

FORTH

The AST.01 and AST.01X Standard Definitions

Kitt Peak National Observatory
Terrel Miedaner
June 1977

()

(

()

FORTH VOCABULARY

The following pages describe the complete AST.01 and the AST.01X Extended Forth Vocabulary, together with a few words recommended for future inclusion. These words comprise a basic Forth system supporting arithmetic and logical operations on integers, block I/O to either tape or disk, and a single terminal.

The words appear in essentially the same sequence as their numerically sorted identifier codes. The action of each word is described in abbreviated form: A string of symbols indicating which parameters are to be placed on the stack before executing the word; the word itself; then, any parameters left on the stack by the word. In this notation, the top of the stack is to the right.

Symbols are used as follows:

b	Block number.
c	7-bit ASCII character code.
f	Flag: 0=False, non-zero=True. All words which return a flag return 0=False or 1=True.
m n p q r	16-bit integers.
u v w	Double-precision (2 cell) numbers.
x	Pseudo-vector index.
nnnn pppp	The name of a word.
ssss	A string of characters.
vvvv	A vocabulary name.

Compiler directives, that is, words which are executed when encountered in a colon-definition, are underlined.

Preceding a verbal description of each word, certain characters may appear within parentheses. These denote some special action or characteristics, as follows:

C	The word may be used <u>only</u> within a colon-definition. A following digit (C0 or C2) indicates the number of memory cells used when the word is compiled, if other than one. A following + or - sign indicates that the word either pushes a value onto the stack or removes one from the stack during compilation. (This action is not related to its action during execution.)
E	The word may not normally be compiled within a colon-definition.

FORTH Vocabulary

- K The word is defined only in KPNO systems, and is not currently part of the standard. Its definition is retained in this document for reference only.
- L The word causes loading and possible execution of one or more blocks.
- N Non-reentrant; may not be used within an interrupt-handler word.
- T Tape systems only.
- X The word is part of the AST.01X extension.

FORTH

The AST.01 and AST.01X Standard Definitions

- ! m p ! Stores m at address p.
- !BLOCK b !BLOCK p Obtains a core buffer for Block b, leaving the first buffer cell address. The block is not read from disk, and is automatically marked as updated.
- " " ssss" Transmits a message of up to 63 characters delimited by " " to the selected output devices.
- #TER #TER m (X)Returns the physical unit number of the terminal device.
- ' ' nnnn p Leaves the address of the parameter field of nnnn. A compiler directive, ' is executed when encountered in a colon-definition: The address of the following word's parameter field is found immediately (at compilation), and stored in the dictionary (after the address of LIT) as a literal to be placed on the stack at execution time.
- e.g. The sequence: ' nnnn is identical to:
LIT [' nnnn,] within a colon-definition.
- ((ssss) Ignores a comment of up to 63 characters delimited by a right parenthesis. A single blank between parentheses is not allowed.
- ()DIM m ()DIM nnnn (K) Defines an array m+1 cells in length, named nnnn. The sequence: i nnnn leaves the address of the ith cell on the stack. The index i should be in the range $0 \leq i \leq m$, but no check is made for values exceeding this range.
- * m n * q 16-bit integer multiply.
- */ m n p */ q Leaves $q = m*n/p$. Retention of an intermediate 31-bit product permits greater accuracy than the otherwise equivalent sequence: m n * p /.
- + m n + q 16-bit integer addition.
- +! m p +! Adds integer m to value at address p.

FORTH - The AST.01 and AST.01X Standard Definitions

+BLOCK	m +BLOCK b	Leaves the sum of m plus the number of the block currently being interpreted.
+DP	n +DP	(K) Change the dictionary pointer by the signed value on the stack.
+LOOP	m +LOOP	(C) Adds m to the loop index. Exit from the loop is made when the resultant index reaches or passes the limit, if m is greater than zero; or when the index is less than (passes) the limit, if m is less than zero. The value m may be a variable.
,	m ,	Stores m into the next available dictionary cell, advancing the dictionary pointer.
-	m n - q	16-bit integer subtraction (m-n).
.	m .	Prints the value on the stack as an integer, converted according to the current number base.
/	m n / q	16-bit integer divide, m/n. The quotient is truncated; any remainder is lost.
/MOD	m n /MOD r q	16-bit integer divide, m/n. The quotient is left on top of the stack, the remainder beneath. The remainder has the sign of the dividend, m.
Ø	m Ø) q	(X) Toggles the most significant bit of m.
0<	m 0< f	Leaves a true flag if m is negative.
0<=	m 0<= f	(X) True if m is zero or negative.
0=	m 0= f	True if m is zero.
0<>	m 0<> f	(X) True if m is non-zero.
0>	m 0> f	True if m is positive and non-zero.
0>=	m 0>= f	(X) True if m is greater than zero.
1+	m 1+ q	(X) q = m+1.
1+!	p 1+!	(X) Add 1 to the contents of address p.
1-	m 1- q	(X) q = m-1.

