



DOCUMENTATION FOR
microFORTH on RCA COSMAC 05

FORTH, Inc.
815 Manhattan Avenue
Manhattan Beach, CA 90266

(213) 372-8493

microFORTH INSTALLATION INSTRUCTIONS

You will find another copy of the following bootstrap procedure on the diskette's microFORTH label. Turn on the terminal, disk drive, and CPU. Remove the microFORTH diskette from its protective jacket and insert it in Drive 0 (the upper drive) firmly but without forcing it. The elongated slot goes in first, with the label facing up. The diskette should pop into position behind the plastic retainer lip on the right beneath the slot.

1. microFORTH is loaded by using the utility supplied by RCA. This utility operates at any baud rate less than or equal to 1200. To use it, push the RESET and RUN UTILITY switches on the front panel. Then type a carriage return on the terminal to enable the utility to determine your baud rate.

2. The utility will respond with an asterisk (*) on the next line. At this point you should enter:

\$L

followed by a carriage return.

3. The utility will reply:

READ?

to which you reply with a carriage return.

4. microFORTH will load in about one minute. (If you will be operating at a baud rate greater than 1200, you should change it now. See "Baud Rate Instructions" in this Documentation.) When loaded, FORTH will send an OK to the terminal.

5. If the terminal has not responded correctly, recheck your procedure carefully and attempt to boot again. If trouble persists, call FORTH, Inc.

6. If your terminal responds properly, the FIRST thing to do is to create a backup copy of the microFORTH diskette! To do this, follow the "Disk Backup Procedure" in these instructions.

In the following procedures, remember that commands are always separated by one or more spaces and that a command sequence must be terminated by a carriage return. No action will be performed until the carriage return has been issued. Whenever an operation is successfully completed, FORTH responds with an OK .

If a typing error is made in a line before the carriage return has been pressed, the RUBOUT key may be used to move back over the input line so that the items can be retyped. If an attempt is made to RUBOUT at the start of a line, the terminal bell will ring.

Please work through the examples given in the microFORTH Primer at the terminal and try some experiments of your own to become more familiar with FORTH. Any time you want to eradicate any interim definitions you have made, type:

FORGET TASK : TASK ;

microFORTH BAUD RATE INSTRUCTIONS1.0 BAUD RATE SETTINGS

Screens 15 and 17 of your listings contain the source text for ECHO and KEY, the two fundamental terminal I/O words. On Lines 5 and 11 of each screen you will find constants preceded by comment asterisks (*). By editing these constants according to the following table, you can obtain any baud rate.

<u>Baud Rate</u>	<u>Screen 15</u>		<u>Screen 17</u>	
	<u>KEY</u>	<u>ECHO</u>	<u>KEY</u>	<u>ECHO</u>
110	FF	FF	--	--
150	C2	B5	--	--
300	60	59	D9	C9
600	2F	2B	6B	62
1200	17	15	33	2E
1800	0F	0D	21	1D
2400	0B	09	18	15
4800	--	--	0A	08
9600	--	--	03	01

The baud rate after booting is selected by editing Line 7 of Screen 3 to load either Screens 15 or 17 (0F or 11 in hex).

2.0 CHANGING THE BAUD RATE IN MEMORY

Occasionally the user's terminal will not recognize the baud rate generated by the microFORTH system delivered to the customer, and conversely.

If this is the case, it will be necessary to alter those constants in memory that govern the baud rate timing loops. These constants are stored as the data bytes of "load immediate" instructions in the microFORTH code words KEY and ECHO. The words KEY and ECHO are defined in Screens 15 or 17 of your listings (only one screen is selected during bootstrap, depending on the baud rate requested by the customer; Screen 17 is selected only if the customer requested a baud rate greater than 2400). The addresses of these constants are:

	<u>Screen 15</u>	<u>Screen 17</u>
ECHO	A40	A40
KEY	A61	A62

The values in these locations are as listed in Section 1.1.

Boot up following the bootstrap instructions on page 1, then get into the utility by pressing the RESET and RUN UTILITY Buttons, followed by a carriage return.

Now interrogate memory, using the ?M command. For example, to confirm that your diskette loads at 1200 baud using Screen 15, you enter:

?M0A61 01

and should receive the answer 15. Similarly,

?M0A40 01

should be answered by 17.

Having confirmed the locations and values of the constants, you can alter them with the !M command. The symbols !M are followed immediately by the address to be modified, a space, and the two hex digits to be stored there.

Begin by fiddling with the constant for KEY . After fixing KEY , place a slightly smaller value in the constant of ECHO (typically, 1 less if Screen 15 is loaded, 2 less if Screen 17 is loaded), and return to microFORTH by pushing the buttons for the RESET and RUN PROGRAM.

When both constants are set correctly, you have an operational system. You should now edit these constants directly into Screen 15 or 17. To do this, type:

EDIT LOAD

After the EDITOR is loaded, type either 15 LIST or 17 LIST . In the definitions of KEY and ECHO you will see the constant preceded by the comment (*). To alter these lines, type a line number, a space, and a T (for type). The designated line will be typed back, indented two spaces. On the next line type a P (for put), a single space, and a copy of the text that that was typed, changing only the constant.

After changing both constants, type FLUSH , to write them onto the disk.

microFORTH DISK BACKUP PROCEDURE

1. Type:

DISKING LOAD

2. Place a blank diskette in Drive 1 (the lower drive).
3. Copy the entire diskette from Drive 0 to Drive 1 by typing:

BACKUP

This will take several minutes.

