeated until the CONDITION written into the matching UNTIL (4th line from the end of the procedure) is met. In other words the MENU will continue to show and allow the various inputs until you press T. This being option 5 (i.e. greater than 4) the loop is left and the procedure continues with the line following the UNTIL... line.

In the case of this program the next two lines print the total cash and the number of cheques, and the GET (last line) holds the display until you press any key, when the program is finished.

There are one or two important points to note about using the MENU system within a Loop. You MUST have an option on the menu which allows you to escape from the loop, and you are advised (for convenience) to start each option with a different letter, if possible.

Note that options can also start with characters other than alpha, but you will have to access these with the CURSOR KEYS and <EXE>. This feature is quite handy to stop you selecting an option by mistake (see the SCORE: program on page 76 for an example. The Zero option is prefixed with a minus sign to avoid accidentally zeroing the score).

The DO-UNTIL command is one type of loop. The other (on the Organiser) is WHILE-ENDWH. Whereas the controlling factor in a DO-UNTIL loop is contained in the LAST line of the loop, WHILE-ENDWH has this in the first line. This means, in practical terms, that the specified condition is examined at the start of each loop and, if the condition is met, the loop is exited. It could be, therefore, that the lines between a WHILE and an ENDWH are never reached because the WHILE condition was immediately

satisfied. This is invaluable in some cases, especially to save time, when compared with a DO-UNTIL loop, which has by its structure to complete the loop at least once before the condition can be examined (by the UNTIL... line).

I hope this has made Loops a bit more understandable. Next month I will look at IF-ELSEIF-ELSE-ENDIF structures. Here are a couple of programs which use the two loop structures. They are simple demonstrations of how the two functions compare:

```
dtable:
LOCAL num1%,num2%
num2%=2
DO
PRINT
num1%=0
DO
num1%=num1%+1
PRINT num1%,"x",num2%,"=",num1%*num2%
PAUSE 10
UNTIL num1%=12
num2%=num2%+1
UNTIL num2%=13
```

```
wtable:
LOCAL num1%,num2%
num2%=2
WHILE num2%<13
PRINT
num1%=0
WHILE num1%<12
num1%=num1%+1
PRINT num1%,"x",num2%,"=",num1%*num2%
PAUSE 10
ENDWH
num2%=num2%+1
ENDWH
```

See if you can work out how the procedures work, and particularly why the num2% limit must be set at 13 when the num1% can be 12!! Both programs will, of course display the "old style" multiplication tables from 2 to 12. If you have a printer (attached) you can modify either program to print out tables - I have done a set each for my grandchildren!

Finally, if any of you beginners has any comment or query on this series so far, please let me know. It is much easier to put things right if I have some feedback!

Small Ads etc

For Sale

Organiser Model LA, good condition
£70

Apply to Tim Schomer
22a Owen Court
Richardson Road
Newcastle-upon-Tyne
NE2 4BH

Organiser Model LA - £50, Mains Adaptor - £7
Blank 16k datapak - £10, LACE program - £25
Comms Link - £30, Finance Pak - £20

Sell separately or will exchange for printer in good
condition (not Psion)

Phone Jim Kirwan on 0772 614185

Comms Link (with manual) - £45 ono
Contact D.P. Wood
14 Goldsmith Court
Stukeley Street
London WC2B 5LE

32k datapaks - £23.99ea
Harvester Superchip - £43.99
Phone Simon on 01 385 8816

Brand new (boxed) Organiser LA
plus Mains Adaptor, Pocket Spreadsheet
AutoScribe Plus, and Mike Shaw Book
£130 ono

Phone: Chris Clifford on 01 226 0822

Exchange

Pocket Spreadsheet (w/manual)
Travel Pak (w/manual)
exchange for 128k datapak
Contact: Tam Darcy
65 Carlton Avenue
Gillingham Kent
Tel: 0634 576455

Program Listing Blunders

A gremlin crept into the listing of A.R.P. Spencer's excellent FINDW: program (page 94), last month. The last five lines, commencing with IF LEN(c\$)... should be removed and tagged onto the end of the same procedure (after the line POSITION(f+1) - first line on page 96

Wanted

Comms Link and Mains Adaptor
please write to:

J. Leaper, NCISS, BFPO 8

Anyone with Genealogical programs
for the Organiser
please phone: Nigel Penton Tilbury
on 0480 693000

IPSO London Group

The February meeting of the London Group was again very poorly attended. For those who did attend, the evening's talk was provided by **Steve Rogers** who is Psions Business Manager for the MC range computers. Steve gave us a short slide show, covering the features of the MC range and the differences between the three machines that make up the range.

Steve was then subjected to a thorough grilling from those present, especially the two members who already have MC400s. There was then a chance to try out the MC400 and the MC600. We very fortunate to see the MC600 as it is not yet in the shops, unlike the MC400 which now seems to be freely available. Thanks, Steve, for an interesting evening.

John Lambert, who is co-sysop of the Psion section of the **Kybernesis Bulletin Board**, will be our guest speaker for the next meeting. John wrote the software for the tone dialler add-on available from Transform and will be demonstrating it to us. This will be on Tuesday, 20th March. Please contact me on 01-568 4138 for details of how to get to the London Group meetings.

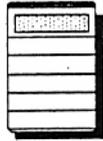
Jonathan Hurwitt

Swindon Psion Organiser User Group

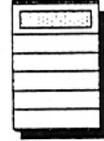
The Group continues to flourish. Meetings are held at 19.30 on the first Monday of every even-numbered month. Anyone is invited to attend, especially those within easy reach of Swindon.

Contact:

Jeremy Holt
14 Belmont Crescent
Old Town
Swindon Wilts SM1 4EY



IPSO FACTO



The Newsletter of the International Psion Organiser User Group

Volume IV . No. 2 . April 1990

Editorial Office: 130 Stapleford Lane . Toton . Beeston . Nottingham . NG9 6GB . UK

Phone: (0602) 735482 . Microlink Mailbox MAG11839

IPSOMEET 90

Will all those who are planning on coming to IPSOMEET 90 **BUT HAVE NOT YET SENT THEIR PAYMENT** please do so without delay (£8.00 per person). Cheques to be made payable to IPSO and sent to the address at the head of this page. The reason for the urgency is the need for me to pay the caterers in advance, so please take note.

Hardware Project

The hardware project, which was started in the last issue and continued in this needs some (quite long) programs to complete it. These programs were judged to be too long to be practical to print in this newsletter. However, anyone who wants printed listings can obtain them from me by sending an SAE and £1 to cover printing cost.

STOP PRESS

I have just received a copy of MACHINE CODE PROGRAMMING ON THE PSION ORGANISER by Bill Aitken. The book is available from either your local dealer, bookshop or from the Publisher,

Kuma Computers Ltd,
Pangbourne, Berks -
Tel: 0734 844335.

The price is £14.95. A full review will be featured in the next newsletter.

For Sale

- Organiser Fax Leather Case (brand new in box) - £42
- Psion Printer with extra paper - £140
- Mike Shaw File Handling Book - £10
- Diary Link - £3
- Organiser Developer (two line version) - £24
- Widget Filemaster (latest version) - £35
- Widget AutoScribe Plus - £33

Contact: Kevin Ash
15 Alexandra Road
Ashford Common
Middx TW15 1TW
or phone: 0860 382475

Pak Trouble

I have heard of several cases where members have had trouble FORMATTING datapaks. In one case the problem was so bad that Psion agreed to change the Formatter. It seems that finally the trouble has been traced to the fact that different makes of EPROM (the heart of the datapak) require differing times to properly erase. Some require DOUBLE the recommended time to erase. Anyone having problems can usually cure them by formatting the pak a second time. Personally, I have NEVER had a pak which didn't erase first time on my UVIPROM Eprom Eraser (cost £21.95!) after only 8 minutes.

When the RAMPAK was first introduced it was stated that the battery should last two to three years, so some of the original RAMPAKS should be approaching the end of their battery life. What is not clear (and several members have asked about this) is what happens when the battery needs changing. Although the battery is a fairly standard mercury cell it is not meant to be changed by the user. Maybe the best solution would be to have an exchange system where dealers would take in your old RAMPAK and give you one with a new battery for a small fee. The only problem then would be what about any data stored on the pak? Perhaps some enterprising member would be prepared to offer a complete RAMPAK service temporarily storing any information, changing the battery, restoring its contents and returning it to its owner.

Contents

Editorial.....	9
Hardware Project ..	10
Progs & Procs.....	12
Beginner's Page....	15
Product News.....	16

A Hardware Project

Part 2

Communicating with the Outside World.

by Mike Davies

In this article we shall explore how to time events. As a specific example we shall write a program that will turn the Organiser into a bicycle speedometer and trip meter. It is assumed you have read the previous article, and can now read data from a reed switch and display if it is open/closed etc.

The hardware for this project is as before. The reed switch is taped to the front fork of the bicycle, and the magnet is taped to one spoke. Make sure that the magnet is sufficiently close to the reed to switch it on/off, and that it cannot slide outwards from centrifugal force. I used the switch between RTS and CTS of the Comms Link adaptor, but you can use Rx data or DSR as mentioned before. Such a setup will give a pulsing signal as the bicycle wheel turns.

When it comes to timing events, there are two ways of doing it. You can either count the number of pulses over a set time, or, time the length of a single pulse. It is easiest simply to count pulses over a set time, because if there aren't any we simply return with zero. However, if we do count pulses, there is a danger of missing one whilst we display/calculate the results from the last iteration. In our bicycle speedometer we will time a single pulse, this is more complex but we cannot afford to miss any data.

When timing a single pulse, a timeout has to be built into the count loop, otherwise the program could wait for a pulse that never comes. For this application a timeout of 5 seconds seems adequate. A 27" (700mm) diameter wheel (which we will use in all our examples) should give 1 pulse at 1 MPH in that time. The other thing that should abort any timing loop is a keyboard hit. Failure to do so will give the appearance of a mis-hit, and it is tempting to hit the keyboard again.

What we need to do now is decide what units of time we will count it. Whatever units we decide to count in, our timed event will always be +/- 1 unit, because it is almost impossible to synchronise the start of our event with any free running counter within the Organiser. There are two such free running counters we could use. At \$20CB is a word counter that is incremented every 0.05 seconds. At \$0009 is a free running counter incremented every 0.00000108506 seconds.

If we decided to choose the counter at \$0009 then in less than 1/28 of a second we would get integer overflow. Using the counter at \$20CB is no good either. If we are really travelling at 19 MPH a pulse comes about every 0.25 secs (5 clock ticks). If we lost a tick or gained a tick then the display would waver between about 24 MPH and 15 MPH. Not very satisfactory either.

Another solution is to use our own "clock". We do this by actually counting the machine cycles each instruction takes, and count the number of times that each group of instructions is executed. In the attached machine code listing you will find the relevant instructions annotated. The routine increments the X-reg every 75 machine cycles which is about every 0.0000813795 seconds and integer overflow will occur after 5.33 secs, but since we timeout after 5 secs this should be no problem. The routine PAUSE: simply exists to extend the machine code to 75 cycles, without it the unit would be every 40 cycles and that would cause problems with a slow moving wheel. The nested BSR calls is the most effective way of wasting time in minimum memory without register corruption.

All these timings, have to be taken "with a pinch of salt". Other things are happening within the Organiser - interrupts, looking for alarms, keyboard scanning etc. ; which can extend a timing cycle. Interrupts can be turned off, but I have never liked doing this. Because our unit of time measurement is so small, the routine returns quite a large count, so the laws of statistics come into play and reduce the interruptions caused by the interrupts (no pun intended) to a minimum.

The machine code source listing is well commented, and the attached OPL program runs quite well in practice. The only problems experienced are at high speed where I assume extra vibration through the bicycle forks can cause the reed to bounce causing a false reading now and again. This is quite a common event when testing switches and there are several ways of getting around the problem. In this specific example we ignore all timed intervals of 0.8 milli secs or less.

The program displays the current speed, total distance travelled, the current trip distance, and the average speed travelled during the trip. All statistics are saved in the BIKEDATA file and restored and dated each run. The available keyboard commands can change the display from Imperial to S.I. units, and the "T" key zeroes the trip meter.

(continued on next page)

A Hardware Project (cont).

I hope these two articles have opened your eyes to other potential applications. Actually making the Organiser output signals to control devices such as stepper motors etc, requires some knowledge of electronics, but if anyone would like to try, I can recommend the book "Electronic Circuits for the Computer Control of Robots" by R.A. Penfold in the Bernard Babani series.

(see Page 9 for information on the programs)

Copying a procedure from Organiser to PC

by Mike O'Regan

1. Make sure that you are in the PSION directory on the PC.
2. On the COMMS menu choose TRANSMIT and PROCEDURE. Key in the name of the procedure and press <EXE>.
3. Leave the name the same on the Organiser bottom line.
4. Key in CL <CR> on the PC
5. Finally press <EXE> on the Organiser and the procedure will be transmitted to the PC.
6. Type DIR <CR> on the PC and you should see the name of the procedure which you have just transmitted, which now has the suffix .OPL.

Writing an OPL procedure on the PC

1. Use your word-processor to write the procedure as a NON-DOCUMENT (or ASCII) file.
2. Write the procedure exactly as you would on the Organiser (Note that you can use characters - such as ? ' etc, which are not normally available from the Organiser keyboard).
3. Save the file with any valid filename, plus a .OPL suffix.
4. If necessary, COPY the file to the PSION directory.
5. Make sure that the COMMS LINK is connecting the Organiser and the PC.
6. Select RECEIVE and PROCEDURE from the COMMS MENU on the Organiser.
7. Key in the name of the procedure held in the PSION directory on the Organiser and press <EXE>.

8. The bottom line on the Organiser will then suggest the same procedure name, which you may change, if you wish.

9. Key in CL <CR> on the PC.

10. Press <EXE> on the Organiser and the procedure will be transferred.

11. You can now TRANS and SAVE the procedure, just as if you had keyed it in in the normal way!

NOTE: The secret of using the COMMS LINK software on a PC is to have it IN THE SAME DIRECTORY AS THE FILES YOU WISH TO TRANSFER

Next month I will explain how to transfer a datafile from PC to Organiser.

For Sale

Organiser LA Model, box, Manuals £50
Widget Filemaster, latest version £30
DIP Pure Finance Pack £25
Harvester Superchip £45
Phone Call Cost Calculator £8
128k Datapak £70

Phone TONY WEBSTER on 0280 812915
or Telecom Gold/Microlink MAG80005

Organiser LZ 64
4 line Pocket Spreadsheet
FNKEY v3.1
Comms Link

Write with your best offer (around £250) to:
R.A. Stewart
23 Parmington Close
Callow Hill
Redditch
B97 5YL

Tandy CGP115 Plotter/Printer
c/w 6 rolls of paper, dust-cover,
printer-computer lead,
4 packets of pens (unopened) (3 colour, 1 Black)
£50 (plus p&p)

Write to J.J. Vincent
14 Cambridge Road
Bromley, Kent
BR1 4EA

Progs & Procs - 1

DIR

by Neil Draycott

This is the first MEMORY RESIDENT program for the XP and LZ models (at least I have never seen any published).

The program, called "DIR", gives a directory of any type of file (data files, diary files, OPL programs etc.) and remains resident in the "hidden" top area of RAM to be called whenever required from the main menu.

Because the code employs software interrupt vectors the installed program is only 84 bytes long and takes only 0.7% of the RAM space on an XP model. The installation program takes note of any other memory resident programs such as FNKEY and should not be affected or interfere with them.

"DIR" will remain resident until the Organiser "crashes" or the battery is removed.

TYPING THE PROGRAM

Enter the program EXACTLY as listed, remember mistakes can erase ALL records. Re-check, it is easy to be impatient. Translate and save the program, it must be called "DIR".

INSTALLATION

Insert "DIR" into the main menu, this program MUST be run from the main menu.

If "DIR" cannot be found in the main menu then a warning will be given and the installation will stop.

When the message "<DIR> installed" is displayed then the typed-in program is no longer needed and can be saved to a datapak for future installations or just erased.

USING THE PROGRAM

Now whenever "DIR" is selected from the main menu; firstly the PAK is chosen by using the <MODE> key (<CLEAR> key will exit) then the type of files to be listed are chosen from a menu, pressing <MODE> at this point will return to the selection of PAK and pressing <CLEAR> will exit. All the type of files selected on the chosen PAK are then displayed while EXE is pressed until "END OF PAK" is displayed or <CLEAR> is pressed.

ABOUT THE PROGRAM

If "DIR" is removed from the main menu then the installation program will have to be run again when "DIR" is re-inserted. Every time the installation program is run a further 84 bytes are lost from the RAM, this could have been prevented but I decided the extra program length was not worth it. The installation program uses memory 9 of the calculator as an external variable, so any data in M9 is trashed. The selection of PAK also tests which PAKs are present. An interesting feature is the inclusion of a menu which can be exited by pressing <MODE>, try that in OPL!

EXPLANATION OF CODE

For the machine code buffs a step-by-step explanation of the main directory part of the program is included below. Perhaps someone could explain why the installation has to run twice?

CODE	MNEMONIC	NOTES
7F 2187	CLR RTB_BL	Clear the run-time buffer leading byte count.
CE T%	LDX PROMPT\$_A	Load X with the address of prompt string.
CC 0000	LDD #0	A=0, B=0 (single line edit, 0 max input).
3F 68	SWI #104	Edit filename.
25 18	BCS #24	Exit if <CLEAR> pressed.
CE T%+4	LDX MENU\$A	Load X with address of menu string.
CC 1003	LDD \$1003	Exit key bit mask- allows EXE, CLEAR & MOD to exit selection in menu
3F 50	SWI #80	Menu.
96 42	LDA A \$42	Load A with selection No. (0-4).
88 81	ADD A \$81	Add \$81 to make a valid file type.
97 42	STA A \$42	Store result at \$42.
C1 01	CMP B #1	Compare key pressed to key 1.
27 06	BEQ #6	Exit if <CLEAR> pressed.
C1 02	CMP B #2	Compare key pressed to key 2.
27 DD	BEQ #35	Goto start if <MODE> pressed.
3F 7B	SWI #123	Display device catalogue.
39 00	RTS	Return to system.

Although "DIR" will run on the LZ model there is not really any point; the same program exists within the UTILS function and I have not included the LZ files such as Pager, Xdiary etc.

```
DIR:
LOCAL T%,P$(45),X%,Y%,B%(19),X$(1)
IF M9<>-999
M9=-999
T%=PEEKW($2065)-84
POKEW $2065,T%-1
DIR:
ENDIF
```

(listing continued on next page)

Progs & Procs - 2

```

PRINT"Installing.."
T%=PEEKW($2065)+1
[P$="3DIR5Files005Diary003Op1005Comms006Spr
ead0000" ]
DO
X$=MID$(P$,X%+1,1)
Y%=ASC(X$)
IF Y%<57
Y%=VAL(X$)
ENDIF
POKEB T%+X%,Y%
X%=X%+1
UNTIL X%>44
X%=PEEKW($2002)
Y%=0
DO
IF PEEKW(X%+Y%)=$0344
IF PEEKW(X%+Y%+2)=$4952
POKEW X%+Y%+4,T%+45
GOTO C::
ENDIF
ENDIF
Y%=Y%+2
UNTIL Y%>255
CLS
PRINT" Please INSERT"
PRINT" <DIR>";CHR$(16)
GET
STOP
C::
B%(1)=$7F21 : B%(2)=$87CE
B%(3)=T% : B%(4)=$CC00
B%(5)=$003F : B%(6)=$6825
B%(7)=$18CE : B%(8)=T%+4
B%(9)=$CC10 : B%(10)=$033F
B%(11)=$5096 : B%(12)=$428B
B%(13)=$8197 : B%(14)=$42C1
B%(15)=$0127 : B%(16)=$06C1
B%(17)=$0227 : B%(18)=$DD3F
B%(19)=$7B39
X%=1
Y%=0
DO
POKEW T%+45+Y%,B%(X%)
X%=X%+1
Y%=Y%+2
UNTIL X%=20
CLS
PRINT" <DIR> installed";CHR$(16)
PAUSE 10
STOP

```

(Editor's Note: Neil points out that this is probably the first publication ANYWHERE of a MEMORY RESIDENT program for the Organiser!)

STRCALC
by David Green

This program was written to overcome the Organiser's refusal to accept simple numeric equations in response to an INPUT command. This is a stand-alone procedure to illustrate the sequence, but it could be adapted so that it could be called from another procedure if the feature was needed.

STRCALC will run on any model, but the U: procedure is required for the XP/CM models (this has been previously published and is in the Handbook also)

```

strcalc:
LOCAL x,a$(50),l%,p%,q,x%,y,z
ONERR l1::
u:(0,0,0,1,2,20,8,0,0)
l1::
KSTAT 3
AT 1,1
IF q=0
PRINT "Input Num+OP+Num"
ELSE
PRINT "Input OP+Number"
ENDIF
AT LEN(GENS$(q,30))+1,2
INPUT a$
IF a$=""
RETURN
ENDIF
x=0
DO
x=x+1
IF MID$(a$,x,1)=CHR$(41)
AT 2,2
PRINT MID$(a$,1,x-
1);CHR$(0);MID$(a$,x+1,LEN(a$)-x)
PAUSE 20
ENDIF
UNTIL x=LEN(a$)
a$=GENS$(q,30)+a$
x%=1
l%=LEN(a$)
DO
x%=x%+1
IF MID$(a$,x%,2)="*"
y=VAL(MID$(a$,1,x%-1))
z=VAL(MID$(a$,x%+2,l%-x%-1))
q=y**z
GOTO l2::
ENDIF
UNTIL x%=l%
x%=0

```

(listing continued on next page)

Progs & Procs - 3

```

DO
x%=x%+1
p%=ASC(MID$(a$,x%,1))
IF p%=37 OR p%=41 OR p%=42 OR p%=43 OR
p%=45 OR p%=47
y=VAL(MID$(a$,1,x%-1))
z=VAL(MID$(a$,x%+1,1%))
ENDIF
IF p%=37
q=(y/100)*z
ELSEIF p%=41
q=z**(1/y)
ELSEIF p%=42
q=y*z
ELSEIF p%=43
q=y+z
ELSEIF p%=45
q=y-z
ELSEIF p%=47
q=y/z
ENDIF
UNTIL x%=1%
l2::
PRINT q;
GOTO l1::

```

TIMER

by David Green

This is yet another of David's TIMER programs - with a new feature by popular request. It is possible to PAUSE the timer and resume at a later time. Pressing P for PAUSE toggles this feature.

```

timer:
GLOBAL a,b,c,d,h,m,s,e$(2),f$(2),m%
u:(0,4,4,4,4,21,14,4,0)
DO
d=DAY
h=HOURL
m=MINUTE
s=SECOND
e$=MID$(DATIM$,17,2)
f$=MID$(DATIM$,20,2)
a=d*86400+h*3600+m*60+s
m%=MENU("Start/Stop,PAUSE,Intermediate-
Time")
IF m%=0
RETURN
ENDIF
CLS
AT 1,2
t:

```

```

b=INT(a/3600)
c=a-b*3600
AT 1,1
PRINT "S";CHR$(0),"T";"=";b;";";INT(c/60);";";c-
INT(c/60)*60
AT 15,1
PRINT CHR$(0);"F"
AT 5,2
PRINT "hrs"
AT 10,2
PRINT e$+f$+"hrs"
UNTIL GET=1

```

HUNT

by David Green

Hunt is *almost* like SEARCH on the Utilities Menu of the LZ. It is meant to run on the CM/XP models and its operation is self-explanatory.

```

hunt:
LOCAL str$(20),p$(1),fil$(16)
KSTAT 1
PRINT "FIND:";
INPUT str$
p$="A"
DO
fil$=DIR$(p$)
DO
IF KEY$="Q"
RETURN
ENDIF
CLS
PRINT fil$
[OPEN fil$,a,a$,b$,c$,d$,e$,f$,g$,h$,i$,j$,k$,l$,
m$,n$,o$,p$ ]
DO
IF KEY$="Q"
RETURN
ENDIF
IF FIND(str$)
CLS : PRINT fil$ : PAUSE(10)
AT 1,2
IF DISP(-1,"")=81
RETURN
ENDIF
NEXT
ENDIF
UNTIL EOF
CLOSE
fil$=DIR$("")
UNTIL fil$=""
p$=CHR$(ASC(p$)+1)
UNTIL p$="D"

```

Beginners Page

by Mike O'Regan

Notes for Beginners on the use of the Organiser's built-in database.

I was recently reminded by looking at a new member's database that using FIND and SAVE in the most efficient way is not always understood. The Organiser's facilities are certainly one of its major advantages over other machines. It is very easy to just use the SAVE option to enter data in a haphazard fashion.

The same is not true of most databases on other computers, where it is necessary to give some thought to the STRUCTURE of the database before designing and using it. The Organiser, on the other hand, lets you enter data immediately (with no thought of anything like a recognisable structure). The Organiser then lets you use FIND to get you out of this self-imposed chaos! You can use any clue to FIND any record containing the clue ANYWHERE, regardless of lack of structure.

Other - more structured - databases require you to state which FIELD(S) will be used with a FIND clue. Let me give you an example.

Use your Organiser to enter the following in A:MAIN. Don't worry if you already have some records in A:MAIN - these trial records can be deleted later.

Switch your Organiser on and make sure that you are at the top-level menu. Press S (for SAVE) and key in the following records:

```
Smith, Jonathan <DOWN CURSOR>  
Tel: 01-123-4567 <DOWN CURSOR>  
Brain surgeon <EXE>
```

```
Jock McTavish <DOWN CURSOR>  
Haggis salesman <DOWN CURSOR>  
Glasgow 987654 <EXE>
```

```
Dentist, Birmingham 34567, I. Tuggem <EXE>
```

```
PIN Number 9393939 <EXE>
```

The first thing to notice about this little "database" is how easy it was to enter the data!

The second point is just how easy it is to use FIND to get at the data. Just try keying in BRAIN as a clue. This will immediately get you the first record - Smith, Jonathan ...etc - in spite of the fact that the clue was on the third line (FIELD) of the record. This would be true of any of the records.

However, if you wanted to use one of the utilities to SORT this FILE into alphabetical order, then the chaotic nature of the file becomes immediately apparent. Sorting works on the FIRST field (that is the first characters of the record).

It would probably help if you could imagine that your Organiser is a card-index file of the manual kind - that is a little box of cards where each card has 16 lines to accept information. Usually, to remind you how to enter the information, the blank cards have LABELS on each line, such as SUR-NAME, PHONE NO. etc.

Of course your Organiser has no such labels unless you are using either a commercial database system or you have written your own in OPL. So it is a good idea to actually write out a small card and keep it handy whenever you are entering new records.

This brings me to another point. Many users of the Organiser complain about their information scrolling across the display. This is usually because they enter data as a single line, probably with the various bits just separated by a comma. Then if the data is longer than the display length it will scroll. What you are actually doing is entering records which have only one FIELD (line) when there are actually 16 FIELDS kindly provided by Psion for your use.

If you use them properly, it is possible in some cases to avoid scrolling altogether. Also, when used in this way (that is a separate line for each bit of information) it is possible to so arrange things that the bits of data which are used most often - e.g. Name and Phone Number - are visible on a non-scrolling display as soon as FIND is used.

If you have a LIZZY then you can have four lines showing in this way.

Finally, do not forget to DELETE the trial records which you now have in A:MAIN

Next month we will start looking at writing your own database system in OPL and we shall examine the advantages of doing things this way, even on the new models with their extra file-handling facilities..

Don't forget that I welcome feedback about any topic raised in this series.

Product News

by the Editor

Another Assembler for the Organiser.

A new member has sent me a very interesting Assembler on datapak. Although the program is not as all-embracing as the Assembler recently reviewed, it is nevertheless a very interesting development (according to Les Ball, who has had a quick look at it). The authors of this pak describe it as a FULL SYMBOLIC TWO-PASS ASSEMBLER, WITH BUILT-IN EDITOR. A full manual was not yet available (the authors are just completing it). It is planned to sell the package for £29.95.

For further information please contact:
Sam Walker on 0453 883497.

Harvester Cease Trading?

As far as I have been able to ascertain, **Harvester Information Systems**, who were a major supplier of Organiser software, have ceased trading (at least for the time being).

I have had numerous enquiries from both members and others who have had no success contacting Harvester after sending payments for software. Members are advised to wait for more positive news which I will publish as soon as I find out any more.

New Psion Catalogue

To coincide with the WHICH COMPUTER SHOW, which is on at the NEC later this month, Psion will be publishing their latest catalogue of third-party products for the Organiser. You should be able to get a copy from your local dealer, or direct from Psion.

Psion have, by the way, recently changed their address and telephone numbers. They are now as follows:

Psion PLC (or Psion UK PLC)
Alexander House
85 Frampton Street
London NW8 8NQ

Tel:01-262-5580

Fax:01-402-3144

Telex: 263258 PSIONC G

Technical Support are at the same address, but with their own phone number: 01-262-8188

Note that the above phone numbers will change from 01 to 071 from 6 May 1990

Psion Production & Service Centre
17-19 Bristol Road
Greenford
UB6 8UP

Tel: 01-566-6162 (01 changes to 081 from 6 May)

The Engineering & Development Departments remain at Psion House, Harcourt Street, and Huntsworth Mews

SUMS - an arithmetic trainer for children.

I have received a review copy of SUMS, a neat little trainer in basic arithmetic for children. It operates on the lines of the "Little Professor" and rewards right answers with a little tune. It certainly was popular with children I have tried it on!

SUMS is available (at the very reasonable price of £5 plus £1 p&p) from:

ZIP Software
40 Lennox Road
Gravesend
Kent DA11 0ER
Tel: 0474 365206

ZIP will also supply a list of their other budget software on request.

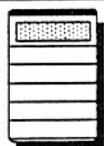
DATABOX

I have had news of another "alternative-to-datapak" device. The Databox is a similar device to that developed by Pete Littlewood, but with some important differences:

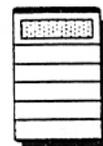
1. It uses 64k eproms, costing around £10 each
2. It uses the widely available 12.5v type (not 21v)
3. It will be cheaper - around £40 or £45 with ZIF socket - price depending on demand.

In other words the first 64k of storage will be around £55 with each additional 64k for just £10. The box is 49mm square, so it is quite pocketable.

Unfortunately the information sheet has become detached from THE NAME OF THE DEVELOPER so will he please get in touch so that I can print the details in the next issue!!



IPSO FACTO



The Newsletter of the International Psion Organiser User Group

Volume IV . No. 3 . May 1990

Editorial Office: 130 Stapleford Lane . Toton . Beeston . Nottingham . NG9 6GB . UK

Phone: (0602) 735482 . Microlink Mailbox MAG11839

Looking Towards Europe

One of our members is a professional translator and has commercial contacts in Italy. If any IPSO members require translation or help in marketing their products in Italy, then he has offered his services at a reduced rate. He is also interested in getting in touch with other members in Italy with a view to forming an Italian Group.

Anyone interested in either matter should contact:

Sandro de Rosa
Via G. Colombina, 1
I-35134 PADOVA PD
ITALY

Editorial

IPSOMEET 90

As planned, IPSOMEET 90 was held on Sat 5 and Sun 6 May '90. A total of 153 members and guests attended over the two days (somewhat less than we had expected and planned for, but more of that later). First of all, I must thank our friends **Herman van Hauwe** and **Astrid van Loveren**, of PocketMinded, and **Theo van Groenderbeek** and family all of whom made the trip from Holland and Belgium respectively

With their usual reliability, **Mike Davies and Maggie Dean** (of ROVOREED) provided the hard and software sales stand for the two days, and also thanks to Denwood Design for displaying their range of excellent leather cases. A very special "Thank you" to **Peter Houppermans** (from Psion) who provided many interesting "teach-ins" on both the Organisers and the MC 400, and to **John Seymour** (Psion) who brought a precious one-off Thesaurus pak for us to examine.

Finally, thanks to **Mike Leigh** (Cubsoft), who came on Sunday and explained some of the mysteries of the well-known FNKEY and XBASE programs. I think everyone who came enjoyed the excellent buffet provided by our caterer **June** and her Ladies

There was just one blot on the proceedings. The week before IPSOMEET, my wife and I spent some time telephoning members who had expressed an initial interest in coming, but from whom we had heard nothing further. Some TWENTY people confirmed that they would be present and would "pay on the day". We took these people on trust and provided buffet for each of them. On the day, these twenty didn't show up (with no explanation whatever!) leaving us picking up the extra £160 bill for wasted buffet etc. I leave this matter up to the conscience of these members as to whether or not they pay what they owe!!

Finally, I must apologise to those who came on the Sunday - I was not at my best, having contracted some kind of virus which laid me low for the next few days.

The Spectrum Connection

I have had a number of members enquiring about the ins and outs of connecting an Organiser and a Sinclair SPECTRUM. If any member has the necessary information, please contact the Editor with a view to writing a short article for the Newsletter to explain how it is done (in a simple way). This would be much appreciated.

Contents

Editorial	17
IPSO Review	18
Progs & Procs	20
Beginners Page	22
Product News	23
Machine Code	24

IPSO Review - 1

MACHINE CODE PROGRAMMING ON THE PSION ORGANISER

by Bill Aitken

Book review by Neil Draycott

Since my first ventures into machine code on PCs I have always been frustrated by the various text books available on machine code programming in general or the very un-informative manuals aimed at specific machines. I had no such problems with the Organiser as there has never been any such publications to date. I therefore greeted the arrival of Bill Aitken's book from KUMA Press (the people who brought you Mike Shaw's 2 books) with a heavy heart. However within an hour I was quite enjoying the read. This publication is over 300 pages long, a size which promised plenty of information without weakening the coffee table, and so with an old XP by my side I turned to page one.

The first chapter "Ancestry of the Psion Organiser" is in fact a brief history of the development of computers in general. Although interesting, I would have preferred some facts about the old one line Organisers, Psion in general, emergence of the XP model etc.

Next we move on to the structure of the Organiser and in this chapter Bill has produced a nice analogy between computers and the human body, with the workings of the Organiser being explained simply and clearly. At this stage this book seemed to be aimed at the first time machine code "dabbler".

The next two chapters confirmed this by giving an in-depth explanation of the number systems and terms essential to any such programmer and included enough "training exercises" and examples to make sure the points were driven home

A few chapters into the book and we actually start programming machine code, and this is where things went a little haywire. Included in the Annex are listings of a simple dis-assembler, number converter and RAM accessing programs to be used to run machine code programs. The misprints in my copy were so bad that it took a long time to sort out what these programs were meant to do.

