



4. Documentation requirements

When it is impossible or infeasible for a system or program to define a particular behavior itself, it is permissible to state that the behavior is unspecified and to explain the circumstances and reasons why this is so.

4.1 System documentation

4.1.1 Implementation-defined options

The implementation-defined items in the following list represent characteristics and choices left to the discretion of the implementor, provided that the requirements of this Standard are met. A system shall document the values for, or behaviors of, each item.

- aligned address requirements ([3.1.3.3](#) Addresses);
- behavior of [6.1.1320](#) EMIT for non-graphic characters;
- character editing of [6.1.0695](#) ACCEPT and [6.2.1390](#) EXPECT;
- character set ([3.1.2](#) Character types, [6.1.1320](#) EMIT, [6.1.1750](#) KEY);
- character-aligned address requirements ([3.1.3.3](#) Addresses);
- character-set-extensions matching characteristics ([3.4.2](#) Finding definition names);
- conditions under which control characters match a space delimiter ([3.4.1.1](#) Delimiters);
- format of the control-flow stack ([3.2.3.2](#) Control-flow stack);
- conversion of digits larger than thirty-five ([3.2.1.2](#) Digit conversion);
- display after input terminates in [6.1.0695](#) ACCEPT and [6.2.1390](#) EXPECT;
- exception abort sequence (as in [6.1.0680](#) ABORT");
- input line terminator ([3.2.4.1](#) User input device);
- maximum size of a counted string, in characters ([3.1.3.4](#) Counted strings, [6.1.2450](#) WORD);
- maximum size of a parsed string ([3.4.1](#) Parsing);
- maximum size of a definition name, in characters ([3.3.1.2](#) Definition names);
- maximum string length for [6.1.1345](#) ENVIRONMENT?, in characters;
- method of selecting [3.2.4.1](#) User input device;
- method of selecting [3.2.4.2](#) User output device;
- methods of dictionary compilation ([3.3](#) The Forth dictionary);
- number of bits in one address unit ([3.1.3.3](#) Addresses);
- number representation and arithmetic ([3.2.1.1](#) Internal number representation);
- ranges for n, +n, u, d, +d, and ud ([3.1.3](#) Single-cell types, [3.1.4](#) Cell-pair types);
- read-only data-space regions ([3.3.3](#) Data space);
- size of buffer at [6.1.2450](#) WORD ([3.3.3.6](#) Other transient regions);
- size of one cell in address units ([3.1.3](#) Single-cell types);
- size of one character in address units ([3.1.2](#) Character types);
- size of the keyboard terminal input buffer ([3.3.3.5](#) Input buffers);
- size of the pictured numeric output string buffer ([3.3.3.6](#) Other transient regions);
- size of the scratch area whose address is returned by [6.2.2000](#) PAD ([3.3.3.6](#) Other transient regions);
- system case-sensitivity characteristics ([3.4.2](#) Finding definition names);
- system prompt ([3.4](#) The Forth text interpreter, [6.1.2050](#) QUIT);
- type of division rounding ([3.2.2.1](#) Integer division, [6.1.0100](#) */, [6.1.0110](#) */MOD, [6.1.0230](#) /, [6.1.0240](#) /MOD, [6.1.1890](#) MOD);
- values of [6.1.2250](#) STATE when true;
- values returned after arithmetic overflow ([3.2.2.2](#) Other integer operations);
- whether the current definition can be found after [6.1.1250](#) DOES> ([6.1.0450](#) :).

4.1.2 Ambiguous conditions

A system shall document the system action taken upon each of the general or specific ambiguous conditions identified in this Standard. See [3.4.4](#) Possible actions on an ambiguous condition.

The following general ambiguous conditions could occur because of a combination of factors:

- a name is neither a valid definition name nor a valid number during text interpretation ([3.4](#) The Forth text interpreter);
- a definition name exceeded the maximum length allowed ([3.3.1.2](#) Definition names);
- addressing a region not listed in [3.3.3](#) Data Space;
- argument type incompatible with specified input parameter, e.g., passing a flag to a word expecting an n ([3.1](#) Data types);
- attempting to obtain the execution token, (e.g., with [6.1.0070](#) ', [6.1.1550](#) FIND, etc.) of a definition with undefined interpretation semantics;
- dividing by zero ([6.1.0100](#) */ , [6.1.0110](#) */MOD, [6.1.0230](#) / , [6.1.0240](#) /MOD, [6.1.1561](#) FM/MOD, [6.1.1890](#) MOD, [6.1.2214](#) SM/REM, [6.1.2370](#) UM/MOD, [8.6.1.1820](#) M*/);
- insufficient data-stack space or return-stack space (stack overflow);
- insufficient space for loop-control parameters;
- insufficient space in the dictionary;
- interpreting a word with undefined interpretation semantics;
- modifying the contents of the input buffer or a string literal ([3.3.3.4](#) Text-literal regions, [3.3.3.5](#) Input buffers);
- overflow of a pictured numeric output string;
- parsed string overflow;
- producing a result out of range, e.g., multiplication (using *) results in a value too big to be represented by a single-cell integer ([6.1.0090](#) * , [6.1.0100](#) */ , [6.1.0110](#) */MOD, [6.1.0570](#) >NUMBER, [6.1.1561](#) FM/MOD, [6.1.2214](#) SM/REM, [6.1.2370](#) UM/MOD, [6.2.0970](#) CONVERT, [8.6.1.1820](#) M*/);
- reading from an empty data stack or return stack (stack underflow);
- unexpected end of input buffer, resulting in an attempt to use a zero-length string as a name;