FORTH - The AST.01 and AST.01X Standard Definitions

2* m 2* q (X) q = 2*m.

2/ m 2/ q (X) q = 2/m.

2DROP m n 2DROP (K) Drop the two topmost values from the stack--that is, drop a double-precision number.

2DUP m n 2DUP m n m n (K) Duplicate the two topmost values from the stack--that is, duplicate a double-precision number.

:

 : nnnn Create a dictionary entry for a colon-definition, set compilation mode, and set the context vocabulary equivalent to the current vocabulary.

:ORx :ORx (K) Begin an anonymous colon-definition, placing its address in pseudo-vector x for subsequent compilation by ADOx. See P-Vx.

;

 ; (C) Terminates a colon-definition and stops compilation.

;;

 ;; (C) Terminates a defining word nnnn, which can be subsequently executed to define a new word pppp. Subsequent use of pppp will cause the words between ;: and ; to be executed with the parameter-field address of pppp on the stack.

;CODE

 ;CODE (C) Stops compilation and terminates a defining word nnnn. Switch the context vocabulary to ASSEMBLER in anticipation of a machine-code sequence. When nnnn is subsequently executed to define a new word pppp, the execution-address of pppp will point to the machine-code sequence following the ;CODE of nnnn. Then, subsequent use of pppp (or any other word defined by nnnn) will cause this machine-code sequence to be executed.

;EXIT

 ;EXIT (X) Terminate a colon-definition when encountered at execution-time; compilation is not stopped.

;S

 ;S (E) Stops interpretation of a symbolic block.

<

 m n < f True if m < n.

<=

 m n <= f True if m ≤ n.

=

 m n = f True if m = n.

<>

 m n <> f True if m ≠ n.

FORTH - The AST.01 and AST.01X Standard Definitions

>	m n > f	True if m > n.
>=	m n >= f	True if m <u>></u> n.
>R	m >R	(C) Pushes m onto the top of the return stack. See I and R>.
>IM	>IM nnnn	Set the precedence bit of the following word, making it a compiler directive.
?	p ?	(N) Prints the value contained at address p in free format, according to the current base.
?DEF	?DEF nnnn m	Returns the first memory cell address of nnnn if nnnn can be found in the context vocabulary; zero otherwise.
?TER	?TER c	(X) Returns the character code of the last character entered at the terminal, or zero if no character has been typed.
@	p @ q	Leaves the contents q of memory address p.
[[Stop compilation. The words following the left bracket in a colon-definition are executed, not compiled. Typically, left and right brackets are used in conjunction with the interpreter-level conditionals IFTRUE-IFEND to control compilation.
]]	Resume compilation. Words following the right bracket are compiled.
^	^ nnnn	Return the compilation address of the following word; that is, the address which would be compiled in a colon-definition. Abort if nnnn is not found.
ABORT	ABORT	Enter the abort sequence, clearing all stacks, printing an informative message, and returning control to the terminal.
ABS	m ABS q	Leaves the absolute value of a number.
ADOPT	m <u>ADOPT</u>	(C) Stores m into the next available dictionary cell, advancing the dictionary pointer.
ADOx	<u>ADOx</u>	(K) Adopts a code or colon orphan by placing the address contained at pseudo-vector x into the dictionary. See P-Vx.
AND	m n AND q	Bitwise logical AND of m and n.