Having corrected all the mistakes the operation of the "Hex loader" was nothing like the description in the book and, because of bad OPL techniques, ran so slowly that entering code became a tedious affair. However I persevered and was instructed in the precise art of machine code programming in a logical fashion with each example building on the knowledge gained from the previous one.

All the very basics were covered, addition and subtraction, use of the stack, display and keyboard control, decision making and redirecting programs, but I was a little disappointed with the examples of complete programs. These were just number crunching code sequences which, with a little more work, could have involved more screen activity and therefore been more satisfying.

Once the first time programmer has grasped the principles behind some of the code (and probably "crashed" the Organiser more than once by experimenting) then the author suggests other avenues that can be explored with the techniques covered.

With this in mind I turned to the next section; "The Organiser Operating System" which shows how some of the functions of the OS can be called by the programmer via the Software Interrupts. This is one of the most interesting uses of machine coding and the most rewarding. Included within this chapter are a few examples to get you started on what can be an addictive "messaging about" with the Organiser operating system. The author has sensibly chosen some of the "safer" vectors as examples to guarantee some initial success.

I then turned the page to get some answers to my own problems with the OS and came to a halt, with less than half the book read the rest is a reference section including a whole chapter on physical details of the microprocessor (surely only of use to hardware developers) and a memory map (available in the Organiser user's handbook).

What is useful is a well laid out list of the instruction set in both alphabetical and numerical order, but the list of vectors also included is full of misprints and in some cases are in error.

Conclusion

As the author himself explains, this book is pitched somewhere between extremely dry and too funny for words and includes many good explanations of what can be difficult-to-grasp subjects. The level of understanding is aimed at the first time machine code programmer wishing to get into coding. However, the included programs are only worth typing in because there are no cheap and cheerful assembler programs available at present to do the work (could this be rectified by an IPSO member?) [SAM WALKER seems to have done this. Ed] and the book does stop short of some of the more interesting things that can be done with machine code.

(concluded on next page)

IPSO Review - 2

For the experienced programmer this book is useful as a condensed version of the Psion Technical Reference manual (but without LZ and I/O updates), for a novice I would say 'be patient'!

Machine code programming on the Psion Organiser is available from:

Kuma Computers Ltd.
Pangbourne
Berkshire.

Price £14.95

Re Machine Code Programming by Bill Aitken

I read in IPSO FACTO that you are about to review the Machine Code book by Bill Aitken. I have had the book for some months now and I write to offer my thoughts and comments on the book.

I have found that the flow of the text, from starting with a very simple routine, and progressing through ever-increasing more complex concepts, useful and logical. Unfortunately, there are some negative aspects of the book which, in my view, seriously detract from his work.

The text, when introducing his machine code loading/viewing routines, page 49, I think, suggest that his routines contain a discrete easy method to load machines code routines into the reserved high RAM space. This is far from true; the only method that I have found to load machine code is to access the reserved space through the <view> option and to use the <alter> option. Hardly, I think, what is suggested by his text or, I suspect, by his intentions. Similarly he suggests that there is a simple <disassable> option; again not entirely true - this option has to be accessed through a very convoluted route.

An analysis of his OPL procedure suite shows the declaration of a number of variable never used, and indeed, some variables given values but where no action seems to depend on their values. I am inexorably drawn to the conclusion that the OPL procedure suite published is not the final version intended for publication.

I have noticed one or two misprints in the procedures, mostly affecting display, but one is the <DECODE> routine and leads to a misrepresentation of a disassembly. Further, I think that the disassembly of code

associated with SWI (hex 3F) is, in fact, wrong. PSION use SWI as an operating system call, calling built-in routines and the byte following the SWI is a the number of the OS routine called i.e. SWI is a 2 byte instruction; Aitken treats it as a one byte instruction and disassembles the second byte as a normal machine code instruction.

Lastly the listing of the OZ routine calls stops well short of all the routines built into the XP, and the extra routines in the LZ are not mentioned at all.

I have, over the last two months, used Bill Aitken's OPL procedures as a base, and have "corrected/extended" them to remedy the "deficiencies" discussed above. I have produced a 5.25" disk containing all the OPL/ODB routines. Perhaps these could be offered to members, if permission can be obtained from KUMA?

Tony Spencer

A Final Word from the Editor

Many thanks to Tony for this generous offer. I have spoken to KUMA, who admit that there are some mistakes in the first print of Bill Aitken's book. Errata sheets have been produced and can be obtained directly from them by sending a S.A.E. There is also a supplement covering the LZ additions available. Finally, KUMA have kindly agreed that we can distribute the disk mentioned above, which I will now arrange.

For Sale

LZ64, plus Comms Link, all supporting manuals "Using & Programming.." book, 32k datapak, Finance Pak, Oxford Spell Checker
all for £220 ono

Tel: **John Cornforth** on (0423) 886928

UVIPAK Eprom Erasers

I have had numerous requests to re-publish the source of the UVIPAK Eprom Eraser (cheap substitute for the Psion Pak Formatter) which was mentioned in last months newsletter. There are 3 models (all suitable) and many members use this equipment to their full satisfaction. The UVIPAK is obtainable from the makers, who operate a credit card ordering system:

Ground Control Ltd
Tel (0702) 230324

Progs & Procs - 1

London by Adrian Pegg

I had several programs submitted to convert the old 01 London telephone codes to the new 071 and 081 codes, but Adrians is the best. It is fully automatic.

Press <ON/CLEAR> when you have finished. This version will run on ANY model. There is also an LZ version, which is available as a listing, or on pak or disk if required. Lack of space prevents publishing both.

```
london:
GLOBAL ex%,nc%,g%,x%,cn%
ONERR error::
CLS
start::
CLS
PRINT"Exchange number";CHR$(63);
AT 1,2
PRINT "01- "
KSTAT 3
AT 4,2
cn%=0
x%=0
ex%=0
DO
  tt::
  x%=GET
  IF x%=1
    RETURN
  ELSEIF x%=8
    ex%=ex%/10
    cn%=cn%-1
    AT 4,2
    IF ex%>0
      PRINT ex%;
    ENDIF
    PRINT " ";
    GOTO tt::
  ELSEIF x%<48 OR x%>57
    x%=0
    GOTO tt::
  ELSE x%=x%-48
ENDIF
ex%=(ex%*10)+x%
AT 4,2
PRINT ex%;
cn%=cn%+1
UNTIL cn%>2
IF ERR=206
  RETURN
ELSEIF ex%<200 OR ex%>998
  PAUSE 5
  BEEP 10,50
  ex%=0
  GOTO start::
```

```
ENDIF
AT 16,1
PRINT " ";
lonchk:
AT 1,2
PRINT "0";nc%;"1-";ex%
KSTAT 4
g%=GET
IF g%=1 OR g%=8
  RETURN
ELSE ex%=0
  AT 1,2
  PRINT" ";
  AT 1,2
  POKEB $76,g%
  GOTO start::
ENDIF
error::
ONERR OFF
RETURN
```

```
lonchk:
nc%=8
IF ex%>=200 AND ex%<=209
ELSEIF ex%>=290 AND ex%<=305
ELSEIF ex%>=307 AND ex%<=314
ELSEIF ex%>=316 AND ex%<=319
ELSEIF ex%=330
ELSEIF ex%=332
ELSEIF ex%>=335 AND ex%<=337
ELSEIF ex%>=339 AND ex%<=349
ELSEIF ex%>=360 AND ex%<=368
ELSEIF ex%>=419 AND ex%<=424
ELSEIF ex%>=426 AND ex%<=429
ELSEIF ex%>=440 AND ex%<=453
ELSEIF ex%=455
ELSEIF ex%=456
ELSEIF ex%>=458 AND ex%<=464
ELSEIF ex%>=466 AND ex%<=472
ELSEIF ex%=475
ELSEIF ex%=478
ELSEIF ex%=479
ELSEIF ex%>=500 AND ex%<=509
ELSEIF ex%=514
ELSEIF ex%>=517 AND ex%<=521
ELSEIF ex%=523
ELSEIF ex%=524
ELSEIF ex%>=526 AND ex%<=536
ELSEIF ex%>=539 AND ex%<=547
ELSEIF ex%>=549 AND ex%<=579
ELSEIF ex%>=590 AND ex%<=599
ELSEIF ex%>=640 AND ex%<=695
ELSEIF ex%>=697 AND ex%<=699
```

(listing cont. on next page)

Progs & Procs - 2

```
ELSEIF ex%>=740 AND ex%<=752
ELSEIF ex%>=754 AND ex%<=756
ELSEIF ex%>=758 AND ex%<=771
ELSEIF ex%=773
ELSEIF ex%>=776 AND ex%<=778
ELSEIF ex%=780
ELSEIF ex%=781
ELSEIF ex%>=783 AND ex%<=789
ELSEIF ex%>=800 AND ex%<=811
ELSEIF ex%>=840 AND ex%<=859
ELSEIF ex%>=861 AND ex%<=864
ELSEIF ex%=866
ELSEIF ex%>=868 AND ex%<=871
ELSEIF ex%>=874 AND ex%<=900
ELSEIF ex%>=902 AND ex%<=910
ELSEIF ex%=913
ELSEIF ex%=914
ELSEIF ex%>=940 AND ex%<=954
ELSEIF ex%>=958 AND ex%<=961
ELSEIF ex%>=963 AND ex%<=969
ELSEIF ex%=974
ELSEIF ex%=975
ELSEIF ex%=977
ELSEIF ex%>=979 AND ex%<=989
ELSEIF ex%>=988 AND ex%<=998
ELSE nc%=7
ENDIF
RETURN
```



Last month's program TIMER needs the following extra procedure to run:

```
t:
IF m%=2
IF m4>0
STOP
ELSE
m6=a
a=a-m9
m9=m9+a
m4=a
GOTO l1::
ENDIF
ENDIF
IF m4>0
m5=a
a=m4
```

```
m4=0
GOTO l1::
ELSE
a=a-m9
a=a-m5+m6
m9=m9+a+m5-m6
m5=0
m6=0
ENDIF
l1::
IF m8<10
PRINT "0";m8;
ELSE
PRINT m8;
ENDIF
IF m7<10
PRINT "0";m7;
ELSE
PRINT m7;
ENDIF
IF m%=3
m9=m9-a
RETURN
ENDIF
IF (m%=2) AND (m4>0)
RETURN
ENDIF
m8=VAL(e$)
m7=VAL(f$)
RETURN
```

Long Lines in Listings

Because of the two-column format of IPSO FACTO, it is inevitable that some program lines will not fit on a single line within the column. In this case, I have the convention of enclosing the line between SQUARE BRACKETS []. These are NOT TO BE KEYED IN, but are just used as markers of the beginning and end of a long line. In any case, the square bracket, although included in the Organiser extended ASCII set, is not directly useable from the Organiser keyboard.

Procedures on Disk or Datapak

I am still getting requests for listings on disk or datapak (which I have mentioned a couple of times). The offer at the moment is restricted to those progs and procs printed in this Volume (4). Eventually, I hope to complete the task of putting ALL IPSO's published programs on disk, and when this is complete they will all be made available.

Beginners Page

by Mike O'Regan

PC Communications

Transferring database files between the Organiser and a PC.

One question I get asked more than perhaps any other is about the process of transferring database files between the Organiser and other computers. Although this is usually quite straightforward, the instructions in the Comms Link Handbook are not exactly easy to follow. These instructions are adequate if you already know the basics of inter-computer data transfer, but are rather confusing for the beginner.

Although the following information is related to data transfer between the Organiser and an IBM PC (or clone), the principals are basically the same for any computer.

Most database systems on a PC allow the creation of what is known as ASCII EXPORT FILES. This just means that they can reduce the information to a file where the various FIELDS are separated by a specified ASCII character (usually a COMMA - ASCII code 44) and each RECORD is separated by another ASCII character combination (usually a LINE FEED <LF> - ASCII character 10 and CARRIAGE RETURN <CR> - ASCII character 13).

These "default" characters CAN be used without any changes to produce a file which the Organiser can recognise. There is just a small drawback - all fields in each record will appear on the Organiser as a single line, with the fields separated by a comma. This is because (unless you change it) the FIELD SEPARATOR character on the Organiser is not a comma - it is a TAB character (ASCII code 9).

Most PC databases will allow you to change the field separator from the comma to another character of your choice, so you should be able to change this to a TAB character (by changing this to a figure 9). Now when you transfer this file to the Organiser each field will appear on a different line (subject, of course, to a maximum of 16 lines).

This brings me to another point. Quite often a SECTION of the PC database is what is required as the basis of the Organiser database. Let me give you an example. I keep the main IPSO database (which contains ALL the information on each member) in a (very large) database on my PC. It would be nice to have all this information available on the Organiser, but the file is too big for the biggest datapak. However, I keep a file which contains only NAME and TELEPHONE NUMBER on the Organiser and this fits on a 32k RAMPK (at present) and can be updated

every week from the PC file.

To do this I produce an EXPORT FILE which contains only the NAME and TELEPHONE NUMBER fields. The whole updating process of producing this file and transferring it to the Organiser takes less than 5 minutes per week.

Transferring data with the Comms Link. Assuming you have adjusted the FIELD and RECORD separators as mentioned above and produced your export file all that remains is to COPY this file to the Directory which contains your Comms Link software.

Remember when you do make the transfer, first install the Comms Link properly connecting the PC and Organiser. Ensure that the PROTOCOL option on the Organiser SETUP menu is set to PSION.

All that needs to be done is to type CL<CR> on the PC and select RECEIVE and FILE from the Comms Link menu on the Organiser. After adjusting the filenames (if necessary) on the Organiser, pressing <EXE> will transfer the file in (usually) less than a minute. The Organiser BEEPs when the process is complete.

Although it is obviously quicker to receive the data into RAM (either A: or a RAMPK) it is also possible to receive the file directly onto a datapak - it just takes a bit longer and the PC will wait while the Organiser "pumps up the action".

Of course it is a good plan to plug in the Organiser's mains lead when writing anything to a datapak.

Editor's Note - If the Beginners Section is to continue, I require much more FEEDBACK from you Beginners out there. Come on, put pen to paper, and let me know what STILL puzzles you about your Organiser. I will not print your name if you so wish. No query should be though too trivial.

LIZZY Tip

When using FIND, either on a database, or whilst in the DIARY, if you press <SHIFT> and <EXE>, you move backwards through the file. Not a lot of people know that!

Pete Slipple

Product News

by the Editor

Psion Thesaurus-Spell Checker-Anagram Solver.

By the time you read this, the new T-SC-AS pak should be available. The pak was announced some time ago, but has taken some time to appear. A prototype was on show at the Which Computer Show in April, which attracted quite a lot of interest. We also had one at IPSOMEET (Sat. only) which was kindly brought by John Seymour.

This pak breaks new ground for Organiser products. The pak contains a special 265k (yes I do mean 256k!) surface mounted chip which stores an almost unbelievable amount of information. Let us hope that this is the start of a crop of "superchips" which could give the whole Organiser system a shot in the arm.

I had time to give the pak a good run-through at IPSOMEET (although there was no manual yet). The use of the pak is more or less self-explanatory. WORDS auto-boots to the end of the top-level menu. The default mode is spell checking and this allows various mis-spellings to be checked. I found it was slightly better at checking words entered phonetically than ones which had one or two letters misplaced (especially the first letter) which are likely to occur as "typos" when entering text at speed (as in a word-processor situation). As the spell checker is supposed to be fully accessible from OPL then this will be one of its obvious uses.

The Thesaurus was more impressive (and possibly easier to program!). It produced about 16 synonyms for OAF, including some rather naughty ones and a (most unlikely) FOOLNINCOMPOOP (sic).

The Anagram Solver has an unknown application (although it works well). I thought that Crossword buffs took solving anagrams as a challenging part of the action. It seems a bit of a cheat to have a Computer do it for you, even if it is a novelty on the Organiser.

Overall the pak is very impressive and I look forward to giving it a proper "seeing to" when my promised example finally arrives in the IPSO office. The other good news is the price - £49.95 (for a 256k superpak).

CODEX by Adrian Pegg

This is a new program which will take your Organiser datafile, seek out all the OLD London telephone numbers and automatically convert the 01 to either 081 or 071 as appropriate. Versions are available for LZ and XP (state which with order). Send £5 together with a data or RAMPak to:

Adrian Pegg
37 Osborne Road
London N13 5PT.

Jaro DATABOX

The device which I mentioned in the last issue is now available. It is called the Databox. Basically the device is in the form of a small box (with or without a ZIF (zero insertion force) socket. A short narrow ribbon cable connects the box to the Organiser (through either of the side sockets). The socket accepts standard 64k 12.5 volt UV-EPROMS and once inserted the box behaves exactly like a 64k datapak, to be initiated, written to, read from, and subsequently erased and re-used in the same way as a standard datapak.

The Databox with standard socket and one 64k EPROM costs £45.00

A ZIF Socket (recommended) costs an additional £6.00

Further 64k EPROMS are available from Jaro at £10.00 for a single, or £25 for 3 (supplied in a special storage/carrying case), or £40.00 for 5 (with a free large case which will take 8 EPROMS). All prices include p & p. Apply directly to:

Jaro Computer Services
9 Brighton Close
Addlestone, Weybridge
Surrey KT15 1PP

Sightmaster Navigation Program from Waypoint Software

Member, **MIKE HARRISON** has developed a new ocean navigation program for the Organiser. It is suitable for both commercial navigators and yachtsmen and two different display versions are available for the 2 and 4 line machines. Further details from:

Waypoint Software
25 Roosevelt Avenue
Lancaster
LA1 5EJ
Tel: (0524) 62205

Routewise - from Titan

This guidance system for the motorist, with over 250,000 routes throughout the British Isles is available to members at a special price of £35. If you don't like the program, you can format the pak and have a 64k datapak! If you ONLY want a blank 64k pak, these are available at £25. Contact:

Titan Computers Ltd
3 Cleveland Road
Bradford BD9 4BP
(tel: (0274) 544586

Machine Code Section - 1

Operator Precedence Copyright

(c) 1990

Russ Beinder

A subtle, but important, peculiarity came up while I was using my LZ to do some statistics calculations. After squaring a negative number I was surprised to find that I got a negative result. I checked this out in as many different ways as I could, but it seemed that the only way to get a correct answer was to enclose the negative number in parentheses.

I assumed that it was a bug. However, I checked the manual to be sure, and it is not a bug. The manual for the LZ model shows that the power (**) operator has a higher precedence than the unary minus, or NOT operator. The same page in the XP manual shows that the unary minus and NOT operator have a higher precedence than the power operator.

After some experimentation on several different machines, I determined that the LZ manual is correct and the XP manual must have a misprint. This still does not answer my original dilemma. It would appear that the Organiser operates differently from a standard calculator with regard to the unary minus. On a calculator you would press the unary minus button followed by the "x" squared button and always get a positive result. On the Organiser it is necessary to put the negative number in parentheses before squaring it.

This may seem trivial to some, but this problem occurs regularly when calculating variances, and other functions. It turns out, though, that it is not so odd that Psion would choose to implement it this way; many programming languages use the same operator precedence. It's just odd for a calculator.

Using ESCAPE OFF/ON

Copyright (c) 1990

Russ Beinder

As you may know, turning ESCAPE OFF can be a dangerous thing in your program. If you accidentally create an endless loop in your program you must remove your battery and reset the machine to continue. For this reason I do not like to turn it off, even in commercial applications. Instead I enable error trapping, and I look for ERROR No. 206. This works in most circumstances, but not all.

The following commands do not function well with escape on:

PAUSE

BAR\$

USR (when calling top level functions)

The problem is that, if the user presses the <ON/CLEAR> key while the program is waiting for keyed input, nothing appears to happen. Of course the Organiser thinks you are trying to break out of the program, and is waiting for you to press the "Q" key.

In the case of calling top-level functions from OPL, you cannot return directly to OPL - you must press <ON/CLEAR> twice. All these problems can be fixed by turning ESCAPE OFF. Now you can think about my original comment, "What about endless loops?".

The obvious answer is to turn ESCAPE ON after the problem command. The problem is that this does not work. It would appear that the internal <ON/CLEAR> break flag remains in effect even after the command where it occurred, or I simply cannot remove my finger fast enough from the key. So when the ESCAPE ON command is given OPL immediately generates ERROR 206.

Fortunately there is a fix for this little problem - execute the commands in the following sequence:

ESCAPE OFF	or	ESCAPE OFF
PAUSE -300		BAR\$(4,-1,1)
KEY		KEY
ESCAPE ON		ESCAPE ON

I do not have a solid fix for the top level function calling problem. However, variations on the above solutions appear to provide workable solutions for each of the functions.

For those of you who do not care if ESCAPE is OFF, turn it off once and everything else will work just fine.

Trap at ??

Did you ever wonder what that nasty TRAP message is really trying to tell you. In latter versions of the operating system. 3.6 and up I believe, the TRAP screen contains a couple of cryptic numbers. The screen in question looks something like this:
TRAP AT pppp
rrrr

Where the "pppp" is the memory location where the TRAP occurred, and the "rrrr" is the return address from the current subroutine. To illustrate this I have provided the following OPL program. Beware, these programs will intentionally crash your machine with a TRAP. Make sure you have backed up all your valuable data before trying them. Also, the TRAP vector at \$2046 must contain the address

Machine Code Section - 2

of the normal operating system TRAP routine. To ensure that this is the case, remove your battery and press the <ON/CLEAR> key to wipe your machine clean.

trapdemo:

```
LOCAL a%(3)
a%(1)=$8D00 :a%(2)=$0101
a%(3)=$0039
PRINT HEX$(ADDR(a%()))
GET
USR(ADDR(a%),0)
```

The following is the 6303 assembler code present in the OPL program:

```
0000 8D 00 bsr 0002
0002 01      nop
0003 01      nop
0004 00      ill
0005 39      rts
```

The code doesn't really do much, but it does demonstrate a method for working with the program counter register. Few microprocessors provide instructions for directly working with the program counter, so indirect methods must be used. The "bsr" instruction pushes the current PC+2 onto the stack, and continues executing instructions, not really what was meant, but it works. The result of all this is that the TRAP handler will now display information that we can otherwise pre-determine:

```
TRAP AT 0004
0002
```

Where the numbers displayed are relative to the address printed in the OPL program. So now when you are testing your machine language programs, and you get a TRAP, the resulting frozen machine will be slightly less annoying.

6303 Family Tree

by Russ Beimer (c) 1990

I always find it helpful in understanding how things work by knowing where they came from, or what motivated their design. To give some clarity to the Psion situation I have provided a short account of where the 6303 came from.

The 6800 comes from a very versatile chip family, where each new processor follows a mostly upwardly compatible instruction path. This compatibility extends to the 6809 and even the 68000 line (now there's a nice processor to program). For these reasons the 6800 legacy has endured to be implemented in a wide variety of computers and controllers, even some Ford and General Motors automobiles.

Some hybrid chips have been designed around the 6800, including the 6801, 6803, and others. These chips typically include some RAM and ROM instructions. They are meant to replace several support chips normally used with the 6800, and they are often used in embedded controller applications. Hitachi, in their infinite wisdom, decided that the Motorola design is so wonderful that they would produce a family of compatible chips, only better. This resulted in the 6303 that is in every Organiser. It includes several instructions not found in the 6803, fewer cycles per instruction in many cases, more control lines and interrupts, and a lower power surface mount package. It also contains 192 bytes of RAM from \$20 to \$FF.

This compatibility means that books written with the 6800 family in mind are, with the exception of the new instructions, compatible with the 6303. To help anyone out there who is struggling with machine code and/or assembly language I have provided a list of a few books on the 6800 at the end of this article. It would be best to get a book detailing either the 6801 or 6803, as these contain several instructions not in the 6800. The principal of machine code, however, remains the same. Although this is off this subject I thought it should be mentioned anyway. A method for avoiding TRAPs has been presented as:

```
POKEW $2046,$8000
```

This is only meant to function on XP or CM model Organisers - it will crash the LZ. Instead you may wish to try:

```
POKEW $2046,PEEKW($2056)
```

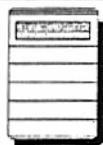
This will force a warm start instead of the TRAP message. I have not studied what actually happens using either of these methods, so although they appear to work, you should count your blessings, backup your data, and properly reset your machine (i.e. remove the battery).

Books for reference

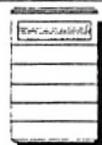
6800 Assembly Language Programming, by Lance A. Leventhal, Osborne 1978, paperback 480pp Code 0-931988-12-8

6800 Software Gourmet Guide & Cookbook, by Robert Findley, Hayden 1976, paperback, 211pp Code 0-8104-6281-8

Using the 6800 Microprocessor, by Elmer Poe, Howard W. Sams & Co. Inc. 1978 paperback 176pp Code 0-672-21512-8



IPSO FACTO



The Newsletter of the International Psion Organiser User Group

Volume IV . No. 4 . June 1990

Editorial Office: 130 Stapleford Lane . Toton . Beeston . Nottingham . NG9 6GB . UK

Phone: (0602) 735482 . Microlink Mailbox MAG11839

Editorial

The Organiser Is Still the Tops

After my customary periodic look at the competition (other pocket computers), I have come to the conclusion that the Organiser is still the clear leader in the field. The main contenders have all got serious drawbacks, as far as I am concerned.

The Microwriter AGENDA

This machine looked good at the outset, with all sorts of goodies promised, most of which never materialised. I even bought one in order to give it an extended trial. The built-in features (apart from the Text Editor/Word Processor) were quite fiddly, to say the least. The Calculator was abysmal. There was a promise of an add-on to be the equivalent of a Hewlett-Packard Scientific Calculator, but this never appeared. There was also much said about a powerful programming language (the lack of a built-in language is a serious omission). This was promised at various times over more than a year. When it did arrive, in the form of ABC (Agenda Basic Compiler) it turned out to be totally unusable unless the User owned, or had constant access to, a PC, as the language was written on a PC format floppy disk. There was also a lot of noise made about the Microwriter Keys. I found these to be quite an attractive novelty at first (and quite easy to learn), but some of the Microwriter key combinations for anything other than text were an absolute pain in the neck, requiring mental as well as digital gymnastics. Finally, I disposed of the machine, rather disillusioned.

The Atari PORTFOLIO

At first glance, this machine appears to be streets ahead of the rest, with its big (8 x 40 char) display and a full set of QWERTY keys. Another point in its favour is the built-in facilities: Text Editor, Address Book, Diary, and Spreadsheet. These facilities are, indeed, excellent. For instance the Text Editor allows word-wrap, search & replace, etc. and the amount of text visible on the screen at any time is great (compared with the Organiser or the Agenda). The Address Book (database) is quite clever in some ways - it automatically slots in new entries in alphabetical order. It has a silly file format, when transferring files between itself and a PC. The

"end of field" markers (<CR> and <LF>) are the same as the "end of record" markers, making for fun and games when transferring data from a normal PC database. The Spreadsheet is quite good, too. The Portfolio is wierd when it comes to COMMS. The standard connection is called a SMART PARALLEL INTERFACE. This really does connect to the PARALLEL socket on a PC and the accompanying programs work well. When it comes to SERIAL COMMS, however, the situation is rather different. The SERIAL INTERFACE comes without any software, which means that you are expected to find your own amongst the plethora of PC Comms programs. I tried the Psion CL programs, but they won't work. Just how you load any that do work is a mystery, unless you have BOTH COMMS INTERFACES, as the SERIAL one would need some software to load any suitable software, if you see what I mean! I think that the Portfolio is really spoilt, again by the lack of a built-in language. Separate languages in the form of software add-ons are all right for big static machines, such as the PC, but hardly the thing for a pocket computer. I have had a Portfolio for a few months now, and I use it mainly for writing articles, letters, etc. At the moment, I am also using it, in conjunction with a downloaded database, as an automatic phone dialler (which is a useful built-in facility). I will compare this with the Psion add-on Dialler in the next few weeks. If only the Portfolio had OPL built-in, it would be a serious contender..

Contents

Editorial . . . 26

Beginners Section . . 27

Progrs & Procs . . . 29

Product News . . 32

Feedback . . 32

Beginners Page - 1

by Mike O'Regan

This month, partly through lack of response to my plea for more feedback from you beginners out there, and partly from a genuine need, I am including a series of tips about keying in programs, ways of checking them, and, if necessary putting them right.

First of all, let me assure you if you are at all afraid of altering any programs, that the methods I will describe are not of the kind which will cause your Organiser to fatally "crash" (needing a hard reset).

Before we go any further, let me explain how I prepare programs which are printed in IPSO FACTO. Some of these programs arrive from their authors on data/RAMPaks, others on IBM compatible disks, and others either as printouts or even hand-written listings. I have both an XP and a LIZZY 64, so I usually test the programs on the appropriate machine.

The easiest ones to handle, of course are those which arrive on pak with full explanations in printed form. These can usually be tried out without any further ado (and for this I am eternally grateful to such authors). After running the programs to assure myself that they DO work, even if it is not possible to try out all possible running conditions, the listings are transferred via the Comms Link directly to my PC to await further use.

I keep all OPL listings in a directory called PSION on the PC. This directory also contains the Comms Link programs, for obvious reasons.

Programs which arrive on disk are almost as easy to handle. These are copied into the PSION directory (being careful that program NAMES do not clash with existing programs, as they would be overwritten). These programs are then sent down the Comms Link into the appropriate Organiser, TRANSLated and SAVEd, then RUN in the same way as previously described.

Programs which arrive as printed (or hand-written) listings are handled in two different ways, depending on the length of the program. Short routines are keyed in directly to the Organiser and then treated as above. Longer listings are typed into WordPerfect on my PC, saved as ASCII files in the PSION directory, and fed into the Organiser, via the Comms Link for testing.

At this stage any new programs are, as a matter of course, fed through Tony Spencer's excellent IPSO-CONV program (which also lives in the Psion dir-

ectory). This program magically converts listings into the format I prefer for printing in the newsletter. It does this by converting all OPL words into upper-case (capital letters), all VARIABLE names into lower-case, and, if necessary, indenting LOOPS (to make them more understandable).

Sorry if the above explanations were a bit long-winded. They were only meant to show that I take a fair amount of trouble to ensure (at last) that listings which appear in IPSO FACTO are correct. Here I must admit that, once or twice, programs have got slightly "garbled" in the process of laying them out on the pages of I.F., but I hope I have eliminated this now.

Here, at last, are the promised tips:

1. **Count the number of lines** - this might sound a bit basic, but I assure you that it is good advice, especially with programs which contain many similar lines. I have had numerous occasions when members have phoned me at their wits end, only to discover, when I have suggested a line count, that they have missed out a complete section of the program.

Looking for Errors - after keying in the program(s), attempting to TRANS them is sometimes impossible. In this case the Organiser is usually very helpful with messages which point to the error. It is worth pointing out at this stage that programs which are RUN from the top-level menu (and not from the PROG menu) will *not* point to the error, so, if you are DE-BUGGING a program, always RUN it from the PROG menu.

2. The most common mistake is to get one or more VARIABLE names wrong, either on the second line of the program (where your *declare* your variables, either LOCAL or GLOBAL), or in the body of the program where these VARIABLES are referred to. If you get a message such as "Missing GLOBAL in ...", then this indicates that you should check that your variables match.

3. The next most common error is "DO" without its matching "UNTIL", or, more common still, "IF", "ELSEIF", "ELSE" without a matching ENDIF. This is usually signalled by the Organiser as a SYNTAX ERROR.

4. **MISSING PROCs** - many programs consist of more than one procedure. If any of these is missing, it may not be immediately apparent until the parent program actually "calls" the missing one. If

Beginners Page - 2 (etc)

the parent program calls others from multi-choice menu then it could RUN without any hitch until you choose an option which has no matching procedure.

5. Programs which have ERROR TRAPPING - these are often the most difficult to de-bug, as the error-trapping which is included to help the program work often hides any mistakes in the listing keyed in. If the program doesn't seem to be working, without any indication why, then it is usually possible to remove the error-trapping lines to enable the program to point out the errors, and replace them when all is well.

6. Unfortunately, OPL (unlike many other languages) has no TRACE mode. This is because it is not constructed with *numbered lines* and so cannot indicate by a TRACE just where the program has reached. If you are running a program, especially a complex one, which is "hanging-up" at some indeterminate point (which may be in any one of a number of procedures) then a good tip is to use the Organiser's BEEP to help you trace the fault.

You can insert a line, such as BEEP 100,100, almost *anywhere* in *any* procedure and, if the Organiser produced this BEEP, you at least know that it has reached that point in the procedure. You can then move it (the BEEP line) further along in the procedure until you fail to get a BEEP when you RUN. This, of course, will indicate that the portion of the procedure which is causing the problem is BETWEEN the point where you previously inserted the BEEP line and its present position.

Finally, let me dispel the mix-up which sometimes exists with the two words PROCEDURE and PROGRAM. A Procedure can be said to be a Program where it is self-contained - in other words it works on its own, without the need to "call" another Procedure. It follows that a PROGRAM can be either ONE or A NUMBER OF RELATED Procedures. I hope this clears up this point, which is one which is the source of confusion for many Beginners.

For Sale

- 1-16k Datapak - £10
- 1 - 32k Datapak - £20
- 1 - 64k Datapak - £25
- 1 - Psion FINANCE PAK - £15
- 1 - Harvester GAMES PAK - £15
- 1 - Comms Link (2line) -£35

Phone Seamus Docherty on (0628) 31224 (evenings)

Getting Rid of the Border

As you know, when a program written for an XP/CM is run on the LIZZY, a "fancy" border appears around the old 2 Line x 16 Char display area. If you wish to get rid of this border then key in, and run, the following little program:

```
border:
POKEW $2099,32
```

This little routine just puts the ASCII character 32 (a SPACE) into the memory slot which holds the border pattern. If you wish, you may use ANY character from 32 to 255. A curious feature of this program is that, if it is slightly extended then the full four-line display is used in the program named. For instance, if you make it:

```
border:
POKEW $2099,32
MATHS:
```

having inserted an old MATHS PAK, the opening menu not only takes up all four lines, but it is no longer in ALL CAPS, but in the LIZZY Main Menu format!

Perhaps one of our machine code boffins can tell us why!

IPSO London Group

Owing to pressure of work, Jonathan Hurwitt is unable to carry on running the London Group. However, another member has taken over the Group, so that it can keep up the good work. Any offers of help from other members is welcomed.