The following specific ambiguous conditions are noted in the glossary entries of the relevant words:

- [>IN](#) greater than size of input buffer ([3.4.1](#) Parsing);
- [6.1.2120](#) RECURSE appears after [6.1.1250](#) DOES>;
- argument input source different than current input source for [6.2.2148](#) RESTORE-INPUT;
- data space containing definitions is de-allocated ([3.3.3.2](#) Contiguous regions);
- data space read/write with incorrect alignment ([3.3.3.1](#) Address alignment);
- data-space pointer not properly aligned ([6.1.0150](#) ,, [6.1.0860](#) C.);
- less than u+2 stack items ([6.2.2030](#) PICK, [6.2.2150](#) ROLL);
- loop-control parameters not available ([6.1.0140](#) +LOOP, [6.1.1680](#) I, [6.1.1730](#) J, [6.1.1760](#) LEAVE, [6.1.1800](#) LOOP, [6.1.2380](#) UNLOOP);
- most recent definition does not have a name ([6.1.1710](#) IMMEDIATE);
- name not defined by [6.2.2405](#) VALUE used by [6.2.2295](#) TO;
- name not found ([6.1.0070](#) ' , [6.1.2033](#) POSTPONE, [6.1.2510](#) ['], [6.2.2530](#) [COMPILE]);
- parameters are not of the same type ([6.1.1240](#) DO, [6.2.0620](#) ?DO, [6.2.2440](#) WITHIN);
- [6.1.2033](#) POSTPONE or [6.2.2530](#) [COMPILE] applied to [6.2.2295](#) TO;
- string longer than a counted string returned by [6.1.2450](#) WORD;
- u greater than or equal to the number of bits in a cell ([6.1.1805](#) LSHIFT, [6.1.2162](#) RSHIFT);
- word not defined via [6.1.1000](#) CREATE ([6.1.0550](#) >BODY, [6.1.1250](#) DOES>);
- words improperly used outside [6.1.0490](#) <# and [6.1.0040](#) #> ([6.1.0030](#) #, [6.1.0050](#) #S, [6.1.1670](#) HOLD, [6.1.2210](#) SIGN).

4.1.3 Other system documentation

A system shall provide the following information:

- list of non-standard words using [6.2.2000](#) PAD ([3.3.3.6](#) Other transient regions);
- operator's terminal facilities available;
- program data space available, in address units;
- return stack space available, in cells;
- stack space available, in cells;
- system dictionary space required, in address units.

4.2 Program documentation

4.2.1 Environmental dependencies

A program shall document the following environmental dependencies, where they apply, and should document other known environmental dependencies:

- considering the pictured numeric output string buffer a fixed area with unchanging access parameters ([3.3.3.6](#) Other transient regions);
 - depending on the presence or absence of non-graphic characters in a received string ([6.1.0695](#) ACCEPT, [6.2.1390](#) EXPECT);
 - relying on a particular rounding direction ([3.2.2.1](#) Integer division);
 - requiring a particular number representation and arithmetic ([3.2.1.1](#) Internal number representation);
 - requiring non-standard words or techniques ([3.](#) Usage requirements);
 - requiring the ability to send or receive control characters ([3.1.2.2](#) Control characters, [6.1.1750](#) KEY);
 - using control characters to perform specific functions ([6.1.1320](#) EMIT, [6.1.2310](#) TYPE);
 - using flags as arithmetic operands ([3.1.3.1](#) Flags);
 - using lower case for standard definition names or depending on the case sensitivity of a system ([3.3.1.2](#) Definition names);
 - using the graphic character with a value of hex 24 ([3.1.2.1](#) Graphic characters).
-

4.2.2 Other program documentation

A program shall also document:

- minimum operator's terminal facilities required;
 - whether a Standard System exists after the program is loaded.
-



[Table of Contents](#)



[Next Section](#)