FORTH - AST.01 and AST.01X Standard Definitions

ASSEMBLER	<u>ASSEMBLER</u> (X) Switch the context vocabulary pointer so that dictionary searches will begin at the Assembler Vocabulary. The Assembler Vocabulary is always chained to the current vocabulary.
BASE	BASE p (N) An integer pointing to the current conversion base value.
BEGIN	<u>BEGIN</u> (C0+) Mark the start of a BEGIN-END loop. The words between <u>BEGIN</u> and its corresponding END will be repetitively executed until the END-condition is satisfied. Loops may be nested.
BELL	<u>BELL</u> (X) Activate terminal bell or noisemaker as appropriate to device.
BLK	BLK p (N) An integer, pointing to the number of the block being listed or edited.
BLOCK	b BLOCK p Leaves the first address of Block b. If the block is not already in memory, it is transferred from disk or tape into whichever core buffer has been least recently accessed. If the block occupying that buffer has been updated, it is rewritten on disk or tape before Block b is read into the buffer.
CASE	m n <u>CASE</u> ... <u>ELSE</u> m ... <u>THEN</u> (C2+,X) If m equals n, m is dropped from the stack, and the words immediately following CASE are executed until the next ELSE or THEN. If m does not equal n, m remains on the stack and the words after ELSE (or THEN if no ELSE is used) are executed. The value n is always dropped. m n <u>CASE</u> ... <u>THEN</u> m
CHAIN	<u>CHAIN</u> vvvv (X) Connects the current vocabulary to all definitions that might be entered into Vocabulary vvvv in the future. The current vocabulary may not be FORTH or ASSEMBLER. Any given vocabulary may only be chained once, but may be the object of any number of chainings. For example, every user-defined vocabulary may include the sequence, CHAIN FORTH.
CODE	CODE nnnn Creates a dictionary entry for a code definition named nnnn, and sets the context vocabulary to Assembler.
COM	m COM q Leaves the one's complement of m.
CON	m CON nnnn (X) Abbreviation of CONSTANT.

FORTH - The AST.01 and AST.01X Standard Definitions

- CONSTANT m CONSTANT nnnn Creates a word which when executed pushes m onto the stack. Since the "constant" m may be modified by the sequence: q ' nnnn ! it is oftentimes advantageous to define a variable as a constant, particularly if the variable is accessed more often than it is modified.
- CONTEXT CONTEXT p (X) An integer that indicates at which vocabulary dictionary searches are to begin.
- CONTINUED b CONTINUED Continue interpretation at Block b. The block currently being interpreted is marked as least-recently-accessed, so that its buffer will be used for storage of Block b, and the contents of the alternate block will remain in memory.
- COUNT p COUNT (m) n (C) The count-byte n is extracted from the first memory cell of a message string beginning at address p, and left on the stack. The character-address m of the first byte of the message is typically left on the stack or in a register. Whatever, COUNT is to be used in conjunction with a following PRINT or TYPE.
- CR CR Transmit carriage return/line feed codes to the selected output devices.
- CUR CUR p (T,K) An integer pointing to the physical record number before which the tape is currently positioned. REWIND sets CUR = 1.
- CURRENT CURRENT p (X) An integer that indicates the vocabulary into which new words are to be entered.
- DECIMAL DECIMAL Sets the numeric conversion base to decimal mode.
- DEFINITIONS (vvvv) DEFINITIONS (X) Sets the current vocabulary (into which new definitions are placed) to Vocabulary vvvv (the context vocabulary). vvvv need not be specified explicitly.
- DISCARD DISCARD (N) A null-definition intended for use as a standard REMEMBER word, as some version of DISCARD can always be found in the dictionary.
- DLIST m DLIST (K,E) List the dictionary, starting at the entry beginning at location m. The listing may be terminated by any terminal key.

FORTH - AST.01 and AST.01X Standard Definitions

- DO** *n m DO* (C) Begin a loop, to be terminated by LOOP or +LOOP. The loop index begins at *m*, and may be modified at the end of the loop by any positive or negative value. The loop is terminated when an incremented index reaches or exceeds *n*, or when a decremented index becomes less than *n*. Within a loop, the word I will place the current index value on the stack.
Loop indices are available to three levels of nesting. Within nested loops, the word I always returns the index of the innermost loop that is being executed, while J returns the index of the next outer loop, and K returns the index of the second outer loop.
Execution of DO places three parameters on the return stack: The starting location of the loop, the index limit, and the index.
- DPL** *DPL p* An integer, pointing to a number-conversion parameter: The number of digits following the fractional point on input or output. A negative value at D indicates that no "," was entered on input, or that none is to be printed on output.
- DROP** *m DROP* Drop the topmost value from the stack.
- DUMP** *m DUMP* (E,K) Dump the contents of memory cells starting at address *m*. Addresses are output in the current conversion base; memory contents are output in octal. The dump is terminated by any terminal key.
- DUP** *m DUP m m* Returns a duplicate of the topmost stack value.
- EDIT** *b EDIT* (L,X) The Editor Vocabulary is loaded, if not already in the dictionary, becoming the context vocabulary. Block *b* is listed.
- EDITOR** EDITOR (X) The name of the Editor Vocabulary. If that vocabulary is loaded, EDITOR establishes it as the context vocabulary, thereby making its definitions accessible.
- ELSE** ELSE (C2) Precedes the false part of an IF-THEN-ELSE conditional or the continuation of a CASE-type conditional.
- END** *f END* (C2-) Mark the end of a BEGIN-END loop. If *f* is true the loop is terminated. If *f* is false, control returns to the first word after the corresponding BEGIN.
- ERASE-CORE** *ERASE-CORE* Marks all block-buffers as empty, without affecting their actual contents. Updated blocks are not flushed.

