

## Installation Instructions

## DISK INSTALLATION

This version of fig-FORTH is designed to be used with the FDC-1 disk controller on the SYM-1, using double-density 5.25" disks, formatted for 1024-byte sectors. The delivered version is configured for a 32K-RAM system and 40-track drives. Cassette linkages are also included.

The fig-FORTH Installation Manual is required and the Source Listing is recommended. These are available from the FORTH Interest Group, PO Box 1105, San Carlos CA 94070. In Canada, contact WEDJ Micro Products, 39 Rockfield Cres, Nepean Ont K2E 5L6.

Before you load FORTH into RAM, you must first set up SYM-DOS to use 1K sectors, with its buffer at the top of memory. Enter the following code at \$7B00:

```
7B00- 20 06 90 JSR $9006
7B03- 20 86 8B JSR ACCESS
7B06- A9 7B LDA #7B
7B08- 8D 2B A6 STA $A614 ; buffer is at $7B80
7B0B- 85 FD STA $FD ; set pointer too!
7B0D- A9 28 LDA #40 ; for 40-track drives only!
7B0F- 8D 14 A6 STA $A614
7B12- 60 RTS
```

Now do a .G 7B00. The disk system will be initialized with the workspace at \$7B80 and the buffer at \$7C00. Load the system disk in drive 0 and do a L3 FORTH 0. Cold start FORTH with a .G 200. The warm start is .G 204.

For a 24K system the following changes will be necessary. First, change the 7B at location 7B07 to 5B and load the code above at \$5B00. Load the FORTH system as above, and immediately make the following changes (before doing .G 200).

```
0211:59 0897:4D 08A3F:59 13C5:4051 13DF55:51
```

Location \$020E contains the character that FORTH will recognize as the DELETE character for input, and that will automatically be echoed as a backspace. The delivered system assumes the ASCII backspace (\$08). You should set this location to the value of the key you will use as a deleting backspace.

If your drives are not 40-track, the following change is necessary so that the switch from drive 0 to drive 1 is made properly. Change locations \$1C6C,D to \$00AF for 35-track, or to \$0190 for 80-track (low byte first!). FORTH stores its source code on the disk in 1024-byte "screens". In a 40-track system, drive 0 contains screens 0 to 199, and drive 1 contains screens 200 to 399.

The accompanying assembler listing gives the major changes to the fig-FORTH Release 1.1 source listing. Other modifications have been made throughout the code to patch in the SYM I/O routines and to accommodate the 1024-byte sectors.

## USE

The FDC-FORTH disk-map is as follows. Tracks 0 through 6 (FOP screens 0 through 34) contain the standard SYM-DOS directory and boot-up FORTH object code. The exception is screens 4 and 5 (trk0:sec5 and trk1:sec1) which contain the system error messages. The size of the object-code block has been deliberately made larger than the actual object code. This is so that an expanded FORTH boot-up file can be stored over the old file by setting FIXBLK (\$A624) to non-zero (see SYM-DOS guide, section 5.5). The remainder of the disk (screens 35 through 199 in a 40-track system) should contain FORTH source screens. The system makes no distinction between disks recorded in drive 0 or 1: track 7/sector 1 is screen 35 in drive 0 and screen 235 in drive 1. This feature makes shuffling screens between drives very easy. A simple disk-copy utility can be written to copy screens 0 through 199 to screens 200 through 399, although the copying process takes over 9 minutes!

Screens 100 through 122 on the disk supplied contain the source code for all additions to the fig model that are included in the boot-up object file. The fig-model editor source is on screens 102 through 113, and the source for Bill Ragsdale's assembler is on screens 114 through 119. For details on the operation and use of this amazing assembler, see the September 1982 issue of Dr. Dobb's Journal (No. 71) or FORTH Dimensions Vol III Number 5.

Screens 130 to 135 contain the source code for a simple guessing game. Type 134 LOAD (ret) and when the OK prompt appears, type GUESS (ret).

