IBM

Series/1

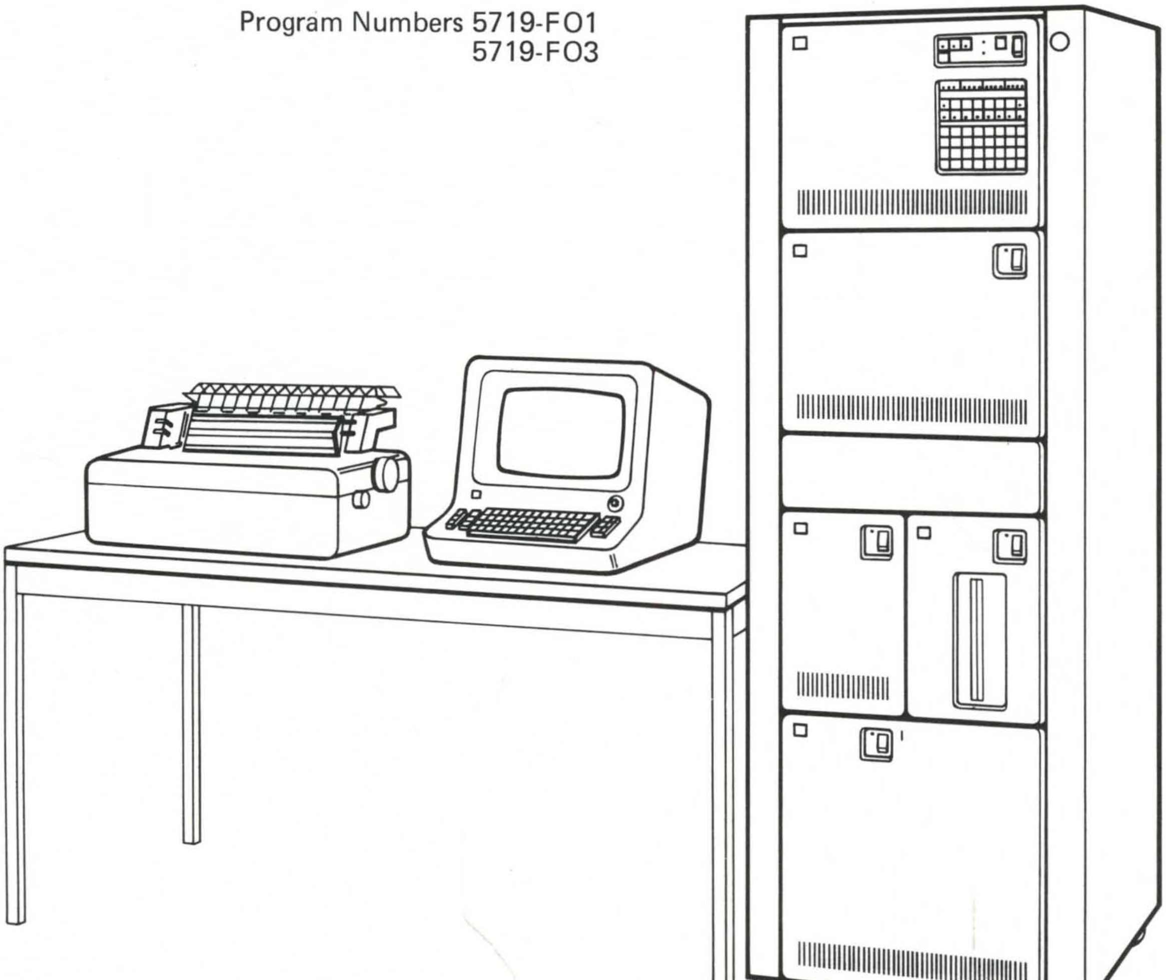
GC34-0132-0

S1-25

PROGRAM
PRODUCT

IBM Series/1
FORTRAN IV
Introduction

Program Numbers 5719-FO1
5719-FO3



FORTRAN IV INTRODUCTION

IBM

Series/1

GC34-0132-0

S1-25

PROGRAM
PRODUCT

FORTRAN IV INTRODUCTION

IBM Series/1
FORTRAN IV
Introduction

Program Numbers 5719-FO1
5719-FO3

This publication is for planning purposes only. The information herein is subject to change before the products described become available.