FORTH - AST.01 and AST.01X Standard Definitions

EXIT	EXIT (C) Force termination of a DO-loop at the next opportunity by setting the loop limit equal to the current value of the index. The index itself remains unchanged, and execution proceeds normally until LOOP or +LOOP is encountered.
F	F p (K) An integer pointing to the field length reserved for a number during output conversion.
FLUSH	FLUSH Write all blocks that have been flagged as "updated" to disk or tape. Return when output is completed.
FORGET	FORGET nnnn Delete nnnn and all dictionary entries following it. Although nnnn must be in the context vocabulary to be found, the words that follow it are deleted no matter which vocabulary they belong to. Normally, FORGET should not be used within a colon-definition, as it is not a compiler directive. For such applications, use a word defined by REMEMBER.
FORTH	<u>FORTH</u> (X) The name of the primary vocabulary. Execution makes <u>FORTH</u> the context vocabulary. Since FORTH cannot be chained to anything, it becomes the only vocabulary that is searched for dictionary entries. Unless additional user vocabularies are defined, new user definitions normally become part of the Forth Vocabulary.
GCH	GCH c Get a character from the terminal, i.e., return the ASCII code of the next character typed.
GO-TO	m GO-TO (E,X) Interrupt interpretation of a block, resuming at line m of the current block. GO-TO may only be used during loading of a block.
HEAD	HEAD p Returns a pointer to the first location of the last word defined in the current vocabulary (equivalent to CURRENT @).
HERE	HERE p Return the address of the next available dictionary location.
HEX	HEX (X) Switch the base to hexadecimal.

FORTH - AST.01 and AST.01X Standard Definitions

- I m (C) Push the topmost return stack value onto the user stack without disturbing the return stack. Typically, I is used to return the index of an innermost DO-loop, but it can also be used to access values pushed onto the return stack by >R.
- ID. p ID. (K) Prints the identifier of the word at address p.
- IF f IF ... ELSE ... THEN (C2+) IF is the first word of a conditional. If f is true, the words following IF are executed and the words following ELSE are not executed. The ELSE part of the conditional is optional. If f is false, words between IF and ELSE, or between IF and THEN when no ELSE is used, are skipped. IF-ELSE-THEN conditionals may be nested.
- IFEND IFEND (E,X) Terminates a conditional interpretation sequence begun by IFTRUE.
- IFTRUE f IFTRUE ... OTHERWISE ... IFEND (E,X) Unlike IF-ELSE-THEN, these conditionals may be employed during interpretation. In conjunction with the words [and], they may be used within a colon-definition to control compilation, although they are not to be compiled. These words cannot be nested. See GO-T0.
- IM> IM> nnnn Clears the precedence bit of nnnn. Words with the precedence bit set are compiler directives.
- INTx m INTx (E,K) Define a pseudo-vector variable, initialized to m, indexed by x (see P-Vx). The value m may be accessed by RCLx or modified by STRx, which must appear only within a colon-definition. These words are designed for core and speed efficiency, not to enhance code readability.
- J J m (C) Execute J within a nested DO-loop to return the index of the next outer loop.
- K K m (C) Execute K within a nested DO-loop to return the index of the second outer loop.
- LAST LAST p An integer pointing to the address of the last dictionary entry made, which is not necessarily a complete or valid entry.
- LINE m LINE p Leaves the character address of the beginning of line m for the block whose number is contained at BLK.

FORTH - AST.01 and AST.01X Standard Definitions

LINELOAD m b LINELOAD (K) Begin interpreting at line m of Block b.

LIST b LIST (K) List the ASCII symbolic contents of Block b on the selected output device.

LIT LIT m (C) Automatically compiled before each literal encountered in a colon-definition, execution of LIT causes the contents of the next dictionary cell to be pushed onto the stack.
NOTE: On the CDC 6400, LIT is implemented so as to access the next 15-bit parcel of a 60-bit word. This implementation is recommended for other machines of large word size.)

LOAD b LOAD Begin interpretation of Block b. The block must terminate its own interpretation with ;S or CONTINUED.

LOOP LOOP (C) Increment the DO-loop index by one, terminating the loop if the new index is equal to or greater than the limit.