## MODIFICATION FOR CASSETTE OPERATION

Several more changes are necessary to operate with cassettes. Location \$021A should be set to 00. This is the flag that tells the operating system whether or not the error messages are on screens 4 and 5. Also, location \$1AF4 should be set to 00. This is the contents of the variable D/C which is used by the routine R/W to switch between disk and cassette. These two changes can be made from the FORTH environment by typing CASSETTE (ret) and 0 26 +ORIGIN ! (ret). Lastly, location \$1A9B should be set to your terminal's clear-to-end-of-line code. If your terminal does not support clr-to-eol, set this location to zero. This is the last character EMITTED by the routine CLMSG, the source for which is on screen 100.

If you have both disk and cassette, you can switch between them using the commands DISK and CASSETTE, which merely alter the value of the variable D/C.

## CASSETTE INSTALLATION

The cassette version contains the object file first, followed by the source screens for the cassette and FDC linkages and the start-up patch. The next five screens are the guessing game. The cassette object file is already modified as outlined above, with the exception of the clr-to-eol code. The file number for the object code is 71, which is displayed during a load as an "F". The screens are numbered starting at 4.

(may (or may not be!))

## CASSETTE USE

When the cassette is used for mass storage, the operator is prompted to start the cassette. For example, issuing the command "3 LIST" will result in the message "READ SCR 3" and the sync indicator will come on. You should position the tape to the start of screen 3 and press PLAY. The next file encountered will be read in -- the file (screen) number is ignored. For writes the message is "WRITE SCR 3" and the system waits for a keystroke before beginning the write. You should position the tape, press record and play, then hit a key.

FORTH uses the "virtual memory" concept to provide a large workspace in a small amount of memory. Basically, data (in this case the FORTH source screens) are shuffled onto and off of mass storage into the RAM buffer areas as needed. This works fine for disk, but can become tedious with cassettes. The operating system knows when the information in a given buffer has been changed since it was read off the mass storage device, and will write it back before using the buffer for something else. With cassette, this means that the user can be shunting the tape back and forth, reading and writing blocks under the direction of the operating system.

One way to make this much less tedious is to use more than the three buffers the standard system is set up for, and to anticipate when the operating system will want to write blocks to the cassette. With a 32K cassette-based system you could have as many as ten buffers, and you can squeeze 6 buffers into a 24K system. The locations to change and the calculations are as follows.

MEMTOP = top of memory (e.g. \$8000)  
 0210/11 ← MEMTOP - \$80  
 0896/97 ← [0210/11] - (\$0404 \* #BLOCKS) (e.g. \$5158)  
 0BAE/AF ← [0210/11]  
 13C4/C5 ← [0896/97]  
 13CF/D0 ← [13C4/C5] + \$0404 (e.g. \$555C)

If you are using ten buffers, it is convenient to input, test and compile your programs in chunks of 10 screens or less. That way, your entire source for the chunk is in RAM (no cassette operations) so compilation and editing proceed very rapidly. When the chunk of 10 screens has been debugged, they can be written to cassette with a FLUSH command, and a new chunk of source screens created. I personally used this system with a 6-buffer 24K system for over a year with good results. I adopted a scheme where each screen began at 10 times it's screen number on the tape counter. While this wasted a lot of tape, it made over-writing blocks very easy, and still got over 20 screens on one side of a C-30.

## FDC-FORTH

### Addendum to Installation Instructions

The FDC-FORTH diskette is shipped with a write protect tab installed to prevent accidental "clobbering" of the screens supplied. It is suggested that you make a back-up copy, as follows:

First format a blank diskette in drive 1 with .89 1,1,3 <cr>.

Then, after entering FDC-FORTH with a .G 200, enter the following sequence of instructions:

```
EMPTY-BUFFERS <cr> ( ALWAYS after cold start )
136 LIST <cr> ( just to "study" what it will do! )
136 LOAD <cr> ( to compile : DUPLICATE )
DUPLICATE <cr> ( to do the copying )
```

Enter the <cr> when requested, and the back-up copy will be generated.

In the event that you wish to prepare a "customised" version of FDC-FORTH, enter your new object code as the first and only file on a properly formatted diskette, and use 137 LOAD to copy the original screens to the new diskette. You may, of course, modify the range of screens to be copied by entering the appropriate values in Screen 137.

Revision 0: 3 June 1983 - SYM Users' Group