First Edition (February 1977)

This edition applies to IBM Series/1 FORTRAN IV (compiler and object support library), Program Number 5719-FO1, and IBM Series/1 Realtime Subroutine Library, Program Number 5719-FO3.

Significant changes or additions to the contents of this publication will be reported in subsequent revisions or Technical Newsletters. Requests for copies of IBM publications should be made to your IBM representative or the IBM branch office serving your locality.

A form for readers' comments is provided at the back of this publication. If the form has been removed, send your comments to IBM Corporation, Systems Publications, Department 27T, P. O. Box 1328, Boca Raton, Florida 33432. Comments become the property of IBM.

Contents

Preface v

Series/1 FORTRAN IV Program Product 1

Resources for Series/1 FORTRAN IV 3
- Minimum System Configuration 3
- Secondary Storage Estimates 3

Libraries 3
- FORTRAN IV Object Support Library 3
- FORTRAN IV Realtime Subroutine Library 4
- Mathematical and Functional Subroutine Library 5

FORTRAN IV Diagnostics 5

FORTRAN IV Error Handling 5

FORTRAN IV Language Elements 6
- Multiple Program Support 6
- Logical and Relational Operations 6
- Bit-Level Operations 6
- Direct-Access Input/Output 7
- Additional READ and WRITE Parameters 7
- Device-Independent Input/Output 7
- List-Directed Input/Output 7
- Double-Precision 7
- Carriage Control for Reports 7
- Six-Character Names 7
- Expanded Character Set 7
- A Comparison of FORTRAN IV Features 8
- FORTRAN IV Language Statements 10

Index 13

Preface

This manual introduces the Series/1 FORTRAN IV program product (compiler and object support library) by providing an overview of the Series/1 FORTRAN IV language elements, the FORTRAN IV-supplied object support library, and the FORTRAN IV Realtime Subroutine Library and the Mathematical and Functional Subroutine Library program products. The resources required for the preparation and execution of your FORTRAN IV application programs are summarized and a list of the minimum system configuration is provided.

The language is a subset of American National Standard FORTRAN, X3.9-1966, and includes the American National Standard (ANS) Basic FORTRAN, X3.10-1966, with the exception of object time formats, adjustable dimensions, COMPLEX data type, G-format specifications, and two-level FORMAT parenthesis. Also included are IBM extensions to the language.

This publication is intended to be used by management personnel as a planning and evaluation aid and by programmers as an overview of the features and characteristics of Series/1 FORTRAN IV.

Associated Publications

IBM Series/1 FORTRAN IV: Language Reference, GC34-0133
IBM Series/1 System Summary, GA34-0035
IBM Series/1 Realtime Programming System: Introduction and Planning Guide, GC34-0102
IBM Series/1 Program Preparation Subsystem: Introduction, GC34-0121
IBM Series/1 Mathematical and Functional Subroutine Library: Introduction, GC34-0138

Series/1 FORTRAN IV Program Product

The Series/1 FORTRAN IV program product is designed for programming applications on the IBM Series/1 computer and consists of a compiler and object support library. The FORTRAN IV compiler produces code with emphasis on compact storage and execution speed. FORTRAN IV and your Series/1 computer together allow you to:

- Perform calculations and make decisions
- Read information from sensor devices and send commands back to controlling devices
- Schedule programs and input/output operations
- Create reports and disk data sets

FORTRAN IV is a high-level, mathematically-oriented language designed to manipulate numerical data and format input/output operations. In addition to being easily learned and understood, applications can be programmed without knowledge of the Series/1 assembler language. Programming productivity is increased since high-level languages require less coding than assembler languages. An example of the difference in these two languages is:

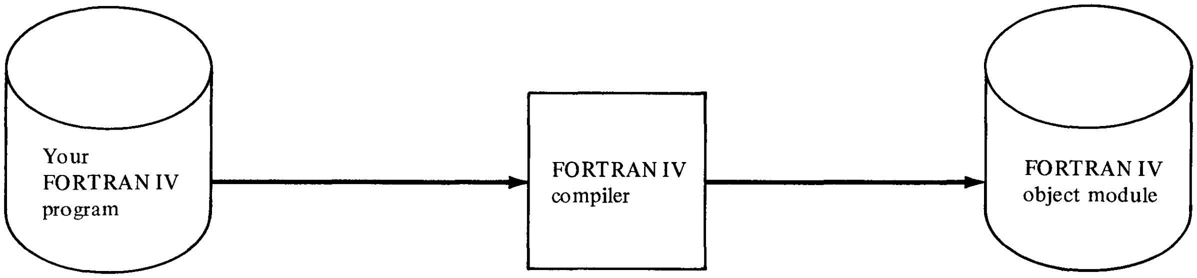
Mathematical formula: I = J + K - L

FORTRAN IV solution   Series/1 assembler solution

```
J = 3                MVW  J,R7
K = 2                AW   K,R7
L = 3                SW   L,R7
.                    MVW  R7,I
.                    .
.                    .
I = J + K - L        .
                     I    DS   F
                     J    DS   F
                     K    DS   F
                     L    DS   F
```

The FORTRAN IV compiler is a serially reusable, single task set which executes in a batch environment under the IBM Series/1 Program Preparation Subsystem (hereafter referred to as Program Preparation Subsystem) job stream processor. During execution, the compiler operates under the control of the IBM Series/1 Realtime Programming System (hereafter referred to as Realtime Programming System) facilities required for your application programs. FORTRAN IV object modules are combined with library routines by the Program Preparation Subsystem application builder to form load modules which execute under the control of the Realtime Programming System. Figure 1 describes the preparation of Series/1 FORTRAN IV programs.

Compiling - translating FORTRAN IV source statements into an object module.



Application building - combining FORTRAN IV object modules with library modules to form an executable task set or a reprocessable load module (suitable as input to the application builder for subsequent processing with other load modules to form a task set).

