

CHAPTER 17. WORDS

The source code in this chapter is in file UTILITY.BLK, screens 3 and 5.

The command WORDS (VLIST in older Forth systems) displays all the words defined in a vocabulary. It is very useful in inspecting the dictionary contents, and also in determining the progress of compilation when a large application is being loaded. The implementation of this command in F83 system is slightly more complicated because the vocabulary is hashed into four threads.

17.1. OUTPUT FORMATTING COMMANDS

To display a sequence of variable length names on the CRT terminal or printed on a printer, it is necessary to wrap a whole word around the end of a line instead of breaking a word. A few commands allows us to detect the end of line condition and insert a carriage return before printing the last word.

VARIABLE LMARGIN	The column number of the left margin. 0 LMARGIN !
VARIABLE RMARGIN	The column number of the right margin. 70 RMARGIN !

```

: ?LINE          ( n --- ) If the current line does not have space for n more characters,
                    move to the next line.
    #OUT @          Current character count.
    + RMARGIN @ > IF Add n to it. If it is greater than the right margin,          CR
                    Move to the next line.
    LMARGIN SPACES Align to the left margin.
    THEN ;

: ?CR            ( --- )      Move to next line if we had passed the left margin already. 0    ?
LINE ;

```

17.2. WORDS

In the parameter field of the definition of a vocabulary command like FORTH, EDITOR, or DOS, etc., there are four addresses pointing to the ends of four thread in the dictionary. They are the link field addresses of the four newest definitions defined in the vocabulary. To print out the entire list of names in the vocabulary, we have to trace through these four threads and print out word names in the descending order according to the addresses. These four addresses are first moved to the top of the dictionary. The largest address is first used to print the name which was define last. This address is replaced by the next address in its thread, and the process is repeated until all names are printed.

```

: LARGEST ( addr n --- addr' val ) Given an address and a number on the stack, return
    the address and the value of the largest entry in the array.  OVER 0 SWAP
    Add 0 and addr on the stack.
ROT 0 DO Scan through the array.
  ≤DUP @ U< Compare contents of an array entry with the current largest
    value.
  IF If the entry is greater,
    -ROT 2DROP discard the old address and its value,
    DUP @ OVER and replace them with the new address and its value.
THEN
  2+ Next address to be scanned.
LOOP DROP Discard the address used for scanning.
;

```

```

: WORDS ( --- ) List the words in a vocabulary. It can be interrupted by
    pressing any key.
CR LMARGIN @ SPACES Align to the left margin.
CONTEXT @ The context vocabulary.
HERE #THREADS 2* CMOVE Copy the array of thread ends to the word buffer.
BEGIN Begin the printing loop.
  HERE #THREADS LARGEST Pick the largest address in the thread array.  DUP
WHILE If it is not zero, print a name. Otherwise, the vocabulary is
    exhausted.
  DUP L>NAME From the link field, move to the name field.
  DUP C@ 31 AND The length of the name.
  ?LINE Make sure there is enough room at the end of a line.
  .ID Print one name.
  SPACE SPACE Add 2 spaces.
  @ SWAP ! Replace the largest address in the thread array by the address
    of the next word in the thread.
  KEY? IF EXIT THEN Exit if a key is pressed.
REPEAT Continue until all names are printed.
2DROP Discard the address and a value, which is zero.
;

```

ROOT DEFINITIONS WORDS must also be defined in the ROOT vocabulary.

: WORDS WORDS ; WORDS can now be accessed any where in the F83 system.

FORTH DEFINITIONS