MAPØ MAPØ p (T) An integer pointing to the first location in the tape map.

MAX m n MAX q Leaves the greater of two numbers.

MIN m n MIN q Leaves the lesser of two numbers.

MINUS m MINUS -m Negates a number by taking its two's complement.

MOD m n MOD r Leaves the remainder of m/n, with the same sign as r.

MOVE p q n MOVE Moves the contents of n memory cells beginning at address p into n cells beginning at address q. The contents of p is moved first; overlapping of data can occur. (0 10 ! 10 11 4 MOVE clears locations 10 through 15.)

MSEC n MSEC (K,N) Delay for n milliseconds, +2%.

NAND m NAND n (X) Logical not-and.

NOR m NOR n (X) Logical not-or.

NOT m NOT f (X) Equivalent to 0=.

NUMBER NUMBER Convert a character string left in the dictionary buffer by WORD as a number, returning the result in registers, internal temporary locations, or on the stack. The appearance of characters that cannot be properly interpreted will cause an error exit appropriate to the installation.

FORTH - AST.01 and AST.01X Standard Definitions

O.	m O. (K) Convert and output in octal mode, unsigned, and preceded by a blank. BASE is unchanged. Format specifications are observed.
OCTAL	OCTAL Set the number-conversion base to octal.
OR	m n OR q Bitwise logical inclusive OR.
ORCx	ORCx (K) Begin an anonymous code-definition, as per :ORx.
OTHERWISE	OTHERWISE An interpreter-level conditional word. See IFTRUE.
OVER	m n OVER m n m Push the second stack value.
P-Vx	P-Vx p (K) Returns the address p of one element in a temporary storage area known as the pseudo-vector table. Sixteen addresses are available. Addresses are indexed, during interpretation or compilation, by the binary value of the least significant four bits of the character code of x. (E.g. either P-V1 or P-VA returns a pointer to the second element of the zero-indexed pseudo-vector table.) These temporary locations are used during compilation, usually to retain and access the addresses of anonymous dictionary entries.
PAGE	PAGE Clears the terminal screen or performs an action suitable to the output device currently active.
PICK	n PICK Returns the nth stack value, not counting n itself. (2 PICK is equivalent to OVER.)
PCH	c PCH Transmit a character to the selected output printer device. See TCH.
PRINT	(m) n PRINT (C) Transmit n characters to the selected output printer starting at character address m, which will have been placed on the stack or in an internal register by COUNT.
PRINTER	PRINTER (X) Select a hard-copy PRINTER as the output device for all output directed through PCH or PRINT. See TERMINAL.
QBLOCK	b QBLOCK p (X) Like BLOCK, but may return while previous contents of block are still being written to output device.

FORTH - AST.01 and AST.01X Standard Definitions

R>	R> (C,X) Pop the topmost value from the return stack and push it onto the user stack. See I and >R.
RCLx	RCLx m (C,K) Recall the variable stored at a location established by INTx. See P-Vx.
READ-MAP	READ-MAP (T) Read to the next file mark on tape, constructing a correspondence table in memory (the map) relating physical block position to logical block number. The tape should normally be rewound to its load point before executing READ-MAP.
REMEMBER	REMEMBER nnnn Define a word nnnn which, when executed, will cause nnnn and all subsequently defined words to be deleted from the dictionary. The word nnnn may be compiled into and executed from a colon-definition. The sequence DISCARD REMEMBER DISCARD provides a standardized preface to any group of transient blocks.
REWIND	REWIND (T) Rewind the tape to its load point, setting CUR = 1.
ROLL	$u_n u_{n-1} \dots u_1$ n ROLL $u_{n-1} \dots u_1 u_n$ Extract the nth value from the stack, leaving it on top and moving the remaining values into the vacated position. (3 ROLL is equivalent to ROT; 1 ROLL is a null operation; 0 ROLL is undefined.)
ROT	m n p ROT n p m Rotate the topmost three stack values.
SET	m p SET nnnn Defines a word nnnn which, when executed, will cause the value m to be stored at address p.
STRx	m STRx (C,K) Store m at the location established by a preceding INTx definition. See P-Vx.
SWAP	m n SWAP n m Exchange the topmost two stack values.
TCH	c TCH Transmit a character code to the terminal, irrespective of output-device selection. See PCH.
TERMINAL	TERMINAL Select the terminal as the only output device, cancelling previous selection of PRINTER.
THEN	<u>THEN</u> (CO-) Terminates an IF-ELSE-THEN conditional sequence.
TYPE	(m) n TYPE (C) Transmits n characters to the terminal, irrespective of output device selection, starting at the character address m. See COUNT, PRINT.