The next meeting of the Group is on Tuesday 19 June with a talk on the "Trials & Tribulations of Using the Psion", given by the new Chairman.

Any enquiries about this meeting, or about the Group in general should now be addressed to:

David Gray
39 Twyford Avenue
Acton
London
Tel: 081 993 5702

Progs & Procs - 1

Workday

by Geoffrey F. Miller

LZ

This program "gets inside" the Lizzy ROM. It was inspired by what is at first sight a little throw away paragraph in Appendix C (page C-1) of the LZ Programming Manual, giving the address and contents of the byte which holds the definition of the work-days in the alarm clock.

I get up later on Wednesdays than on other weekdays, as well as sometimes having to go to work at week ends. This program suite allows the user to set all, any, or none of the days of the week to activate the "wrkday" option from the ALARM menu.

WORKDAY: contains the menu, allowing instantaneous resetting of the system defaults. After any alteration of the settings, the current settings are reviewed and displayed by WORKDAYV:

WORKDAYS: allows the user's choice of work days to be selected. As a precautionary measure, if you select NO workdays, the DEFAULT will be reset, to avoid unpredictable (possibly nasty) results.

```
WORKDAY:
LOCAL m%
GLOBAL d$(20),w$(21),n%,x%,r%,e%
ONERR error:
d$="Mo Tu We Th Fr Sa Su"
menu::
CLS
UDG 0,4,14,14,31,31,4,0,31
UDG 2,0,0,0,0,0,0,31
PRINT CHR$(0);REPT$(CHR$(2),14) :CLOCK(1)
m%=menun(2,"VIEW,-SET,DEFAULT")
CLS
PRINT " ";CHR$(184);" G F Miller"
PRINT
PRINT "Please wait..."
IF m%=0 :RETURN
ELSEIF m%=1 :workdayv: :goto menu::
ELSEIF m%=2 :workdays: :workdayv: :goto menu::
ELSEIF m%=3 :POKEB($20A7),(31)
workdayv: :goto menu::
ENDIF
goto menu::
```

```
WORKDAYS:
LOCAL a%,p%,d%,g$(1)
ONERR error:
CLS
```

```
PRINT CHR$(24);"Set Workdays Y/N"
AT 1,3 :PRINT d$;
PRINT " ";
a%=0 :r%=0 :p%=0
DO
s::
PRINT CHR$(63);
g$=UPPER$(GET$)
PRINT CHR$(8);
IF g$="N" : PRINT "- ";
ELSEIF g$="Y"
PRINT CHR$(255)+" ";
r%=2**a%
p%=p%+r%
r%=0
ELSE GOTO s::
ENDIF
a%=a%+1
UNTIL a%=7
IF p%=0
p%=31
ENDIF
CLS
PRINT " ";CHR$(184);" G F Miller"
PRINT
PRINT "Please wait..."
POKEB($20A7),(p%)
RETURN
```

```
WORKDAYV:
LOCAL d%,p%,a%,f%
ONERR error:
w$=""
p%=PEEKB($20A7)
a%=6
DO
d%=0
r%=2**a%
d%=d%+r%
f%=(p%>=d%)
IF f%=-1
p%=p%-d%
w$=" "+CHR$(255)+" "+w$
ELSE w$=" "-w$
ENDIF
a%=a%-1
UNTIL a%=-1
w$=LEFT$(w$,20)
PRINT CHR$(24);"Workday Alarms set:-";
AT 1,3 :PRINT d$;w$
GET
RETURN
```

Progs & Procs - 2

```

ERROR:
CLS
PRINT "Error",e%
IF e%=230
PRINT "Printer not on"
ELSE PRINT ERR$(e%)
ENDIF
GET

```

WORK

by Geoffrey F. Miller

LZ

Work (with its sub-routines ANALYSE:, TIDY:, and ERROR: - listed above) is a time analysis routine. First write a datafile names WORK.ODB (usingXFILES). This represents the various tasks which you might perform and can have as many activities as you like, provided that they are one-line entries.

I append a copy of the one I use (I am a surgeon).

The program WORK: will locate the device A:/B:/C: on which this task file resides and then writes the record file to the same device. Thus the program suite can reside in B: and the data can be recorded on a RAMPAK in C:, if so desired.

The rest of the program is menu-driven, using a 24 hour clock. The time spent on any activity is recorded to the nearest whole minute and will work through midnight, assuming that you change the logged activity before the following midnight! The ANALYSE option gives a print-out of your timesheet for the period.

The vital point about this timer program is that, once toggled, you can get on and use the rest of the Organiser as normal, so you are not deprived of access to datafiles, etc.

```

WORK:
[LOCAL t$(21),d%,m%,s%,h%,hh%,mm%,p%,v%,h$(2),
m$(2)]
GLOBAL d$(1),e%,f$(10)
ONERR error:
t$=LEFT$(DATIM$,21)
d%=0
L1::
d$=CHR$(d%+%A)
IF EXIST (d$+"WORK") :goto L3::
ELSE d%=d%+1
IF d%>3 :goto L2:: :ELSE goto L1::
ENDIF
ENDIF
L2::

```

```

[PRINT CHR$(12+"DEVICE ERROR"+CHR$(14)
+"NO WORK FILE"+CHR$(16))
GET :STOP
L3::
UDG 0,0,14,28,10,1,0,0,31
UDG 2,0,0,0,0,0,0,0,31
CLS
PRINT CHR$(0);REPT$(CHR$(2),14) :CLOCK(1)
[s%=MENUN(2,"START,FINISH,ANALYSE,
TIDY")]
IF s%=0 :RETURN
ELSEIF s%<4 :goto L4::
ELSEIF s%=4 :goto D::
ELSEIF s%>4 :goto L3::
ENDIF
L4::
IF EXIST(d$+"WORKTIME")
OPEN (d$+"WORKTIME"),B,b%,b2$,b3$
LAST
IF B.b3$<>" " :goto L5:: :ENDIF
IF B.b2$="" :goto L5:: :ENDIF
h%=VAL(MID$(B.b2$,17,2))
:m%=VAL(RIGHT$(B.b2$,2))
t$=LEFT$(DATIM$,21)
hh%=VAL(MID$(t$,17,2))
:mm%=VAL(RIGHT$(t$,2))
IF mm%<m% :mm%=mm%+60
hh%=hh%-1 :ENDIF
IF hh%<h% :hh%=hh%+24 :ENDIF
hh%=hh%-h% :h$=FIX$(hh%,0,-2)
IF LEFT$(h$,1)=" "
h$="0"+RIGHT$(h$,1)
ENDIF
mm%=mm%-m% :m$=FIX$(mm%,0,-2)
IF LEFT$(m$,1)=" "
m$="0"+RIGHT$(m$,1) :ENDIF
B.b3$=h$+" "+m$
UPDATE
ELSE
CREATE (d$+"WORKTIME"),B,b%,b2$,b3$
ENDIF
L5::
IF s%=2 :STOP :ELSEIF s%=3 :goto ana::
ELSE OPEN (d$+"WORK"),A,a$
PRINT CHR$(12);CHR$(24)
AT 1,1 :PRINT "Select Activity..."
ENDIF
DO
v%=view(3,A.a$)
IF v%=3 :IF pos>1 :back :ELSE last :ENDIF
ELSEIF v%=4 :next :IF eof :first :ENDIF :ENDIF
p%=pos
UNTIL v%=13
USE B

```

Progs & Procs - 3

```
B.b%=p%
B.b2$=t$
B.b3$=""
APPEND
CLOSE
error:
D::
f$=d$+":WORKTIME"
tidy: :STOP
ana:: :CLOSE :ANALYSE:
```

```
ANALYSE:
LOCAL c%,g%,g$(1),h%,m%,p%
ONERR error:
PRINT CHR$(12);" This uses"
PRINT " CALC MEMORY"
PRINT CHR$(25);"Proceed? Y/N"+CHR$(16)
g$=UPPER$(get$)
IF g$="Y" :goto L1:: :ELSE STOP :ENDIF
L1::
CLS
AT 1,2
PRINT "ANALYSIS IN PROGRESS"
OPEN (d$+":WORK"),A,a$
LAST :p%=pos :first
OPEN (d$+":WORKTIME"),B,b%,b2$,b3$
g%=0
M9=0 :M8=0
DO
M7=0 :M6=0
USE A
c%=pos
LPRINT A.a$
USE B :first
DO
IF B.b%=c%
LPRINT B.b2$;" ";B.b3$
h%=VAL(LEFT$(B.b3$,2))
M6=M6+h%
m%=VAL(RIGHT$(B.b3$,2))
M7=M7+m%
ENDIF
NEXT
UNTIL EOF
m%=M7 :h%=M6
GOTO sum::
total::
LPRINT "TOTAL ";
IF h%<100 :LPRINT " "; :ENDIF
IF h%<10 :LPRINT " "; :ENDIF
LPRINT h%;";";
IF m%<10 :LPRINT "0"; :ENDIF
LPRINT m% :LPRINT
```

```
M9=M9+m%
M8=M8+h%
USE A
NEXT
UNTIL c%=p%
m%=M9 :h%=M8 :g%=1
sum::
h%=h%+INT(m%/60)
m%=m%-INT(m%/60)*60
IF g%=0 :GOTO total:: :ENDIF
LPRINT "GRAND TOTAL ";
IF h%<100 :LPRINT " "; :ENDIF
IF h%<10 :LPRINT " "; :ENDIF
LPRINT h%;";";
IF m%<10 :LPRINT "0"; :ENDIF
LPRINT m%
LPRINT CHR$(12)
```

```
TIDY:
LOCAL g$(1)
PRINT CHR$(12);" DELETE FILE"
PRINT "Delete ";f$
IF NOT EXIST(f$)
AT 1,4 :PRINT " FILE NOT FOUND";CHR$(16)
GET
RETURN
ENDIF
PRINT CHR$(25);"Confirm? Y/N"
g$=UPPER$(GET$)
IF g$<>"Y" :RETURN
ELSE
AT 1,1 :PRINT " DELETE FILE"
PRINT "Delete ";f$
PRINT REPT$(" ",20);
PRINT "Delete... ";f$
DELETE f$
PAUSE 20
ENDIF :RETURN
```

Sample WORK.ODB

```
WARD ROUND
OUT-PATIENTS
FRACTURE CLINIC
OPERATING
A+E DEPARTMENT
ADMINISTRATION
TEACHING
WAITING TIME
TRAVELLING
ON CALL
```

Product News & Feedback

from the Editor

A Problem with the Psion Printer

One or two members have had problems with the Psion Printer "dropping out" in the middle of a printout. This has been traced to a low state of charge in the built-in Nicad accumulator. The fault is NOT rectified by plugging the unit into the Mains. The way to avoid this annoying occurrence is to keep your printer fully charged at all times - you cannot overcharge it.

Another New Book on the Organiser

Hot on the heels of the Machine Code book, KUMA have not announced "The Psion Organiser Deciphered", by Gill Gerhardi, IPSO member Vic Gerhardi and Andy Berry.

The book is described as a beginners guide for all Psion Organiser Owners, giving an in-depth introduction to the LZ, LZ64, CM and XP models. The features and functions of these versatile hand held computers are introduced and explained in a non-technical way. The differences between the various models are also explored.

Written in a sensitive, highly readable, form the text is liberally illustrated with handy hints, tips and practical advice together with a wealth of examples. The result is a quality informative book, ideal for all new Organiser owners.

The authors have extensive experience in running computer training courses and wrote the widely acclaimed book on the Cambridge Z88 portable computer "Z88 Magic", published by Kuma in November 1989

"The Psion Organiser Deciphered" will be available from June 14 at the retail price of £12.95. The ISBN number is 07457 0139 6

RAM.PAGE Paks 1

28k, 256k, 512k, and 1Megabyte RAM.PAGE paks are available from the sole UK distributor, Widget Software. Full details, prices, availability etc on demand from Widget on 0438 815444

Further News of the Jaro Databox, etc

Robin Crorie, of Jaro, tells me that he has widened his DATABOX offer as follows. He is prepared to transfer ANY programs onto a DATABOX EPROM, provided that he supplies the EPROMS, and, in the case of Commercial software, he will also need a signed accompanying letter stating that the customer is the original purchaser of the software concerned.

Can I also remind you that the DATABOX is extremely easy to use, needing no outside power supply. It is just a case of "plug-in and go", just as if using a RAM or Datapak.

Robin says that he will supply Databoxes to members in Europe for the same price as that to UK customers.

Pocket Spreadsheet Tip

from Dick Morgan

Loading the Spreadsheet into my LZ64 and choosing PLAN on the main menu caused an "ERROR - OUT OF MEMORY" message to be displayed.

I tried clearing all NOTEBOOK files, etc out of A: until there was nothing left, but the problem still persisted. The solution, given me by Psion Technical Support, was to reset the Organiser and the point to remember - one not stressed in the Spreadsheet manual - is NEVER TO LEAVE THE SPREADSHEET WITHOUT USING THE QUIT PATH. I

It appears that leaving the Spreadsheet without using the QUIT path leaves an invisible spreadsheet still resident in the DEVICES section of RAM, blocking the space required by the same datapak on reinsertion, and causing the OUT OF MEMORY message. (I assume I must have left the Spreadsheet open, the Organiser switched itself OFF automatically, I swapped the Spreadsheet pak for AutoScribe while the machine was OFF and later, when reinserting the Spreadsheet, activated the ERROR message).

(continued on next page)

Product News & Feedback - 2

When using RESET in the UTILS menu, I noticed that the Organiser clock stayed at the current time; had I reset by removing the battery, of course, the clock would have reverted to zero. I wonder whether there is an OPL routine that would clear the DEVICES memory without having to reset?

A Pat on the Back

from Michael Bruce Walsh

I finally bought my LZ64 2 weeks ago from ROVOREED, as you suggested. I received it on the Thursday morning after sending off my cheque on the Tuesday night.

I only had it one day when the display started fluctuating slightly, so I sent a letter to Rovoreed explaining the fault. After a few days, I received a replacement, plus a few stamps towards returning the faulty LZ64, which I did on Saturday.

Well done Rovoreed.

A few questions I would like to ask:

Q1. Does this fault happen often?

A1. (I have never heard of it before. Ed)

Q2. Is the RS232 cable to connect the LZ to other computers a special one, or is it just a standard cable obtainable at electronic shops, such as Tandy and, when you mention files can be transferred from Organiser to PC would that include pocket computers such as the ATARI Portfolio.

A2. The Psion COMMS LINK/RS 232 is special in that the communications software is built-in to the plug and further software for the PC is supplied on 5.25" (or 3.5") disk. Compared with Comms devices supplied for other pocket computers, the Comms Link is exceptionally good value. For instance, the RS 232 device for the Atari Portfolio is supplied with no software at all - they expect you to provide your own. If you have this, then the Portfolio could talk to the Organiser by connecting their respective Comms Link cables. Ed.

Amendments to London Telephone Codes

Adrian Pegg has sent me a few amendments to the LONCHK procedure published last month. Before I printed this, I checked with a supposedly "impeccable" source - the Telecom Panel in Exchange & Mart, but

apparently some of these were wrong. Here are the additions to LONCHK to enable LONDON to produce the right 071 and 081 codes.

The lines containing the underlined figure should be amended to:

```
ELSEIF ex%>=529 AND ex%<=536
ELSEIF ex%>=874 AND ex%<=894
ELSEIF ex%>=940 AND ex%<=944
ELSEIF ex%>=979 AND ex%<=981
```

The following lines should be INSERTED in the procedure:

```
ELSEIF ex%>=390 AND ex%<=399
ELSEIF ex%=526
ELSEIF ex%=527
ELSEIF ex%>=897 AND ex%<=900
ELSEIF ex%>=946 AND ex%<=954
ELSEIF ex%>=983 AND ex%<=986
```

Finally, delete the line:

```
ELSEIF ex%=975
```

For Sale

Tandy CGP-115 Colour Plotter/Printer - £35

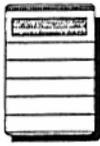
Collector's Item Psion Organiser 1
in excellent condition w/Finance Pak & Manuals
Offers invited

Phone David Cantor on 0932 63187

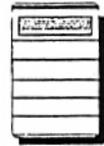
Epson HX20 Laptop Computer
Built-in Printer, Micro-cassette,
BASIC, Monitor
Spare Printer Ribbons, Cassettes
c/w all manuals, hard carrying case
All in perfect condition - £150 ono
and

Epson P40 Thermal Printer
c/w handbook, adaptor to Comms Link
£50

Phone Mike O'Regan on (0602) 735482)



IPSO FACTO



The Newsletter of the International Psion Organiser User Group

Volume IV . No. 5 . July 1990

Editorial Office: 130 Stapleford Lane . Toton . Beeston . Nottingham . NG9 6GB . UK

Editorial

Posting items to IPSO

A few days ago one of our members had a nasty shock. He lost a 128k datapak in the post. It had to come. Although we have constantly reminded you all that datapaks you send us should be sent by RECORDED DELIVERY, we continue to get paks of all sizes in all sorts of packets by normal post. The extra cost of posting items by recorded delivery is only 40 pence and a small Jiffy-bag a few pence extra - a fraction of the value of even an 8k pak. The pak which went missing had been sent in an ordinary envelope, with no packing whatsoever! The envelope arrived with a hole in the side where the pak had escaped. Subsequent enquiries with the Postal Authorities were fruitless - no-one could say just where in the postal chain the pak had gone missing. I hope that this little story (which is true) will help avoid disasters like this in the future. The moral is use a Jiffy-bag and send your precious pak by recorded delivery.

256k Superpaks Unassailable

I have had a number of calls and letters asking if the new Thesaurus/ Spell Checker can be erased and then used as a 256k datapak, although I did mention that the new pak contained a custom 256k surface mounted ROM. The ROM is programmed once and is then totally committed - that means that there is no way of altering the data contained in it. Normal datapaks contain EPROMS (Erasable Programmable Read Only Memory) which accounts for their ability to be erased by exposure to UV light and subsequently re-used. I agree that it would be marvellous if the Thesaurus chip was an EPROM, but this is, unfortunately not the case. I hope that this will put the record straight once and for all.

Machine Code Articles

Machine Code Buffs will have noticed that there is a slight hiccup in the promised articles. I hope to get this sorted out and resume the articles as soon as possible..

An Apology



I think an apology is due for the unforgiveable errors which I introduced into Geoffrey F. Millers excellent Work: and Workday: programs last month. I thought I would be clever and save space by replacing all Geoff's ONERR calls with a reference to a separate procedure called ERROR:. This caused problems with the programs, which had each contained their own error-trapping routines. I then completely hashed up things by not including an ERROR: procedure. I'm afraid it was a case of me altering programs after having tested them in their original form. I hope I have learned my lesson for the future!

A quick way to get these programs up and running is to remove all lines which contain the word "error". If anyone would like a copy of the original listings, please ask me. My apologies all round.

Contents

Editorial	34
Beginners Page . . .	35
Program Page. . . .	36
IPSO Review.	37
Feedback.	38
Product News. . . .	40

Beginners Page

by Mike O'Regan

PEEK, POKE, and the Organiser Memory.

This is not meant to be a complete treatise on Machine Code - we have a separate section for this. However, in response to numerous requests (from Beginners) I will just give you a few pointers about the use of the OPL functions PEEKB, PEEKW, POKEB, and POKEW.

First of all, it will be helpful if you can think of these in terms of VERBS and NOUNS. POKES are Verbs - PEEKS are Nouns. Verbs, (as we all no doubt remember), are DOING words. Verbs (in the form of POKES) are DANGEROUS! On the other hand Nouns (in the form of PEEKs) are HARMLESS. In other words, "POKEing about" in the Organiser's memory (if you are unsure of what you are doing) is not only dangerous, it can be FATAL. That is why we keep issuing dire warnings which accompany any OPL procedures which use POKE. However, you can think of PEEK as a sort of VARIABLE (in other words a box labled with an ADDRESS and whose contents may change).

PEEK is used to examine (but not change) the contents of any particular location. The B and W after either PEEK or POKE stand for BYTE and WORD respectively, and a WORD is simply 2 BYTES. A single memory location can only contain one BYTE, so, if we want to work with two consecutive locations then we use the PEEKW or POKEW. Of course PEEKB or POKEB work with a single memory location.

Usually PEEKs and POKES are written into an OPL procedure. However, it is possible to use both of the PEEKs directly from the CALC option. Let us try this out now - remember that PEEK is safe, so have no fears.

Switch on your Organiser and select CALC from the top-level menu. Key in:

```
PEEKB($20CD) <EXE>.
```

The display should show the number 1 (i.e. the present contents of memory location \$20CD). Press <EXE> once again and edit to read:

```
PEEKW($20CD) .
```

You should see the figure 300, which is the SUM of the two figures presently in memory location \$20CD and the following one (in other words the TWO BYTES or WORD starting at location \$20CD).

This just happens to be the number of seconds to automatic switch-off. Using our (safe) PEEKs we have had a look at the contents of first one, then two memory locations.

Now let us have a look at the dangerous world of the POKE. To illustrate this we will use the same locations. This time we must write a short OPL program, as POKES are not allowed from the CALC function.

Key in the following procedure (from the PROG Menu):

```
swoff: POKEW $20CD,25
```

TRAN and SAVE the procedure, but DO NOT RUN JUST YET!

Key in another short procedure:

```
norm: POKEW 20CD,300
```

and TRAN and SAVE it.

Before you RUN swoff:, a word of explanation (if you haven't yet realised what it does). It replaces the standard 300 second switch-off time with 25 seconds. Now you can see why norm: had to be written first. Norm: restores the full 300 seconds. With only 25 seconds, there would be no time to write and run the norm: program.

Now see if you can EDIT swoff: so that you can enter a switch-off time when you run the program. If you are not sure, I will list the whole program in next month's issue. By the way, the shortest possible switch-off time is 15 seconds.

Notes on Entering Location Numbers when using PEEK and POKE

The numbers which are generally used to indicate memory locations with PEEK and POKE are written in HEXADECIMAL. This is simply a numbering system which uses the numbers 0 to 15. As our normal decimal system uses only 0 to 9, we need a way of writing the extra numbers (10 to 15). The HEXADECIMAL system uses the letters A to F to represent these numbers, so that you may expect to see any combination of 0(zero) 1 2 3 4 5 6 7 8 9 A B C D E or F. The Organiser knows you are using this numbering system when the number is prefixed with a \$. The A-F may be UPPER or Lower case

Program Page

This program is not just another PASSWORD program. It is the first to reside in memory, inconspicuously, and works automatically without any third party even knowing that a password exists.

SETUP by M.J. Siddiq.

setup:

```
LOCAL n%(3),pw$(3),l%
POKEW $2065,$7f00
DO
  CLS
  PRINT"Password";CHR$(63),": ";
  INPUT pw$
UNTIL LEN(pw$)=3
DO
  l%=l%+1
  n%(l%)=ASC(MID$(pw$,l%,1))
UNTIL l%=3
POKEW $2058,$7f00
POKEW $7f00,$3f0e
POKEW $7f02,$4f5f
POKEB $7f04,$cc
POKEW $7f05,$803a
POKEB $7f07,$fd
POKEW $7f08,$2058
POKEW $7f0a,$3f0c
POKEW $7f0c,$3f48
POKEB $7f0e,$c1
POKEB $7f0f,n%(1)
POKEW $7f10,$26f8
POKEW $7f12,$3f48
POKEB $7f14,$c1
POKEB $7f15,n%(2)
POKEW $7f16,$26f2
POKEW $7f18,$3f48
POKEB $7f1a,$c1
POKEB $7f1b,n%(3)
POKEW $7f1c,$26ec
POKEW $7f1e,$4f5f
POKEB $7f20,$cc
POKEW $7f21,$7f00
POKEB $7f23,$fd
POKEW $7f24,$2058
POKEW $7f26,$3f0e
POKEB $7f28,$39
CLS
PRINT"Customised OFF"
PRINT"Now installed"
GET
```

CM and XP
ONLY

Notes

lowers top of language stack
to \$7F00

re-routes SWOF to \$7F00

BEEP

clears register D

) loads D with \$803A

)

) store contents of D at

) \$2058

switch off

get key

) compare with 1st letter

) of password

if not equal GOTO \$7F0A

) get second key press and compare

) with second letter of password

)

if not equal GOTO \$7F0A

) get third key press and compare

) with third letter of password

)

if not equal GOTO \$7F0A

clear Register D

) loads D with \$7F00

)

) stores contents of D at \$2058

)

BEEP

RETURN from sub-routine

*(Editor's Note: BE SURE TO REMEMBER YOUR
PASSWORD otherwise YOU will be locked out of
your machine)*

IPSO Review

by Mike O'Regan

The Organiser De-ciphered

by Gill Gerhardi, Vic Gerhardi & Andy Berry

I have mentioned before that there was a new Organiser book in the process of being released. The book (with the above title) is now available.

This book will be welcomed by all those (90% plus) Organiser users who want to make the most of the Organisers (any Model II from the CM to the "FAT LIZZY" (LZ 64), but don't wish to experience the joys of OPL programming. So this book fills a gap in Organiser literature, in that there is only ONE small program in the whole book! - and more of that later.

The book is written in the style of a "spy code book" which you either like or hate. If the latter is the case, try to overcome your initial reaction, as the book (slim volume that it is - some 190 pages) contains much that will be of enormous help to the tyro.

I like the way that the Organiser is likened to a spy headquarters, with corridors and lifts connecting the different levels and departments. This is a novel approach, which I tried out on one or two new Organiser users to see if they could find their way through the various sections.

It is always difficult to describe functions which are similar on each model of the Organiser without too much repetition, but without omitting the essential differences. Overall, I would say that the (to us old hands) obvious drawbacks of the CM model were rather glossed over throughout the book - indeed the CM gets much more mention than the XP/LA, and sometimes confused with it.

Now to one or two gripes. The book strongly advises against using a re-chargeable ni-cad accumulator instead of batteries. If sensible use is made of ni-cads, there is no reason why they should not be used, with a consequent substantial saving in the running costs of your Organiser. I have used ni-cads personally for years with no trouble whatsoever. I keep a spare in the case with my Organisers, just in case.

There is a mention of BROWSING through a database (on Page 54). This could have mentioned that it is possible to browse BACKWARDS through a database, using SHIFT/EXE.

At various points in the text there are references to an Appendix. Unfortunately, in the first print run, the printers forgot to include what would be a most useful Appendix (especially as IPSO has a mention - for which my thanks to the Authors and Publishers).

Page 61 gives the impression that MAIN is the only data file available on the CM/XP models. Of course, using a suitable program suite (such as XBASE or FILEMASTER) or even writing your own in OPL, you may work with as many files as you wish.

Finally, as mentioned above, there is only one OPL example in the whole book - a short VAT routine. This is a poor example as it contains a variable declaration - VAT - which is superfluous. The program could also use some FIX\$ to limit the number of pence to 2. Worst of all, the program is in a CM/XP section of the book, despite that fact that it will ONLY RUN ON AN LZ!

Taken all-in-all this book will be a boon to the beginner facing his new Organiser for the first time, especially if he wants to make full use of the extensive built-in facilities which grace even the smallest of the Organiser IIs.

The Psion Organiser Deciphered is available from all good bookshops, Psion dealers, or the publishers direct:

Kuma Computers Ltd
12 Horseshoe Park
Pangbourne
Berks RG8 7JW

Tel:0734 844335
Price £12.95

(Editor's Note: I am obliged to Kuma for making a review copy available so soon after publication))

For Sale

Citizen 120D Printer in VGC, little used
£100.00
Contact:

Denis O'Regan on 0923 857764 (evenings)

Feedback - I

Readers Letters

Q. The clock in my Psion LZ64 persistently loses. Can it be regulated via the software and, if so, how. Or is it a matter of regulating the hardware?

B.A. Watkin

A. *As far as I know, the real-time clock in the Organisers is not capable of being regulated through either software or hardware. Does any member know any better?*

Q1. Can anyone tell me why, when using SORT from XFILES, there are often some items that re-appear at the start of the sorted database as well as in the correct sequence.

Q2. Does anyone know the PEEKs for ascertaining the size of a datapak from within a procedure?

Q3. Is there a (machine code?) Procedure that can remember and re-load my customised Main-Menu after a crash/reset?

Pete Sipple

A3. *You would have to save the configuration of your menu to a file on a RAM/DATA PAK (otherwise your CRASH/RESET would get rid of everything from RAM). KPROG (from Kirsta) has a facility for saving the whole of the current configuration (including MAIN MENU, UDGs, SOUND, AUTO-OFF, KEY CLICK and CURSOR SCROLLING).*

Q. I have recently treated myself to a LZ64 (thanks to ROVOREED) and find many of its new facilities excellent, if apparently copied from the superb CUBSOFT XBASE package. I do have my gripes, however. In the DIARY, the advantages of the four line display are ignored by having those "<Free>" messages. Do you know of a way to suppress these so that only useful information is displayed? Also, can the initial weekly summary be avoided?

Ian McPherson.

Getting Rid of the Border

On receipt of the latest issue of IPSO FACTO, I typed in and ran the procedure "border:" only to find that it does not work. The code being a single byte, the command should have been POKEB instead of POKEW. (Oops, Ed) In any case, it is not necessary to use dangerous machine code routines to disable the border. The border is only

operative when a procedure is called from the top-level menu. To get rid of it you only have to call the procedure from another procedure. Such a procedure can be very simple, as, for example, to call the TRAVEL PAK:

```
trav:  
Travel:  
RETURN
```

By the way, when are we going to get an updated TRAVEL PAK, with a more useful vocabulary than the measly 700 words included in the current one. Pocket Translators with more than 40000 words are now available on the market.

Percy Johansson, Sweden

Q. I have just got WINDOWS v3.0 for my PC and thought it would be an ideal environment for dumping LZ files while getting on with other things on the PC - e.g. using SENDALL Unfortunately, I cannot get CL to run successfully for more than one file transfer before WINDOWS shots everything down with "Application has violated system integrity". Does anyone know if it is possible to run CL software under WINDOWS v3.0 (386 enhanced mode). If so, with what settings.

Answers to:

Richard Masters
24 Homeleigh Road
Nunhead
London SE15 3EE
Tel: 071 732 4889

If it is technically possible, I would like to make the suggestion that you turn off proportional printing for listings in IPSO FACTO. I usually find it useful to match blank spaces in a string with characters on an adjacent line in order to recalculate how many to enter, but you simply can't do that with proportional spacing as it takes away any relationship between characters.

(This is a good point. I will try to use the COURIER (mono-spaced) typeface for all future listings. Ed)

(cont. on next page)

Feedback - II

With regard to RUSS BEINDER's (machine code) article, I have a couple of questions. First what is BAR\$ It appears to be OPL from his article, but I have no idea what it does and it is not in the manual. Secondly, he mentions calling top-level functions from OPL. Is this really possible, and if so how is it done?

(Over to you, RUSS. Ed.)

My final question is this. I would very much like some of my larger programs to AUTO-BOOT into the top-level menu, but not being machine code minded, I have no idea how to achieve it. Can anyone enlighten me please.

Adrian Pegg

(I would be pleased to print such a procedure. Ed)

Paralink Bugs

Sometimes my program would work and sometimes it would send apparent garbage to the printer. To cut a very long story short, I found that the fault was not mine and could be reproduced by a simple OPL program:

```
tst:
start::
LPRINT "ABC"
GET
GOTO start::
```

This works fine, until you let the Organiser go to sleep in the GET. When it wakes up it will print "EFG" and not "ABC" next time. WHY?? because bit 2 is being held "on".

I called **Widget** (from whom I bought the Paralink) and they (again with concern) put me on to the Author, **Chris Fearnley at Silvertree**. He rapidly fixed the problem and was most appreciative and helpful. It seems that the problem is really a fault of the OPERATING SYSTEM of the Organiser. Anyway, in case it affects you, the fix is:

```
POKEB $10,4 : REM sent the value of 4
to port 16
```

Chris Fearnley also gave me a performance improvement for his software. I understand that **Widget** is upgrading all future Paralinks with both these fixes.

Whilst it is infuriating to waste time finding bugs in other people's products, one is disarmed by the friendly and helpful manner of most people involved in Organiser products. The response to bug reports is also gratifyingly quick (though Psion Tech Support have hey to fix

the bug I found in the Developer. They said: "I'm afraid that the programmer who wrote that bit has left...")

Bob Gilchrist

Bulletin Board News

I run a BBS linked to the International FIDO-PSION-ECHO, and a file-area to download some Organiser PD software. Information is as follows: SKANDALON BBS, D-6200 WIESBADEN / WESTERN GERMANY Tel: 0049-6121, 300-2400 Baud, 8,N,1 24hr online.

Further information from:

Andreas Mann
Roemergasse 16a
6200 WIESBADEN
W. Germany Tel: 06121/422673

Psion to the Rescue

A somewhat offbeat Psion application came to light this Spring when the twin screw motor-cruiser "TANTALUS" was crossing the North Sea in order to appear at the East Coast Boat Show in England. When TANTALUS left Flushing for the Ipswich based boat show, it was found that, due to an electrical fault, her main steering compass was not functioning. Undeterred, the skipper set out using a small magnetic hand compass. The cruiser made good progress for the first 60 miles of the 100 mile journey, when both engines failed leaving the crew helpless halfway across the North Sea. They had a small VHF radio which they used to broadcast for assistance, explaining that they were drifting some 4 miles to the north east of the Galloper Bank. The P&O European Ferries "CERDIC FERRY" was crossing from Holland to Felixstowe when the broadcast was heard. As the "CERDIC FERRY" was only a few miles from the position given, a lookout was posted and communications established between the ship and the cruiser. The CERDIC FERRY duly passed through the given position without seeing the cruiser. As the visibility was in excess of 10 miles, it was evident that the position given was wrong.

A Psion Organiser with a Sightmaster Astro-Navigation program installed came to the rescue. The skipper of the cruiser was invited to use his magnetic compass to take a bearing of the Sun, giving a result of 270 degrees. The Psion rapidly calculated the true bearing as 240 degrees, giving an error of some 30 degrees (enough to have arrived in London rather than Ipswich). With the compass error established it was a simple matter to rework the position of the cruiser as being some 12 miles to the South. The Harwich lifeboat town the crippled cruiser in during the following morning.

Product News - I

ODBTRAN

from Sycom Services

ODBTRAN is designed to perform file transfers between the Organiser ODB and dBase DBF file formats. A large number of application programs allow the import of dBase format files directly so this actually gives quite a wide spread of possibilities.

