OPL for the Pocket Book

Lee Calcraft reviews Acorn s Palmtop Programming Pack.

The OPL programming language was developed by Psion for their Organiser range, and appears in an enhanced form in their Series 3 palmtop. It is one of the features of the Series 3 which are not supplied as standard with the Pocket Book. But Acorn have just launched OPL as an add-on for their clone. It comes on a flash EPROM card, and fills the remaining drive bay in the Pocket Book. The accompanying 320 page manual, which follows Psion s almost to the letter, is well organised and easy to use.

Figure 1.
OPL on the Desktop

So what is OPL, and who needs it? The letters OPL stand for Organiser Programming Language. It is a sophisticated high level structured language, which is very like BBC Basic in many ways, while at the same time owing something to the influence of C. With more than 200 dedicated keywords it provides ample graphics commands for controlling the screen display, including full support for on-screen windows, a range of string and file handling functions, and even high level database functions - not to mention event-handling functions which allow your programs to multi-task with the suite of resident applications. And if you are having trouble remembering keywords, the OPL Pack automatically provides a searchable database of keywords and their meanings, accessed from the Cards icon.

INSTALLATION

Installing the OPL Pack is simple. You just plug the OPL card into the spare drive bay, and switch on. Then call up the Desktop, click on Menu, and move across to Apps, and select Install application. Once you have done this the OPL icon will appear on the Desktop. Selecting this (by pressing Enter) takes you directly to the OPL editor. This is a full screen editor with many features of the Write word processor. It provides an excellent development environment, and includes facilities like search and replace, as well as the ability to translate and run programs interactively.

Figure 2.
The OPL editor in use

Note the term Translate: OPL, as implemented by Psion/ Acorn, is a compiled language. This means that you cannot just type in your program and run it directly, as you can with BBC Basic. An intermediate stage, which Psion call translation, must be performed to create a piece of executable code. In practice this is quite painless: the translator can be called up from the OPL editor s Menu, or by the Acorn-T key combination. The compiler works quickly, and if errors are spotted you are taken to the appropriate point in the editor so that you can fix the problem. Once translation has been

```
PROC test:
 gborder 3
 gat 20,40
 gfill 100,30,0
 print "TEST PROGRAM"
 tt:
 gat 150,10
 gclock on, $25
 get
ENDP

PROC tt:
 print
```

Figure 3.
An Example Program

performed, you are offered the option to run the program, and away you go.

A REAL EXAMPLE

To give a feel for the way that OPL works, we will take a real example. The following program demonstrates the way in which procedures are used, and shows one or two graphics commands. When it is run it creates the display shown in figure 4, containing a bordered screen, some plain text, a filled rectangle and a clock.

In a similar way to the C language, OPL programs do not have numbered lines, and must begin with a procedure which is automatically called when the program is run - though unlike C, you can specify its

Figure 4.
Running the test program

name. In the example, we have called our starting procedure PROC test. This begins by drawing a shadowed graphics border around the current window (full screen) - the 3 specifies the type of border.

The graphics cursor is then moved to position 20,40 (the origin is at the top left of the screen, and the full screen is 240 x 80 pixels in size). We then draw a filled box 100 pixels by 30 (the third parameter specifies the type of plot - set, clear and invert are offered), and then we print the words TEST PROGRAM. Note that this is printed at the text cursor, and so appears at the top left-hand corner of the screen.

Next we call the second level procedure PROC tt. This simply prints a second line of text to let us know that the procedure was successfully called. Finally we place the graphics cursor at 150,0, and put a clock on the screen. The parameters supplied with the gclock on command specify that it should be a large analogue type with a moving seconds hand - yes, the clock is continuously updated without any specific support from the program.

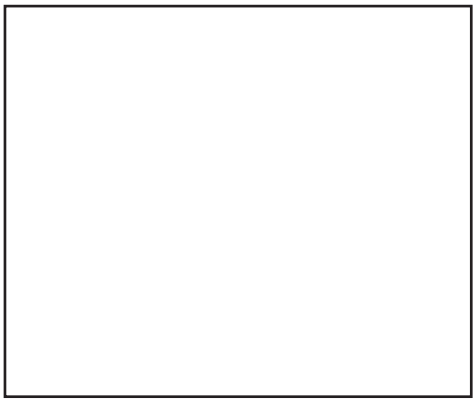
The final get in the program just waits for a keypress before the program terminates. If you have run it from the editor, you are taken back there as soon as you press a key.

Once you have successfully translated a program you will see that a second OPL icon appears on the Desktop screen. This is the icon for runnable programs. From here you can directly run any program that has already been translated. You can also install any such program with its own icon on the Desktop, and with its own special keyboard shortcut.

CONCLUSION

As you will appreciate, the OPL Programming Language (as Acorn s pack is called)

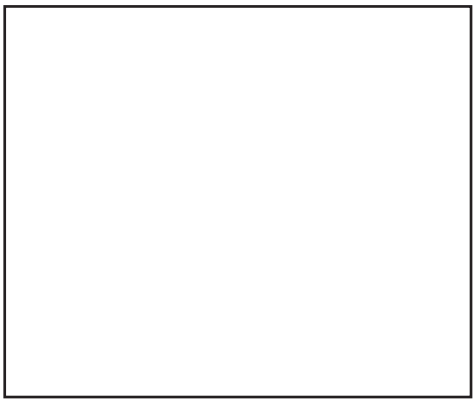
confers great power and flexibility to Pocket Book users. If there is some application which your Pocket Book lacks, then you can just run one up to fit the bill. Moreover, you can even create special functions for the Pocket Book calculator in exactly the same way. And if you find the keyboard a bit fiddly for programming, then you can use the Arc. Create your program in Edit, and then use the A-Link to translate and run it on the Pocket Book.



caption

Product
OPL Programming Language
Price £69.95 inc. VAT
Supplier Your Acorn Dealer

Calling OPL programmers:
if you have any interesting OPL programs for the Pocket Book or Series 3, please drop us a line.



caption