More comfortable than basic

64'er Ausgabe 7/July 1984, p. 66-67

The Forth programming language is available for more and more microcomputers, and not without reason. Forth is a powerful language that in a sense combines the positive sides of Basic, Pascal and Assembler. For the Commodore 64, one of the most powerful Forth versions is now available as a plug-in module: the 64 Forth from Human Engineered Software.

This Forth version for the C 64 is a FIG-Forth with a greatly expanded language scope. The abbreviation FIG stands for Forth Interest Group and refers to an independent group of Forth enthusiasts who have set themselves the goal of standardizing and disseminating Forth. With a language like Forth, which can be expanded and changed by the user at will, such a standard also seems urgently necessary, because otherwise the exchange of software between users would be almost impossible. The 64 Forth adheres very closely to the FIG standard, which directly benefits the user: Most Forth programs developed for other computers are, unless special hardware properties are exploited, immediately run on the C 64 with minimal changes.

Forth is generally already characterized by a large supply of commands. In 64 Forth, however, the basic vocabulary of the FIG standard was again greatly expanded. Over 500 commands are available. To keep track of this, these commands are divided into four vocabularies: FORTH, EDITOR, SYSTEM and ASSEMBLER. A vocabulary is simply called up or activated by specifying its name.

At the beginning, only the normal Forth vocabulary is accessible. If you want to enter longer forth programs or texts, then you simply call up the editor vocabulary and can now use all existing commands to edit texts. The system vocabulary contains commands to access the operating system level.

Although Forth programs usually run faster than basic programs, time-critical situations can occur from time to time. In this case, you call up the Assembler vocabulary and immediately have a 6502 macro assembler available.

Already at a first look at the table of contents of the (unfortunately only English) manual it is noticeable: The 64 Forth has a whole range of commands in addition to the FIG-Forth standard to support special C 64 properties.

With BGROUND and BORDER, for example, the background and the frame color are selected. With NEWSPRITE you can design new sprites very comfortably within a screen mask. Sprite data can be written into or read from special sprite files. With the SHOW command, sprites are made visible, with HIDE they disappear from the screen again. With

other commands you can move, color and enlarge sprites across the screen. Of course, the multi-color mode is also possible.

Anyone who has ever struggled with the various POKE commands in Basic to move a few sprites over the screen will appreciate these possibilities of the 64 Forth very much.

It is very similar with the sound effects: Where in Basic you need many lines with POKE commands to play a few notes, it is possible with 64 Forth in plain text and much clearer. The command VOICE1, for example, selects the sound generator 1, with TRIANGLE, SAWTOOTH, SQUARE or NOISE the waveform is selected. Other commands control frequency, envelope, volume, and other factors. In total there are 40 (!) Commands to control the synthesizer.

Also the control of external devices is no problem with 64 Forth. In the SYSTEM vocabulary there are various commands for data communication with printer, floppy or cassette. For example, the PRON command replaces the basic command sequence "OPEN 1,4:CMD1". To send commands to the floppy, it is not necessary to open a command channel, but the CMD command is sufficient. For example, to format a disk, write in 64 Forth: CMD N:Name,ID.

In addition, most of the kernel routines of the operating system can be easily called up by name, resulting in a high degree of flexibility in data exchange via the serial bus.