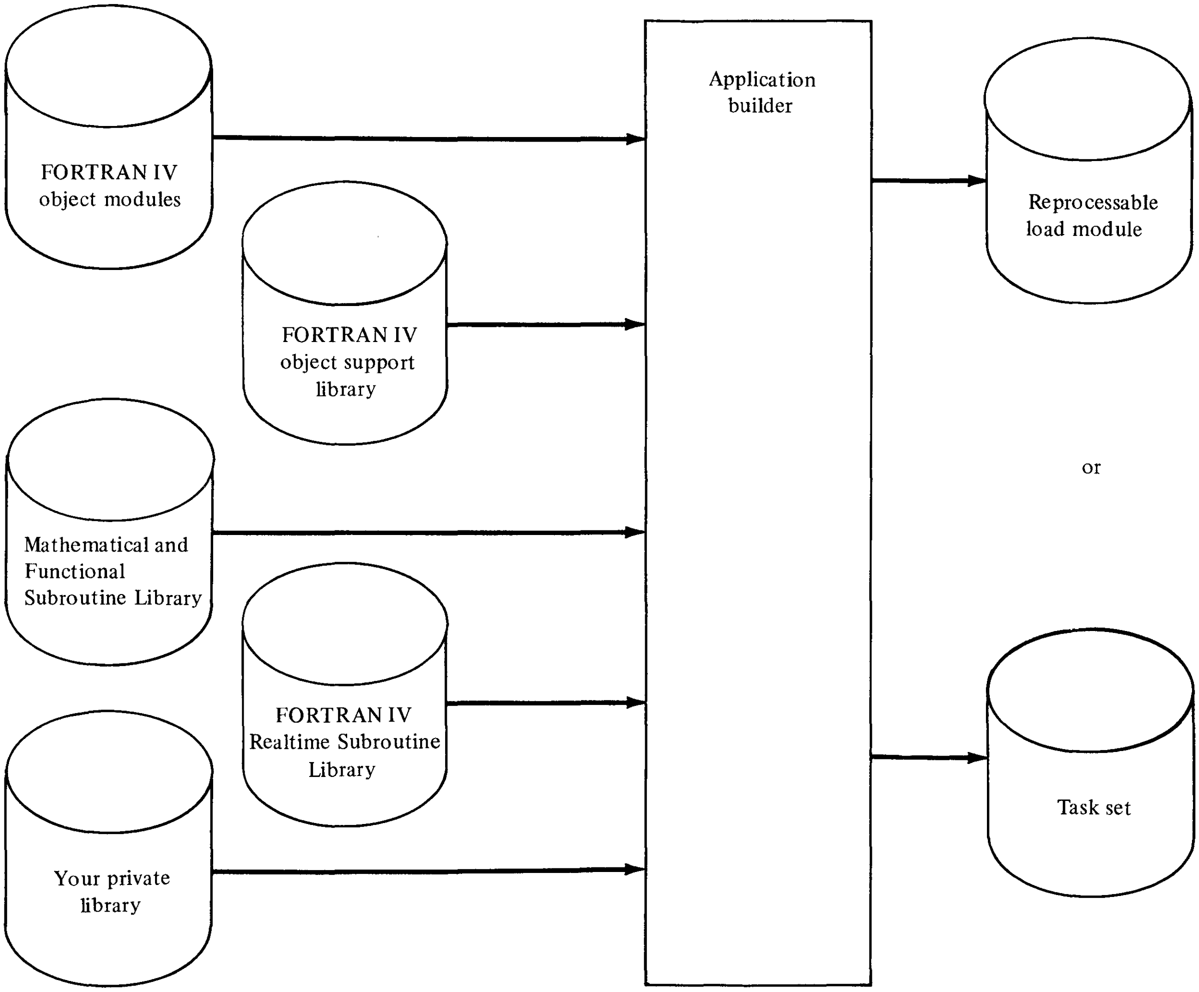


Figure 1. Preparation of Series/1 FORTRAN IV programs

Resources for Series/1 FORTRAN IV

IBM Series/1 FORTRAN IV compiles under the control of the Program Preparation Subsystem (Program Product 5719-AS1). Object code executes either in the Program Preparation Subsystem environment or under control of the Realtime Programming System (Program Product 5719-PC1). Floating-point support, either hardware or emulation package, is required for object execution if the user performs floating-point (REAL) operations.

Minimum System Configuration

The minimum Series/1 system configuration for preparation and execution of your FORTRAN IV program is as follows:

- Operator Station
- 4953 or 4955 Processor with storage for the Realtime Programming System and 16KB user partition for the compiler.
- 4962 Disk Storage Unit
- 4964 Diskette Unit
- Device for output, for example: 4962 Disk Storage Unit, 4964 Diskette Unit, 4974 Printer

Secondary Storage Estimates

- Compiler: 450 sectors (see Note)
- JSP environment lists: 45 sectors
- Intermediate file: 400 sectors (see Note)
- Object support library: 360 sectors
- FORTRAN IV Realtime Subroutine Library: 200 sectors
- MFSL requirements: See *Mathematical and Functional Subroutine Library: Introduction*
- Source input file: Depends on program to be compiled
- Object output file: Depends on program compiled

Note. Block size of 256 bytes is required. For optimum performance, sector size should be 256 bytes.

Libraries

Series/1 FORTRAN IV provides you with modular implementation of all library routines, thereby reducing program size. Library routines are re-entrant and can be used by multiple application programs.

FORTRAN IV Object Support Library

The FORTRAN IV object support library contains subroutines which are combined, as necessary, with the object modules produced by the FORTRAN IV compiler. The object support library routines perform:

- Input/output processing
- Error handling
- Explicit and implicit service operations

When you need a library routine, the compiler generates a call to it. During processing by the Program Preparation Subsystem application builder, a copy of the library routine is made part of the storage load. The storage load module is then ready to execute on your Series/1 computer.

FORTRAN IV Realtime Subroutine Library

The FORTRAN IV Realtime Subroutine Library: Program Product 5719-FO3, provides realtime system support. The subroutines contained in this library conform to the Instrument Society of America (ISA) Standard, S61.1-1976, and are available to your programs via the CALL statement. The procedures include:

- *Executive function* subroutines, which provide the ability to start, stop, or delay the execution of programs. These subroutines are:

START	execute
TRNON	execute at a specified time
WAIT	suspend execution

- *Process I/O* subroutines, which access analog and digital points for both input and output. These subroutines are:

Analog

AISQW	reads input points in sequence
AIRDW	reads input points in user-specified sequence
AOW	writes output registers in user-specified sequence

Digital

DIW	reads input registers
DOMW	sets and resets user-selected DO points
DOLW	sets or resets user-selected DO groups

- *System service interface* subroutines, which access system services. These subroutines are:

$ATACH	attach task
$AWAIT	wait on completion of event
$CON	connect task to PI interrupt
$DEQUE	remove element from queue
$DFNEV	define event
$DFNQU	define storage queue
$DFNRS	define resource
$DISCN	disconnect from PI interrupt
$DLTEV	delete event
$DLTQU	delete queue
$DLTRS	delete resource
$DTACH	detach (terminate) task
$ENQUE	add element to queue
$MDSST	modify system scheduler table
$MDSTT	modify system task set table
$POST	post completion of event
$RDTOD	read time-of-day
$RELRS	release resource
$REQRS	request resource
$SERXT	set task error exit routine
$SETRL	set ROLLIN/ROLLOUT status
$TSQUE	queue task set for execution
$TSSTP	terminate task set execution
$WRTOD	write time-of-day

- *Time* subroutine, which determines the current time of day. This subroutine is:

  TIME

- *Date* subroutine, which determines the current calendar date. This subroutine is:

  DATE

Mathematical and Functional Subroutine Library

The Mathematical and Functional Subroutine Library: Program Product 5719-LM1 is a prerequisite and provides the following subroutines:

- *Mathematical*: arc tangent, one or two arguments; cosine; exponential function; exponentiation; hyperbolic tangent; logarithms, common and natural; maximum value; minimum value; modular arithmetic; positive difference; sine; square root; transfer of sign.
- *Conversion*: EBCDIC to floating point; EBCDIC to integer; floating point to EBCDIC; integer to EBCDIC
- *Error-checking*: function test; floating-point divide exception; floating-point overflow/underflow.

FORTRAN IV Diagnostics

FORTRAN IV-provided diagnostic aids or services assist in program creation and debugging. Source statements are verified for proper syntax during compilation. For faster analysis, statement syntax can be checked without producing object code. These statements produce a log of program activity (such as a trace of routine labels) while your program is executing. You may request a variety of maps and listings from the FORTRAN IV compiler:

- Listing of source statements
- Statement label map with relative addresses
- Map of storage locations for variables and arrays
- Hexadecimal listing of the object deck
- Cross-reference index for symbols and labels

The FORTRAN IV library contains routines which provide diagnostic information in case error conditions occur during program execution. You may use these routines or supply your own.

The debug facility consists of a DEBUG specification statement, an AT debug packet identification statement, and the TRACE statement. These statements are used to specify the desired debugging operations for a program.

FORTRAN IV Error Handling

While your FORTRAN IV program is executing, errors may result from invalid data usage or hardware failures. FORTRAN IV assists in detecting these errors and taking appropriate action.

The mathematical routines check the arguments passed to them for validity. If an argument is invalid, corrective action is taken and processing continues uninterrupted. For example, if you pass a negative argument to the ALOG (natural algorithm) function, the absolute value of the argument is used. You can check within your FORTRAN IV program to determine whether corrective action has been taken.

Other errors can be checked within your FORTRAN IV program. These include:

- Overflow — Arithmetic result greater than the maximum valid real number
- Underflow — Arithmetic result less than the smallest valid nonzero real number
- Divide check — Attempt to divide a real number by zero
- Function code — Illegal argument/invalid data

For I/O device errors, FORTRAN IV offers several levels of error handling. The lowest level of support is the printing of an informative message prior to program termination. If you want your program to take some particular action prior to termination, you can specify an error handling routine (using the ERRXIT subroutine) to receive control when I/O errors occur. An additional level of error handling is provided by the ERR= and END= parameters on I/O statements to specify different error-handling routines for different devices and situations.

FORTRAN IV Language Elements

Series/1 FORTRAN IV is a subset of American National Standard (ANS) FORTRAN, X3.9-1966, and includes American National Standard Basic FORTRAN, X3.10-1966, with the exception of object time formats, adjustable dimensions, COMPLEX data type, G-format specifications, and two-level FORMAT parenthesis. Series/1 FORTRAN IV also has many language elements not provided by ANS Basic FORTRAN; these are referred to collectively as the Series/1 FORTRAN IV extensions.

The Series/1 FORTRAN IV extensions adapt FORTRAN to the Series/1 environment and provide greater programming flexibility. The significance of several extensions follows.