The program is menu-driven and therefore simple to use.

The price of this program is normally £35.00 but Sycom have offered to sell it to IPSO members at £25.00 +VAT (incl p&p within UK mainland). Sycom also operate a RAMPAK battery replacement service which is available to members.

For more details, if required contact:

Sycom Services
9 Museum Hill
Haslemere
Surrey GU27 2JR

Organiser Powalight

Have you ever tried to use your Organiser in poor light conditions - on the train, in the car, in a badly lit store-room, on your boat, out of doors, on an aircraft, perhaps. Anywhere, the Powalight can solve the problem. Designed to run from its own rechargeable power source, the Powalight simply clips on to the Organiser, making it a simple, yet highly effective light source that illuminates both the Organiser display and the keyboard.

For more details contact:

Mike Taylor
12 Timmis Close
Cinnamon Brow
Warrington WA2 0BU

Denwood Design New Phone Number

Anyone who has trouble contacting Denwood Design on their old number should note the new number 071 278 3906

SAMKA Assembler Update (etc)

Sam Walker informs me that his (very reasonably priced) ASSEMBLER has been radically updated.

The following features have been added:

- **Error handling
- **A De-bugger to examine memory
- **Memory allocation features
- **A print option
- **Improved speed
- **Many other useful features

Despite all the improvements, the price is still being held at £29.95.

Sam will also perform a copying service if you wish to supply your own pak for the Assembler. The pak should have a minimum of 8k free memory. The cost of this service is £19.95

SAMKA also produce a Games Pak, priced at £19.95 (or £9.95 if you provide the pak). They also have some new products:

1. **Hangman:** the famous word game brought to the Organiser. It contains a large dictionary, sound effects and graphics. You can play against the Organiser or against another person. Not available on datapak, but can be copied onto an 8k (or larger) pak for £4.95.
2. **Menagerie:** as the name suggest, this pak contains 8 mini program of assorted types. It contains various usilties, such as an on-screen UDG designer, where a small cursor can be move around and pixels plotted or unplotted. Also Menagerie contains two games, one of which is COUNT-DOWN, a word skills game. It represents good value for money at only £3.95 (send your own 8k or larger pak).

For any further details on any of the above, or to order contact:

Sam Walker
3 Tetbury Street
Minchampton
Glos GL6 9JG
Tel: 0453 883497

Product News - II

Cause for Concern?

Dear IPSO FACTO

I feel I must complain about the 'Routewise' pak and the Company supplying this item, TITAN COMPUTERS (see Product News - May).

Not believing that a pak could perform such a detailed task as plotting a route from A to B, anywhere in the UK, showing route to either screen or printer, O/Survey, A routes, etc, I rang them to check for myself. The gentleman I spoke to said that the pak would allow you to enter a town, and in London, areas, as a FROM, then enter another town as a TO. The pak would then sort out, and I quote, "...a detailed route using O/Survey co-ordinates, motorway junctions and turn offs..". I was also told that it would give you a mileometer reading giving the distance between given co-ordinates.

Too good to be true, I thought, and so purchased one.

When the pak eventually arrived, it was put to the test. It was at this point that my disbelief became disgust. I entered LONDON as the FROM and GRANTHAM as the TO. After a sorting time, the "detailed route and mileometer reading" appeared on the screen...

"TAKE A1 TO GRANTHAM"

No mileometer reading, not what I or anyone could call a "detailed route" and subsequent trials proved to be just as bad, giving no better route than jumping in your car and driving, just following the signs.

Tony Lee.

Tony's letter details a long and complicated story about how he went about sending the program back and claiming a refund. At the time of writing the letter, he had had no satisfaction, but phoned me later to say that, after much further trouble he managed to get a refund. I have heard of one or two other cases involving TITAN COMPUTERS, where payments were made but no goods received. As things stand at the moment, I can only say that you would do well to treat this firm with the greatest caution. Although they appear in the new Psion User Catalogue, the telephone number quoted is more often than not unobtainable.)

In fairness to all concerned, I must also say that one or two members, having dealt with this company, have expressed to me their full satisfaction with the service and products purchased

For Sale

Psion Spell Checker (c/w manual & box) £15.00
Phone Dave on 0702 76205

Carefully used Organiser LZ64
plus Pocket Spreadsheet, Printer Link
Mains Adaptor, Case, Manuals, Handbooks etc.
Offers around £200 to:
Gillian Bryan Tel: 0327 61631

Tandy CGP 115 Printer/Plotter
c/w spare pens & paper, manual
Comms Link & lead to connect Organiser
£85.00 ono
FNKEY - Latest Version c/w manual £35.00
Phone Neil on 0332 880663 after 6pm

Organiser XP(LA) - £60.00
Psion Printer w/Denwood Leather Case - £170.00
CUBSOFT XBASE - £35.00
32k Datapak - £20 16k Datapak - £12
Mains Adaptor - £8
P. Littlewood Databox £30
Selection of blank EPROMS - offers
All in perfect condition and with manuals
Contact: Jim Kirwan on 0772 614185

Contacts wanted

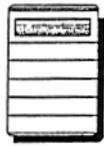
Anyone in the Radio Industry interested in receiving and/or swapping procedures, please contact:

Pete Sipple
51 Eastwood Boulevard
Westcliff-on-Sea
Essex SS0 0BY

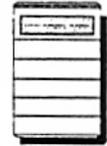
Disclaimer

I would like to make it quite clear that the inclusion of news of products in this Newsletter does not mean that IPSO endorses either the products mentioned or the companies supplying such products. From time to time some of the items featured will be the subject of reviews, which I try to ensure are fair and unbiased. Members are well advised to play safe when dealing with Companies which are unknown to them, especially when it comes to submitting relatively large amounts for products they have not seen.

Mike O'Regan



IPSO FACTO



The Newsletter of the International Psion Organiser User Group

Volume IV . No. 6 . August 1990

Editorial Office: 130 Stapleford Lane . Toton . Beeston . Nottingham . NG9 6GB . UK

Editorial

The Thesaurus & Spell Checker

I have at last been supplied with a review copy of this revolutionary pak. It is hoped that this is the first of a number of SUPERPAKS which could make use of the large capacity (256k) surface-mounted ROM. The Thesaurus is certainly a clever piece of programming, showing just how much can be crammed onto a single pak by using data compression techniques. However, I started thinking about Thesauruses (Thesauri?) and Spell Checkers in general. I have used quite a few spell checkers on a range of different computers, and there is no doubt that they have their uses. However, they also have quite a few drawbacks. For instance, I hardly ever use a spell-checker to check "spelling"; rather to check any silly typing mistakes, such as transposed letters within a word. Most spell-checkers will stop and highlight words which are unrecognised, but rarely suggest the correct word under these circumstances, especially if the first two letters of the word have been accidentally transposed. It could be said that careful proof-reading would be equally effective. The computerised Thesaurus is a fairly recent addition to the tools of the word-processor. Although it appears to have a useful function, I have yet to find ANYONE who actually uses one. If you think about it, anyone whose grasp of the language is good enough to select the "mot juste" from the proffered list so readily supplied by the Thesaurus, is probably also capable of

providing his own alternatives. Conversely, for the unwary, it is all too easy to select a wrong word. To return to the Psion pak and its capabilities. Although, in theory, it would be possible for the Organiser to use the pak to spell-check a complete document, say produced by AutoScribe or another word-processor, in practice it would be unacceptably slow if used in this way. It is more useful to just check the odd word "manually". The Thesaurus is equally effective when used in this way, although, in common with other examples, it has a few quirks. The Anagram solver is quite useable (if you have a use for it). It would be possible to increase its value by prudent use of the new commands within an OPL program. We use the Spell Checker to adjudicate any dodgy words in Scrabble!!

Contents

Editorial . . .42

Beginners Page . .43

Machine Code . . .44

Progs & Procs . . 47

Product News . . 49

Beginners Page

by Mike O'Regan

Handling Data on the Organiser - Part I

What all computers do best is to handle data, and the Organiser is no exception. In fact our little machines handle and allow access to data remarkably well - in many aspects much better than any other machine of any size, and certainly better than other pocket devices.

The Organiser has something that no Desk-Top computer has; that is a built-in powerful data handling capability. With just the two commands FIND and SAVE it is possible to do much on the CM and XP models. Even restricted to the single file MAIN it is possible to create and handle a single database on each device, therefore different databases automatically exist on each data or RAM pak as well as internally in A:

If you have a LIZZY then these facilities, plus a few more are available with either MAIN or any other file that you CREATE.

I think that it is time for a few words about the nature of databases in general, and their use on the Organiser in particular. First a look at the manual way of doing things which has direct parallels in computerised data handling. A manual database can be in the form of a box full of cards on which the data is hand-written. The box could be labelled "MAIN" to use our familiar name. If each card in the box is completely blank at the start, then we have a rough equivalent of the Organiser's built-in system.

It would be possible, but not very effective, to start filling in the cards (entering data) without any thought to arranging it in a standard way (this is often done on the Organiser!). Let us say that our cards are not completely blank, but that each card has 16 lines drawn on it to help us write out our data with one piece of data on each line.

If we started to write at random on these lines, the first card could have, say, "Mr John Smith" on Line 1 and "10 Main Street" and "Anytown" on the 2nd and 3rd lines respectively. There is nothing to stop us filling in the second, or subsequent, cards with information written in different places, say Line 1 "Anytown", Line 2 "Mr Joe Brown" and perhaps Line 3 "Tel. No: 12345".

What is more likely, is that the cards would be marked with "field names" so that each line on each card would then contain information in the same order, which would make using the cards much more efficient. For instance, each card could have the 1st line marked

"NAME", the 2nd line "Address", 3rd line "Telephone Number" etc., so that subsequently, not only would the information be entered in the same position on each card, but the "field names" would be reminders, so that we did not forget to enter individual items. So we could finish up with a number of cards, each containing information.

Each card is called a "record" and each of our marked 16 lines is called a "field". The whole collection of cards is called a "file" or "database".

Having filled in our cards, the next thing to consider is the ORDER in which we will STORE the cards in the box. This is very important if we are to be able to FIND the information later. For instance, we may decide, if our file deals with details about people, that the cards would be SORTed so that they were in alphabetical order by Surname, with "Abel" being the first card and "Zahler" the last. It would then be possible to FIND, say, "Davies" by shuffling through the "A's", "B's", etc until we arrive at the "D's" and ultimately "Davies".

But what if we wanted to find all the cards which had "London" in the line marked "TOWN"? As our cards are in "SURNAME" order, FINDing all cards marked "London" would involve looking at EVERY card, and perhaps finding after all that that no card was marked "London"! If we regularly need to FIND all cards marked "London", then we could save time by SORTing the cards so that they were in alphabetic order by "TOWN". However, should we then wish to find a card with the SURNAME "Davies" our old problem has returned.

Another solution would be to have TWO complete sets of identical cards, but SORTed in two different orders. If we applied this principle to the extreme on our set of 16 line cards, then we could find ourselves shuffling any one of 16 sets to find just one piece of information!

By now you will be beginning to realise why our Organiser database is so powerful. We only need ONE set of electronic "cards", from which we can FIND any "card" (or "cards") without having to SORT any of them. If we have a LIZZY, (or a CM or XP with a bit of simple programming) we can even ask it to FIND all the "Dr's whose surname is "Davies" and who live in "London". Try doing that in one go on a manual file!

(continued next month)

Machine Code Pages - 1

An Alternative Keyboard
by Neil Draycott

CM &
XP ONLY

The Organiser has one of the strangest keyboards in the computing world. As well as being alphabetical instead of qwerty (like a typewriter keyboard) it also requires non-standard key sequences to change between capitals and lower case letters (with the usual shift key method producing numbers). Also many useful characters cannot be accessed via the keyboard (such as the £ and ?).

In everyday use these limitations are not a real problem, but an alternative keyboard can be programmed into the operating system of the Organiser to enhance it's operation.

THEORY

When the Organiser is on, the operating system scans the keyboard all the time to see if a key has been pressed and then reports the key presses to a special program called the keyboard translator. The scanning of the keyboard does not start in one corner and then move logically through as one would expect but instead scans in a pattern which allows a quicker recognition of the keys, the actual key represented is then looked up in a table of key numbers held in the Organiser memory.

We now have the possibility of changing the keyboard by two methods; writing our own keyboard translation program and installing it in place of the existing or changing the table of key numbers held in memory. Because no secrets are kept by the Psion operating system, both these methods can easily be achieved.

CHANGING THE KEYBOARD TRANSLATOR

To allow a quick change between the alternative keyboard and the normal one a "hot key" is used, I have chosen SHIFT+SPACE. All the replacement keyboard translation program needs to do is check if SHIFT and SPACE are pressed together; if they are then change between the alternative table of keys and the normal table, if the key pressed is not SHIFT+SPACE then just pass the key onto the usual translator.

A record of where the keyboard translation program is stored is kept at memory location \$205C and the record of where the table of keys is stored is kept at location \$205E. Changing the translator program can introduce such effects as macro control (used by CUBSOFT's "FNKEY" program) or auto-booting of Paks (such as with the Psion program Paks).

CHANGING THE KEYBOARD LOOKUP TABLE

I have chosen the alternative keyboard to be lower case letters normally with capital letters selected when SHIFT is also pressed and some special characters available by pressing SHIFT+MODE, <--, -->, and EXE.

The installation program copies the normal keyboard table to a reserved section of memory changing the values to suit on the way, thus resulting in two versions of the table being available.

INSTALLING THE PROGRAM

CAREFULLY type in the installation program, translate and then run it. A small portion of memory will be lost and any other memory resident programs such as "FNKEY" will not be affected (except some of the features of "FNKEY" will be switched on and off via SHIFT+SPACE). The installation program is now not needed and can be saved to a Pak or simply erased.

USING THE PROGRAM

The Organiser operates normally until SHIFT and SPACE are pressed together, a beep is then sounded and the alternative keyboard is then in use. All letter keys are in lower case with capitals selected by pressing SHIFT and a letter together (the CAP and NUM keys are disabled and produce no effect).

Pressing SHIFT and MODE gives a "?", SHIFT and <-- gives a "£", SHIFT and --> gives a ",", and SHIFT and EXE gives a "." No numbers can be typed.

To return to the normal keyboard just press SHIFT and SPACE again, the usual keys are returned to with the CAPs status set to capitals. With a little practise this alternative set becomes very useful.

FOOTNOTE

After much experimenting I settled on memory location \$FF to use as a flag storage, this is a transient storage area that MAY be used by other applications. Does anyone know of a better zero page storage area or know of any functions that use \$FF?

(continued on next page)

Machine Code Pages - 2

EXPLANATION OF MACHINE CODE

For those interested there follows an explanation of the machine code used in the translation routine;

```

7B 01 FF  BTST #$FF  Test bit 0 of $FF (alternative keyboard on/off flag)
26 06    BNE #6     Jump forward if alternative keyboard not on
71 80 7B  AND $80,$7B Remove NUM flag from keys status
72 01 7B  BSET      Set lower case keys in key status
81 0B    CMP A #11   SPACE pressed?..
26 05    BNE #5     ...if not then jump to normal key translation
7B 80 7B  BTST $7B   SHIFT pressed?
2B 03    BMI #3     If SHIFT pressed carry on, otherwise...
7E 84AA  JMP $84AA  ...go to normal key translation.
3F 0E    SWI #14    Beep
7B 01 FF  BTST $FF   Is alternative keyboard on?..
26 07    BNE #7     ...no, so jump pass next instructions
86 01    LDA A #1   Load A with 1
CE 858B  LDX #$858B  Load X with usual key table
20 04    BRA #4     Jump forward 4 bytes
4F      CLR A      Clear A
CE Altkeys LDX #Altkeys Load X with address of new key table
97 FF    STA A $FF  Store flag at $FF
FF 205E  STX $205E  Store address of table at $205E
86 02    LDA A #2   Load A with 2
97 7B    STA A $7B  Make key status = capitals
4F      CLR A      Clear A
97 74    STA A $74  Number of keys presed now = 0
39      RTS      Return to system
    
```

newkeys:

```
LOCAL x%,y%(72),z%(72),tor%,y%
```

```
x%=1
```

```
z%(42)=46:z%(47)=32
```

```
z%(57)=8:z%(67)=63 REM other numbers may be
```

```
z%(68)=3:z%(69)=4 REM introduced here to
```

```
z%(70)=237:z%(71)=44 REM include other shift characters
```

```
z%(72)=1
```

```
IF m6<>-999
```

```
POKEW $2065,PEEKW($2065)-130
```

```
m6=-999
```

```
newkeys:
```

```
ENDIF
```

```
PRINT"Installing..."
```

```
m6=0
```

```
tor%=PEEKW($2065)
```

```
DO
```

```
y%(x%)=PEEKB($858a+x%)
```

```
IF z%(x%)>0
```

```
y%(x%)=z%(x%)
```

```
ELSEIF x%>36
```

```
y%(x%)=ASC(UPPER$(CHR$(y%(x%-36))))
```

(continued on next page)

Machine Code Pages - 3

```

ENDIF
POKEB tor%+56+x%,y%(x%)
x%=x%+1
UNTIL x%=73
z%(1)=$7b01 : z%(2)=$ff26
z%(3)=$0671 : z%(4)=$807b
z%(5)=$7201 : z%(6)=$7b81
z%(7)=$0b26 : z%(8)=$057b
z%(9)=$807b : z%(10)=$2b03
z%(11)=$7e : z%(12)=PEEKW($205c)
z%(13)=$3f0e: z%(14)=$7b01
z%(15)=$ff26: z%(16)=$0786
z%(17)=$01ce: z%(18)=PEEKW($205e)
z%(19)=$2004: z%(20)=$4fce
z%(21)=tor%+57:z%(22)=$97ff
z%(23)=$ff20: z%(24)=$5e86
z%(25)=$0297: z%(26)=$7b4f
z%(27)=$9774: z%(28)=$3900.
x%=1
DO
  IF x%=11
    POKEB tor%+y%,z%(x%)
    y%=y%-1
  ELSE
    POKEW tor%+y%,z%(x%)
  ENDIF
  x%=x%+1
  y%=y%+2
UNTIL x%=29
POKEW $205c,tor%
CLS
PRINT"   NEW KEYS           INSTALLED";CHR$(16)
PAUSE 10
STOP

```

*Editor's Note: Although it should be no longer necessary, I am once more reminding you that programs (such as the above) which directly address memory are **dangerous**, and can cause your Organiser to either **lock up** or **crash fatally**. It is in your own interest to make sure that you have backed up anything important in A: before attempting to run such programs. I am saying this because it is all too easy to get carried away, especially if you are particularly keen to get the program up and running. **You have been warned!***

This is the second procedure of FORMAT (RAMPAK), continued from Page 48

```

loadmc:(address%,mcode$)
[LOCAL addrcnt%, count%,
hi%,lo%]
addrcnt% = address%
count% = 1
DO
  hi% =
  [ASC(UPPER$(MID$(mcode$,
COUNT%,1))) - %0]
  [IF hi% > 9 :REM check
for character a to f]
  hi% = hi% - 7
  ENDIF
  lo% =
  [ASC(UPPER$(MID$(mcode$,
COUNT% + 1,1))) - %0]
  [IF lo% > 9 :REM check
for character a to f]
  lo% = lo% - 7
  ENDIF
  [POKEB addrcnt%,hi%*16
+ lo%].
  addrcnt%=addrcnt% + 1
  COUNT% = COUNT% + 2
UNTIL COUNT% >
LEN(mcode$)
RETURN

```

For Sale

LZ64, as new, with
boxes and manuals
Any reasonable offer considered

Contact:
Karen Yates
20 Linden View
Hednesford
Nr Cannock
WS12 5UA
Tel: 05438 78708

Progs & Procs - 1

Adding & Deleting Items to/from Top Level Menu

by Mike Davies

The following two programs will add/delete items to the Top Level Menu. The first (ADDTLM:) adds an item and takes 3 parameters:

1. "item\$" which hold the string to be placed in the menu
2. "pos%" is the position in the menu where the item is to be placed - 0 for first, 1 for after first, and -1 for before "OFF"
3. "type%" must be zero on XP/CM. On the LZz can also be 1 (for Notes) or 2 (for File).

Note that if an item already exists with the supplied name (item\$), then this new item will NOT be added, even if "type%" is different. Also note that this routine is not capable of restoring the built-in functions such as FIND or SAVE. This can only be done using the MODE key whilst in the Top Level Menu.

```
ADDTLM:(item$,pos%,type%)
LOCAL param%(3),a%,p%(22),str$(8)
```

```
[p%(1)=$183C : p%(2)=$4FE6 :
p%(3)=$005C : p%(4)=$DD4I]
[p%(5)=$EE00 : p%(6)=$CC21 :
p%(7)=$873F : p%(8)=$6D38]
[p%(9)=$3CA6 : p%(10)=$05CE :
p%(11)=$2187 : p%(12)=$E600]
[p%(13)=$3A6F : p%(14)=$01A7 :
p%(15)=$0238 : p%(16)=$EC02]
[p%(17)=$3F65 : p%(18)=$2404 :
p%(19)=$4F18 : p%(20)=$2003]
p%(21)=$CE00 : p%(22)=$0039
```

```
[IF(PEEK($FFE8) AND 8)=8 AND
(PEEK($FFCB) AND $80)=$80]
    REM We are an LZ so "2" is max
    value for type%
    a%=2
ENDIF
IF type%<0 OR type%>a%
    RAISE(247)
ENDIF
```

```
str$=item$
[param%(1)=ADDR(str$) :
param%(2)=pos% : param%(3)=type%]

a%=USR(ADDR(p%()),ADDR(param%()))
if a%
    RAISE(a%)
ENDIF
```

The following procedure deletes an item and takes just 1 parameter - "item\$" holds the name of the item to delete. If the named item does not exist in the menu, then no error is returned.

```
DELTLM:(item$)
LOCAL p%(2),str$(8)
p%(1)=$183F : p%(2)=$6739
str$=item$
USR(ADDR(p%()),ADDR(str$))
```

MOT & SPEL

by Mike O'Regan

The following two programs are the first I am publishing which use the new Thesaurus/Spell Checker. Note that without the presence of the pak, (which must be booted), the programs will not work.

MOT (French for "word", don't you know!), will accept up to 9 letters and then produce ALL the known ANAGRAMS of these letters (using all 9). Just enter your letters and press <EXE>. Press <ON/CLEAR> to leave the program.

```
mot:
LOCAL w$(9),w2$(9)
CLS :PRINT "Letters - up to 9"
INPUT w$
CLS :PRINT "I've found..."
PRINT SPELL$:(6,w$),
WHILE LEN(SPELL$:(7))>0
    PRINT SPELL$:(7),
ENDWH
PRINT :PRINT "...no more"
IF GET=1:STOP
ELSE mot:
ENDIF
```

Progs & Procs - 2

SPEL accesses the Spell-Checker part of the pak. If you enter a correctly spelled word, then this is accepted and signalled as "good". If you enter an incorrectly spelled word or other jumbled letters, then the program will say "I think you mean ..." followed by a list of (numbered) possible words. It will also search any supplementary words which you have added to the main dictionary using USERDICT. You will find that the pak produces some very unlikely words, including Proper nouns, but it is good fun.

```
spel:
LOCAL sp$(15),cnt%
CLS
PRINT "spell your word"
INPUT sp$
IF SPELL$(1,sp$)<>" "
CLS
PRINT sp$
PRINT "is right"
GET
ELSEIF SPELL$(2,sp$)<>" "
CLS
PRINT "I think you mean"
cnt%=1
PRINT cnt%,SPELL$(2,sp$),
PRINT "...or"
WHILE SPELL$(3)<>" "
cnt%=cnt%+1
PRINT cnt%,SPELL$(3)
PRINT "or",
ENDWH
PRINT "what?"
ELSE
PRINT "I don't know it!"
ENDIF
IF GET=1 :STOP
ELSE SPEL:
ENDIF
```

Format (RAMPAK)

by L.J. Piper

The following two programs will FORMAT a RAMPAK, on either an XP or CM. This facility is built-in on the LZ's UTILITY Menu and so is not required for the LZ Models.

```
format:
LOCAL m%, m cadr%
LOCAL pack$(2), mcode$(88),
code$(176)
start::
CLS
[m% =
MENU("Format_RAM_Pack:,B:,C:,Exit
")
IF m% < 1 OR m% > 3]
RETURN
ELSEIF m% = 1
GOTO start::
ENDIF
pack$ = "0" + HEX$(m% - 1)
ask::
CLS
m% =
MENU("Are_you_sure_?,Yes,No")
IF m% < .1 OR m% > 3
RETURN
ELSEIF m% = 1
GOTO ask::
ELSEIF m% = 3
GOTO start::
ENDIF
CLS
PRINT "CHECKING PACK -"
PRINT "WAIT."
m cadr% = ADDR(mcode$) + 1
[code$ = "0C4FC6" + pack$ +
"3F62254BDE94A6008402261786FF9741
CC0001CE00413F61"]
[code$ = code$ + "25354FC6" +
pack$ +
"3F62252E202E3F6F0C104E4F54205241
4D20"]
[code$ = code$ +
"5041434B0D0A70726573732073706163
65206B6579003F493F48C12027"]
code$ = code$ +
"0426F63F203F5B39"
loadmc: (m cadr%,code$)
USR(m cadr%,0)
GOTO start::
```

The second program,
LOADMC: is listed on
Page 46

Product News

Widget FileMaker

For a long time now I have been convinced that, if the Organiser owner has access to a PC (IBM personal computer or clone), then the best (in terms of security and cheapness) way to store both data and procedure files from the Organiser is on the PC's disk (either floppy or hard). Using the simple programs supplied with the Comms Link, this is quite quick, clean and simple. Sending data to the PC is one thing. Reversing the process and sending data from the PC to the Organiser takes a little more thinking out. There have been some attempts to provide PC software which makes this a bit easier. **FileMaker**, from **Widget** is the latest to appear on the scene.

FileMaker is a PC program which will enable you to create and manipulate database files suitable for use on the Organiser II (all models). It features Import and Export commands to make it possible for the Organiser to exchange files with other PC software such as Dbase, Lotus 123, Dataease etc.

Besides being able to create new files for the Organiser, it can also manipulate existing files created on the Organiser, such as MAIN, etc. A powerful feature is that, after creating or loading a data file, it can SORT the file into Ascending or Descending order on ANY field. Once this has been done to your satisfaction, the sorted file can be returned to the Organiser, either as a separate file or to replace the original one.

FileMaker is supplied with a very easy to follow 16 page booklet, which explains how everything works, even for a beginner. The program itself, being menu-driven, is very easy to use and the handbook becomes superfluous after using it a couple of times.

I tried out the program with one or two large databases, and it worked perfectly every time. However, I am quite used to transferring both data and procedures between my PC (Opus) and Organisers, for which I find the Comms Link programs quite adequate. When it comes to databases, I transfer selected fields of my Membership File (which is maintained, using Superbase Professional). This is done by creating an appropriate Export file within Superbase. It is possible to define the field and record separator characters to match the Organiser's standard <TAB> and <CR><LF>, so that no further translation (i.e. to convert a standard comma-separated file) is necessary. The main file on the PC is kept in alphabetic order by Surname, and this is the way it is sent to the Organiser. By the way, it takes about 45 minutes! to transfer the

file (which is too big to fit in even an LZ64's A:). It is written directly to a 64k datapak.

FileMaker is available NOW directly from

Widget Software Ltd
121 London Road
Knebworth
Herts SG3 6EX
Telephone Orders to:
(0438) 812320 Price £59.95

UDGED from Vincent Yelland

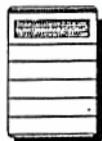
In one way, a UDG editor is essential for the LIZZY. This is because, unlike the CM and XP, the LIZZY uses all 8 UDGs every time you switch on (this is unfortunate, as it would have been much better, say, to make at least 32 UDGs available, or much better allow ANY standard character to be re-defined!) Anyway, it is useless to dream of what might have been, and we are stuck with the LIZZY the way it is.

UDGED uses all four lines of the display to display - and manipulate - UDG designs. A similar program is available on the **KIRSTA KPROG** pak. Both versions have pros and cons. Both use the cursor keys to move around and set or delete PIXELS using <EXE>.

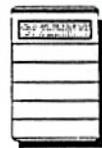
What both of these programs fail to do is to actually CHANGE the UDGs permanently - they only give the codes to be used with the new UDG command. The **KPROG** program can be used with any model from CM to LZ64, whereas **UDGED** is specific to the LZ.

The price of the program has not yet been fixed - it depends on the potential sales. For more information contact:

Vincent Yelland
20 The Mille End
Wlathamstow
London, E17 5QE
Phone: 071-236 2489



IPSO FACTO



The Newsletter of the International Psion Organiser User Group

Volume IV . No. 7 . September 1990

Editorial Office: 130 Stapleford Lane . Toton . Beeston . Nottingham . NG9 6GB . UK

The IPSO FACTO Production Line

I have had quite a few letters asking how the newsletter is produced and so here is a full description.

In the early days (Volume I) IPSO FACTO was written on a BBC Master Micro, using Wordwise, and it was of necessity quite a simple affair. The masters were, at first produced on an Amstrad DMP printer, but this was soon replaced by a Citizen 120D.

Volume II saw a change to an Atari 520 STM, using various DTP programs. The printer was still the Citizen, but some masters were printed on an Atari Laser printer (by the ST User Group).

Volume III saw other major changes. The computer became an Opus PC IV (XT clone), with EGA monitor and 20mb hard-disk. The printer was also changed to a 24 pin Epson LQ 500. Source material for the newsletter comes in various forms - on datapaks, on 5.25" disks, handwritten, typed or printed, or as EMAIL (rare). Most text is fed into 1st Word Plus for editing, etc. Program listings are first fed through Tony Spencer's excellent IPSOCONV program on the Opus (to convert them into our preferred format).

All material is then fed into Logitech FINESSE (latterly upgraded to Version 3.1, with Spell-Checker etc. - thank God). Master pages are now printed on a Hewlett-Packard Deskjet Plus Inkjet printer (300 dpi). The Organiser Logo was created with GEM Draw. The typefaces used are Swiss (Helvetica), Bitstream Charter for text, and courier for listings

The masters go off to our very reliable printers, who print, punch and collate the whole of the issue. Finished magazines are folded and put into envelopes by myself and my long-suffering wife. Labels are printed (now on Laser Label sheets) from the IPSO members database (Superbase Professional) on the Opus.

That's about it! I am, on the whole, quite glad that I learned to type as a teenager - all those years ago (groan).

The MIDI Link

As I am a musician (when I'm not slaving over a hot IPSO FACTO!), and have a few electronic instruments which feature MIDI (Musical Instrument Digital Interface), I have long been interested in computer programs which allow me to program the instruments in various ways.

In the days of my Atari, the interfaces were built in. However, it should be possible to use the Organiser to program these instruments. The obvious advantage is the portability of the Organiser.

If anyone could come up with a program to do this, I am quite sure that it would find a ready market. Many musicians actually carry a full Atari system around with them just to program their instruments. Is there anyone out there who could write the necessary software?

Contents

Editorial50
Beginners Page .	.51
Progs & Procs . .	.53
IPSO Review57

Beginners Page

by Mike O'Regan

Handling Data on the Organiser - Part II

Last month we had a look at most of the features - and drawbacks - of a card-based database. This month I am going to examine the built-in database of the CM/XP and the possibility of multiple databases on the LZs (or CM/XP, with the help of some simple software written in OPL)

You have probably noticed (if you have used the Organiser's built-in database facilities) that the Organiser treats all data entered through the SAVE command as ALPHA STRINGS, even if numbers form part of some or all records. This means that, although the two commands SAVE and FIND work very well, it is not possible to manipulate any NUMERIC data.

For instance, if our database was, for example, a membership list, it is not possible to TOTAL a field named "subscription" to arrive at a "total subscription" figure. We shall see, however, that it is possible to do this, fairly easily, if we write our own OPL database handling routines. In fact, OPL was written especially to simplify data handling (unlike most other computer languages).

Before we make a start on writing the first procedure, one last look back at the most efficient methods of using the FIND, SAVE and MAIN database, without the necessity of writing any OPL.

Let me first remind you that I have seen very many INEFFICIENT DATABASES on Organisers! I realise that I have mentioned this several times before, but it is very important.

One of the strengths of the Organiser when handling data is also a weakness, used wrongly. That is that the Organiser will literally SAVE ANYTHING, in ANY ORDER, without so much as a BEEP of protest. What is more, FIND will work on any bit of data ANYWHERE in the record. However, if you are searching through a database with FIND it is, at best, a nuisance if, say, the top line (field 1) of each record contains different information - on one a telephone number, on the next a Christian name, with a third showing "town" etc. If you are in a hurry (someone on the phone from Barbados!) it can be a real nightmare trying to winkle out information under these circumstances.

Now a little thought, BEFORE ANY DATA IS ENTERED, will reward you well later on. For instance,

writing out a real card (from your now-redundant cardbox) with FIELD NAMES from 1 to 16, if required. If you don't want to carry the card with you, or you are afraid you will lose it, then use the Organiser itself to store the format. Just enter as the first record in your file:

Title & Initials:
First Name:
Telephone No:
Subscription: (etc.)

If you forget what the format is, then FIND (with no clue entered) will put your first record (format) on the display.

Incidentally, if you are using a LIZZY, it is very important that your data is in the right order, as the "wild-card" FIND will not operate properly unless the clues are entered in the right order. For instance, if your file structure was similar to the short four field one shown above, then "FIND 081*smith" will not turn up any Smiths living in Outer London because the "telephone no" prompt should follow the "name" one.

Now to begin writing our first OPL data-handling procedure. This is to be a very simple database, as it is really only meant to demonstrate the principles. However, once these are grasped, you should be able to devise your own additions to the procedure to do the things which you would like YOUR database to feature. Now key in the following procedure:

```
dbstart:  
IF NOT EXIST("A:member")  
CREATE "A:member", A, name$, phone$, subs  
ENDIF  
PRINT "Database called"  
PRINT "MEMBERS exists"  
GET
```

That wasn't too painful, was it? Here is a short explanation of the procedure:

Line 1 is the procedure name (all our proc names will start with "db..."). Do not key in the ":" - the Organiser does this for you.

Line 2 is almost plain English - it looks to see if a file called "MEMBER" exists in the Organiser memory.