4. Verify that no errors occurred by typing:

250 500 SWEEP

This will read all screens on the new diskette successively, reporting any errors encountered.

5. If errors are encountered, the following remedial procedures are recommended:

- a) Repeat the process starting with Step 3 above.
- b) Try another diskette.
- c) Consult RCA about possible hardware malfunctions in the disk drive.

6. When a good copy has been produced, mark the backup diskette with an identifying label and put it in a safe place.

microFORTH SYSTEM RESTART

The COSMAC microFORTH system comes with a resident restart routine that will return control to the terminal whenever the program counter is RESET to zero or one. This can be achieved by depressing first the Reset and then the RUN PROGRAM buttons on the front panel of the development system.

The microFORTH system will stop whatever it was doing, reinitialize Registers R, U, F (the location of NEXT), and I, reset both stacks, echo whatever word it had last interpreted, and output the message: RESTART! The block number from which the last word has been interpreted will be placed under the stack as for error messages.

microFORTH RECONFIGURE MEMORY

The purpose of the Reconfigure screen (Screen 28) is to alter the microFORTH precompiled program on disk so that a subsequent boot will occupy either more or less memory and/or contain a different number of block buffers. The memory size and number of buffers are each controlled by the values of a specific constant on Line 1. To change either of these parameters, perform the following procedure:

1. Perform a simple verification of the new memory. (You can use MOVE to make one pass each of two bit patterns through every location and then inspect the last location. For example, use all zeros first and then all ones.)

2. Be sure that you are familiar with the DISKING and EDITOR procedures and that you have a bootable backup system disk to fall back on in case of an error.

3. With the disk to be altered in Drive 0, use the EDITOR to change Line 1 of Screen 28 to the appropriate value (in hex) for the number of buffers to be allocated (NB) and/or for the number of bytes of memory to be used (LIMIT). For example, to allocate three buffers and 9K bytes of memory, you would use:

```
3 CONSTANT NB and 2400 CONSTANT LIMIT .
```

4. Double check your changes. Then type 28 LOAD . The system will type out three numbers, then OK . Write the numbers down because they will be used in subsequent steps. At this time, the precompiled program for the disk on Drive 0 will have been altered. Before rebooting, you must make several source changes on the altered disk.

5. Reload the EDITOR and list Screen 3. Edit Screen 3 to change the literal start address and ERASE count in the definition for ERASE-CORE on Line 12. The new values are, respectively, the first two numbers typed out by 28 LOAD in Step 3. For the current example (i.e., 16 buffers and 32K), the new definition should be:

```
: ERASE-CORE 2274 18C ERASE ;
```

6. If the number of buffers has been changed, alter the loop limit in the definition of FLUSH in Line 13. For the current example the definition would be:

```
: FLUSH 3 0 DO 0 BUFFER ! LOOP ;
```

7. If the number of buffers has been changed, also edit Screen 24 (DISKING), Line 1, to change the value of the constant NB .

8. Double check all changes. When you are satisfied that they have been made correctly, FLUSH , and reboot.

9. If the altered disk will not boot, try configuring your disk for its original configuration by repeating the procedure from Step 1. Take extra care to verify each change. If you can do this properly, your disk should match the backup (see MATCH in DISKING).

STRING EDITOR OPERATING INSTRUCTIONS

List the screen to be edited and type `EDITOR` .

The user variable `R#` holds the relative location of the editing cursor within the current screen. The cursor may be thought of as lying between two characters of a line or before the first character of a line.

The command `TOP` positions the cursor before the first character of the current screen.

The `EDITOR` command `T` has been defined to position the cursor before the first character of the numbered line.

The string editor itself (Screen 21) has the following commands:

`M` `F` `N` `B` `C` `X` and `TILL`

The last step in the execution of each these commands is to show the current line that holds the cursor in the following format:

HERE IS THE CURSOR_ FOLLOWED BY MORE LINE 7

The position of the cursor is indicated within the image of the line by the ASCII code 5F (hex), which prints either as an underline or a back-arrow. Immediately after the last character of the line, the line number is printed (with no intervening spaces). Note that this format differs from that of `T` .

The command `M` moves the cursor by the signed amount on the stack (i.e., negative motion is backward, positive is forward). For example, `-10 M` backs the cursor up over ten characters.

To advance the cursor just past a designated string of characters, type the command `F` (for Find), a single space, the desired string, and a carriage return. If the string does not occur on the current line, succeeding lines will be searched up to the end of the screen.

To repeat the `F` operation on the same string of characters as previously entered, type `N` (for Next).

To back up over the last string of characters found, type `B` (Back). This leaves you positioned before the string rather than after it.

To insert a string of characters at the current location, type `C` (for Change), a space, the exact string of characters, and a carriage return. The change only affects the current line. Characters are lost off the end of the line; succeeding lines are not affected. You may not insert more characters than remain on the current line; the attempt to do so will generate an error. Please note that all errors in the string editor automatically return you to `TOP` , so it may be necessary to reposition the cursor after an entry error.

To delete the next occurrence of a string, type `X`, a space, the exact string to delete, and a carriage return. If the string does not occur on the current line, succeeding lines will be searched up to the end of the screen. The deletion only affects the line in which the string is found. The end of that line is filled with as many blanks as the number of characters deleted. If the string cannot be found, an error is given and the cursor returns to `TOP`.

To delete all characters from the cursor up to and including a designated string, type `TILL`, a space, the designated string, and a carriage return. The end of the line will be blank-filled as for `X`. Unlike `X`, `TILL` may only be used to delete characters on the current line.