Multiple Program Support

Three FORTRAN IV statements (PROGRAM, INVOKE, and GLOBAL) permit optimal use of the interrupt scheme. The PROGRAM statement assigns a unique name to each of several main programs thereby establishing multiple entry points for execution in response to interrupts. The INVOKE statement permits you to select the combination and sequence of programs to be loaded into main storage. The GLOBAL statement establishes a data area common to two or more main programs. These three statements help you model your programming support according to your interrupt configuration and allow you to pass information and control between various program elements as conditions warrant.

Logical and Relational Operations

To aid in making decisions, you can define logical variables and assign them the values true or false. Furthermore, you can evaluate a logical or relational expression with the logical IF statement and take appropriate action according to the logical value of the expression. For example, the expression A.GT.B (A greater than B) can be evaluated to be either true or false. In this example, .GT. is a relational operator. The valid logical and relational operators are listed under the section "FORTRAN IV Language Statements".

Bit-Level Operations

With most FORTRAN languages you work at the word or character level. To work with sensor-based activities you need to interrogate and manipulate the bits that correspond to sensor-based input/output points. Series/1 FORTRAN IV includes these functions for bit-level operations:

• ICOMP or NOT	Logical *complement* of an argument
• IAND	Logical *and* of 2 arguments
• IOR	Logical *or* of 2 arguments
• IEOR	Logical *exclusive or* of 2 arguments
• ISHFT	Logical *shift* of an argument
• BTEST	Logical *test* of a specified bit
• IBSET	*Set* a specified bit
• IBCLR	*Clear* a specified bit

Direct-Access Input/Output

Series/1 FORTRAN IV supports both formatted and unformatted direct-access data sets on the 4962 Disk Storage Unit. The direct-access support is implemented through the DEFINE FILE, READ, WRITE, and FIND statements.

Additional READ and WRITE Parameters

The ERR= and END= parameters on the READ statement and the ERR= parameter on the WRITE statement give you added flexibility and control. ERR= specifies the label of a routine to handle errors during either a sequential or direct-access READ or WRITE. END= specifies the label of a routine to handle end-of-file during a sequential READ.

Device-Independent Input/Output

Your Series/1 FORTRAN IV programs may be written so that assignment of physical devices to many data sets can be deferred until you are ready to execute the program. These assignments can then be changed as required for subsequent use without recording and recompiling the program.

List-Directed Input/Output

List-directed I/O simplifies data entry by freeing you from FORMAT statement restrictions. Data may be entered (or is written) without regard for column or line boundaries.

Double-Precision

Statistical applications often require greater accuracy than single-precision (6+ places decimal) can offer. The Series/1 FORTRAN IV compiler, therefore, includes a double-precision capability.

Carriage Control for Reports

You may specify carriage control characters in report data sets created by your FORTRAN IV program; these characters cause the printer carriage to skip lines or start new pages while the data set is being printed.

Six-Character Names

Names of up to 6 alphameric characters may be used for variables, arrays, functions, and subroutines within Series/1 FORTRAN IV programs. This technique, which allows you to use more meaningful names, improves program documentation and simplifies maintenance efforts.

Expanded Character Set

The Series/1 FORTRAN IV character set includes the dollar sign ($) as an alphabetic character and the ampersand (&) and apostrophe (') as special characters in addition to the basic FORTRAN character set, which includes the letters A,B...Z, the numbers 0,1...9, and the characters plus (+), minus (-), slash (/), equal (=), asterisk (*), the left and right parentheses, and the blank.

A Comparison of FORTRAN IV Features

FORTRAN Features	*Series/1 FORTRAN IV features not supported in IBM Basic FORTRAN IV*	*Series/1 FORTRAN IV features not supported in ANS FORTRAN*	*Series/1 FORTRAN IV features not supported in IBM System/360 and System/370 full FORTRAN IV*	*Series/1 FORTRAN IV not available in IBM System/360 and System/370 full FORTRAN IV*	*Series/1 FORTRAN IV not available in FORTRAN IV H extended features*
Adjustable (object time) dimensioning				X	
ASSIGN	X				
ASSIGN GO TO	X				
Asynchronous I/O					X
BLOCK DATA	X				
COMPLEX				X	
DATA	X				
Debug facility	X				
Debug with UNIT, INIT, or SUBCHK				X	
Direct-access I/O statements		X			
DISPLAY				X	
ENTRY	X	X			
ERR & END parameters in a READ	X	X			
ERR parameter in a WRITE	X	X	X		
EXECUTIVE functions	X				
Exponentiation or function reference in subscripts appearing in I/O lists				X	
Extended precision					X
EXTERNAL extension					X
FORMAT with G specification, or more than 1 level of parenthesis, or adjustable (object time) format specification				X	
Function names in type statements		X			
Generalized subscript form	X	X			
GENERIC					X
GLOBAL		X	X		
Hexadecimal constant		X			
IMPLICIT	X	X			
Initial data values in explicit specification statements	X	X			
Integer *2 data type		X			
INVOKE	X	X	X		
L & Z Format Codes	X				