Line 3 is what happens if the file DOES NOT EXIST - a file called "member" is CREATED in the Organiser memory A:. The other A (between "A:members" and "name\$") is known as a LOGICAL FILE NAME. It is usually A, but I will explain more about this later in the

Small Ads & Feedback

series. The "member" file has only THREE FIELDS. Two are STRING fields - name\$ and phone\$. The third field, "subs" is a NUMERIC field (no \$ sign is used)

Line 4 (ENDIF) is the end of the IF which is in Line 2

Lines 5,6 and 7 are only to inform you that the file exists. They are not essential to the program operation. Pressing any key will terminate the program.

As we are taking it in very easy stages, this is all for this month. If you have ANY queries about anything so far, please get in touch.

Small Ads For Sale

Organiser, Model CM
with Handbook and AutoScribe Plus
all boxed and in mint condition
Any reasonable offer accepted

Phone Nicholas Holmes on
071 260 1571(daytime)

Organiser XP(16k)
£50

Phone Marc Caplan on 081 954 6367

Atari Portfolio Pocket Computer
c/w Serial & Smart Parallel interfaces
plus 32k RAMPACK
Books, etc., all mint

£200

Casio fx-7000G Scientific Calculator
mint - £20

Yamaha SHS-10 MIDI Keyboard
mint, boxed, manual £50

phone Mike O'Regan
on (0602) 735482

For Sale (cont.)

Concise Oxford Spell Checker - £20ono
Phone Ray Waterhouse on
0695 576101 (evenings)

Organiser XP/LA 32k model - £50
Comms Link & Software - £30
Denwood Filofax Type Leather Case - £20
Pocket Spreadsheet - £20
Letter Organiser - £10
AutoScribe II(1988) - £20
Mike Shaw Book I - £10
2 x 64k datapaks - £60
1 x 32k datapak - £15
1 x 16k datapak - £10
1 x 32k RAMPak - £20

Phone Ruth Corbett on:
071 938 1957 (day) or
081 992 0119 (eve) (W.London)

Upgrade Tips

I don't know if your readers are aware, but the Comms Link can be upgraded (from 2 line to 4 line) by returning the old product to Psion and enclosing £34.95. Also the Developer Pack can be upgraded for £19.95

Marc Worsfold

Lizzy Bug

A small OPL problem I have encountered. The DAYS function returns the number of days since 1/1/1900. If this is used to evaluate today's date by:

$y\% = \text{DAYS}(\text{DAY}, \text{MONTH}, \text{YEAR})$

an integer overflow error is returned (Error #195), as $y\% > 32767$. My solution is to subtract 28 from YEAR, since calendars repeat every 28 years.

Julian Water

Progs & Procs - 1

FSIZE & FDUMP

by Peter Houppermans

Fsize is a rather tiny but useful program to quickly determine how much space a certain file takes on a pack (or in RAM), as PSIONs 'dir' command in the LZ only shows how many records are available, not how much space they need.

Since the 'trick' that forms the core of this program is applicable to both XP and LZ, I wrote a procedure for both machine types (actually, I adapted one to run on the other machine).

The thing that makes this program work is that the entire record is always stored in a special filebuffer, and the program just reads the byte which indicates the length of it. As a result, a file can be scanned rather quickly and you can easily ask features as 'max, min and average record length'.

```
fsize:
LOCAL file$(10),size,k%
CLS
CLOCK(1)
UDG 0,31,17,23,19,23,23,31,31
AT 1,1
PRINT CHR$(0);REPT$(CHR$(2),14);
AT 1,2
[PRINT" = FileSize v1.00 = by Peter
Houppermans";]
PAUSE 20
start::
REM ==> GET filename (+ pack)
AT 1,3
PRINT CHR$(22);"File => ";
TRAP EDIT file$
IF ERR OR file$=""
RETURN
ENDIF

REM ==> test file
TRAP OPEN file$,a,a$
IF ERR
AT 1,4
[PRINT CHR$(23);"ERROR:File not
found";]
GOTO start::
ENDIF

REM ==> GET size
[REM file header takes always 11 by-
```

```
tes]
size=11
AT 1,4
[PRINT CHR$(23); :REM clear 4th line
WHILE NOT EOF]

REM read length in filebuffer "A"
REM AND add "2" for the record header
size=size+2+PEEKB(PEEKW($2016))
AT 1,4 :PRINT size,
NEXT
ENDWH
[AT 1,4 :PRINT CHR$(23);size, :REM
was NOT printed IF file empty]
PRINT"bytes",COUNT,"rec";
CLOSE
GOTO start::
```

```
fsize
LOCAL file$(10),size,k%
CLS
PRINT"=FileSize v1.00=by P Houpper-
mans";
PAUSE 20
start::
REM ==> GET filename (+ pack)
AT 1,1 :PRINT"File => ";
TRAP EDIT file$ :REM GET file to
scan
IF ERR OR file$=""
RETURN
ENDIF

REM ==> try to OPEN file
TRAP OPEN file$,a,a$
IF ERR
AT 1,2
PRINT"File not found !";
BEEP 99,99
GOTO start::
ENDIF

REM ==> start counting
AT 1,2
[PRINT CHR$(15); :REM clear SECOND
line]
REM file header takes always 11 by-
tes
size=11
WHILE NOT EOF
[REM read length in filebuffer "A"
AND]
[REM add "2" for the record header
```

**CM & XP
only**

**LZ
only**

Progs & Procs - 2

```

size=size+2+PEEKB(PEEKW($2016))]
AT 1,2
PRINT size,
NEXT
ENDWH
AT 1,2
PRINT CHR$(15);size, :REM was NOT
printed IF file empty
PRINT"b/",COUNT,"rec";
CLOSE
GOTO start::

```

The next program is a short and universal one to dump the contents of a text file to a printer. It will adapt itself to any number of fields you might have been using, and by its 'modular' structure you can take any part you like to use in your own routines too

The program checks for the presence of a file, for the presence of a device capable of handling the 'LPRINT' command (Comms Link, ParaLink or Printer II), and you can add as many features as you would like (you will do anyway), the basic cores you need are in this program.

```

fdump:
LOCAL buffer$(255),file$(10)
LOCAL a%,boot%(2),swap%
CLS :PRINT"File:";
TRAP INPUT file$
IF ERR OR file$=""
RETURN
ENDIF

swap%=$1839 :REM xgdx, rts
boot%(1)=$3f17
boot%(2)=$3901
[USR(ADDR(boot%()),0) :REM autoboot
devices]

REM test device presence
ONERR test::
LPRINT"";
test::
ONERR OFF
IF ERR
CLS :PRINT"device missing"
PAUSE -30
[WHILE KEY :ENDWH :REM empty key-
board buffer]
RETURN
ENDIF

TRAP OPEN file$,a,a$
IF ERR

```

```

AT 1,1
PRINT"file NOT found"
PAUSE -30
[WHILE KEY :ENDWH :REM empty key-
board buffer]
RETURN
ENDIF

[REM initialise printer here IF you
wish]

WHILE NOT EOF
REM read entire record

[buffer$=USR$(ADDR(swap%),PEEKW($201
6))]
[WHILE a% :REM as long as tabs
EXIST]
[buffer$=LEFT$(buffer$,a%-
1)+CHR$(124)+MID$(buffer$,LEN(buf-
fer$)-a%)]
a%=LOC(buffer$,CHR$(9))
ENDWH
LPRINT buffer$
NEXT
ENDWH
CLOSE

```

Editor's Notes:

1. Square Brackets, [], are used to mark the beginning and end of long lines and should not be entered (you can't get them from the standard Oragniser keyboard anyway!)

2. Blank lines, REM (remark) statements, and indents are added for clarity, but need not be entered as they do not affect the programs in any way.

Adjustment to "Alternative Keyboard" Proc. (page 44)

The storage area used (SFF) causes the alternative keyboard to switch on unexpectedly.

Some suggested other storage areas are:

\$7C Auto switch-off flag - works fine until Comms Link is connected, then switches alternative keyboard on when call made to the Link

\$78 Auto repeat flag - works OK but keys are slowed down when alternative keyboard is on.

\$69 scrolling speed flag - works OK but speeds up any scrolling messages.

To use any of these values, simply replace the value SFF in the array zz%(2)=, z%(15)=, z%(22)=, in the installation program with the chosen value and re-run the program.

Progs & Procs - 3

Load & Store Menu

by M.J. Siddiq

In response to requests for routines which will load a user's customised menu, the following procedures will enable a use to load his/her procedures into the top-level menu without all the usual palaver. THESE PROGRAMS WERE WRITTEN PRINCIPALLY FOR THE XP. The machine-code routine slots into the application area starting £2020.

When the Organiser is switched on from a reset, the Main Menu looks like this:

```
FIND SAVE DIARY
CALC PROG ERASE
TIME INFO ALARM
COPY RESET OFF
```

Suppose, before deletion of any of the system procedures (such as DIARY) you wanted the Top Level menu to look like this:

```
FIND SAVE DCODE
ASSM FDEL DIARY ...etc.
```

The added procedures will be the ones that need to be inserted by the program LDMEN.

The required procedures along with their position in the Main Menu should be saved using STMEN.

Going from left to right and then line by line with the initial position representing zero then DCODE will be in position 2, ASSM in position 3 and FDEL in position 4 and so on. This information is stored in RANK ORDER (lower positions first) using STMEN. As the order is important, stored files are overwritten rather than edited.

It is preferable to either write or copy the file MENU, which stores the information, to datapak. Providing that the appropriate datapak is fitted the user procs can be loaded into the Main Menu using LDMEN

```
ldmen:
LOCAL
[ps,a,x,p$(30,10),z,p(30),n,fn$(10),
g$(1)]
CLS
KSTAT 1
PRINT"MENU-LOADER"
PRINT"PAK A, B or C";CHR$(63);
```

```
DO
g$=GET$
UNTIL g$="A" OR g$="B" OR g$="C"
fn$=g$+": "+"MENU"
IF EXIST(fn$)
OPEN fn$,a,p$,p
ELSE
PRINT"ERROR: NO SET-UP"
PRINT" FILE"
GET
STOP
ENDIF
n=0
DO
n=n+1
POSITION(n)
p$(n)=a.p$
p(n)=a.p
NEXT
UNTIL EOF
DO
x=x+1
z=LEN(p$(x))
POKEB $2187,z
ps=0
DO
ps=ps+1
POKEB
$2187+ps,ASC(MID$(p$(x),ps,1))
UNTIL ps=z
ps=ps+1
POKEW $2187+ps,p(x)
POKEB $2020,$c6

POKEB $2021,p(x)
POKEB $2022,$3f
POKEB $2023,101
POKEB $2024,$39
USR($2020,0)
UNTIL x=n

stmen:
LOCAL p$(10,p,g$(1),fn$(8),n,pg$(10)
CLS
KSTAT 1
PRINT "MENU-SAVER"
PRINT "PAK A, B or C"
DO
g$=GET$
UNTIL g$="A" OR g$="B" OR g$="C"
fn$=g$+": "+"MENU"
IF EXIST(fn$)
PRINT "MENU FILE EXISTS"
PRINT "OVERWRITE (Y/N)"
DO
```

Progs & Procs - 4

```
g$=GET
UNTIL g$="Y" OR g$="N"
IF g$="N"
  STOP
ELSE
  DELETE fn$
  CREATE fn$,a,pg$,p
ENDIF
ELSE
  CREATE fn$,a,pg$,p
ENDIF
DO
  n=n+1
  CLS
  PRINT"OPL PROG:",
  INPUT pg$
  PRINT"Position:",
  INPUT p
  a.pg$=pg$
  a.p=p
  APPEND
  DO
    PRINT "Another entry";CHR$(63)
    g$=GET$
    UNTIL g$="Y" OR g$="N"
  UNTIL n=30 OR g$="N"
  IF NOT g$="N"
    PRINT "SORRY, file full"
    PAUSE 10
  ENDIF
  PRINT "closing file...."
  CLOSE
  PAUSE 15
```

The Simplest VAT Program

by Mike O'Regan

I have been asked several times for a REALLY SIMPLE VAT program (sometimes from people who bought the Psion Formulater pak for this purpose!). The following procedure couldn't be much simpler. It will do repeated VAT conversions both from NET-GROSS and GROSS-NET until 0 (zero) is entered at the "Sum" prompt.

```
vat:
LOCAL num
DO
  CLS
  PRINT "Sum:",
  INPUT num
```

```
IF num=0 :STOP :ENDIF
CLS
[ PRINT "+VAT",FIX$(num*.15,2,6)
, FIX$(num*1.15,2,6) ]
[PRINT "-VAT",FIX$(num/1.15,2,6),
FIX$(num-(num/1.15),2,6) ]
GET
UNTIL num=0
```

I make no apology for including this program, despite its simplicity. I have supplied copies to quite a few people who regularly need VAT conversions, and they all swear by it! It will run on ALL MODELS.

A Challenge to Programmers from the Editor

01 LBL COMP	10 RCL 01
02 0	11 +
03 STO01	12 STO01
04 LBL 01	13 GTO 01
05 0	14 LBL 02
06 TONE 0	15 RCL 01
07 PSE	16 STOP
08 X=0?	17 GTO 01
09 GTO 02	18 END

DO NOT KEY IN THE ABOVE PROGRAM! (even if you can). It was written for another machine (which HP owners will recognise) - a HEWLETT PACKARD 41C Programmable Calculator. This does not mean that IPSO is branching out to include other machines. What is interesting is the purpose of the program. It is a short (27 bytes only) routine to work as an AUTOMATIC ADDING MACHINE. When the program is run, a low-level BEEP signals "ready for next number". At this stage ANY number can be keyed in (at a reasonable speed). There is then no need to key in "+" (or anything other than the next number. If, on the BEEP prompt, no number is keyed in, the "total so far" is displayed and the program then STOPS and waits for the R/S (EXE) to be pressed, when it will carry on as before. The secret of why it works with ANY NUMBER of ANY LENGTH is that the HP 41's keyboard remains LIVE during a PSE (pause).

The challenge is for you to write a program for the Organiser which will perform the same functions. There will be a small prize for the best entry (if it is at all possible!).

IPSO Review

by Mike O'Regan

IFILES & Index Manager

from EMF in Control

In the world of desktop computers, data handling is usually done in quite a different way to the way the Organiser handles the job. Most database software requires a fair amount of work on the part of the user before any data at all can be entered, and even more work before any meaningful use can be made of the data once it has been entered. However, the effort required to "setup" a database is usually worthwhile (see recent articles in the Beginners Section).

If data is entered at random on the Organiser (which is usually the case), then its nature is quite apparent if we just "browse" through the records from start to finish. Unless it has been possible to "SORT" the records, they will be more or less in the order in which they were entered, minus any which may have been subsequently amended in any way (where such records are tagged on the end, as we all know).

Data entered into "desktop" databases is usually just as random, except that the user is usually unaware of this. The reason is that such databases have one or more INDEX FILES, one of which is in the "driving seat" at any particular time. For instance, if the database is in the form of a personnel file, then the DEFAULT or KEY INDEX could be ALPHABETIC by SURNAME. So, whenever any work is done on the file, such as entering new records, deleting, amending, etc., every time we use the database the records "appear" to be in alphabetic-by-surname order.

It is also quite usual to have at least one (often more) alternative INDEXES. For instance, a personnel file might also have an index based on STAFF NUMBER. It would then be possible to call up this index, when required, and while it is in operation, all records "appear" to be in numerical-by-staff-number order. I say "appear" to be, because the order of the actual records has not changed. It is just that an index has been used to make the order seem to be different.

What is the point of all this, I hear you ask! Well, up to now, it seems that INDEXES have been almost totally ignored in Organiser-database circles. It seems remarkable that no-one has really thought of applying the INDEX principle to Organiser databases. Until now.

The IFILES programs seem to have filled a gap in Organiser software. IFILES is intended to be a replacement for XFILES on the LZ, and to provide XFILES facilities (and more) for all models. The programs also provide a number of language extensions (to OPL) to enable the more experienced user to write their own database software. IFILES allows you to

build an INDEX SYSTEM into your database, either before you enter data, or, to a limited extent, apply the INDEX principle (and much more) to an existing data file. It is possible, for instance, to enter "prompts" (field names) to assist when both entering and using records. Fields may be designated as text, integer or floating-point, so that numerical fields can be further processed later.

Once the indexes have been created they can be used to FIND records in a more precise way. For instance, if the SURNAME index is operating, then the SURNAME field ONLY is scanned on each record when FIND is used. If the current index is on a numeric field, then even more precision is possible, using >=, =, <=, etc.

The real advantage of this system of indexes is that enables the user to keep secure data files on datapaks, adding to or deleting as necessary, but keeping the INDEX files (which are in a constant state of flux) in either RAM or on a RAMPACK.

The OPL extensions (which are used in IFILES) are accessible by programmers through the Index Manager, which is part of the system. The extensions are as follows:

Index creation - INDEX

Index manipulation - OPEN, VOPEN%, USE, CLOSE, VALID%, INDEXES%, INDEXNO%, INDFLAG%, FIELDS%

File manipulation - EXIST%, DELETE, COPY, DIR\$, DIRW\$

Performance tuning - TESTSIZE, TESTTYPE

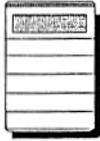
System functions - SWI%

The language extensions generally function in a similar manner to their OPL equivalents. All standard OPL file manipulation procedures, except for APPEND, UPDATE, and ERASE, may be used in parallel with the index extensions.

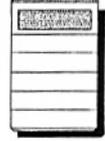
This software has many more really useful features too numerous to mention in full. I think that the package represents many truly innovative features to greatly extend the Organiser's data handling facilities.

IFILES and Index Manager is available on a 32k datapak from:

**EMF in Control
Falcon House
533 Buxton Road
Great Moor
Stockport
Cheshire SK2 7HJ
(Tel 061 483 6881)
Price £49.95**



IPSO FACTO



The Newsletter of the International Psion Organiser User Group

Volume IV . No. 8 . October 1990

Editorial Office: 130 Stapleford Lane . Toton . Beeston . Nottingham . NG9 6GB . UK

Why do *YOU* own an Organiser?

What makes anyone lay out a fair amount of money on an Organiser? There must be some common factor somewhere, (or maybe a number of them) or Psion would not have sold more than half a million of the little beasts. I have put this question to many Organiser owners and received almost as many different reasons. One which stands out above all others is that it was bought as a reliable pocket data handling device which should be of great practical value to its owner. The built-in facilities seem at first glance to cater for every need. Some owners (who didn't invest in some form of data pak or RAM pak) found to their cost that information which they so carefully typed into their little friend could, very easily, disappear by accident - I have had many a phone call of people almost literally in tears having done something which crashed the machine and lost forever their valuable data. One or two have given up at this stage and reverted to the old style filofax type of Organiser, which was more reliable, if not quite as versatile. This is a pity when it happens, as, when treated properly the Organiser behaves itself and becomes a really useful tool.

For myself, I own my Organiser(s) for a variety of reasons - not least that I have a fascination for tiny electronic devices which can be made to do almost anything, with a bit of coaxing. I also count myself among those who like to PROGRAM for its own sake - and who can say we are wrong. I have been programming first calculators, then computers since the late '70s and have never lost my fascination for this side of the hobby. Amongst the pocket machines I have owned (or still do own!) are a Sinclair Enterprise, Casio 501s and 502s, Sharp 1211, PC 1500, Atari Portfolio, and Microwriter Agenda. I also owned an Organiser Model I, (which was pathetic), before moving on the Organiser II. My Organisers earn their keep by holding at least two pretty large databases and many appointment schedules (all regularly backed up against accident, I might say!)

Going back to my original question - why do *YOU* own an Organiser. Write in and let me know. I will publish the most interesting tales and perhaps offer a small prize to the most original one.

A New Source of Organiser Equipment

One of our newest members is offering Organiser equipment at discount, especially to members. This small firm already deal in ACORN and COMMODORE machines and offer "a full range of hardware and software" for these machines. They are just starting to stock Organiser items and would welcome enquiries for further information to:

Leopard Technology
7 Oakhill Avenue
London
NW3 7RD
Tel: 071 435 0005

Contents

Editorial	58
Beginners Page.	59
Progs & Procs	61
IPSO Reviews	64

Beginners Page - 1

by Mike O'Regan

Handling Data on the Organiser - Part III

Last month, after comparing manual card-based databases with those on the Organiser, we wrote our first OPL program, which checked to see if a file called "MEMBER" existed and CREATED one if it didn't.

At this stage, did you see the deliberate mistake in the procedure? That's right you eagle-eyed observers, the program prints the message "Database called MEMBERS exists". I did this for a special purpose and that is to illustrate that, although the printed message is wrong (the file is called MEMBER - without the "s"), nevertheless the program will still run faultlessly. This shows that any computer (including the Organiser) can be made to show false information without flinching. We could equally have written PRINT "3 + 5 = 10". The program would have printed this, despite the fact that $3 + 5 = 8$.

Anyway, having run this procedure, we now have a file called MEMBER on device A: (built-in memory). Before we go any further, let us examine a peculiarity of the Organiser, when it comes to databases. If you own an LZ, having run this procedure, it is possible to access the file which we have created by using XFILES. However, if we do, then the FORMAT of the file which we carefully created to have just 3 fields - two ALPHA and one NUMERIC - is totally ignored by the Organiser! Although it is possible to OPEN the file, thereafter it behaves exactly like a file created by the XFILE NEW option, with a full 16 fields (all ALPHA). It is also possible to enter correctly formatted records from XFILES if we carefully use only the first three fields, with the third field entered deliberately as a number by using the SHIFT/NUM (or SHIFT only, held down).

If your machine is a CM or XP, then the last paragraph does not apply. In any case, the next stage in our procedure writing will be designing and writing a procedure to actually enter data in our file. To make the data entry process more comfortable each field will have a "prompt" to make sure that you know just what order to enter the information.

This may seem superfluous with just 3 fields, but the principle is the same no matter how many fields are in the database (up to the Organiser's 16 field maximum). Another advantage of entering data in this way is that, when we come to a numeric field, the Organiser automatically does the "SHIFT/NUM" for

us so that we have yet another reminder that the data expected is numeric.

At this stage, let us look a bit closer at the second and third fields of our data file. The second (phone\$) is reserved for telephone numbers. Note that we have made this a \$ (alpha string) field. This is because, although telephone numbers mostly consist of figures, it is possible to have alpha parts. For instance, my telephone number could be either "0602 735482" or "Nottingham 735482".

If we try to enter alpha data in a field reserved for numeric data, then we will get an error, or at least a ? to tell us that the Organiser was expecting a number. The best way to think of the difference between alpha data which contains numbers and pure numeric data is to think of the purpose of the data. Our telephone number is alpha because the numeric part is purely incidental (as far as the data file is concerned). In other words, we are not likely to total all the telephone numbers in our file and then divide the answer by the number of records to find an "average telephone number"!

On the other hand, our "subs" field is deliberately numeric so that we can - later on - use it for genuine numeric purposes. This last point is one of the great advantages of writing data handling procedures instead of using the built-in ones. Oh, and one last point before we list the next procedure; the "subs" field is called a "floating-point" field, because it can be used to enter numbers which have a decimal part (if necessary), so that we can enter, for example, £12.50. If we had designated the field as an "integer" field (by adding % to its name thus, subs%) then we would only be able to deal in whole numbers.

So on to our next procedure which will be used to enter information into our data file (MEMBER):

```
dbenter:
OPEN "A:member",A,name$,phone$,subs
DO
PRINT "Title,Inits,Name"
INPUT a.name$
IF LEN(a.name$)=0
STOP
ENDIF
PRINT "Telephone No:"
KSTAT 3
INPUT a.phone$
PRINT "Subs amount",
```

(continued on next page)

Beginners Page - 2

by Mike O'Regan

```
INPUT a.subs
KSTAT 1
APPEND
UNTIL 0
```

What the program does

This procedure is a very basic data entry procedure. Nevertheless, it does have one or two points worth a mention. As in ALL.DO...UNTIL loops, it is **essential** to have a "getout" (otherwise there is not way of stopping the program continuing and it would be necessary to do a HARD RESET to get out!). This is contained in the lines which start "IF.." and finish with "ENDIF". What this bit does is to look at the entry we have made in response to the "Title,Inits,Name" prompt. If, instead of entering the next Title, Initials and Name (e.g. "Mr F Smith") we instead just pressed the EXE key, then the LEN (length) of our input would be 0 (zero) and this would STOP the procedure and return us to the top level menu (at this stage). Notice also the use of KSTATs. The KSTAT 3 will make the Organiser operate as if we have pressed the SHIFT and NUM keys (our entry will be numeric). This is because it is better to enter telephone numbers as all numeric. However, the field is still ALPHA, so it is still possible to enter "Nottingham 735482" by using the SHIFT key for the "Nottingham" bit.

The KSTAT 1 puts the mode back to ALPHA CAPS for the next entry.

Before I finish this month's offering, I must say that I have been "underwhelmed" by your response to this series. Is anyone out there actually following this?!

Small Ads

Psion Printer c/w Mains Lead,
Paper & Manual
all in perfect condition
£150
Tel: **N Poskett on:**
Guildford 36448 eves/weekends
or Guildford 68161 ext 2456 daytime

Small Ads may be submitted by Members and will be inserted free of charge as a service to all members. This does NOT apply to Commercial Advertising, for which special arrangements must be made. Please contact the Editor for details.

Feedback

...Speaking of simplicity, the FSIZE program in last months newsletter can be reduced to;

```
LOCAL F$(10)
INPUT F$
OPEN F$,A,X
PRINT RECSIZE
CLOSE
```

and the Load and Store Menu program used machine code routines almost twice as long as required, I hope this is not the start of a bad trend. Part of the attraction of the Organiser is its simplicity, lets not make it more complicated by employing over complicated routines.

Along the same theme, I recently wanted to type a few letters at work to download back into WordStar at home and thought I would use the Organiser.

A program such as IPSO's AutoScribe would be no use because of the editing required to remove the formatting codes and so I ended up just using the Organiser's normal PROG function and regarding the letters as programs. Obviously no TRANSLATION should be applied (in fact the TRANS option can be removed from the menu via a bit of machine code) and the produced letters can be exported with no problems and can even be able to be printed from the Organiser. I now use this magic little editor for all sorts of uses apart from programming.

(to be fair to AutoScribe, it should be mentioned that it is capable of producing "export" files, with no "formatting codes" for the express purpose of being compatible with other machines. Ed)

OPEN LETTER TO MACHINE CODERS;

Does any one know where the memory area for editing block files (such as programs) is, the technical reference is a little vague on the subject.

Neil Draycott

Progs & Procs - 1

The Challenge

Last month I issued a challenge for programmers to match a tiny but effective program which I had written and used for some time on the Hewlett-Packard 41c Programmable Calculator/Computer. One or two members phoned me to ask if this was a joke - they said that the proposed program was "very easy" to write and hardly a serious challenge. Well, those members must have subsequently realised that the apparently simple program was not quite so simple after all, as neither of them came forward with a workable program. In fact I have only had three programs, all totally different in approach, which would do what I asked.

Let me first remind you what was required and then we will look at the solutions submitted. What the H-P program did was to allow me to simply add columns of figures by the simple process of keying in the figures - any figures - and nothing else. The use of PSE (pause) on the H-P, during which the keyboard was still "live" and capable of accepting keystrokes, was the key element in the program. I did say that the program should be capable of dealing with ANY number, (from a single digit to a complex real number with decimals). To make this possible the H-P automatically EXTENDS the pause as each digit is added. Although only one of the submitted entries met this point, nevertheless all three programs have each got something to recommend them and all three will do the job required.

It just so happened that I had a one-off job which largely consisted of adding and checking a complete ledger of some 100 pages with 66 entries on each page, so I was able to give each program extensive "field testing". At the end of this period, I decided that the best tool for the job was still the H-P, but the reason was not the program, but rather the superior keyboard which has widely spaced numeric keys with a positive tactile feedback which makes left-handed working by touch very easy.

Here are the three programs. I am listing all three, because they show how this task was approached by three expert OPL programmers. As I said above, all three are useable, and the program has real practical value as you will see if you have to total masses of figures.

Comp

by Frank Cook

Author's Notes: The time delay may be increased or decreased to taste by adjusting the figure for t%. t%=350 is just about right for the Author.

```
comp:
LOCAL n$(12),s,t%
DO KSTAT 3
  DO BEEP 50,4000 :CLS :n$=""
:t%=350
  WHILE t% :t%=t%-1
:n$=n$+KEY$
  AT 1,1 :PRINT CHR$(126);n$
  ENDWH
  ONERR x:: :s=s+VAL(n$)
  x:: ONERR OFF
  UNTIL n$="" :CLS :PRINT "=";s
UNTIL GET<>13
```

Autoadd

by David Green

Author's Notes: After a number is entered, the total is displayed indefinitely until next entry. Negative numbers are accepted. BEEP is not really necessary (I found that is was. Ed). Press <EXE> once to exit program. Suitable for both all models of Organiser.

```
autoadd:
LOCAL
a,b,h,h2,k$(1),m,mm2,n$(100),s,s2,t
KSTAT 3
l1:
PRINT "  ADD:",
PAUSE 0
h=HOUR
m=MINUTE
s=SECOND
l2:
h2=HOUR
mm2=MINUTE
s2=SECOND
t=h2*3600+mm2*60+s2
IF t-h*3600-m*60-s<3
  REM adjust to suit yourself
  k$=KEY$
  IF ASC(k$)=13
    RETURN
```

(continued on next page)

Progs & Procs - 2

```

ENDIF
  PRINT k$;
  n$=n$+k$
  GOTO 12::
ENDIF
IF n$=""
  RETURN
ENDIF
a=VAL(n$)
b=b+a
AT 1,2
PRINT "Total:",b
BEEP 100,200
PAUSE 0
n$=""
CLS
GOTO 11::

```

Plus

by Nick Frank

Author's Notes: This program includes a couple of embellishments (mostly borrowed from my RESULT program). The calculation total is displayed permanently and not just when no number is keyed in. Percentages are allowed using the % key. The number of entries is counted and may be displayed by pressing EXE. You can go back and edit the last entry by pressing the Up Cursor key (just in case you hesitated too long when entering the number). If you enter an illegal number (e.g. two decimal point), the display will flash until you put it right. Probably more difficult was deciding on the length of pause before the automatic addition takes place. After trying several alternatives, I settled for half a second, plus an additional 1/20 second for every digit in the number. This can, of course be adjusted by the user. I also decided that there were five situations where it was useful to have an indefinite pause (a PAUSE 0 in the programming languages). These were:

- If no number has been entered yet (obviously)
- If the last key pressed was any of the following:
 - a decimal point
 - the Up Cursor key (initiating an Edit)
 - the DEL key (which deletes the digits in the number, one character at a time).
 - the % key (to allow the user time to note what the percent figure is).

The above is achieved by the line in the program:
 [PAUSE (10+LEN(input\$))*(input\$>"
 AND RIGHT\$(input\$,1)<>". " AND k%<>3
 AND k%<>8 AND k%<>37)]

```

plus:
[LOCAL INPUT$(14),exinput$(14),
total,exttotal,n,c%,flag%,k%]
KSTAT 3
ESCAPE OFF
start::
CLS
AT 1-(total>=0),1
PRINT GEN$(total,15);CHR$(8);
IF c%
  AT 1,2
  PRINT"+";INPUT$;
ENDIF
loop::
[PAUSE (10+LEN(INPUT$))*(INPUT$>"
AND RIGHT$(INPUT$,1)<>". " AND k%<>3
AND k%<>8 AND k%<>37)]
k%=KEY
a::
IF k%=0
  b::
  ONERR error::
  n=VAL(INPUT$)
  ONERR OFF
  BEEP 10,1000
  exttotal=total
  exinput$=INPUT$
  total=total+n
  c%=c%+1
  INPUT$=""
  flag%=0
  GOTO start::
ENDIF
[IF (LOC("1234567890.E-",CHR$(k%))
OR (LOC("+",CHR$(k%))AND RIGHT$(IN-
PUT$,1)="E")) AND LEN(INPUT$)<14]
INPUT$=INPUT$+CHR$(k%)
PRINT CHR$(k%);
GOTO loop::
ENDIF
IF k%=1
  IF LEN(INPUT$)
  INPUT$=""
  ELSEIF total=0
  RETURN
  ELSE total=0
  exttotal=0
  exinput$=""
  c%=0
  ENDIF
  GOTO start::
ENDIF
[IF k%=3 AND INPUT$="" AND flag%=0
AND c%>1]

```

(continued on next page)

Progs & Procs - 3

```
INPUT$=exinput$
total=extotal
c%=c%-1
flag%=1
GOTO start::
ENDIF
IF k%=8 AND LEN(INPUT$)
INPUT$=LEFT$(INPUT$,LEN(INPUT$)-1)
PRINT CHR$(8);" ";CHR$(8);
GOTO loop::
ENDIF
IF k%=13 AND INPUT$=""
AT 1,2
IF c%=1
PRINT"1 entry";
ELSEIF c%>1
PRINT c%,"entries";
ENDIF
GET
GOTO start::
ENDIF
IF (k%=13 OR k%=43) AND LEN(INPUT$)
GOTO b::
ENDIF
IF k%=37
ONERR error::

[ INPUT$=GEN$(VAL(INPUT$)*total/100,
14) ]
ONERR OFF
GOTO start::
ENDIF
GOTO loop::
error::
ONERR OFF
DO
PRINT
CHR$(15+(c%=0));CHR$(43+11*(c%=0));
PAUSE -10
PRINT INPUT$;
PAUSE -10
k%=KEY
UNTIL k%
GOTO a::
```

After much deliberation, because all three programs are good, I decided that **Frank Cook's** program wins my prize. The program is compact and well-written, but more to the point it does exactly what I asked for and no more. I thank the other two authors for their submissions. I do not apologise for printing all three listings - they warrant further study by any up and coming programmers.

Small Ads

LZ64 Organiser c/w Handbooks - £95 (incl p & p)
Phone Mike O'Regan on (0602) 735482 anytime

LZ/32k Model with leather case & manuals £100
Panasonic P3131 daisywheel printer with ribbons
and spare daisy wheels - £200
Paltex 9 pin printer with serial interface £45
Apply to: **David Gray**
39 Twyford Avenue
Acton
London W3 9PZ
Tel: 081 993 5702

XP Organiser, Psion Printer, Datapak Formatter
Comms Line, Datapaks (1x128k, 2x32k,4x16k)
Mains Adaptor, Manuals Cost £640
Price £350
Phone: **John Ward on 0239 820 706**

Readers Letters

Dear IPSO,
As a Psion LZ64 owner, I have made several small structural/nautical programs in OPL. However, to expand to the more complicated computations, I need matrix notations and easy access to matrices (2 dim arrays).

