POWER PACKS

We look at four spreadsheet-based programs for home micros — Micro Swift, Practicalc II, PS and Vizastar — packages that some believe prove that 'the ordinary home micro has enough power to compete with the bigger business systems'.

Micro Swift, Practicalc II, PS and Vizastar belong to a new breed of enhanced spreadsheet-based packages, which have clearly had their inspiration from Lotus's integrated 1-2-3 package and its successor, Symphony (see page 644). But whereas the Lotus 1-2-3 and Symphony packages were written for the IBM PC and compatible machines (1-2-3 requires 296 Kbytes of user memory to run, and Symphony demands at least 320 Kbytes), the new packages are designed for home micros. In some respects, the four packages we look at here have wrought miracles in compressing many of the features available on the larger packages into the 30 Kbytes or so of memory available to the user of micros such as the Commodore 64.

However, as yet, these 'mini-Symphony' packages can offer only two of the four options that make the more powerful (and expensive) packages so attractive. Given current hardware limitations, to try to incorporate all four options — spreadsheet, database, word processor and programmability — would undoubtedly necessitate the sort of compromises that have made the Three-Plus-One ROM-based software of the Commodore Plus/4 something of a disappointment (see page 709).

RELATIVE STRENGTHS

Let's consider some of the options offered by these four programs, to compare their relative strengths. PS, Micro Swift and Vizastar are all programmable, to a greater or lesser extent. This is an extremely valuable facility, since it allows the user to automate functions that would otherwise require many keystrokes to carry out — in the same way that keyboard macros are used with Lotus 1-2-3 (see page 784). The three programs do this in different ways, and we will consider their separate approaches in turn.

Modules are programmed on the PS package using familiar BASIC commands. These modules are then saved by pressing <f3> and executed using <U>, or, alternatively, they can be auto-executed on loading by SAVEing them to disk with a full stop after the program name. The package has a range of helpful programming facilities: for example, it can GOSUB to a subroutine in a

program from a formula within a cell, simply by inserting the GOSUB command within the formula. Functions can be defined using the FN function, and the program also has the facility to pass string, row and column, and numeric values.

Micro Swift can be programmed by simply placing a list of commands in column Z — the first command giving the name of the program, preceded by a hash sign (#), and the last line containing the command @QUIT. Let's consider a simple example:

```
Z1 #SUM
Z2 @SUM(A1,A3)
Z3 @ASSIGN(Z2,A4)
Z4 @QUIT
```

This program will add the values contained in cells A1, A2 and A3, and then assign the value, now found in cell Z2, to cell A4. The program is called with the instruction #SUM.

Of all the packages considered here, perhaps the simplest to program is Vizastar, since the commands consist of the initial letters that would be pressed to execute them manually. Thus, to use a specific database, you would press the CBM key followed by D(ata), U(se), D(atabase) and the name of the database. Finally, you would press <RETURN>. In programming, the slash sign (/) is used in place of the CBM key, so that / DUDname[RET] will execute the action if <f8> is pressed. Function and editing keys are programmed by pressing <CTRL> plus the appropriate key, and this letter is printed out when the function is used in a program. However, when the cursor keys are programmed in this way, they are printed as [up], [down], [left] or [right].

Vizastar's database is a powerful implementation, actually using a section of the notional sheet (rows 1,000-plus) not otherwise available to the user, to store record formats. Each record can consist of up to nine screens, and they can be accessed by the Key or Next, Prior, First, Last or Current commands (each utilising the initial letter from a command menu). Records may also be Added, Replaced (modified) or Deleted.

The fields have letter names, starting with A and finishing with BK, which relate to the columns of that name in the spreadsheet. Therefore, as an example, search criteria can be set up on a blank line of the spreadsheet. A is always the key field — the field on which data is sorted.

Practicalc II is a spreadsheet that uses a 'long label' facility, which allows text to spill over from one cell across any blank adjacent cells. This facility allows the program to operate as a word processor with a maximum line length of 100