FORTH - AST.01 and AST.01X Standard Definitions

Interim recommendations and afterthoughts, coming from various meeting participants, for consideration:

1. The "B" prefix should be used for byte-addressing operators - B@, B', BMOVE, etc.
2. The "C" prefix should be used for character-processing operators.
3. DECIMAL should be the normal radix.
4. QUERY might accept an end-of-line character, buffer address, and buffer length on the stack, for greater generality.

FORTH ASTRONOMICAL USERS GROUP

The first meeting of the FORTH Astronomical Users Group was held at Kitt Peak National Observatory, Tucson, Arizona, on May 26-27, 1977. The intent of this meeting was to define a standard for the FORTH language that would satisfy current users in the astronomical community, and which would also be of sufficient generality and quality to interest users outside the community.

The following individuals attended. Those whose names are marked by an asterisk contributed but did not hold forth during the entire meeting.

- *James Brault - KPNO
- Peter Collins - Steward-MMT
- Wayne Hammond - Cal Tech
- *Mike Hollis - NRAO
- Dave Jenner - UCLA
- Matt Johns - McGraw Hill
- Steven Maker - Dartmouth-McGraw Hill
- Terrel Miedaner - KPNO (Chairman)
- Jeff Moller - KPNO
- David Rogstad - JPL
- Tom Sargent - Steward
- Paul Scott - KPNO
- *Richard Stevens - KPNO
- *Don Wells - KPNO

We have defined a basic standard and an extension. The standard, designated AST.01, is not yet complete. Although many difficult points were resolved, no universally acceptable output-formatting scheme was chosen, leaving that important matter awaiting our next meeting.

The extension to the standard, AST.01X, defines words which are not essential to the language but are used widely enough so that their names ought to be reserved and their definitions fixed. Larger or visitor-oriented installations such as KPNO will implement the extensions in optional source blocks if they are not already included in their system.

Additionally, 4-character plus count identifier coding schemes will be regarded as an option. Obviously, programs written under such systems will not be compatible with the standard unless suitably restricted. Standard FORTH programs will be transferable to AST.01 systems with the 4-character option subject to avoidance of the : WORM WORD ; type of redefinition.

The AST.01 standard meets the following requirements:

- 1) It will run on any 8K 16-bit machine and on longer word-length machines. A minimum of 511 blocks of random-access mass storage is assumed. An AST.01 subset can be adapted to smaller word-length machines and less powerful configurations.
- 2) Any program written in AST.01 (excepting CODE definitions) will run without modification on any installation supporting it, independent of machine, assuming identical word-length. Such programs will be upward compatible with longer word-length machines.

- 3) Availability of memory greater than 8K and mass storage space of more than 511 blocks is not specified by the standard, nor is any particular style of implementation.

Although stand-alone FORTH is recommended because of its superior I/O efficiency, FORTH has been run quite effectively under both general and special-purpose operating systems. The standard will run under either mode.

The AST.01X extension is designed to:

- 1) Define standards for tape-only systems.
- 2) Fix the definitions of certain specialized or potentially ambiguous words for readability.
- 3) Accommodate the availability of specialized hardware and installation-dependent features.
- 4) Standardize useful but unessential words.

It is intended that installations will provide these extensions as options whenever feasible and refrain from alternate definitions of the extension words.

AST.01 FORTH

Structure

2-level precedence (a single precedence bit). It is recommended that the precedence bit be located in the most-significant position of the count byte, in accordance with current practice, although this is not specified as part of the standard.

Identifier - first 3 characters + character count. (The extended standard permits 4 characters.)

Characters - 7-bit ASCII.

Block length - 1024 characters (not necessarily 512 words) logically divided into 16 64-character lines of symbolic information. Binary data formats are currently unspecified. (The standard is defined such that any program can determine block length in terms of memory cells.)

Terminal Control

RUBOUT - Backspace (delete) one character.

CTRL/U - Delete the entire input line. (This character is specified in place of the more common BREAK key because

- a) The BREAK key was intended to be a transmission-stop code, not a character, and does not always work well on some terminals.
- b) The position of the BREAK key varies from terminal to terminal.

CTRL/X - Recommended use as Escape Key (return control to terminal) unless terminal or system constraints dictate otherwise.

CTRL/Z - Recommended use: reset dictionary to basic FORTH system. (This key would be recognized from the terminal query loop only.)

CTRL/S/Q - Recommended: Stop/Resume output.

Physical Block Standards

A standard FORTH system will be presumed to have a random mass storage device (disk or floppy) containing at least 511 blocks. The physical structure of blocks with respect to the medium is not defined.

