Arithmetic #

Table of Contents

Stack Notation #

The following words hauptsählich described in its individual function. In this form of description that you have already seen, the effect of a word on the stack is given in brackets and in the following form:

((((Before - after))))

before - values on the stack before execution of the word after - Verte on the stack after execution of the word

In this notation, the top element of stack (tos) is written always right. Unless stated otherwise, all stack notations refer to the later version of the word. In immediate terms, the impact of the word on the stack is specified at compile time. Words are also characterized by the following symbols:

C	This word kana are used only during compilation of: definition.
I	This word is an immediate word, which is running in compile state.
83	Forth83 This word is defined in the standard and must work on all standard systems are equivalent.
U	Features a Uservariable.

If the pronunciation of a word of English pronunciation from the natural, it is indicated in quotes. Occasionally, following a German translation. The names of the stack parameters follow, unless suggestive names have been elected, the following schedule. The names can be provided with a subsequent digit.

	Stack notation		Number type		value range in decimal		minimal width of the field
Flag	logical value	0 = flash, else = true	16bit				
True (tf)	logical value	-1 (as a result)	16bit				
False (ff)	logical value	0	16bit				
B	Bit	0 .1	1bit				
Char	character	0 .127 (0 .255)	7 (8bit)				

8b	Any 8 bits not applicable		8bit
16b	16 arbitrary bits not applicable		16bit
N	Number of weighted, bits	-32 768 .. 32 767	16 bit
+ N	positive value	0 .. 32 767	16 bit
U	unsigned number	0 .. 65 535	16 bit
W	number, n, or u	-32 768 .. 65 535	16 bit
Addr	address how u	0 65535	16 bit
32b	32 arbitrary bits not applicable		32bit
D	double exact number	-2,147,483,648 2,147,483,647	32bit
+ D	positive double-precision number	0 .2,147,483,647	32bit
	Ud unsigned, double-precision number	0 .4,294,967,295	32bit
Sys	system-value	not applicable	Not applicable

People Forth arithmetic #

-1 (- -1) \ \ 0 (- 0) \ \ 1 (- 1) \ \ 2 (- 2) \ \ 3 (- 3) \ \ 4 (- 4) \ \

Frequently used numbers were made constant. Defined in the form:

(((N))) n Constant

This Syeicherplatz savings and reduces the execution time.

- 1 +
- 1 -
- 2 +
- 2 -
- 2 *
- 2 /
- 3 +
- Abs
- Not

- Negate
- Even
- Max
- Min
- +
- -
- *
- /
- Mod
- / Mod
- * /
- * / Mod
- W / mod
- Umax
- Umin

People Forth logic words #

- True
- False
- 0 =
- 0 <>
- 0 <
- 0>
- =
- <
- >
- U <
- U>
- And
- Or
- Xor
- Uwithin
- Case?

People Forth 32bit words #

- Extend
- Dabs
- Dnegate
- D +
- D-
- D *
- D =
- D <
- D0 =
- M *
- * For
- M / mod
- Ud / mod
- Um / mod

Stack Operations #

Conventional programming languages contain more or less pronounced, the concept of PROCEDURES: For certain program functions necessary operators are summarized in designated parts of the program to enable this program functions in several places within a program on their name. Since FORTH procedurally without any restriction, is also FORTH makes no distinction between procedures and functions or its operators. Everything is described as a WORD.

FORTH LANGUAGE as programming is so out of words. Thus can be FORTH-words:

1. Data areas
2. Algorithms (commands)
3. Programs

To use appropriate procedures, can know the most languages, PARAMETER: These are data that a procedure be passed when called for processing. Data that are only used within a procedure called, LOCAL about this procedure, in contrast, is known as data that are outside of certain procedures available and can be accessed by all procedures with all operators, GLOBAL.

The first way of passing parameters between procedures, the agreement of named GLOBAL variables. These global variables are for the entire duration of the program, existing static and can be manipulated by all procedures.

Another way of passing parameters is to set up a storage area in which to call a word during the unnamed parameter dynamically managed. This mechanism for named local variable is standard FORTH not available because the organization of these local variables are associated with a loss of both the local data after the execution of the word as well as a loss in the execution speed of the word.

For the volks4TH was presented in the VD 1 / 88 is an implementation of named local variables.

FORTH used for the mutual surrender of the main parameters of words STACK, a particular memory area in which the words expect their parameters. These parameters will not name names, but their interpretation is clear from the position within the stack memory area. This results in the large number of operators to change the stack location of a value for the FORTH is famous / notorious.

Thus it is clear which and how many parameters need to be a word, widely used STACK COMMENTS:

The opening parenthesis followed by a list of parameters. This is the parameter that is expected as the top stack element, far right. Then follows a "--", symbolizes the execution of the word. Then, the state of the stack after the execution of the word is presented, with the top stack element is back on the right. The closing parenthesis ends the stack comment.

A word SQRT, which provides the square root of an integer, would in FORTH so named and described: (((sqrt (number - sqrt))))

If this new word is called, all contained words run, edited and may be provided parameter passed consequent results on the stack.

The call of procedures done in FORTH is done implicitly by the mention of the name, as well as the data transfer between words usually implicit.

- Drop
- 2drop
- Dup

- ? Dup
- 2dup
- Swap
- 2swap
- Nip
- Over
- 2over
- Under
- Red
- -Red
- Roll
- -Roll roll
- Pick
- . S
- Clear stack
- Depth
- S0
- Sp!
- @ Sp-fetch