So far I understand the lack of 2 dim arrays to be a major setback of OPL. The MATHS PAK has matrix computations, although seemingly not accessible in a program.

If there is anybody who knows of a simple method to solve these two problems, would they please let me know

G.J. van Ommen
Ter Borch 8
9472 RB Zuidlaren
Netherlands
Phone: (0)5905 4099

News & Reviews - 1

Transform DTMF Dialler

One of the "gimmicks" built-in to the Atari Portfolio was the DIAL facility, which allowed the computer to dial any touch-tone telephone. I say "gimmick" because this facility was quite limited - the number to be dialled had to be part of the first field of any database, and the feature did nothing else but just dial the phone, albeit working perfectly every time.

There have been two telephone diallers available for the Organiser for some time now. One of these has recently been discontinued (for technical reasons, I think). The one remaining is the Transform DTMF Dialler, and I have had one on evaluation for a few days now.

This device plugs into the top port of ANY Organiser and is styled to match the Organiser so that it looks like part of the machine. Transform can now supply an extended case so that the DTMF Dialler can be kept plugged in. There are many advantages to this, as you will see.

The DTMF Dialler certainly does dial phone numbers for you - but it does much more. The self-contained unit features quite a lot of very handy file-handling additions, a phone-call timer and logger, OPL extensions, access to Home Banking, quick dialler for 10 most used numbers, etc.

File Handling

All the features work with existing files, either MAIN or any other filename, on any device. When using the auto-dial facility, the program very cleverly finds the field containing a telephone number no matter what its position in the record is (if there is more than one telephone number, it uses the first one). Any file may be allocated a set of 3 character "prompts" (shortened field names) and once these are allocated they are used by the program to refer to the fields. The SEARCH facility allows you to use just one field of your choice, or do the usual Organiser global search (which uses all fields). This is an excellent timesaver. The whole file may be reordered in ascending or descending alpha order on any field of your choice. This process is quite fast and does not use excessive memory in the process. When using a file, it is possible to go very quickly to the FIRST record, LAST record, or back and forth one record at a time. Records can be ADDED or EDITed directly from this mode. These facilities have to be tried to appreciate them in full.

Call Timing & Logging

This facility enables you to (optionally) time and/or log any call you make. The unit price can be adjusted, if necessary. When your call is answered, then a simple key press starts the timer, which displays the elapsed time and the cumulative cost of the call. When the call is finished another single key press stops the timer. If you have previously CREATED a file called DTMFLOG (on any device), you have the option of keeping a log of all your calls, detailing who you phoned, when, exactly how long you were connected, and the cost of the call. This can be invaluable if you are in any business where it is necessary to charge calls to the client. The program uses the Organiser's clock/calendar to apply the appropriate rate for the time of day and day of week.

OPL Extensions

Three basic extensions to OPL are included with the DTMF Dialler. They are TONE:, DIAL:, and PARSE\$. TONE:("123") will produce the dial tones for 123. The DTMF Dialler can generate tones for 16 different characters - digits 0-9, letters A,B,C,D, Hash(#) and Star(*). DIAL:(anyold\$) will try to extract a recognisable number from "anyold\$", and dial this number (with the option of including a pause in the dialling). Thus DIAL:("call Fred on 123") will generate the tones for 1,2,3. The PARSE\$ function takes a string of characters and returns the string that would be dialled using the DIAL function. It is useful as a quick check that the string actually contains a "diallable" number.

Mercury Code.

The DTMF Dialler has a facility to store in encrypted form, the codes needed to dial a number via the Mercury network. When used, it automatically decrypts the number to dial, but will not display the number, thus maintaining security.

Connections

There are basically two ways of actually using the unit to dial calls. It is possible to simply hold the speaker close to the telephone handset (I have found that this works every time with standard phones, but will not work with my cordless one). The alternative is to use the (optional) plug-in speaker extension which you can attach to your phone, leaving your hands free.

News & Reviews - 2

Documentation

A very comprehensive 58 page pocket-size manual is provided, but I found that, as with all good software, the various modes of operation soon become second nature - the screen displays are all helpful - and the handbook can soon be left on the shelf (with all those other manuals which you may wish to refer to from time to time)

The DTMF Dialler is available direct from:

Transform Ltd
7c Station Approach
Hayes
Kent
BR2 7EQ

Telephone: 081-462-4666 Fax 081 462 3971

Prices are: DTMF Dialler £69.95
Extended Leather Case (Organiser/Dialler) £12.95
Speaker Extension £9.95

HB Games Pak

We have reviewed several Games Paks from time to time and here is another. However, this is no ordinary games pak. All the games on the HB Games pak have been written totally in Machine Code, so that, at last we can have some really competitive arcade games which run very smoothly and efficiently on any Organiser (yes even the CM and XP versions are highly competitive!).

The games, not surprisingly, are already well-known to enthusiasts on other machines. What makes them special is that HB have actually managed to make them work on the Organiser's tiny keyboard (with continuous pixels, they would have been even more impressive, but alas, the Organiser's display is split into 8x5 matrices).

There are six games to choose from - Defender, Breakout, Worms, Snake, Shootout, and Mind.

Defender is a splendid implementation of the quick-fire arcade game. On the LZ it is uncannily reminiscent of the game of the same name which was very popular on the BBC micro a few years ago. You fly your jetplane over rough terrain, seeking out and shooting down aliens. You can speed up or slow

down your plane and even change direction. The only feature missing from the original is the tiny men bailing out. This is a very competitive game.

Breakout is the game where you knock bricks out of a wall by bouncing a ball against them. This is perhaps the most difficult game on the pak and takes real skill. It is also a bit hard on your eyesight, as the ball is TINY.

Worms is a delightful game, with segmented worms crawling at high speed up and down the screen. You must shoot all segments before any of them reach your position. This takes real timing skill.

Snake is a rather unusual dice throwing game. It takes quite a bit of practice to get the hang of the game. The only criticism I have is that the dice are a bit difficult to read.

Shootout is a real favourite with children. The Organiser becomes a shooting gallery where two lines of animals come under your gunsights. When you hit one, it disappears with a suitable sound. You have a limited number of shots to dispose of screens full of, first, Ducks, then Teddybears and Dogs. You have to be very accurate with your shots. I found it easier to work with the sound switched off.

Mind is a neat version of Mastermind (or Cows & Bulls).

High score tables.

The pak maintains a file of the highest scores achieved in each game. I tried to edit the file (to fiddle the scores!) and found that even the scores are in machine code, making fiddling impossible.

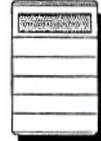
HB Games Pak is the best arcade games pak I have seen so far for the Organiser. It is available from:

HB Consultants
Freepost
Sawbridgeworth
Herts
CM21 9BR

Price £29.95



IPSO FACTO



The Newsletter of the International Psion Organiser User Group

Volume IV . No. 9 . November 1990

Editorial Office: 130 Stapleford Lane . Toton . Beeston . Nottingham . NG9 6GB . UKPhon: (0602) 735482 .

Another Competition

The last challenge was so well taken up that I have decided to have another competition. On the Progs & Procs pages of this issue you will see a version of the Towers of Hanoi. As it stands, the program works quite well, but, with a bit of ingenuity, I am sure that some of our ace programmers can produce other versions - perhaps making full use of the LZ four-line display.

So the challenge is this. To write a version of the Towers of Hanoi which not only let you play, but also let the Organiser display its skills in solving the problem in the least possible moves.

I will again give a small prize to the best program, and print any which show especial merit.

The Towers of Hanoi Legend

There is a monastery just outside Hanoi where the original rings and posts reside. Reports on the number of rings vary, but the best information is that there are 64. The monks have the job of moving the rings from one post to the other, observing the strict rule that no ring may be placed on top of a smaller one. The rings are moved at a constant pace, day and night, without a pause and, when the task is complete, it is the end of the world!

The deadline for the above competition is 31 December.

Incidentally, may I say thank you to a couple of late contributors to the previous challenge who sent in entries rather too late to be considered for a prize.

PSITALY Takes Off

One of our Members has started a Club for Italian Organiser Users. Called PSITALY, the Italian Psion Organiser Club, it will feature a monthly newsletter in the Italian language.

Any Italian members (or anyone else) who are interested in this venture should contact the Editor:

Renato Buzzi
Via Filadelfia 200
10137 TORINO
Italy

Flash EPROMS for Organiser

PSION have announced that the successful FLASH EPROM storage systems, introduced with the MC Range, are now available for the Organiser. These devices are a "cross" between RAMPAKS and DATAPAKS and feature EEPROMS (Electrically Erasable Programmable Read Only Memory). More news next month

Contents

Editorial	66
Procs & Progs.	67
Beginners Page.	70
Feedback.	72

Progs & Procs - 1

CHDIR & DELOFF

by Neil Draycott

This program continues the theme started in previous months of loading and saving the Organiser main menu, but by first 'stripping' the main menu the effect of sub-directories can be created on the Organiser.

Anyone familiar with the operation of a disk based PC will know how useful the concept of SUB DIRECTORIES is, all the files or programs relating to a subject can be stored in a sub directory which has a name relevant to the subject.

The following program attempts to simulate this utility by loading program (or in the case of the LZ model, FILES and NOTEBOOKS) names into the main menu after first stripping the main menu of it's existing names. In this manner 'sub directories' of program names (or file names or notebook names) can be loaded into the menu, without the menu becoming cluttered, by simply naming the 'sub directory' file.

The program has been structured into machine code routines separated by OPL commands, the machine code sections are complete utilities in themselves.

INSTALLING THE PROGRAM.

Type in and translate the program, mistakes can crash your Organiser so be careful.

Prepare the sub-directory files. For an XP model a simple program to create and append a data file will need to be written, for an LZ model use the 'XFILES' option.

The sub-directory files (a separate file for each sub directory) should contain a series of two fields, the name to be inserted and the type of insertion; for an XP the type is always 0 (OPL) but for an LZ model either 0 (for OPL), 1 (for a data file) or 2 (for a notebook file) can be used.

Psion program names cannot be used. The names are recorded in the order they are required in the menu, the loading program will reverse the file. One file should contain no records (called 'ROOT'?).

The 'CHDIR' program will strip the main menu until it finds 'OFF', by moving the OFF function part of the menu can remain at all times, this is useful because the

Psion program names cannot be inserted and so if OFF is moved to ahead of the Psion names then only the part of the menu before the Psion functions will be affected.

To move the 'OFF' function use Mike Davies 'DELTLM' program (page 47) or my 'DELOFF' program, listed below to delete the OFF and then manually insert it back in the menu where required.

USING THE PROGRAM

Select 'CHDIR' from the main menu (inserted to the right of OFF?). Name the directory file (using MODE to change Pak). Entering no name or pressing CLEAR will abort the program but with the default drive set as displayed (like the previously published 'GRABBA'GRABB' etc.)

The main menu will then be stripped down to 'OFF' and the names recorded in the selected file inserted. It all happens very quickly.

ABOUT THE PROGRAM

The machine code sections contain several interrupts, the first A%() array is a self sufficient routine to name a file and allow Pak selection by pressing MODE. The others contain calls to compare two buffers (to find 'OFF') and to delete and insert menu names. I have not yet managed to insert Psion names, any ideas?

Instead of using data files to contain the names to be inserted an alternative is to use separate programs to load each sub directory and keep the program names inserted to the right of 'OFF', like having all your sub directories displayed in the ROOT directory of a PC.

```
chdir:
LOCAL a%(11),p$(5),f$(8),x%
ESCAPE OFF
p$="CHDIR"
a%(1)=$7f21 : a%(2)=$874f
a%(3)=$01ce : a%(4)=ADDR(p$)
a%(5)=$3f68 : a%(6)=$ce21
a%(7)=$8739
f$=USR$(ADDR(a%),8)
IF LEN(f$)
  IF NOT EXIST(f$)
    AT 1,2
    PRINT"< FILE MISSING >";CHR$(16)
    PAUSE 15
    STOP
ENDIF
```

(cont. on next page)

Progs & Procs - 2

```
ELSE
  STOP
ENDIF
p$="OFF"
a%(1)=$fe20 : a%(2)=$02cc
a%(3)=ADDR(p$) : a%(4)=$dd41
a%(5)=$8603 : a%(6)=$c603
a%(7)=$3f72 : a%(8)=$2704
a%(9)=$3f67 : a%(10)=$20eb
a%(11)=$3900
USR(ADDR(a%()),0)
OPEN f$,a,n$,a%
LAST
DO
  p$=a.n$
  POKEB $2187,LEN(p$)
  x%=1
  DO
    POKEB
    $2187+x%,ASC(MID$(p$,x%,1))
    x%=x%+1
  UNTIL x%>LEN(p$)
  POKEW $2187+x%,a.a%
  a%(1)=$3f65 : a%(2)=$3900
  USR(ADDR(a%()),0)
  IF POS=1
    CLOSE
    STOP
  ENDIF
  BACK
UNTIL 0
```

```
deloff:
LOCAL a%(4),p$(3)
p$="OFF"
a%(1)=$01ce : a%(2)=ADDR(p$)
a%(3)=$3f67 : a%(4)=$3900
USR(ADDR(a%()),0)
```



Don't keep tearing your hair out - there was an error in Peter Houppermans' FDUMP program in September (Peter's not mine!). The seventh line from the end of the program should read:

```
[buffer$=LEFT$(buffer$,a%-1)
+CHR$(124)+RIGHT$(buffer$,
LEN(buffer$)-a%)] :REM SORRY
```

The FSIZE Saga

Many people wrote to me to point out that Neil Draycott's "simplified" FSIZE procedure was nothing of the sort! In fact, all Neil's procedure did was to give the size of the first record in the database.

One member did something about it - he sent me a program which adds bits to Neil's program. The result is a procedure which gets the sums almost right:

Jsize

by Marc Worsfold

```
jsize:
LOCAL f$(10,c%
AT 1,2 :PRINT "File: ";
INPUT f$
OPEN f$(a.x
c%=RECSIZE+11
DO
  AT 2,3 :PRINT c%
  c%=c%+RECSIZE
  AT 7,3 :PRINT "Bytes"
NEXT
UNTIL EOF
AT 2,4 :PRINT COUNT
AT 7,4 :PRINT "Records"
CLOSE
GET
```

Although this program is still smaller, it still doesn't give the same results as FSIZE. I wonder if anyone can explain the technical differences of the two programs?

Towers of Hanoi

by M.J. Siddiq

This is an Organiser version of the old "Towers" puzzle. It was written for the XP, but will run on the LZ (if your eyes can stand the strain). For the uninitiated, the object is to move "rings" of different sizes between "pegs". The game starts with 5 "rings" on peg A. You, as the player, are required to move all 5 rings onto any other peg (B,C, or D) in the least possible moves. The only stipulation is that you are not allowed to place a large ring above a small ring during the process. The author says that his best score is 81 moves. Good luck

Progs & Procs - 3

```

hanoi:
GLOBAL a%(8),b%(8),c%(8),d%(8)
LOCAL g%,pa%,pb%,pc%,pd%,go%,w%
KSTAT 1
a%(8)=31 :b%(8)=31 :c%(8)=31 :d%(8)=31
a%(6)=31 :a%(5)=15 :a%(4)=7 :a%(3)=3 :a%(2)=1
pa%=1 :pb%=6 :pc%=6 :pd%=6
CLS
AT 1,2 :PRINT"a b c d";
AT 9,1 :PRINT"Go:";
go%=-1
DO
  go%=go%+1
  AT 12,1 :PRINT go%;
  AT 9,2 :PRINT "Wait...";
  move:
  AT 1,1 :PRINT CHR$(0),CHR$(1),CHR$(2),CHR$(3);
  IF d%(6)=31 AND d%(5)=15 AND d%(4)=7 AND
d%(3)=3 AND d%(2)=1
    w%=1
    GOTO tst::
  ENDIF
  inp::
  AT 9,2
  :PRINT"Ready <";
  g%=GET
  IF g%=1
    g%=GET
    IF g%=1
      GOTO end::
    ELSE
      GOTO inp::
    ENDIF
  ELSEIF CHR$(g%)="A" AND pa%<6
    IF pb%<6
      IF a%(pa%+1)>b%(pb%+1)
        GOTO inp::
      ENDIF
    ENDIF
    AT 16,2
    :PRINT "A";
    b%(pb%)=a%(pa%+1)
    pb%=pb%-1
    a%(pa%+1)=0
    pa%=pa%+1
  ELSEIF CHR$(g%)="B" AND pb%<6
    AT 16,2
    :PRINT "B";
    g%=direct:
    IF g%=1
      GOTO inp::
    ELSEIF g%=5
      IF pa%<6
        IF b%(pb%+1)>a%(pa%+1)
          GOTO inp::
        ENDIF
      ENDIF
      a%(pa%)=b%(pb%+1)
      pa%=pa%-1
      b%(pb%+1)=0
    ELSEIF g%=6
      IF pc%<6
        IF b%(pb%+1)>c%(pc%+1)
          GOTO inp::
        ENDIF
      ENDIF
      c%(pc%)=b%(pb%+1)
      pc%=pc%-1
      b%(pb%+1)=0
      pb%=pb%+1
    ENDIF
  ELSEIF CHR$(g%)="C" AND pc%<6
    AT 16,2 :PRINT "C";
    g%=direct:
    IF g%=1
      GOTO inp::
    ELSEIF g%=5
      IF pb%<6
        IF c%(pc%+1)>b%(pb%+1)
          GOTO inp::
        ENDIF
      ENDIF
      b%(pb%)=c%(pc%+1)
      pb%=pb%-1
      c%(pc%+1)=0
      pc%=pc%+1
    ELSEIF g%=6
      IF pd%<6
        IF c%(pc%+1)>d%(pd%+1)
          GOTO inp::
        ENDIF
      ENDIF
      d%(pd%)=c%(pc%+1)
      pd%=pd%-1
      c%(pc%+1)=0
      pc%=pc%+1
    ENDIF
  ELSEIF CHR$(g%)="D" AND pd%<6
    AT 16,2 :PRINT "D";
    IF pc%<6
      IF d%(pd%+1)>c%(pc%+1)
        GOTO inp::
      ENDIF
    ENDIF
    c%(pc%)=d%(pd%+1)
    pc%=pc%-1
    d%(pd%+1)=0
    pd%=pd%+1
  ELSE
    GOTO inp::
  ENDIF
  tst::
UNTIL w%=1
AT 9,2 :PRINT"You win",
GET
end:: CLS
PRINT"> End of game <"
GET
(Procedures continued on Page 71)

```

Beginners Pages - 1

by Mike O'Regan

Handling Data on the Organiser - Part IV

Before we plunge into the next stage with our database handling programs, here is a bit of handy information about what we have done so far.

Our OPL procedures have been used to CREATE, OPEN, and then enter records in a database called "MEMBER". However, owners of CMs and XPs have no way of accessing the data which they may have entered using dbenter:

Now some of the more knowledgeable among you may know that, if we had used the file name MAIN (instead of MEMBER), then the data entered could already be accessed using the FIND option from the Organiser's main menu. Another advantage would be that there is no need to CREATE a file called MAIN - it already exists!

The only drawback to using MAIN is that you may already be using it for other data, and, if this data is not in the format required (with the first three fields as ALPHA, ALPHA, and NUMERIC), then this could cause problems later when we introduce simple "number crunching" to our database.

Of course LIZZY owners could access the data in MEMBER by using XFILES and OPENing the file.

So, for now, we will stay with the file name MEMBER, so that there is no conflict with other data in the Organiser.

I have had some feedback from readers to the effect that I should give more detailed information about the various OPL procedures as they are introduced, so that there is no doubt what they do and how. This is a good idea and I will try to give as much background as possible.

I trust that the explanations so far for the procedures dbstart: and dbenter: were sufficient - at least no-one has asked for more information.

By the way, please notice that I am using lower-case (small letters) in the procedure names. This is so that they comply with IPSO's procedure writing protocol. In our procedures only OPL words are printed in Caps - all others are in lower-case to distinguish them from OPL words.

One of the excellent features of OPL is the easy way it uses MENUS. In case you haven't used a menu before

here is a short explanation.

The menu (just like its counterpart in a restaurant) is a list of options from which it is possible to choose one by either moving the cursor to it or by keying in its initial letter. For the latter, it helps if each option in the menu starts with a different letter of the alphabet, so that the option only needs a single key press to access it.

So our next OPL procedure will be concerned with setting up a menu to control the individual procedures which will make up our complete PROGRAM. Note that this program can be added to at a later date, if we decide to add further items to the menu as our collection of procedures increases. So now to the procedure, followed by an explanation.

```
db:
LOCAL option%,yn$(1)
[option%=MENU("Create,Add,Delete,-
Erase File,Quit")]
IF option%=1
  dbstart:
  ELSEIF option%=2
  dbenter:
  ELSEIF option%=3
  dbdel:
  ELSEIF option%=4
  Print "Are you sure";CHR$(63)
  INPUT yn$
  IF UPPER$(yn$)="Y"
    DELETE "a:member"
  ENDIF
  ELSEIF option%=5
  STOP
ENDIF
```

Notes.

1. db: is a short name, so that it doesn't take up too much room on the top menu. It is also the initial letter of ALL the procedures it controls. This is good practice and could well be copied by all programmers. It makes it very easy to recognise which procedures belong together, especially if you have many other procedures either in memory or on a d-tapak.

2. The LOCAL line declares the two variables needed in this procedure ONLY (hence LOCAL - not needed in any other procedure). Option% has the % to indicate that it is an INTEGER VARIABLE (in other words no decimal places, as we will only use this to access the numbers 1-5). The yn\$(1) is a STRING VARIABLE (alpha). The (1) indicates that it will use

Beginners Pages - 2

only ONE character - either Y or N to indicate Yes or No.

3. Line 3 is the main line in the program. It will set up and display a menu containing all the words which are between the () and double-quotes ". Note that each item is separated by a comma, and that items need not be restricted to single words (although single words are best so that very long menus - which cannot fit in a single display - are avoided). Also notice that option 4 - (-Erase File) has "double-protection" to avoid erasing your file accidentally. The first protection is the minus sign (-) before Erase. This means that the option cannot be accessed by entering -. It is necessary to use the cursor keys to access this option, thus making it different from the other 4 options. The second protection is described below. The last thing to note about our menu is that it contains a QUIT option. This is almost invariably desirable, if not essential, to allow you to reject all other options and get out of the procedure with no action having taken place.

4. The rest of the procedure is concerned with accessing the option chosen from the menu. Notice that the numbers 1-5 (after IF option%...) apply to the number of items in the menu. Also notice that item 3, dbdel: DOES NOT YET EXIST! Any attempt to choose this option will be met by a "Missing Procedure" message from the Organiser. This doesn't stop us, however, from including it in the menu - we will be writing a procedure with this name later.

5. Option 4 needs a little explanation. This is the second protection referred to above. Erasing a file is a serious business, so you are asked to confirm that you really want to take this drastic step by the words "Are you sure?" The CHR\$(63) will print a question mark (?). In reply to this question you are only allowed to enter a single letter. Unless this letter is "Y" (for Yes) the file will not be deleted. The UPPER\$ is a handy device which will ensure that the "Y" or "y" is in UPPER case, so that it will work as planned. 6. This procedure contains a "nested loop" (3 lines starting "IF UPPER\$...."). It is possible to have such a loop within a loop, provided it is finished off with its own ENDIF.

7. Option 5 - STOP - is self-explanatory. It STOPS the procedure and returns you to the top menu.

When you have entered, TRANSLATED and SAVED this procedure, insert it in your main menu by pressing <MODE> and keying in DB.

For Sale

Organiser XP - 3 yrs. old £40
Olivetti DM 100S Printer. Bought new June .90
Barely used since - original ribbon
c/w 2 ribbons Price new £171 Price now £85
**Ring DAVID Puglsey on 071 218 2407
between 8.30 and 18.30.**
All reasonable offers accepted

Pristine LZ Organiser
c/w handbooks £100 (ono)
Phone: David Berg on 071 249 2990

2 x 128k datapaks @ £50 ea
Phone: Terry Triffitt on 0742 375107

LZ64 Organiser c/w Denwood Leather Case £130
Paralink printer interface £30
AutoScribe Plus - £20
Pocket Spreadsheet II - £20
Finance Pak II - £20
CUBSOFT XBase - £10

Phone Jim Kirwan on 0772 614185

These are the two remaining procs for HANOI
(continued from Page 69)

```
move:
[pgd:(0,a%(1),a%(2),a%(3),a%(4),a%(5),a%(6),a%(7),a%(8))]
[pgd:(1,b%(1),b%(2),b%(3),b%(4),b%(5),b%(6),b%(7),b%(8))]
[pgd:(2,c%(1),c%(2),c%(3),c%(4),c%(5),c%(6),c%(7),c%(8))]
[pgd:(3,d%(1),d%(2),d%(3),d%(4),d%(5),d%(6),d%(7),d%(8))]
```

```
direct:
LOCAL x%
AT 9,2 :PRINT"Direct<";
DO
  x%=GET
UNTIL x%=1 OR x%=5 OR x%=6
RETURN(x%)
```

```
pdg:(x%,a%,b%,c%,d%,e%,f%,g%,h%)
POKEB $180,64+x%*8 :POKEB $181,a%
POKEB $181,b% :POKEB $181,c%
POKEB $181,d% :POKEB $181,e%
POKEB $181,f% :POKEB $181,g%
POKEB $181,h%
```

Feedback - 1

Why Do I Own an Organiser? 1

The answer to this is quite (to me) simple. But first a bit of background may not come amiss. My hobbies and work etc. seem to stem from an interest and involvement in the following:

Engineering - I build my own workshop steam-related products ranging from small table type engines to miniature locomotives, which range from 3.5" through to 2" gauge.

Electronics - this includes things not related to computers as such, although I did build from scratch a Nascom with many add-ons etc.

Amateur Radio - I have my own radio station with most of the equipment "home brew".

Greenhouse - I have a collection of cacti which seems to be growing (in both senses) and now numbers some 300 different plants.
Photography & full-size steam railways.

So now back to the question "Why did I buy an Organiser?"

At first as the cactus collections was getting larger, I wanted something pocket sized where I could store a list of all the plants, so that, when I went round the various nurseries, I could tell at a glance whether or not I had the plant in question.

Now as Norfolk is a "dead" area as far as electronic devices are concerned my search for a machine was, to say the least, a bit limited. I had seen such things as a "digital diary" (CASIO) advertised in the paper and being sold by Boots. This was looked at and I soon found that the machine was rather limited and crude for the price. I was told that a larger machine was available (to order only). The staff seemed to know nothing about these machines, with the same story at Tandy and Dixons.

After this it seemed that there were a number of machines about, but who could show you a range of them, and, what is more important, tell you about them. It was at this point that I was shown an Argus catalogue with a few machines listed, with a short blurb about them but not much information. Now, as Argus offer a 16 day trial with money-back guarantee and no hassle, this seemed the way out to me.

So off I went and came home with a SHARP IQ, only to find that, although you could store on it, it was not very easy to expand. Next I tried a CASIO and returned it, as it had a lousy keyboard and no "slots".

I was told by Argus that the only one left to try was the Psion, so came home with an XP. What a change from the others! It could do so much more. "Great" I thought "I will have one of these - pity about the 2-line display, though".

Next came a stroke of luck. Dixons had an offer of the LZ64 with free leather case and 64k datapak. So back goes the XP - money back - and across the road to Dixons. I told the man that I would like an LZ64 with the free goodies. Out it comes, 4 lines and all, and the man said I could see it working IF ONLY THEY HAD A BATTERY TO FIT IT! What a way to sell something! Needless to say, I told him what to do with his Organiser. I came home with my money intact.

What next? The answer was to "let my fingers do the walking" and I came up with a dealer in Norwich who not only stocked Psions (and batteries), but would show me how to work it. It seemed he had his own LZ. The result was a trip to Norwich, a nice chat, and a trial run on the LZ64. So I bought one with the "goodies", plus a mains adaptor, and £10 less than Dixons anyway. The trip was well worth while.

I am now amazed how much you can do with such a small machine, for I had the same capacity on my Nascom 1/2 but in a box about 18" x 9" x 9"!!

(By the way, re the Beginner's Page. Yes, this to me is most useful and I can understand what you are telling us (I might have to read it twice, as I am always in a hurry to try things out). Please keep on with the good work)

I have a Paralink which I use with a loaned Epson FX80 printer, but find it rather odd as it prints in a different format in MAIN database to XFILES or OPL. It seems rather a lot of money for a cable to plug the two machines together. I will have to write an OPL program to do my print format, as I must send control codes to the Epson to allow me to print my plastic pot labels with the correct spacing. The Paralink SKIP will not let me see what I am printing next, nor will PRINT, except "printing Record No..." However, I will eventually sort it all out.

Graham Hewitt, Great Yarmouth

Editor's Note: Perhaps this should have been called "HOW I bought my Organiser"!!

Feedback - 2

Why Do I Own an Organiser? - 2

I built my first computer out of ex-GPO relays to my own design about twenty-five years ago. It required two separate power sources and could add two numbers! But from then on I was hooked. The appearance of the Organiser many years later seemed to be the answer to all those dreams of a machine that was both powerful, programmable, and genuinely portable.

I work in a housing area to the north of Dundee as a Church of Scotland minister. Our congregational roll contains nearly 400 names, addresses, phone numbers and post codes. I also have a personal address list, as well as a limited "Yellow Pages" list of useful companies and contacts. I use the Organiser for at least three reasons:

First - of all, it enables me to gather together in one place (on my Amstrad PC 1640) every name and address I need and then to transfer them to the Organiser. This saves an amazing amount of time. When the 'phone rings and someone asks for a number, I now look in one place (the Organiser), rather than my diary or my address book or a list here or a file there. In fact, people often do 'phone because they know I have information to hand and they can't find their diary!

Second - the Organiser provides me with a genuinely portable machine. It really does go into your pocket and can be carried away from home. When I am visiting hospitals, I can now check the ward numbers of patients and also see if any of our church members are neighbours of the patient. Although I usually work from home and use the Word Processor on my PC every day, sometimes it is necessary to work in the church. One of our members is deaf and I like to provide her with a full manuscript of Sunday services each week. Now I can use the Organiser as a "text input" device away from home. No time need be wasted. I use the Notepad as a "built-in" text input and then transfer the file across to the PC later to tidy it up and print it out. I have tried AutoScribe, but don't really need its extra facilities.

Third - I use the Organiser for relaxation. I enjoy reading books of theology and Bible commentary! But a different discipline is required when thinking through the logic of a program (however simple) that will work on the Organiser. Once again the size and built-in programming facilities mean that this can be done away from home, on holiday, or travelling by train.

Other "portable" have name extra facilities, but the size and power of the Organiser make it a unique facility in the computing world.

Peter Humphris, Dundee

Editor's Note

These are just two of a most encouraging batch of letters I have had in reply to my question last month. I will be printing some more next month.

Wanted & For Sale

Wanted - Coms Link & Word Processor
(AutoScribe or Easywriter) For Psion XP
Will pay £20 each (plus postage)
for good condition items.

**Phone: Jon Harrington
0203 712 572 (Coventry)**

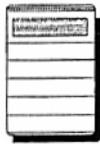
For Sale - Psion Organiser Developer
Boxed, new and unused.
Includes software on 5.25" disk for PCs & manual
Invaluable for writing Organiser software
£35.00

**Phone: Peter Humphris on
Dundee (0382) 458629**

Members Reply

After my little moan about lack of feedback from Members, I am pleased to report that I have had quite a spate of replies, especially from readers of the Beginner's Section. I have also had quite a few constructive suggestions for improvements, and these are always welcome. As the series on Data Handling develops, I would like even more requests for inclusion of options in the little database suite which is developing.

While on the subject, I would like to mention that there has been an increase of members (and even non-members) 'phoning me at all hours, sometimes with nothing specific to ask, but just to have a chat. Curiously enough it is not overseas members in other time zones (who might be excused). I don't mind talking to anybody in trouble at any time, but please try to keep your calls short, and, if possible at reasonable hours.



IPSO FACTO



The Newsletter of the International Psion Organiser User Group

Volume IV . No. 10 . December 1990

Editorial Office: 130 Stapleford Lane . Toton . Beeston . Nottingham . NG9 6GB . UK

A Merry Christmas & Happy New Year

*to IPSO Members
all over the World*

An Apology

I must apologise if this issue is late. A combination of circumstances have led to one delay after another and it is becoming obvious that this will be the latest you have ever received your copy. As many know, we had some power cuts, which made the Organiser itself the only workable computer during this period, and it was not possible to produce IPSO FACTO on the Organiser - in any case, I haven't got a battery-powered printer!

So, apologies again if you have been kept waiting, and I can only hope that I don't get any further holdups in future.

The NZ Organiser

I have just received a copy of a very professional new magazine which has started up for Organiser Users in New Zealand. It is called "*The NZ Organiser*" and will be published four times a year. The Editor is looking for contributors in New Zealand to make the magazine a success. The first (November) issue looks most promising.

Any New Zealand members who are interested should contact:

The Editor
The NZ Organiser
Tuatara Press Limited
3 Mallam Street
Wellington

UK Regional IPSO Groups

There are a number of regional IPSO Groups in the UK, (notably London and Swindon), but I have not had any news of them for some time. If the Secretaries would like to get in touch, I will be please to publish programs, etc, as in the past.

Contents

Editorial	74
Procs & Progs.	75
Beginners Page.	77
IPSO Review.	79
Feedback.	81

Progs & Procs -1

Satellite TV Calibration by A.J. Borthwick

Commercial broadcast satellites are usually to be found in geo-stationary orbit above the equator. They are kept in exactly the same spot above the Earth so their positions can be very accurately calculated in terms of Azimuth (bearing or direction) and Elevation (upwards angle), from a known point on the surface of the Earth.

The programs below will calculate these two coordinates for any geo-stationary satellite from any point on the surface of the Earth. Readers can obtain their position from an Ordnance Survey chart, available from any reputable bookshop, and Satellite positions, usually EAST or WEST, are published in most satellite TV magazines.