A tape system, if implemented, will employ the original FORTH standard, which is capable of making any sequential record device look logically like a (slow) random-access device. At least 511 logical blocks are allowed; the physical order of blocks on tape is not specified. Each block contains 513 cells with the logical block number in the last word. When BLOCK accesses an undefined block, the buffer is zero-filled.

Standard 9-track tape format:

```
a b c d e f g h   (a is the most significant bit)
i j k l m n o p
```

Hardware differences make it difficult to define a suitable 7-track standard. KPNO formats each 16-bit word thusly:

```
0 0 a b c d
e f g h i j
k l m n o p
```

A brief description of the AST.01 standard words follows, by category, with notes describing possible departures from current practice, or other interesting features. Detailed specifications for each word in the standard and extended vocabularies are included in an alphabetical appendix. These are to be studied carefully when implementing words or comparing their action against an existing version of the word.

Do not expect any current version of FORTH to conform to this standard without modification. No installations represented at the meeting had such a FORTH.

Defining Words

: ;

:: The words following are executed with the address of the parameter field of the calling word on the stack.

;CODE CODE Assembler vocabularies are not specified.

CONSTANT CON added in extension.

VARIABLE Replacing INTEGER, for clarity.
VAR added in extension.

SET

WORD In a departure from early FORTH practice, the delimiter is on the stack upon entry to WORD. The word DELIM or DELIMITER is omitted. WORD swallows up to 63 characters, stopping as if a delimiter had been found.

Variable Access

@ ! +!

MOVE

' (tick) is a compiler directive. In a colon-definition it compiles the address of the literal handler and the parameter-field address of the word following ' at compilation time. Abort if the dictionary search is unsatisfied.

Compilation Control

^ (carat) is not a compiler directive. It returns the compilation address of the following word (at execution time). Abort if dictionary search is unsatisfied.

>IM Set the precedence bit of the following word making it a compiler directive.

IM> Clear the precedence bit of the following word.

[] ,

LIT Renaming LIT,

Dictionary Manipulation

HERE Returns address of next available cell. Address must be usable with @ and !.

FORGET

REMEMBER

DISCARD

HEAD LAST

Stack Manipulation

DUP OVER ROT SWAP DROP

PICK Renaming S@. 2 PICK=OVER.

ROLL
0 ROLL=nop
1 ROLL=nop
2 ROLL=SWAP
3 ROLL=ROT

Arithmetic Operators

+ - * / Two's or one's complement.

/MOD MOD Remainder has sign of dividend.

*/

ABS MIN MAX MINUS

Logical Operators

+ < >

0= 0< 0>

<= Less than or equal
>= Greater than or equal
<> Not equal

Masking Operators

AND OR XOR

COM One's complement. Replacing BNOT or NOT in some systems to
remove ambiguity.

Blocks

BLOCK !BLOCK Implies wait for completion. Result of 0 BLOCK unspecified.

UPDATE

FLUSH All buffers.

ERASE-CORE Does whatever is necessary to logically erase all block
buffers. Contents of buffers undefined.

BLK

LOAD ;S

+BLOCK Rename *BLOCK

CONTINUED

Compiled Conditionals (Use of compile-time stack is not specified)

IF THEN ELSE

BEGIN END

DO LOOP +LOOP

EXIT Implementations: change limit to index so as to leave the index intact for the duration of the loop.

I J K

Miscellaneous

() 63 characters maximum. A single space between pairs of parentheses is not allowed.

BASE OCTAL DECIMAL

?DEF Returns address of first cell of dictionary entry if found, zero otherwise.

LINE Returns the character address of the specified line.

Output

. Prints something in the current number base.

?

CR

PAGE Performance related to device.

SPACE Rename BLANK in some systems.

n SPACES

" " 63 character limit.

COUNT COUNT is only specified so as to leave the stack and/or internal register properly set up for a following PRINT or TYPE.

TYPE Input parameters to be established by COUNT.

PRINT Rename WRITE. A switchable word normally initialized so as to direct output to the terminal.

TERMINAL Switches print output to the terminal.

TCH

PCH Rename WCH.

Input

GCH Read an ASCII character from the terminal.

DPL Rename D.

NUMBER Decode the character string left by WORD as a number, if possible, according to the radix contained in the integer BASE.

Many versions of NUMBER return the resultant binary value in internal registers or temporary storage locations. A more general version might return its values on the stack, but at present this is not specified in the standard.