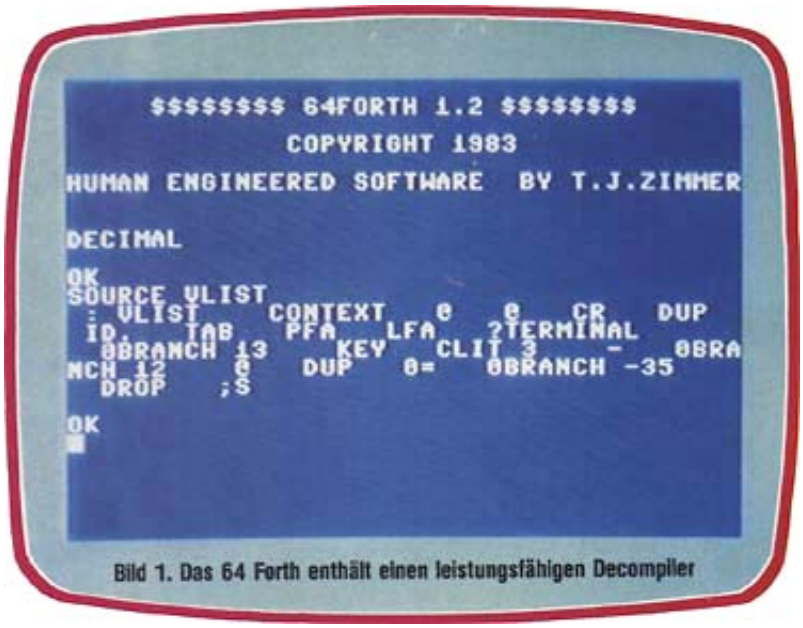

Bild 1. Das 64 Forth enthält einen leistungsfähigen Decompiler

Noteworthy are the quite comfortable test aids provided by 64 Forth. The TRACE, STEP, EMULATE, and CONT commands can be used to monitor the execution of Forth commands. In addition, there is a decompiler that is called with SOURCE and can translate compiled Forth words back again (Figure 1). Such extensive debug functions are usually searched in vain with other Forth compilers.

In contrast to the fairly simple memory management of Basic, Forth uses the concept of virtual memory. Simply put, this means dividing the available memory into small sections, so-

called screens or text boxes, which are accessed via a number. It is not necessary that all screens are in the main memory at the same time. Most screens are therefore created as a relative file on a floppy disk and are only loaded when required.

64 Forth offers the special feature that such virtual memory management is also possible without a floppy. The individual text fields are recorded in the order of their numbering with the Datasette like normal programs. A buffer memory of 16 text fields in the RAM limits the number of cassette operations required.

Each text box can now be edited by itself with the text editor. For this purpose, almost all Forth versions only have a simple line-oriented editor. 64 Forth proves its compatibility by also having such a cumbersome line editor, which is called quite normally with n EDIT, where n is the number of the text field to be edited.

The full-screen editor

In addition, 64 Forth also has a "full-screen editor", as it is known from the basic. In edit mode, press the key combination "Shift" and "INST/DEL", and the entire text field to be edited appears immediately. As usual from Basic, you can now use the cursor keys to move over the entire screen and delete or simply overwrite incorrect text passages.

The function keys are occupied with various helpful functions, for example forward and backward tab and search functions. At the touch of a button you can switch to the previous or subsequent text field. As a further special feature, the editor understands a series of "Wordstar" compatible control functions. Instead of using the "CTRL" button, however, these functions are triggered by pressing the "Commodore" button and a letter at the same time. So if you are used to working with "Wordstar", you will also find your way around with this editor.

Very hard-hard Forth programmers, who already find so much ease of use decadent, still have the opportunity to work with the line-oriented editor instead.

The too lean manual is written in English, a German translation is not planned at the moment.

On 155 small-format pages a short introduction to Forth is given, but this probably won't tell the beginner too much. In short chapters, the text editor and the most important Forth commands are explained. The following are sections on Forth's storage, I/O organization, and the integrated 6502 macro assembler. Listings of a number of Forth utilities and a brief description of all commands round off the manual.

All chapters are quite short and concise, so that one is often forced to either look up other books or simply experiment around. After all, you can find the most important information. Nevertheless, a German, somewhat more detailed manual would certainly be desirable.

Overall, the 64 Forth is certainly one of the best Forth versions currently available for microcomputers. Clear plus points are the excellent editor and the extensive set of

instructions, especially the many commands for sprite graphics and sound generation, which are sometimes sorely missed in the Commodore Basic. The integrated assembler is a valuable help in solving time-critical problems.

Picture 2. The 64 Forth is available as a plug-in module for the C64 and the VC 20

The 64 Forth can be obtained in Germany as a plug-in module (picture 2) for the C 64 via "Die Forth-Quelle" in 7820 Titisee-Neustadt for the price of 198 marks. A version for the VC 20 is also available. Unfortunately, according to the suppliers, the module is not runable on the portable SX 64 due to small hardware differences.

(ev)