Now all IPSO FACTO readers can have a go at installing their own Satellite Dish Antenna systems.

```
geosat:
GLOBAL out$(16)
[LOCAL p,dec,lat,long,sat,k,gha,
lha,sine,elsun,tanf,cosc,el,az]
POKEB $180,72
POKEB $181,12
POKEB $181,18
POKEB $181,18
POKEB $181,12
POKEB $181,0
POKEB $181,0
POKEB $181,0
POKEB $181,0
p=PI/180
k=0.151650447
AT 1,1:PRINT "LATITUDE (DD.MMSS) "
AT 1,2:PRINT "Negative if SOUTH. "
AT 1,4:PRINT " ? "; : INPUT lat
AT 1,1:PRINT "LONGITUDE (DDD.MMSS)"
AT 1,2:PRINT "Negative if WEST. "
AT 1,4:PRINT " ? "; : INPUT long
AT 1,1:PRINT "SATELLITE (DDD.MMSS) "
AT 1,2:PRINT "Negative if WEST. "
AT 1,4:PRINT " ? "; : INPUT sat
AT 1,1:PRINT "+-----+"
AT 1,2:PRINT "| Processing Data |"
AT 1,3:PRINT "| Please Wait ! |"
AT 1,4:PRINT "+-----+"
lat=hmsdd:(lat)
long=hmsdd:(long)
sat=hmsdd:(sat)
gha=sat
```

```
IF gha>0
gha=360-gha
ENDIF
gha=ABS(gha)
lha=gha+long
IF lha>360
lha=lha-360
ENDIF
[sine=SIN(lat*p)*SIN(dec*p)+COS(lat*
p)*COS(dec*p)*COS(lha*p)]
[elsun=ATAN(sine/SQR(-
sine*sine+1))*180/PI
tanf=(SIN(elsun*p)-k)/COS(elsun*p)]
el=ATAN(tanf)*180/PI
cosc=(SIN(dec*p)-
[(SIN(elsun*p)*SIN(lat*p))]/(COS(elsun*
p)*COS(lat*p)))
[az=(-
ATAN(cosc/SQR(cosc*cosc+1))+1.5708)*
180/PI]
IF SIN(lha)>0
az=360-az
ENDIF
BEEP 100,100 : CLS
AT 1,1:PRINT " Satellite Position"
AT 1,2:PRINT "===== "
degout:(ddhms:(el))
AT 1,3:PRINT "EL ";out$;
degout:(ddhms:(az))
AT 1,4:PRINT "AZ ";out$;
GET

hmsdd:(x)
REM
REM
*****
REM degrees,minutes,seconds to deci-
mal degrees
REM
*****
REM
LOCAL y,z,s,t
IF x<0
s=-1
ELSEIF x>0
s=1
ENDIF
t=x*s
y=INTF(t)
z=y+INTF((t-y)*100)/60+((t*100)-
INTF(t*100))/36
RETURN(z*s)
```

(programs continued on next page)

Progs & Procs -2

Editor's Note: Lines which start with REM are for clarity only and should not be entered. Lines longer than a column line are marked by square brackets, [] which are not (and cannot) be entered

```

ddhms: (x)
REM
REM
*****
REM decimal degrees to degrees, minutes, seconds
REM
*****
REM
LOCAL y,z,s,t
IF x<0
  s=-1
ELSEIF x>0
  s=1
ENDIF
t=x*s
y=(t-INTF(t))*60
z=((INTF(t)*100+INTF(y)+(y-INTF(y))*60)/100
RETURN(z*s)

degout:(x)
REM
REM
*****
REM * degrees, minutes, seconds output *
REM
*****
REM
[outs$=FIX$(INTF(x),0,-4)+CHR$(1)+FIX$(INTF((x-INTF(x))*100),0,-3)+"'"]
[outs$=outs$+FIX$(((x*100)-INTF(x*100))*100,2,-6)+CHR$(34)]
RETURN
    
```

PSION

by Neil Draycott

The normal method of accessing Psion routines such as FIND, DIARY etc. is by searching the main menu until the required name is found and then jumping to the address stored just after the name in the main menu. This is fine as long as the routine name exists in the main menu and (in the case of the LZ model) is in the correct language.

To enable a much quicker and "safer" jump to Psion functions extra interrupts have been provided within the LZ operating system. These allow the user to invoke Psion routines as if they had been selected from the main menu. The following program will access all the functions from a single program called "Psion".

Type in and translate the following listing, THIS PROGRAM CAN ONLY BE USED ON AN LZ MODEL and will

crash an XP or CM model.

Delete all the Psion functions from the main menu (Diary, Month, World, Find etc.) and replace them with "Psion" inserted into the menu.

When "Psion" is selected, the top level menu is simulated with all the usual functions returned and can be used normally, pressing CLEAR will return to the actual main menu.

DO NOT run this program from the Prog function and then select "Prog" from the simulated main menu, the resulting loop will crash the system.

The Psion functions can still be inserted into the actual main menu if required. No names can be inserted into the simulated menu.

One option that cannot be entered in this manner is the CALC option. Although an interrupt exists to invoke the calculator unlike the other interrupts ALL operating system variables must be preserved before using it. I have tried calling the CALC option in this way with the D,SP,X and \$41-\$48 variables saved in RAM and it seems to work, but I cannot guarantee it.

```

psion:
LOCAL m%,a%(11),b%(11),c%(2)
ESCAPE OFF
a%(1)=168 : a%(2)=168
a%(3)=167 : a%(4)=164
a%(5)=159 : a%(6)=166
a%(7)=158 : a%(8)=167
a%(9)=175 : a%(10)=168
a%(11)=161
m%=1
DO
  b%(m%)=1
  m%=m%+1
UNTIL m%=12
b%(1)=4 : b%(2)=5 : b%(8)=3
m::
CLS
UDG 0,15,9,15,15,15,15,0,31
UDG 2,0,0,0,0,0,0,0,31
PRINT CHR$(0);REPT$(CHR$(2),14);
CLOCK(1)
AT 1,2
[m%=MENUN(2,"Find,Save,Diary,Time,Notes,World,Alarm,Month,Prog,Xfiles,Utils,Off")]
IF m%=0
  STOP
ELSEIF m%=12
  OFF
ELSE
  c%(1)=$3f00+a%(m%)
  c%(2)=$3900
  USR(ADDR(c%()),b%(m%))
ENDIF
GOTO m::
    
```



Beginners Pages - 1

Handling Data on the Organiser - Part V

Before we move on to write the next procedures in our db: suite (by the way "suite" is a word used to describe a collection of related programs or procedures), I would like to make a few alterations to our existing db: procedure. Some of these are to do with OPENing and CLOSEing files, and our present procedure could cause some problems with this later, so I have modified the procedure to make sure that we don't have such problems. Also, in keeping with current practice, I am exchanging the names "Delete" and "Erase". In our existing procedure, I have used "Erase" to get rid of the whole file, whereas it would be more correct to talk of "Deleting" it. Equally, it is more correct to have a procedure to "Erase" (or remove) a single record which may be no longer required, but leave the rest of the file intact. So, before going any further, edit your db: procedure so that it matches the following listing:

```
db:
LOCAL option%,yn$(1)
OPEN "A:MEMBER",a,name$,phone$,subs
st::
option%=MENU("Add,Find,Erase,-
Delete,Quit")
IF option%=1
  dbenter: :GOTO st::
ELSEIF option%=2
  dbfind: :GOTO st::
ELSEIF option%=3
  dbera: :GOTO st::
ELSEIF option%=4
  PRINT "DELETE FILE"
  PRINT "Are you sure";CHR$(63),
  yn$=GET$
  IF UPPER$(yn$)="Y"
    CLOSE
    DELETE "a:member"
  ENDIF
  GOTO st::
ELSEIF option%=5
  STOP
ENDIF
```

If you know how to do it, the easiest way is to EDIT your existing procedure. Bear in mind the following:

- a) Choose EDIT from the PROG menu
- b) You can move up and down the listing using the UP and DOWN ARROW keys.

- c) The LEFT and RIGHT ARROW keys move you along the lines
- d) DEL deletes the character to the LEFT of the cursor
- e) Any characters keyed in the will be INSERTED at the cursor position
- f) Pressing EXE will produce a new blank line above the cursor position

If you haven't tried EDITing before, try it now. The worst that can happen is that you make a complete mess of your existing procedure, in which case you can always DELETE it and rewrite the whole procedure. But try editing first!

Now a few explanations:

1. I have added the line "OPEN" to open the file for all following options, so that it is not necessary to have a similar line in each procedure. Remove this line from your benter: procedure, so that it now appears as follows:

```
dbenter:
DO
  CLS
  PRINT "Title,Inits,Name"
  INPUT a.name$
  IF LEN(a.name$)=0
    RETURN
  ENDIF
  PRINT "Telephone No:"
  KSTAT 3
  INPUT a.phone$
  PRINT "Subs amount",
  INPUT a.subs
  KSTAT 1
  APPEND
UNTIL 0
```

Notice that I have also added a CLS (clear screen) line which makes the program look a bit neater, especially when running on an LZ model.

2. "Create" has been deleted from the db: menu, as it is easier to keep this as a separate procedure (which is only used to create the file ONCE)
3. The menu has also been changed to include FIND, and get the DELETE and ERASE sorted out (as mentioned above)
4. An extra section has been inserted to access FIND, and the rest of the options have been renumbered.

Beginners Pages - 2

5. The st:: (line 4) is a LABEL, which simplifies the return to the menu after an option has been used. The only option which does not return to the Menu is "QUIT" (for obvious reasons).

The procedures which follow, which allow us to FIND and ERASE any record of our choice, are quite similar. It would be possible, therefore to write a single procedure which would do both jobs, but, for the sake of clarity (at this stage), we will keep them as separate procedures, and perhaps consider later how to combine them.

Here are the procedures:

```
dbfind:
LOCAL search$(15)
FIRST
PRINT "Search for:"
INPUT search$
DO
  CLS
  IF FIND(search$)
    PRINT a.name$
    PRINT a.phone$,a.subs
  GET
  ENDIF
NEXT
UNTIL EOF
PRINT COUNT,"records"
PRINT "NO MORE"
GET
```

This is a very simple program to allow us to enter a search clue, which the program uses to look for any record(s) containing this clue. This is similar to the FIND command from the top-level menu, but, of course, it allows us to examine records in our A:MEMBER file, whereas FIND can only examine the MAIN file. Those of you with LZ models can also use XFILES to OPEN A:MEMBER and then FIND (as with the top-level menu). However, our procedure has a couple of simple additions. First, it will tell us how many records were examined (the total number of records in the file), then it will indicate when there are no more records with the search clue. The line "FIRST" makes sure that the search always starts with the first record. The "GET", after the two PRINT lines, makes sure that the record can actually be read before moving on to the next one. "NEXT" moves on to examine the next record (and determine whether it contains the search clue or not). "UNTIL EOF" means keep doing the above commands UNTIL the EOF (end of file) is reached. The "COUNT" line always counts the number

of records in the file.

```
dbera:
LOCAL search$(15),yn$(1)
FIRST
PRINT "Search for"
INPUT search$
DO
  CLS
  IF FIND(search$)
    PRINT a.name$,a.phone$,a.subs
    PRINT "Erase (Y/N)",
    yn$=GET$
    IF UPPER$(yn$)="Y"
      ERASE
    ENDIF
  ELSE PRINT "NO MORE FOUND"
  GET
  ENDIF
NEXT
UNTIL EOF
```

Most of the comments on the dbfind: procedure also apply to dbera:. As you can see, the procedure has several additions (to enable us to choose whether or not to ERASE the displayed record). This is necessary if we wish to erase only selective records from a number which all contain the search clue. The only line which merits further explanation is "yn\$=GET\$". GET\$ is a single character string produced automatically by our next key press (without the need to press EXE). In this case pressing "Y" (or "y") will erase the record displayed. Pressing any other key will leave the record intact and move on to either the next or the end of file.

Next month we will examine just what has been achieved so far by our little suite of procedures and how we can add to them to give us some useful facilities for data handling which are not available if we solely use the built-in ones.

Attention Lizzy Thesaurus Owners

I have been sent a big program which plays HANG-MAN on the LZ, and uses the SPELL CHECKER/ THE-SAURUS as the source for words. It also has graphics depicting the scaffold, and various levels of play. If any LIZZY.THESAURUS owner would like a copy (either on datapak, disk, or listing) please get in touch. The program is too long to publish.

IPSO Review

LACE II for LZ64 by Beachcomber Software

Reviewed by Graham Foster

1) Introduction

LACE was originally reviewed by Glyn Pollington in April 89 issue of IPSO FACTO. Since then the package has been further developed and many new features added. The original LACE package is still available on a 32K datapak.

LACE II is now supplied on a 64K datapak. There is very little free space available on the pack. Exactly how much has varied from 100 bytes to 2K in the last 6 months! Let me explain...

I bought LACE II in June of this year when several new features had just been added. Unfortunately the package was not 'stable' and I found several bugs very quickly. Steve Knight (Beachcombers director) has been extremely helpful and he was able to supply me with new 'fixed' versions very quickly. I ended up as a Beta site tester for each revision of the code, and Steve took on board my problems and suggestions and produced new versions for testing. At one stage I was getting 1 version a week from Steve!

There have been so many changes and it is such a large package that I can only outline the changes since the last review.

2) Review

The main features reviewed by Glyn (Log, Action, Card, Expenses) still remain in the package but have been enhanced, and new features added.

2.1) Log

As I understand it, very little has changed in this part of the package, and it still time and date stamps 'things to remember' type of items. I don't use this as much as I should!

2.2) Action

This has been changed slightly. You can still 'import' log statements into actions and the dialogue remains much as before (e.g. giver, owner, priority etc). One feature that has been included is that ACTIONS can now be included in the diary with an optional alarm. This is achieved by writing a file that can be MERGED with your diary. This save you entering the action information more than once.

2.3) Card

CARD has been considerably enhanced. It is now a full accounts handling system. In brief the main new items are:-

- 1) Ability to 'Confirm' transactions. i.e. tick transactions off against a statement.
- 2) Tidy accounts on confirmed entries. eg all the items that have appeared on statements can be 'tidied away' leaving the items still to be cleared in your account.
- 3) Interaccount transfers. e.g. Credit your VISA card account from your cheque account only requires 1 transaction and both accounts are updated automatically.
- 4) Standing orders on any accounts. These have been included and extending to include number of months between standing orders so it can cope with quarterly and annual direct debits as well.
- 5) Improved reporting of accounts so you can easily get current balances on any or all of the accounts or categories. e.g. you can find out how much you have spent on petrol over the last 2 months if required.
- 6) Improved browsing and deletion facilities.
- 7) Optional automatic linking of ALL categories to EXPENSES. For example, if you pay for Company items in your own VISA card

then a category of WORK linked to expenses would take you through the CARD and EXPENSES dialogues automatically with the minimum of duplication. Your VISA and Co expenses account would be updated at the same time.

2.4) Expenses

This has been changed somewhat since '89 to allow the user to setup 2 separate expense accounts each with their own categories. It now handles larger amounts of money (e.g. lire) and can cope with any rate of VAT, and overseas exchange rates if required.

The reports can include VAT and currency conversion if required, but the SETUP option can be used to disable these fields and questions if you will never use them. They can be switched on and off as required without affecting the data stored.

3) Completely New Features

3.1) Setup

This is included to reduce the number of questions that are asked at certain times, e.g. VAT and overseas exchange questions disappear if they are not selected on SETUP. Also printer page length and formfeed character are selected here, so do not need to be specified for every report.

3.2) Comms link

The pack now boots the comms link when required, so you can plug it in at the last minute (after selecting the accounts and dates to report over). This is much easier than previously.

3.3) Timesheet

This is a major new section of the package. It can handle Projects, project numbers within projects, categories and activities within project numbers. It keeps track of hours spent in each sub-group and produces neat summary or detailed printouts.

Any category can be linked to expenses, that will allow costing of jobs directly. It also allows the diary to be updated in the same way the action list does. You can therefore fill you diary with the details of your activities.

The package can time jobs for you (without tying up the Organiser) or durations and dates can be entered retrospectively. It is aimed for people who need to cost their time accurately and are busy across multiple projects or tasks. (Accountants!)

3.4) Diary Printing

This facility only exists when the comms link is plugged in, but provides a way to print your diary by date range, matching items, weekly, and even bi-monthly. It can present a 'graphic' representation of the next 70 days of your diary, but you would need to be a bit cryptic and encode the first letter of every diary entry to something you would understand.

The other major advantage of this diary printer is that it will append the list of jobs due to be completed in the date range printed. This means that the actions due in 'this week' will be printed for you along with this weeks diary. It does print the diary resident in memory.

3.5) Phone logger

This is a new facility that is loosely based on the example program in the LZ manual. It is considerably enhanced and can update you expenses files directly. It costs the call and you can define the rates, although it comes with a complete list of the

IPSO Review -2

current BT charge rates for the world. It appears to be fairly accurate. Steve has obviously gone to a lot of trouble to incorporate the same fiddles BT use to rip us off, into his software!

3.6) Improved editing

Steve has provided a nice simple mechanism to include punctuation into any text entry in the package. It simply uses the arrow keys to scroll round alternatives. This does NOT preclude the use of FNKEYS however, and the two packages complement each other well.

4) Problems

As I indicated at the start of the review I have had problems with this software as it has been growing and developed along with me. Steve has been brilliant in updating and fixing the system. It now appears to be stable and I have not encountered any problems with it in the last couple of months.

There are some items that I would like to see addressed, (and that probably will be by the time you read this, if Steve sees the preview copy!)

4.1) Manual

With such a rapidly developing package, the manual has not kept pace with the change. Steve is working on improving the layout and documenting the new features. The main problem I found with it was that although it told you what the package did, it didn't guide on on the best way to use it.

After several months use, I still feel that I could link the Expenses and CARD parts of the package together and give me a more powerful home account tracking system than I am using currently ... but ... I haven't thought through the implications.

Steve was considering providing a programming manual that detailed the main OPL routines in the system. These should help out ordinary OPLers to duplicate some of the neat functions more easily and improve your own code.

4.2) Does not use the LZ display fully, and the ON/CLEAR vs. Quit approach to exiting menus first highlighted in the LACE review is incompletely fixed. Its better but not there fully. Still this is a minor irritation and soon forgotten about after a few days use. At least it is consistent in its own way.

4.3) Timesheet section is over-complex (for my use) and not quite as useful as it could be. It was designed by Steve for his particular application, and perhaps it was a bit more specialist than most peoples needs. I feel that this section would be easier to use if it were more like the EXPENSES or CARD features, and some of the levels of category were removed. I feel that PROJECT and ACTIVITY would be adequate for most peoples use provided a comment field was provided (like ACTIVITY).

I found out that I am not organised enough to use this routinely, and I do not have to use it any more as I now do not need to account for my hours at work!

4.4) Diary printing does not indicate when an item completes - only when it starts.

5) Like to see

Well there is not a lot Steve can put in 1K of space, but some items I would like to see are.. (Are you listening Steve)

a) Redesigned timesheet section that is a bit simpler, and more like the CARD section. (That should save some space!)

b) Include a facility in SETUP to remove the 'spare fuel can' dialogue in the Fuel part of CARD. I don't have a spare can,

and these questions irritate after a while!

c) Include in SETUP a printer initialisation string to set my printer to 96 columns, with a margin. This will allow me to print out the 80 column reports with space to punch them and file them.

d) More consistent use on ON/CLEAR (Its nearly there Steve!)

e) For date and time adjustment, use the arrow keys on the LZ in the same way as the Psion software does.

f) Remove the BACKUP facility and encourage people to get a copy of Jonathan Hurwitt's FM program. It is superb and deserves a review by itself. It appears to have been released to the public domain in May of this year.

6) Summary

I can recommend LACE II to anyone with an LZ. It will probably do more than you would want to do, but the extra facilities can be useful. Steve assures me that one of his customers is using LACE II for run his complete accounts on. I can quite believe it. The package is very comprehensive and is easy to use at any level. The more you know about it the more you find yourself doing on it. The end user support (from Steve) is superb and he must have the patience of a saint to cope with people like me!

Steve needs to update the manual (and probably has done by now) in order to get the best from this package. There are XP versions of this package with some facilities missing. Steve is happy to provide example printouts and further details to any interested party.

**(Beachcombers address is
16 Salter St, Berkeley, Glos)
My phone number is 0296-668698**

For Sale

Husky Hunter Palmtop Computer
Inbuilt GW Basic, Assembler, Text Editor,
Comms Software, Large LCD Screen w/Graphics
capablilty. Water Resistant and rugged.
Re-Chargable power supply
£150 or WHY

DeignCAD 2D CAD software for use on any
640k IBM compatible PC. Unused and unregistered.
Best of the low budget CADs
£150 ono or WHY

Phone Neil on 0332 880663

Feedback

Why I own an Organiser

by Bob Britten

Of course, mainly for all the usual advertised reasons, but being an idle person in my late sixties, I usually toss all names, addresses, hilarious graffiti, etc, that I wish to store and retrieve onto the MAIN kitty of a 32k RAMPACK or datapak because, if I was tidier and had separate files or datapaks for separate subjects, I would never know which one to start retrieving from.

To make retrieval easier, I try not to be too pedantic over the details I store, but always try to add corroborative facts to the basic information, i.e. to retrieve the details of that interesting physicist who was touring with us in Thailand last year, the fact that I had now forgotten his name can be overcome by punching in 'Thailand' or 'physicist'.

Incidentally, another advantage of shoving everything onto the MAIN file of a 32k pak is in the back-up process. I have copied the contents of my half-full 32k pak onto a reserve 64k pak, which always stays on my desk, and save my daily additions to a 16k pak. This I copy periodically to both the 32k pak and the reserve 64k pak so as not to duplicate information or indulge in unnecessary formatting by copying direct from 32k to 64k.

On the subject of OPL, I am a builder of small willow pattern garden bridges and I need to be able to give an instant price estimate to potential customers, who all want bridges of different dimensions. Add to that periodical price changes of some twenty different components and it was soon apparent that I needed to write an OPL program.

No problem, I thought, until I wrote a confirmatory program in BASIC on my Amstrad 8256, which gave me a slightly (about 1%) different answer. This predicament is, I am told, similar to that of a lone navigator in mid-ocean if he has two unsynchronised chronometers aboard - so one must either have one and infinite faith, or three and accept a majority decision (or average). In my case, I decided to create third program by writing a Mini Office spreadsheet on my Amstrad and, at the end of a week's agonised rewriting of all three programs, I achieved synchronisation. I was then able to have total faith in my OPL Organiser program!

More letters on this subject will be printed in future issues. Ed

For Sale

Hardware

Organiser LZ 64	£100.00
Leather Case	£5.00
Comms Link	£30.00
Parallel Converter	£25.00
Epson P40 Printer	£50.00
P40 cable	£5.00

Software

Thesaurus	£30.00
Spreadsheet (LZ)	£25.00
FNKEY V2.2	£25.00
FNKEY V2.1	£20.00
Filemaster+ (LZ)	£25.00
AutoScribe+ (LZ)	£25.00
PSION Games	£20.00
HB Games	£20.00
Harvester Superchip	£20.00
Oxford Spellcheck	£25.00
Finance Pak	£15.00

Storage

32k Rampak	£35.00
128k datapak	£50.00
64k datapak	£25.00
16k datapak	£10.00

Books

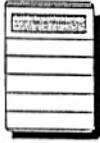
Using & Programming	£5.00
File Handling	£5.00
IPSO FACTO (4yrs bound)	£20.00

or £500.00 for the lot. Prepared to split into reasonable lots (say £100)

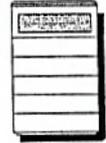
Phone Sean Sales on 071 836 3297 (day) or 081 671 5090 (evenings)

Personal Computer World

For some years now, PERSONAL COMPUTER WORLD, an excellent all round computer magazine, has been printing details of IPSO (free of charge) on their User Group Page. This has been the source of quite a few new members, so it is only right that I blow their horn a bit. Quite often, PCW prints good quality OPL programs and also reviews on Organiser peripherals, as well as news on other pocket computers, so it is well worth a read!



IPSO FACTO



The Newsletter of the International Psion Organiser User Group

Volume IV . No. 11 . January 1991

Office: 130 Stapleford Lane . Toton . Beeston . Nottingham . NG9 6GB . UK
Editor: Mike O'Regan . Phone: (0602) 735482 . Microlink Mailbox MAG11839

Editorial

Subscription Renewal

Once again the time has come for Membership Renewal. You will find a Renewal Form with this issue. **Please complete it and send it back to me as soon as possible, with the appropriate subscription, so that I can begin to plan for next years issues.**

You will notice that I have, reluctantly, had to raise the subscription rates. I did this last year, but did not anticipate the significant increases in all of our expenses for Printing, Stationery, Postage, etc. which have certainly occurred during the year. I thought long and hard, and consulted with some of our long-time members before deciding on the increases. I did, consider as one of the options, stopping publication. This, however, was discounted as I examined the high level of support which I have had and which is, if anything, on the increase. The new subscriptions mean that you (in the UK) will be paying £1.50 per issue (about the price of a pint!), so you must consider whether you think it worth the cost. I hope you will all renew and stay with us for the future.

Vive l'Organiseur Français

As we go to press, I had a pleasant surprise. It was a new book on the Organiser written by French member **Alain Carré**. This is a beautifully produced book of 206 (almost) A4 size pages. From what I can remember of my schoolboy French, it looks a pretty thorough handbook for the French speaking user. It is easy to think that, just because the LIZZY has a FRENCH option, that EVERYTHING will run in French. The Lizzy, quite cle-

verly, runs all the built-in features in the appropriate language, but when it comes to OPL, a fair knowledge of computer English is needed. This book will, I am sure be snapped up by the growing number of French Psioneers, once it becomes known.

Copies are available from the Publisher:

Aware

21 rue Olivier Métra

75020 Paris

Tel: (1) 46 36 46 47

Fax: (1) 46 36 82 54

Price is 250F TTC

...and another one in English

You will see a longish review in this issue of a new book, PSION LZ, A User's Guide to OPL. Since writing the review, I have had a bit more time to examine some of the detail, and I can only say that I consider this the best book so far for both the Beginner and more advanced Psioneer. I am indebted to Dab Press for sending me a review copy in double-quick time!

Contents

Editorial82

Procs & Progs.83

IPSO Review. 86

Feedback.88

Progs & Procs -1

Fuel Economy

by Neil Favager

Procedures (written and tested on an XP) to save and restore the Organiser top-level menu to a data file - very useful to restore a custom menu after getting the dreaded 'TRAP' and having to remove the battery.

The following program -works out fuel economy in m.p.g. given a number of miles travelled and an amount of petrol used.

This method of calculating m.p.g. only works if you always fill your petrol tank to the same level (ie full!) at each fill-up and make a note of your milage at each fill-up. It works with petrol prices in pounds/gallon or pence/litre.

It's a very simple program but is quite useful in it's own way. I don't pretend it's very code-efficient, but it's very easy to see exactly what it does. It makes use of the error trapping command 'TRAP' to check for 'ON/CLEAR' being pressed when a number should be entered. It calls the procedure 'OOPTS:' published previously in IPSO FACTO. (Vol III, so repeated here Ed.)

```
econ:
LOCAL d1,d2,t,c
loop::
CLS
PRINT"MPG CALCULATOR"
PRINT"Miles now:";
TRAP INPUT d1
IF ERR=206
  RETURN
ENDIF
PRINT" then:";
TRAP INPUT d2
IF ERR=206
  RETURN
ENDIF
IF (d1-d2)<0 :REM mileages the wrong way round?
  d1=d2-d1:REM if so take the 'other'difference
ELSE
  d1=d1-d2
ENDIF
PRINT"Total cost:";
TRAP INPUT t
IF ERR=206
  RETURN
ENDIF
PRINT"Unit cost:";
TRAP INPUT c
IF ERR=206
  RETURN
ENDIF
CLS
PRINT"Gallon or Litre Price (G/L)";
IF oopt$:"GL"="L"
  c=c*4.5475/100
ENDIF
CLS
PRINT" MPG is "+FIX$(d1/(t/c),2,6)
```

```
GET
GOTO loop::
```

Saving & Restoring a Customised Top-Level Menu

by Neil Favager

I have read with interest the recent articles in IPSO regarding saving the Oganiser top-level menu and simuating PC-subdirectories on the Organiser.

My particular requirement was very similar to this, but subtly different:

My Organiser top-level menu consists of only 7 of Psions default options; the rest being procedures or sub-menus of my own. If, as has happened from time to time, I have to reset my organiser (or recover after a TRAP) it is very frustrating (and time-consuming) to have to set up the top-level menu all over again.

What I needed was a way to save the Organiser top-level menu options, in the correct order, to a datafile (on a datapack)... and then be able to restore them again at a later date (ie after a crash!)

While I was at it, I also wanted a way to change the case of the options in the top-level menu for 'my programs' only. I like the PSION options to be all in upper-case, and my own options to be in lower case with the first letter capitalised.

So, I set to work, and here is the result (with apologies for the necessary cannibalisation of recent published procedures). The 'system' consists of four procedures and also makes use of the procedure 'OOPTS:' published previously in IPSO FACTO.

MENUS: the entry and menu procedure
MENSUB: procedure to backup the menus to a file
MENUSR: procedure to restore the menus from a file
MENUSC: procedure to change the 'case' of the menu options

Instructions for use:

- o Firstly enter all the procedures (below) and translate them.
- o Save them to a pack, don't leave them in memory (on A:). The procedures use machine code: if you have made a typing mistake you don't want to have to type then all in again as well as resetting your Organiser! Also, if you need to restore the menus it'll be after a crash, so you'll have lost the contents of A: anyway!
- o Next get your Organiser top-level menu exactly as you like it.
- o Take out any 'auto-boot' packs (Spelling Checker/Thesaurus, Dictionary, Spreadsheet, etc) and remove the Comms-Link, if fitted.
- o Press ON/CLEAR to remove the auto-boot options from the top-level menu (Note: you MUST do this!).
- o Select 'PROG' and then 'RUN' the procedure 'MENUS' on

Progs & Procs -2

whichever pack you saved the procedures to.

o You can now run the 'Setcase' option if you like to set OPL procedure menu options to lower case, as I do.

o Select 'Backup' and the system will save the menu-options, their order and associated ROM-address (if appropriate) to a datafile called 'MENUS' on whichever pack you select (press 'Q' to Quit). I suggest that you save the file onto the same pack as the MENU procedures.

o That's it... top-level menus safely saved.

To RESTORE your top-level menu after a crash or power loss proceed as follows:

o Insert the pack with the MENUS procedures and datafile into the organiser.

o Remove all auto-boot packs and the Comms-link, if fitted.

o Switch on and press ON/CLEAR.

o Delete all top-level menu options except 'OFF'. I know that it seems extreme but it's very quick... press 'DEL' and 'EXE' alternately.

o Insert the top-level menu option 'MENUS'.

o Select 'MENUS' by pressing 'EXE' and then choose 'Restore' and enter the letter of the pack which contains the MENU datafile or press 'Q' to quit. Your original top-level menu will now be recreated.

o Delete the 'MENUS' option (the first option) and the top-level menu is exactly as you saved it.

o You can now set OPL menu options to lower case if you wish by running the 'Setcase' option from MENU (by default the menu items are restored as upper case).

o That's it... top-level menus safely restored.

And now here are the procedures:

```

menu:
GLOBAL fn$(8),mm%
fn$="menu"
loop:
mm%=MENU("Backup,Restore,Setcase")
IF mm%=1
  menu:
ELSEIF mm%=2
  menu:
ELSEIF mm%=3
  menu:
ELSE
  RETURN
ENDIF
GOTO loop:

```

menu:

```

LOCAL s,l%,k%,n%,n$(8),a,g$(1)
CLS
PRINT"Backup Menus to (A/B/C) or (Q)";
g$=oopt$("ABCQ")
IF g$="Q"
  RETURN
ENDIF
fn$=g$+":"+fn$
IF EXIST(fn$)
  CLS
  PRINT"File exists. Overwrite (Y/N)";
  IF oopt$("YN")="N"
    PRINT :PRINT"Abandoned.";
    PAUSE 20
    RETURN
  ELSE
    DELETE fn$
    CREATE fn$,b,mn,mn$,ma
  ENDIF
ELSE
  CREATE fn$,b,mn,mn$,ma
ENDIF
CLS
PRINT"Writing";
s=PEEKW($2002):REM start of menus
n%=0
loop:
l%=PEEKB(s) :REM option length
k%=1
n$=""
n%=n%+1
DO
  n$=n$+CHR$(PEEKB(s+k%)):REM option name
  k%=k%+1
UNTIL k%>l%
a=PEEKW(s+k%):REM option address
b.mn=n%
b.mn$=n$
b.ma=a
APPEND
PRINT".";
IF NOT UPPER$(n$)="OFF":REM finished?
  s=s+k%+2
  GOTO loop:
ENDIF
CLOSE
PRINT
PRINT"Completed."
PAUSE 30
RETURN

menu:
LOCAL r1,r2,r3%,r4%,n%(30),n$(30,10),a(30),g$(1)
CLS
PRINT"Load Menus from (A/B/C) or (Q)";
g$=oopt$("ABCQ")
IF g$="Q"
  RETURN
ENDIF

```

Progs & Procs -3

```

fn$=g$+":"+fn$
IF EXIST(fn$)
  OPEN fn$,a,mn,mn$,ma
ELSE
  CLS
  PRINT CHR$(16)+"Menu Backup file not found";
  PAUSE 30
  RETURN
ENDIF
CLS
PRINT"Restoring...";
r1%=0
DO :REM read options into array
  r1%=r1%+1
  POSITION(r1%)
  n$(r1%)=a.mn
  n$(r1%)=a.mn$
  a(r1%)=a.ma
NEXT
PRINT".";
UNTIL EOF
DO
  r2%=r2%+1
  r4%=LEN(n$(r2%)):REM option length
  POKEB $2187,r4% :REM poke into run-time buffer
  r3%=0
  DO :REM poke name do.
    r3%=r3%+1
    POKEB $2187+r3%,ASC(MID$(n$(r2%),r3%,1))
  UNTIL r3%=r4%
  r3%=r3%+1
  POKEW $2187+r3%,a(r2%) :REM poke address do.
  POKEB $2020,$c6 :REM load b with...
  POKEB $2021,n$(r2%) :REM ...position
  POKEB $2022,$3f :REM call organiser ROM...
  POKEB $2023,101 :REM .routine for menu insert
  POKEB $2024,$39 :REM home again
  USR($2020,0) :REM call routine
  PRINT".";
UNTIL r2%=r1%
CLOSE
PRINT :PRINT"Completed.";
PAUSE 30
RETURN

menusc:
LOCAL s,k%,l%,e,w%,o$(8),v$(80),m$(1)
v$="FIND*SAVE*DIARY*CALC*PROG*"
v$=v$+"ERASE*TIME*INFO*ALARM*"
v$=v$+"COPY*RESET*OFF*" :REM standard options
v$=v$+"COMMS*WORDS*KEY*" :REM other possibles
CLS
PRINT"Set case of OPL Items (Y/N)";
IF oopt$:( "YN")="N"
  RETURN
ENDIF
CLS
PRINT"Writing..."
s=PEEKW($2002) :REM find start of menus

```

```

loop::
o$=""
l%=PEEKB(s) :REM length of option
k%=1
DO :REM build option string
  o$=o$+CHR$(PEEKB(s+k%))
  PRINT CHR$(15)+o$;
  k%=k%+1
UNTIL k%>l%
e=s+k%
IF UPPER$(o$)="OFF" :REM finished?
  POKEW e,0 :REM only if own 'OFF' installed
  PRINT
  PRINT"Completed."
  GET
  RETURN
ENDIF
w%=LOC(v$,o$)
IF w%=0
  k%=2
  DO
    m$=LOWER$(MID$(o$,k%,1)):REM set option case
    POKEB(s+k%),ASC(m$):REM poke option back again
    AT k%,2 :PRINT m$;
    k%=k%+1
  UNTIL k%>l%
ENDIF
s=e+2 :REM skip the address locations
GOTO loop::

```

It would be fairly easy to modify MENUSR to make it automatically remove all existing top-level menu options before restoring the saved ones. Perhaps someone will do this?

