draft proposed American National Standard
for Information Systems —

Programming Languages —
Forth
(X3J14 dpANS-6 – June 30, 1993)

Secretariat

Computer and Business Equipment Manufacturers Association

American National Standards Institute, Inc.

WARNING This is a draft of the proposed American National Standard. It is distributed solely for the purpose of review and comment and should not be used as a design document. It is inappropriate to claim compatibility with this draft standard.

Disclaimer

The X3J14 Technical Committee makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The Technical Committee shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Foreword

(This foreword is not a part of American National Standard X3.215-199x)

Forth is a language for direct communication between human beings and machines. Using natural-language diction and machine-oriented syntax, Forth provides an economical, productive environment for interactive compilation and execution of programs. Forth also provides low-level access to computer-controlled hardware, and the ability to extend the language itself. This extensibility allows the language to be quickly expanded and adapted to special needs and different hardware systems.

Forth was invented by Mr. Charles Moore to increase programmer productivity without sacrificing machine efficiency. Forth is a layered environment containing the elements of a computer language as well as those of an operating system and a machine monitor. This extensible, layered environment provides for highly interactive program development and testing.

In the interests of transportability of application software written in Forth, standardization efforts began in the mid-1970s by an international group of users and implementors who adopted the name "Forth Standards Team." This effort resulted in the Forth-77 Standard. As the language continued to evolve, an interim Forth-78 Standard was published by the Forth Standards Team. Following Forth Standards Team meetings in 1979, the Forth-79 Standard was published in 1980. Major changes were made by the Forth Standards Team in the Forth-83 Standard, which was published in 1983.

The first meeting of the Technical Committee on Forth Programming Systems was convened by the Organizing Committee of the X3J14 Forth Technical Committee on August 3, 1987, and has met subsequently on November 11-12, 1987, February 10-12, 1988, May 25-28, 1988, August 10-13, 1988, October 26-29, 1988, January 25-28, 1989, May 3-6, 1989, July 26-29, 1989, October 25-28, 1989, January 24-27, 1990, May 22-26, 1990, August 21-25, 1990, November 6-10,1990, January 29-February 2, 1991, May 3-4, 1991, June 16-19, 1991, July 30-August 3, 1991, March 17-21, 1992, October 13-17, 1992, January 26-30, 1993, and June 28-30, 1993.

Membership

At the time it approved this draft of the proposed American National Standard, the Technical Committee X3J14 on the Forth Programming Language had the following members:

Elizabeth Rather, Chair
Mitch Bradley, acting Vice-Chair
Don Colburn, Secretary
John Rible, Technical Editor
Len Zettel, Vocabulary Representative
Greg Bailey, Technical Subcommittee Chair

Organization Represented	*Name of Representative*
ATHENA Programming, Inc.	Greg Bailey
	Howe Fong (Alt.)
Bradley Forthware, Inc.	Mitch Bradley
Creative Solutions, Inc.	Don Colburn
Ford Motor Company	Leonard F. Zettel, Jr.
FORTH, Inc.	Elizabeth Rather
	Dennis Ruffer (Alt.)
Institute for Applied Forth Research	Lawrence Forsley
	Horace Simmons (Alt.)
Johns Hopkins University, Applied Physics Lab.	John Hayes
Mephistopheles Systems	Dave Harralson
NASA/Goddard Space Flight Center	James Rash
Nomadic Software	John K. Stevenson
Unisyn, Inc.	Gary Betts
	Stephen Egbert (Alt.)
Up and Running	Martin Tracy
Vesta Technology	Jack Woehr
Individual Members	Loring Craymer
	John Rible
	J. E. (Jet) Thomas
X3 Liasons	Clyde R. Camp
	Kathleen McMillan

The following organizations and individuals have also participated in this project as Technical Committee members, alternates, or observers. The Technical Committee recognizes and respects their contributions.

Organizations

British Columbia Inst. of Tech.
Computer Cowboys
Computer Sciences Corp.
Computer Strategies, Inc.
Digalog Corp.
Embedded Sys. Programming Mag.
Forth Interest Group (FIG)
H.B. Pascal & Co., Inc.
Harris Semiconductor
IBM Corporation
IEEE
Kelly Enterprises
Laboratory Microsystems, Inc.
Maxtor Corp.
MCI Telecommunications Corp.
Micromotion
MicroProcessor Engineering Ltd.
National Inst. of Standards & Technology
NCR Medical Systems Group
Performance Packages, Inc.
Purdue University
Robert Berkey Services
Shaw Laboratories
Social Security Administration
Software Engineering
Texas Instruments
The Dickens Company