Integer, fixed-point, and floating-point values will be converted, unambiguously, according to the following rules:

- 1) Numbers may be signed negatively only.
- 2) An imbedded colon denotes sexagesimal conversion.
12:05
- 3) A comma denotes fixed-point conversion with the number of digits following the comma left in DPL after conversion. In 16-bit machines, fixed-point numbers occupy two memory cells.
9,
-,123
12:05:33,
- 3a) Systems which do not support floating-point may elect to use a period (in addition to comma) to denote the radix point of fixed-point numbers. However, given the impending proliferation of machines capable of supporting floating-point, and its attractiveness for many applications, it is recommended that all standard installations switch to exclusive use of the comma for fixed-point as soon as possible.
- 4) Floating-point numbers are denoted by a period; or by the character "E", which may be followed by a signed integer exponent. The number of memory cells occupied by floating-point values is not specified.
9.
-98E-4
.123E50
12:05:33.
- 4a) Installations supporting extended-precision floating-point will denote such numbers by the character "D" in place of "E".
3.14159D
123D-2
- 5) All numbers which do not contain , . E or D are single-precision integers.

6) A character preceded by an ampersand returns the 7-bit ASCII code of the character.

&A (101₈)

-&3 (-63₈)

&3, Fixed-point

&3. Floating-point

It has been recommended that some method for specifying the radix independently of BASE be adopted, as in 6400 FORTH which utilizes a trailing "B" to denote octal. Prefix characters have also been proposed; however, given the internal structure of FORTH, there is no particular implementation advantage to a prefix code (WORD can save the final character) and the suffix seems a more readable convention.

Similarly, a trailing H or Z could denote hexadecimal conversion. Ambiguity is easily resolved by interpreting "B" only as a hex digit when in hexadecimal mode.

These recommendations should be evaluated for subsequent action.

QUERY

It has been recommended that the terminal input handler be redesigned to accept end-of-line code, buffer address, and buffer length on the stack. This appears to be a sound idea, permitting more general use of QUERY.

The KPNO convention of reserving blocks 0-199 for systems may be applicable to other installations.

In the absence of agreed-upon naming conventions for multiple-precision operations, note that KPNO currently uses 2DUP, 3DUP, etc. for multiple-word stack operations (after initially using DDUP, FDUP); and D*, F*, etc. for double-precision fixed-point and three-word floating point, respectively.

Pseudo-vector words might become more popular if they were better understood; for that reason, they are described in the definitions manual.

Recommendations

Time did not permit us to be as thorough as we would have liked. Not all words were evaluated. In the absence of a decision, the following recommendations are offered for consideration.

- | | |
|-------------------------|---|
| +DP | Increment the dictionary pointer by the signed value on the stack. Lacking this, we have no standard way of modifying the dictionary pointer. |
| ()DIM | KPNO's style of allocating dimensioned variable space. |
| @I !I
@K !K
@J !J | Steward's method of accessing array elements within loops -- only the first address of the array is required on the stack. |
| LINELOAD | To make more efficient use of storage blocks. |

AST.01X (Extended)

FORTH
EDITOR
ASSEMBLER

The vocabulary system provides an excellent method of isolating groups of words from one another. In addition to preventing redefinition problems, effective use of vocabularies can reduce compilation time. Even if a full vocabulary system is not implemented, these three vocabularies should be.

VOCABULARY
DEFINITIONS
CHAIN
CURRENT
CONTEXT

Vocabulary defining words recommended for all installations where transported programs should be isolated from in-house applications.

IFTRUE
OTHERWISE
IFEND
GO-TO
ADOPT

Conditional interpretation words offer a powerful method for optional construction of code according to device type, machine, or other configuration constraints. Implementation is cheap.

CASE

A useful conditional. Note that the version adopted differs from, and is more powerful than, the FORTH, Inc. version.

O<=
O>=
O<>

Additional comparison operators

U<
U>
U<=
U>=

Unsigned comparisons in which the parameters are assumed to be 16-bit positive integers

1+
1-
1+!
2*
2/
O)
NOR
NAND

ABORT

BELL

Action appropriate to device.

QBLOCK

Reserve for quick return (write in progress) version of BLOCK.

READ-MAP
REWIND

HEX

CON VAR Abbreviations of CONSTANT and VARIABLE.

NOT Same as 0= but not to replace 0= if used. NOT may have additional meaning in Assembler vocabulary.

PRINTER Switches output to printing device, as applicable.

?TER

#TER Usage reserved, but selection of logical terminal numbers will be an installation option.

>R
R> These are system-level words only, not to be used for transportable code.

;EXIT Internal exit from colon-definition.