The Following procedure was previously published in Vol III, Page 88, but is repeated for those who do not have this Volume.

```

oopt$(o0$)
LOCAL o1$(1),o2%
CURSOR ON
DO
  o1$=UPPER STRING(GET$)
  o2%=LOC(o0$,o1$)
UNTIL o2%>0
CURSOR OFF
RETURN o1$

```

This suite of programs is printed at the specific request of quite a few members. It may seem rather long, but many of the lines are quite short. Of course, as usual, the procedures are available either on 5.25" IBM format disk or can be copied to your own data or RAM pak for a nominal sum, if you don't wish to type them in. The usual warnings for POKES, etc apply. Ed.)

IPSO Review - 1

by Mike O'Regan

Psion LZ A User's Guide to OPL

by Ian Sinclair

This book arrived quietly on the Organiser scene as the fourth book on the Organiser. The previous books have all been published by Kuma Computers - this one is different as it comes from the DAB stable who have already published a few books on other aspects of computers and computing. The author, Ian Sinclair is very well known in computer circles as a respected and best-selling author.

You may well ask "Why do we need yet another book on the Organiser?". Well, in a way, it is a tribute to the machine itself, which is acknowledged by Ian Sinclair to be the leader in the field and the machine against which all others are measured. As I know from members letters, etc., any book which will help the beginner to get into OPL will be welcome. This book, as we shall see, is excellent from this point. At the same time it does not 'talk down' to the reader in a way which would put off the experienced programmer. In other words, there is something for everyone in the book.

Throughout the book, the Author refers to OPL as "OPL BASIC". I would take issue with this term, as OPL, despite some similarities, is certainly not BASIC, being a clever combination of the best of at least three other languages. Most people who have experience of BASIC will know that it has many versions, very few of which can approach OPL in versatility. Although it is possible to get versions which compile, most BASICs are interpreters which makes them fundamentally different to OPL. Anyway, it is only a minor point, and does not detract from the overall good impression of the book.

A quick glance at the Contents pages will show that this book is different in many way to the others. Although the title mentions specifically the LZ models - and there is emphasis on the extra facilities on this model - it is also of great value to the user of the CM and XP models. The book is divided into six chapters:

1. Setting Up
2. Variables
3. Getting Repetitive
4. Menus and Procedures
5. Filing Techniques
6. Finishing Touches

1. Setting Up.

After a useful description of some of the new features of the Lizzy, this chapter goes straight into an outline of OPL and a few - very easily understood - procedures.

2. Variables

The concept of a variable, unless you are a very familiar with Algebra, is often quite difficult to grasp. This chapter

introduces variables in a very natural way and explains them very well with numerous short example procedures. This area has been rather neglected in the existing books and Mr Sinclair certainly puts matters right. An unusual feature is that he uses numerical examples quite early, which illustrates the difference between string and other variables quite well.

3. Getting Repetitive

One of the basic attributes of a computer is its capacity for controlled repetition of commands. This is one of the features which distinguishes a calculator from a pocket computer, along with the ability to "make decisions" and modify the running of the machine accordingly. This chapter explains the different loops available in OPL and again illustrates them very well.

4. Menus & Procedures

This is an important chapter where programming is extensively examined, and the role of OPL's flexible MENU command is introduced. An innovative section headed "Rolling Your Own" looks at the subject of writing your own procedures as distinct from always running those which have been written by someone else. The author is obviously a committed programmer himself, as he states, quite rightly, that programming can be a satisfying occupation in its own right, quite apart from any practical value. This is a point which may have escaped many Organiser users, and one which will surely be echoed by those who have taken the trouble to actually try programming for themselves for the first time.

5. Filing Techniques

The mysteries of file handling are unravelled in no uncertain way in this Chapter. Features include: What is a File? - Knowing the Names - Filing in RAM or Datapak - Creating a File....Statistical Work with Files etc.

6. Finishing Touches

This Chapter looks at the Date Functions, including DAY and WEEK, and some of the practical applications of these. Error Trapping is also examined and a

IPSO Review - 2

short introduction to BEEP and its applications is included.

The book has four Appendices, namely:

- A: PC Link
- B: Boolean Actions
- C: ASCII COdes
- D: DABHAND Guides

There are also two indexes - one for OPL procedures and another general one.

All in all, I found this a refreshing book to read, and a valuable addition to any Organiser User's library. It does not tediously cover previously available information either from the Handbooks or other books.

PSION LZ - A User's Guide to OPL is a must for every LIZZY owner and has much to offer owners of the CM and XP models. It is thoroughly recommended.

The book has 224 pages and is available from good Organiser dealers, the better bookshops, or direct from the publishers:

The Dabs Press
22 Warwick Street
Manchester
M25 7HN

Telephone: 061 773 8290

The price is £12.95

Music Prakpak & Metronome from Mackay Language Software

This is a relatively new music pak, specifically for the LZ models (It will not run on the CM or XP because of the full use of the four-line display, which is essential to the programs). Mackay says that the pak will "...quickly help you to find the right keys on a piano or keyboard and read music with steadily increasing ease, while the Metronome gives you a variable tone or flash at a pace selected by beats per minute or tempo".

The Music Prakpak comes on a 16k datapak, with a booklet of 11 pages of instructions. The booklet can

be dispensed with as soon as the simple menu-driven operation is understood. Practice could then proceed at any time that the pak is plugged in.

The pak autoboots (inserting MUSIC on the top-level menu) by pressing ON/CLEAR twice in the usual way after insertion. Choosing MUSIC produces an opening screen with the top two rows occupied by a tiny keyboard of 34 black and white keys! (What a pity that the Organiser has gaps between the matrices which form the display, as this makes the 'black' keys particularly difficult to recognise).

The bottom line has the menu with Keys, Stave, Metro, Twin-Stave and Quit. I was so intrigued by the 'twin-stave' option (wondering how it could be displayed) that I chose this first. This produced another menu with A-E used to choose delays of different lengths between notes, Other (to set your own delay) and Quit. I chose E, the longest delay and, at first was completely mystified by the display. However, I soon realised that the display was meant to be read by turning the Organiser on its side! This is the first time I have even had to do this. Two five-line staves are then displayed with random notes appearing with their musical names from the low G on the lower stave to the upper F on the treble stave.

At this stage, I thought it appropriate to actually read the handbook, mainly because I didn't seem able to stop the display continuing. <Q> will quit this section of the program.

KEYS lets you test or learn the names of the notes on the standard keyboard and, subject to the limitations of the display of the 'black' notes mentioned above, it is quite effective. After displaying Middle C, a little arrow invites you to name the notes which are randomly selected (in the Test mode) or displays the names of the random notes (in Learn Mode). I found that it took quite a bit of practice to name the notes properly, purely because of the display limitations - 'white' notes rely on their relationship to the 'black' ones to be correctly named. In Test mode, any note incorrectly named gets a high pitched squeak and then the correct name is displayed and you are invited to try again. The arrow stays on the same note, which is a bit silly. It should have moved on to another to save time.

STAVE seemed to be a repeat of TWIN-STAVE, until I realised that the TWIN referred to the number of NOTES displayed simultaneously. This has two options, Scroll or Interact. Scroll produces random notes placed on the stave for set intervals so that you

IPSO Review - 3

can "find them on a piano or keyboard". These are displayed with or without note names, as required. Interact places a note on the staff and waits for you to key in its name (from A to G), signifying right or wrong and then producing the next note.

METRONOME

This produces either an audible or a visual signal at an interval which you choose by either selecting beats-per-minute or a selection of tempi from Largo to Prestissimo. The tone of the audio signal is adjustable by using the up and down arrows.

Method of Working

Two pages of the booklet are devoted to detailed suggestions about how to use the simple programs provided

Conclusion

Whether the Music Prapak can fulfill its promise would take more than the short time available for review, as the programs are meant to be used for an extended period. In any case, as I can already read music, I find it difficult to decide whether the programs would help in the learning program. I should think that they could be used to augment any other music training, but would not provide a complete training aid. This is suggested at the beginning of the booklet.

**Music Prapak is available from:
Mackay Language Software
41 Kensington High Street
London W8 5ED**

Telephone: 071 937 2077 FAX: 071 376 2368

Price is £24.95

More News from Jaro Computer Services

Remember Jaro (of the **Databox** fame). Robin Corrie, the man at the helm tells me that he is now providing a RAMPAK Battery Replacement Service. The replacement battery is a LITHIUM cell which is quite different from a MERCURY or SILVER OXIDE cell. Replacement costs £10, and Robin recommends Registered Post, as Recorded Delivery has a maximum compensation payment of £22 (which is, of course, inadequate to cover the cost of a RAMPAK).

On the Databox front, where a customer wants his

copy-protected software copied to an EPROM, this will normally be a 64k chip which the customer has paid for (copying is free). However, certain proprietary programs have to be copied to 32k chips, otherwise they won't work, because the software assumes the addresses will "wrap around" (which will not happen if copied to a 64k chip. In such cases it should be noted that:

- There is no price difference (although, of course, the 32k EPROMS will also count towards quantity discounts, and
- These 32k chips cannot be used to STORE data with the DATABOX, although, of course, they can be READ as normal. Some programs which behave in this way are the SPREADSHEET and FINANCE PAK II.

Any further information may be obtained from:

**Jaro Computer Services
9 Brighton Close
Addlestone
Weybridge
Surrey
KT15 1PP**

Telephone: (0932) 857398

For Sale

Organiser LZ64. . .£100
Comms Link. . .£35
32k RAMPAK. . .£35
128k Databox. . .45
Leather Carry Case. . .£10
Cubsoft FNKEY V.3. . .£25
or £250 the lot. Boxed in VGC

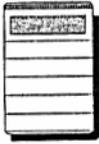
Phone: Peter Humphris on (0382) 458629

Organiser XP Model in VGC
complete with Harvester case (with pak slots)
and Handbook
£60 (ono)

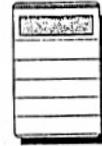
Phone: Mike O'Regan on (0602) 735482

Organiser LZ64 & Denwood Leather Case - £100
UVIPAC Eprom Eraser (formatter) - £20

Phone: Jim Kirwan on 0772 614185



IPSO FACTO



The Newsletter of the International Psion Organiser User Group

Volume IV . No. 12 . February 1991

Office: 130 Stapleford Lane . Toton . Beeston . Nottingham . NG9 6GB . UK
Editor: Mike O'Regan . Phone: (0602) 735482 . Microlink Mailbox MAG11839

Editorial

IPSO is Four Years Old

With this issue, IPSO completes its fourth year. Nobody is more surprised than I am! When I started the Group with just a handful of Psioneers (some of whom are still with us, I am glad to say), I had no experience whatever of either running a User Group or publishing a regular newsletter.

Four years on, I am still getting a lot of support in the shape of material for IPSO FACTO. I would like to thank those who have submitted material over the years, and especially those who have not yet seen their efforts in print. Most of these will be used in due course, as I try to keep an interesting balance in each issue.



In last month's review of the new book "PSION LZ - A Users Guide to OPL, I got the Telephone Number of the Publishers wrong.

**Dabs Press' correct number is
061 - 773 8632.
Their FAX Number is
061 - 773 - 8290**

FNKEY Upgrades

CUBSOFT have informed me that their ever popular FNKEY has just been upgraded to Version 3.5.

The upgrade costs are as follows:

Versions 3.0 & 3.1 - £10
Versions 1.x & 2.x - £20

The new features are:

1. Easier installation
2. Additional utility option
3. date key can now be configured for UK or US format (previously US format only), together with optional time stamping
4. Recordings can now be made which load other FNKEY files (and hence switch banks).

Further details are available from
CUBSOFT on 061 792 2871

Contents

Editorial	89
Procs & Progs.	90
Beginners Page.	93
Reviews & Feedback. . .	95

Progs & Procs - 1

Qustom by David Green

The following suite of programs provide a valuable extension to the range of controls over the various functions of the LZ models, including the saving and restoring of a customised Main Menu. As many of the routines contain lines which directly address memory, great care should be taken when they are keyed in and the warnings about backing up any valuable contents of A: apply. Although there are a fairly large number of procedures, most of them are quite short.

```
qustom:
GLOBAL
a%(11),b$(25,6),m%,p$(6),x%,y%
ONERR 11::
11::
[m%=MENU("AUTO,BORDER,CLICK,HSCROLL,NOISE,STATUS,TURNFF,VSCROLL,ZERO")]
IF m%=1 :auto:
ELSEIF m%=2 :border:
ELSEIF m%=3 :klik:
ELSEIF m%=4 :hscroll:
ELSEIF m%=5 :sound:
ELSEIF m%=6 :status:
ELSEIF m%=7 :shutoff:
ELSEIF m%=8 :vscroll:
ELSEIF m%=9 :zeero:
ELSE RETURN
ENDIF
GOTO 11::
```

```
auto:
REM deletes psion MENU
REM items NOT required
delmenu:
REM inserts your own
REM items into main MENU
chdir3:
[REM adjust following pokes to suit yourself]
POKEB $2099,35 :REM border
POKEB $20c0,0 :REM click
POKEB $6a,4 :REM horizontal scroll
POKEB $a4,1 :REM sound ON/OFF
[POKEB $20cd,120 :REM automatic switch OFF time]
POKEB $6c,10 :REM vertical scroll
[REM resets calculator memories IF
```

```
required]
REM otherwise all are zero
m0=MONTH
m1=WEEK(DAY,MONTH,YEAR)
m9=24
```

```
delmenu:
LOCAL a%(25),p$(6),x%
x%=1
DO
IF x%=1
p$="FIND"
ELSEIF x%=2
p$="SAVE"
ELSEIF x%=3
p$="DIARY"
ELSEIF x%=4
p$="TIME"
ELSEIF x%=5
p$="NOTES"
ELSEIF x%=6
p$="WORLD"
ELSEIF x%=7
p$="ALARM"
ELSEIF x%=8
p$="OFF"
ENDIF
a%(1)=$01ce
a%(2)=ADDR(p$)
a%(3)=$3f67
a%(4)=$3900
USR(ADDR(a%()),0)
x%=x%+1
UNTIL x%=9
```

```
chdir3:
ESCAPE OFF
p$="CALC"
a%(1)=$fe20
a%(2)=$02cc
a%(3)=ADDR(p$)
a%(4)=$dd41
a%(5)=$8603
a%(6)=$c603
a%(7)=$3f72
a%(8)=$2704
a%(9)=$3f67
a%(10)=$20eb
a%(11)=$3900
USR(ADDR(a%()),0)
```

Progs & Procs - 2

```

b$(1)="Y"
b$(2)="WAGE"
b$(3)="VERT"
b$(4)="TKLOCK"
b$(5)="SORT13"
b$(6)="R"
b$(7)="QUSTOM"
b$(8)="OFF"
b$(9)="NOTE"
b$(10)="LEFT"
b$(11)="JOTT"
b$(12)="GKLOK"
b$(13)="EILEX"
b$(14)="DIRY15"
b$(15)="BANK"
b$(16)="ABCD"

```

Note: Insert your
 own CUSTOM
 MENU items in
 these positions

```

y%=1
DO
  p$=b$(y%)
  POKEB $2187,LEN(p$)
  x%=1
  DO
    [POKEB
$2187+x%,ASC(MID$(p$,x%,1))]
    x%=x%+1
    UNTIL x%>LEN(p$)
    POKEW $2187+x%,0
    a$(1)=$3f65
    a$(2)=$3900
    USR(ADDR(a$()),0)
    y%=y%+1
  UNTIL y%=17

```

```

border:
PRINT "BORDER SYMBOL:",
INPUT m%
PRINT CHR$(m%)
GET
POKEB $2099,m%

```

```

klik:
[m%=MENU("Remove Click,Normal Click,
Extend Click")]
IF m%=3
PRINT "SET CLICK LENGTH:",
INPUT m%
m%=m%+1
ENDIF
POKEB $20C0,m%-1

```

```

hscroll:
PRINT "HORIZONTAL SCROLL"
PRINT "DEFAULT VALUE=4"

```

```

PRINT "SET RATE:",
INPUT m%
POKEB $6A,m%

```

```

sound:
PRINT " DO YOU WANT SOUND?"
m%=MENUN(2,"YES PLEASE,NO THANKYOU")
POKEB $A4,m%-1

```

```

status:
IF NOT EXIST("A:STATUS")
  CREATE "A:STATUS",a,i$,j$,k$,l$
ELSE OPEN "A:STATUS",a,i$,j$,k$,l$
ENDIF
a.i$=" BORDER"
a.j$="DEFAULT VALUE:245"
a.k$="CURRENT
VALUE:"+FIX$(PEEKB($2099),0,-3)
APPEND
a.i$=" CLICK LENGTH"
a.j$="DEFAULT VALUE:1"
a.k$="CURRENT
VALUE:"+FIX$(PEEKB($20c0),0,1)
APPEND
a.i$=" HSCROLL"
a.j$="DEFAULT VALUE:4"
a.k$="CURRENT
VALUE:"+FIX$(PEEKB($6a),0,2)
APPEND
a.i$=" NOISE(0=ON/1=OFF)"
a.j$="DEFAULT VALUE:0"
a.k$="CURRENT
VALUE:"+FIX$(PEEKB($a4),0,1)
APPEND
a.i$=" SHUTOFF"
a.j$="DEFAULT VALUE:300"
a.k$="CURRENT
VALUE:"+FIX$(PEEKW($20cd),0,-3)
APPEND
a.i$=" VERTICAL SCROLL"
a.j$="DEFAULT VALUE=10"
a.k$="CURRENT
VALUE="+FIX$(PEEKB($6c),0,-2)
APPEND
a.i$="CALCULATOR MEMORIES"
[a.j$="M0="+GEN$(m0,12)+"M1="+GEN$(m
1,12)+" M2="+GEN$(m2,12)+" M3="+GEN$
(m3,12)]
[a.k$="M4="+GEN$(m4,12)+"M5="+GEN$(m
5,12)+" M6="+GEN$(m6,12)+"M7="+GEN$
(m7,12)]

```

Progs & Procs - 3

```
[a.l$="M8="+GEN$(m8,12)+" M9="+GEN$(m9,12)]
APPEND
FIRST
DO
  m%=DISP(-1,"")
  IF m%=13
    IF POS=COUNT
      FIRST
    ELSE NEXT
  ENDIF
  ELSEIF m%=32
    IF POS=1
      LAST
    ELSE BACK
  ENDIF
  ENDIF
UNTIL m%=1
CLOSE
DELETE "A:STATUS"
```

```
shutoff:
PRINT "SECONDS TO"
PRINT "SHUTOFF";CHR$(63),
INPUT m%
POKEW $20cd,m%
```

```
vscroll:
PRINT "VERTICAL SCROLL"
PRINT "DEFAULT VALUE=10"
PRINT "SET RATE:",
INPUT m%
POKEB $6c,m%
```

```
zeero:
PRINT "ZERO M0 to M9? (Y/N)"
IF GET=%Y
  m0=0 : m1=0 : m2=0 : m3=0 : m4=0
  m5=0 : m6=0 : m7=0 : m8=0 : m9=0
ENDIF
```

***Have you renewed your Membership
yet
If not
this is your last Newsletter!***

For Sale

Kodak Dikonix 150 Portable Inkjet Printer
(serial interface - suitable for Organiser)
c/w re-chargeable batteries & spare print head
(but no charger)

£180 ono or exchange for Psion Printer II

**Contact: Mr P.B. Reece
on (0702) 584302 (after 6 pm)**

Psion Finance Pak £15
Psion Travel Pak £15
Concise Oxford Spell Checker £15
Mike Shaw Book 1 £5

**Contact Simon Ghent
on (0602) 733085
after 9.30 pm Mon - Fri
after 7 pm Sat or all day Sun**

Wanted

Psion MC 400 or MC 200

**Contact Les Ball
on (0602) 289553 (evenings & weekends)**

4 Line Pocket Spreadsheet
(offer of £20)

Contact David Bolton on (0365) 326805

Advertising

Just to remind you that 'small ads' (for Organiser or associated equipment) are inserted free of charge for Members. We maintain our policy of having no commercial advertising in the body of the newsletter, but will accept separate advertising leaflets for distribution for a small charge

Beginners Page - 1

by Mike O'Regan

Handling Data on the Organiser - Part VI

This month, as promised, I will review what we have done so far in this series and see what we are able to achieve by installing our own database system in this way (rather than using the Organiser's built-in database facilities).

Well, first of all, our little program suite is under the control of its own menu, with clear options in the display. The built-in database has just FIND and SAVE (and ERASE on the CM or XP) on the main menu. Some of the other facilities (such as deleting a single record) are not quite so easy to access.

The next advantage of our system is that you are "prompted" for data entry, with a little message for each of the three fields making it clear how the data should be entered. You will have noticed that the SUBSCRIPTION field not only expects NUMERIC input, but it also arranges things so that the Organiser will only accept numeric input in this field. Any attempt to enter ALPHA data will be met with an error message and the invitation to re-enter the SUBSCRIPTION correctly. This is essential, if we are to proceed to the next stage, which is also a significant improvement over the Organiser's built-in feature - it will make it possible to TOTAL some or all of the SUBSCRIPTIONS.

You will also notice that the default status for the entry of TELEPHONE NO is also NUMERIC. The line in dbenter: which arranges this is the KSTAT 3. However, the TELEPHONE NUMBER field is still an ALPHA field. Some beginners find this point a bit difficult to understand, so perhaps a little explanation is necessary.

Although NUMERIC fields only accept numeric input (and, as we have seen above) will soon tell you if your try to enter anything else, ALPHA fields will allow the entry of all alpha characters, plus numeric CHARACTERS and most other characters which can be keyed in directly from the keyboard. Although it is now common practice to enter Telephone Numbers as groups of figures only, some people still use the old method of entering the first part of the number as alpha (e.g. Nottingham 735482 instead of (0602) 735482). It is quite clear that Telephone Numbers, no matter how they are entered, are NOT numeric in the same way as our SUBSCRIPTION field. For instance, it would be a nonsense to ADD all the telephone

numbers together and then divide the sum by the number of records to give us an "average telephone number"! Conversely, the sole reason for making our SUBSCRIPTION field numeric, is precisely so that we can do various numerical calculations with them afterwards. This last point is something that cannot be done directly with data entered using the standard SAVE function of the Organiser.

I hope this has made it clear, then, why it is well worth while taking the (small amount of) trouble to design our little data-handling suite.

One or two people have asked about the number of fields in our database, which I have deliberately restricted to just THREE. There is nothing to stop you experimenting in adapting the database as it stands to either change the names and format of the existing fields, or even adding more fields, up to a maximum of 16. Of course, when you add more fields, you should bear in mind that the total number of characters which can be entered in each RECORD (that is the total for all FIELDS) is 254. Obviously, our little three-field file will never get anywhere near this total, but it should be borne in mind with more extensive databases.

Now, before we design the next procedures in our suite, a quick reminder of what we have already included. So far we have facilities to ENTER records, to FIND a record or number of records, to ERASE (or remove) single records (with suitable safeguards to avoid accidental erasure), and, finally, to DELETE a complete file (again with safeguards).

You will notice that all these facilities are also available as standard functions on the Organiser. The only real difference so far is that ENTERing records is easier because of the "prompts".

The next procedure is NOT available as a built-in feature.

We are now including a procedure to TOTAL all subscriptions and display the sum. Following our conventions so far we will make the procedure name fit in with the others. It is called DBSUBTOT: (using all eight allowed characters for the first time).

(listing on next page)

Beginners Page - 2

```
dbsubtot:
LOCAL stot
FIRST
DO
    stot=stot+a.subs
NEXT
UNTIL EOF
PRINT "Total subs"
PRINT stot
GET
```

Notes

This procedure is fairly self-explanatory. However, one or two notes should be helpful:

a) We need a new variable to keep a running total. This is called "stot" (for Subs TOTal) and, as it only applies to this procedure, it is declared as LOCAL.

b) As in other procedures, the line FIRST positions our proc on the FIRST record in our database, thus ensuring that none of the records is excluded from the total.

c) The line "stot=stot+a.subs" can be simply explained as follows:

1. Open a box (variable) marked "stot".
When we open this box it is always EMPTY (important)

2. Take out the contents of the box and add the first subscription, then replace the new contents back in the box. For the FIRST (and only for the first) the box is empty until the first subscription is added.

3. Repeat this until the EOF (End Of File) is reached.

4. Display the message "Total Subs" and, on the next line, the sum of all subscriptions

5. The GET, as usual, just holds the display for us to read until we decide to press any other key

Now all that remains is to insert our new procedure into the main menu of DB:. For the first time, I am asking you to do this yourself. Have a go, but don't worry, if you can't manage the task I will give you full instructions in the next installment.

Editor's Note:

I must apologise for the one month break in this series and I was quite gratified at the number of urgent messages I received on the subject. In future, I will try to keep the Beginners Page a regular feature. Ed.

The Swindon Group

Readers may have seen occasional references to the Swindon Group in these pages, so I thought that news of what happens in what is reputed to be the fastest growing town in the EEC might interest people.

The Group was formed in 1988 by Simon Webb (a software manager) and me (a solicitor specialising in computer law). The Swindon Group was the first Psion Organiser User Group to be formed outside London.

Perhaps because of the relaxed atmosphere of the meetings, people have been attracted to attend the Group from a wide area. From time to time a prize is awarded to the person who has travelled the furthest. The record to date is held by a man who came from Belgium!

The Group meets every two months at my home (address shown below). The meetings are at 7.30pm on the first Monday of every even numbered month - February, April, June, August, October, and December. There is an informal discussion on new programs, problems that have been encountered and the latest Psion products. The Group has NO membership fee and there is NO admission charge to meetings. The Group has arranged discounts for members in the purchase of Psion equipment. The meetings provide an excellent forum for the sale of secondhand equipment.

If you would like to be included in the mailing list of future meetings please write to me:

Jeremy Holt
14 Belmont Crescent
Old Town
Swindon
Wilts SN1 4EY
(Fax: 0793 612813)

Local Groups

We will publish details of meetings of Local groups, if we receive them in time. We are also quite willing to publish the names and details of anyone who would like to start a Local Group in their area. Anyone who has tried in the past may wish to try again, as the pattern of membership is constantly changing. Ed.

Reviews & Feedback - 1

Mini Reviews Carfax from CLS Software

We have all, I suppose, written small programs from time to time to keep track of our expenses, motoring or otherwise. We may even have promised that, when we have time, we would do the job properly!

Go no further. If you need to properly cost out your car (or cars - up to 20 of the little blighters), then Carfax is for you.

The suit of programs provided present you with facilities for keeping track of:

1. Overall fuel consumption (MPG/MPL)
2. Fuel consumption on the last full tank of petrol (MPG/MPL)
3. Total cost per mile
4. Last month's petrol costs
5. Last year's petrol costs
6. Total mileage
7. Mileage covered in the last year
9. Yearly mileage, estimated on the mileage covered between the last two entries.
10. Yearly mileage, estimated on the mileage covered in the last month.
11. Estimated date of next service.

There are also facilities to record individual vehicle details, including engine & chassis numbers, insurance and MOT details.

The program is menu driven and very easy to use, once a start has been made with a full tank per vehicle.

The program has provision for quite detailed reports, compiled from the data.

The review copy of CARFAX had a slight problem in that it used vehicle registration numbers as filenames. This, of course, could lead to unacceptable filenames where a vehicle registration number started with digits. CLS are correcting this.

All in all, I think that a lot of thought has gone into this pak, and it should prove a godsend to anyone managing anything from a single car up to the maximum 20.

Carfax is available from:

CLS Software
19 Wellington Terrace
Knaphill
Surrey GU21 2AP
Tel: (04860) 87680-

The price is £29.95

256k Flash Datapak reviewed by Kevin Ash

This new device is just like the flash paks that I use on the Psion MC; great to have the data secure and yet no hassle with erasing.

The flash datapak can be placed in either the B or C slot of the Organiser. You can then turn the Organiser on and press the ON/CLEAR key an extra time. This loads some special software (about 4.5k) into the Organiser's device memory. Once the software is copied, a new menu item "Flash" will appear near the end of the Main Menu.

You can then use it just the same as any other datapak. When you delete or update any info on the pak, it is crossed out and you lose the space. In the end the pak will become full with both deleted entries as well as current info (just as on a normal datapak)

When you cant to erase your flash pak, you will need either another flash datapak or a flash datapak formatter program pak. You then simply select "Flash" from the menu, which asks you which device you wish to format (B or C). After this you are given one last chance to change your mind. The display will then show a bargraph, which indicates the progress of the formatting; this takes only a few minutes and then you are ready to start using it again.

More on Flash Datapaks

Since I needed memory expansion on my LZ badly, I bought a Flash Datapak. Maybe you would like to know my first experiences. Well, I am quite happy with it in general. It offers a lot of memory that is for sure but it also costs a lot of money.

I was lucky to have a flash datapak formatter at the same time, because the flash EPROM is not preformatted! Nobody tells you in advance, unfortunately.

Reviews & Feedback - 2

Up to now I have encountered only one problem: downloading paks from the Developer (MAKE or AMAKE). On my PC it is not possible. I figure it has something to do with the booting software that is always present on the flash EPROM, at least that is what the errors are indicating.

**Theo Versnel
Netherlands**

For Sale

2 x 128k datapaks - £45 ea
64k datapak - £25
16k datapak - £7
Harvester Superchip (2 line) - £45
Travel Pak PLUS (2 line) - £45
Oxford Spell Checker - £15
LACE (Log,Action,Credit Card,Expenses) (4 line) -
£25
110v (U.S.) mains adaptor - £8
Nylon Organiser Case - £4
AutoScribe Plus - £25
Filemaster 4 (advanced file handling) - £25
Contact Steve Clack on (0869) 249287

Psion Travel Pak - £20
AutoScribe Plus - £20
(prices include postage within Europe)

**Contact: Percy Johansson
Namndemannav. 170
14557 Norsborg
Sweden**

128k Datapak - £50
Wanted RESULT by Widget (current version)
would consider exchange!

Contact: Brod Mason on 081 469 0566 anytime

Information Wanted

A member would like information on connecting a Commodore Amiga A500 to the LZ

**Anyone who can help should contact:
Robert Lowe on (0923) 268740**

OPL Percentage Functions on the LZ

Recently, whilst writing a suite of PC based software to untranslate OB3 files back to OPL (a disk containing some of my most important OPL source listings was accidentally re-formatted), I came across some OPL functions for the LZ which are rather obscurely documented in the manuals. These are the percentage functions mentioned in the Operating Manual, Page 8-3 as functions that can be used in CALC mode.

Whilst the programming manual on Page 4-2 does allude to these functions in the OPL context, the manual does not describe how to use them.

Thus to add a percentage to a sum, the OPL statement would look like:

$$B=A + (C) \%$$

where B is the result, A is the initial sum, and C is the percentage figure to add. C can be an integer variable (i.e. C%), a floating point variable (i.e. C) or an actual number (i.e. 12, 12.2). If a variable is used the parentheses are essential, otherwise not.

The other percentage functions described in the CALC section of the Operating Manuals are used in the same way

Tony Spencer

Bug in the Thesaurus/Spell Checker

The TSC has a curious bug in the Anagram Solver department.

If you enter a 10-letter anagram, then the last letter is omitted if it occurs 2 or more times.

For instance, if you enter TENDERSITE instead of getting INTERESTED you get TENDERISE.

A more extreme example occurs with OOOOOOOOON (9 Os) which returns NO and ON.

Thanks to **Paul Barrett** for pointing this out.

£49.95

WHY PRESS 254 KEYS? WITH CUBSOFT'S FNKEY YOU DO IT WITH 2!

Keyboard Macros

- * Store and Recall Entire Words, Phrases and lengthy Menu Paths with just 2 Keystrokes!
- * Have you ever pressed the wrong key and had to key in an entire sequence again?
FNKEY users enjoy complete accuracy!
- * Automate repetitive Program Loading and Execution!
- * All the Macros you need stored Permanently and Retrieved instantly!

Cut & Paste

- * Eliminate those tedious programming sequences by Copying and Moving sections:
 - Between Different Programs
 - Within a Program
 - Blocks of Text
- * Ideal for Really Effective and Economic Diary Management!

Punctuation Marks

- * Directly Accessed from your Keyboard
- * Comprehensive Range of Symbols including:
 - Punctuation Marks and £ sign
 - Full Psion Character Set
 - Instant Access to Upper and Lower Case

Memory Resident - No Messing with Data Packs

Comprehensive Manual and Quick Reference Guide

What Experts and Users Both Say:

"What Cubsoft have managed to do is nothing short of a minor miracle!"

"I am sure that this system will take its place among the Organiser best-sellers."

"FNKEY is one of those (rare) programs which, once you have used it, make you wonder how you managed previously without it!"

Various IPSO FACTO Reviews

"I purchased FNKEY from you just a week ago and already I find it indispensable - what a superb program."

Barry Vaughan, Doncaster