Individuals

David J. Angel
Wil Baden
Robert Berkey
Ron Braithwaite
Jack Brown
Chris Colburn
Ted Dickens
John Dorband
Ray Duncan
Douglas Fishman
Tom Hand
Gregory Ilg
Charles Keane
Guy M. Kelly
Andrew Kobziar
Martin Lascelles
Charles Moore
Mike Nemeth
Harry Pascal
Stephen Pelc
Dean Perrine
David C. Petty
Bill Ragsdale
Dennis Ruffer
James Ryland
Dean Sanderson
George Shaw
Gerald Shifrin
Robert Smith
Tyler Sperry
Tom Zimmer

Contents

1. Introduction 1
1.1 Purpose 1
1.2 Scope 1
1.2.1 Inclusions 1
1.2.2 Exclusions 1
1.3 Document organization 1
1.3.1 Word sets 1
1.3.2 Annexes 2
1.4 Future directions 2
1.4.1 New technology 2
1.4.2 Obsolescent features 2
2. Terms and notation 3
2.1 Definitions of terms 3
2.2 Notation 5
2.2.1 Numeric notation 5
2.2.2 Stack notation 5
2.2.3 Parsed-text notation 5
2.2.4 Glossary notation 6
3. Usage requirements 8
3.1 Data types 8
3.1.1 Data-type relationships 9
3.1.2 Character types 9
3.1.3 Single-cell types 10
3.1.4 Cell-pair types 11
3.1.5 System types 11
3.2 The implementation environment 12
3.2.1 Numbers 12
3.2.2 Arithmetic 12
3.2.3 Stacks 13
3.2.4 Operator terminal 14
3.2.5 Mass storage 14
3.2.6 Environmental queries 14
3.3 The Forth dictionary 15
3.3.1 Name space 15
3.3.2 Code space 15
3.3.3 Data space 15
3.4 The Forth text interpreter 17
3.4.1 Parsing 17
3.4.2 Finding definition names 18
3.4.3 Semantics 18
3.4.4 Possible actions on an ambiguous condition 19
3.4.5 Compilation 19
4. Documentation requirements 21
4.1 System documentation 21
4.1.1 Implementation-defined options 21
4.1.2 Ambiguous conditions 22
4.1.3 Other system documentation 23
4.2 Program documentation 23
4.2.1 Environmental dependencies 23
4.2.2 Other program documentation 23
5. Compliance and labeling 24
5.1 ANS Forth systems 24
5.1.1 System compliance 24

5.1.2 System labeling 24
5.2 ANS Forth programs 24
5.2.1 Program compliance 24
5.2.2 Program labeling 24
6. Glossary 25
6.1 Core words 25
6.2 Core extension words 51
7. The optional Block word set 61
8. The optional Double-Number word set 67
9. The optional Exception word set 72
10. The optional Facility word set 76
11. The optional File-Access word set 79
12. The optional Floating-Point word set 89
13. The optional Locals word set 105
14. The optional Memory-Allocation word set 110
15. The optional Programming-Tools word set 113
16. The optional Search-Order word set 120
17. The optional String word set 125
A. Rationale (informative annex) 128
B. Bibliography (informative annex) 181
C. Perspective (informative annex) 183
C.1 Features of Forth 183
C.2 History of Forth 184
C.3 Hardware implementations of Forth 184
C.4 Standardization efforts 184
C.5 Programming in Forth 185
C.6 Multiprogrammed systems 191
C.7 Design and management considerations 191
C.8 Conclusion 192
D. Compatibility analysis of ANS Forth (informative annex) 193
D.1 FIG Forth (circa 1978) 193
D.2 Forth 79 193
D.3 Forth 83 193
D.4 Recent developments 194
D.5 ANS Forth approach 194
D.6 Differences from Forth 83 195
D.6.1 Stack width 195
D.6.2 Number representation 195
D.6.3 Address units 196
D.6.4 Address increment for a cell is no longer two 196
D.6.5 Address alignment 197
D.6.6 Division/modulus rounding direction 198
D.6.7 Immediacy 198
D.6.8 Input character set 200
D.6.9 Shifting with UM/MOD 200
D.6.10 Vocabularies / wordlists 201
D.6.11 Multiprogramming impact 201
D.6.12 Words not provided in executable form 202
E. ANS Forth portability guide (informative annex) 203
E.1 Introduction 203
E.2 Hardware peculiarities 203
E.2.1 Data/memory abstraction 203

E.2.2 Definitions 203
E.2.3 Addressing memory 204
E.2.4 Alignment problems 204
E.3 Number representation 205
E.3.1 Big endian vs. little endian 205
E.3.2 ALU organization 205
E.4 Forth system implementation 206
E.4.1 Definitions 206
E.4.2 Stacks 206
E.5 ROMed application disciplines and conventions 207
E.6 Summary 207
F. Alphabetic list of words (informative annex) 208