FORTRAN Features	*Series/1 FORTRAN IV features not supported in IBM Basic FORTRAN IV*	*Series/1 FORTRAN IV features not supported in ANS FORTRAN*	*Series/1 FORTRAN IV features not supported in IBM System/360 and System/370 full FORTRAN IV*	*Series/1 FORTRAN IV not available in IBM System/360 and System/370 full FORTRAN IV*	*Series/1 FORTRAN IV not available in FORTRAN IV H extended features*
Labeled COMMON	X				
Length of variables and arrays as part of type specification	X	X			
A subset of FORTRAN IV extended list-directed I/O					X
Literal as actual argument in CALL and function reference	X	X			
Literal enclosed in apostrophes		X			
LOGICAL*4	X				
LOGICAL*1				X	
Logical, literal, and hexadecimal constants	X				
Logical IF	X				
Mixed-mode expressions		X			
Multiple exponentiation w/o parentheses to indicate order of computation		X			
NAMELIST				X	
PAUSE with literal	X	X			
PRINT				X	
Process I/O subroutines	X				
PROGRAM		X	X		
PUNCH				X	
RETURN (i) i not a blank	X	X			
READ b list				X	
Subprogram dummy arguments enclosed in slashes				X	
T and Z format codes		X			
Up to 7 dimensions in an array	X	X			

FORTRAN IV Language Statements

Specification Statements

DIMENSION (allowing arrays of up to 7 dimensions)
IMPLICIT
COMMON, labelled, blank
GLOBAL, labelled, blank
DATA
EXTERNAL
PROGRAM
Type statement, with or without length specifications, and allowing data initialization and dimension information.
EQUIVALENCE

Relational and Logical Operators

.EQ.	-	Equal to
.NE.	-	Not equal to
.LT.	-	Less than
.GT.	-	Greater than
.LE.	-	Less than or equal to
.GE.	-	Greater than or equal to
.OR.	-	Or
.AND.	-	And
.NOT.	-	Not

Control Statements

GO TO, unconditional
GO TO, computed
GO TO, assigned
DO
CONTINUE
ASSIGN
IF, arithmetic
IF, logical
PAUSE, with optional message or 5-digit identifying number
STOP, with optional 5-digit identifying number
END

Assignment Statements

Arithmetic assignment, allowing mixed-mode expression and multiple exponentiation
Logic assignment

Input/Output Statements

READ/WRITE, list-directed
READ/WRITE, sequential with optional END= and ERR= parameters on READ; optional ERR= parameter on WRITE
READ/WRITE, direct-access both formatted and unformatted, with optional ERR= parameter on READ and WRITE
FIND
DEFINE FILE
BACKSPACE
REWIND
END FILE
FORMAT, with these codes: A, D, E, F, I, H, L, P, T, X, Z, "literal"

Subprograms and Statement Functions

SUBROUTINE
FUNCTION, allowing type specification of function name and redefinition or arguments
CALL, allowing subprogram name and literal constants as arguments
RETURN
ENTRY, allowing multiple entry points in one subroutine
BLOCK DATA, to initialize labelled COMMON

Constants

INTEGER (INTEGER*4)
INTEGER (INTEGER*2)
LOGICAL
REAL *4
REAL *8

Index

assignment statements 10

bit-level operations 6

carriage control 7
character set 7
compiler 1
configuration, minimum system 3
constants 11
control statements 10

device-independent I/O 7
diagnostics 5
direct-access I/O 6
divide-check 5
double-precision 7

END= parameter 7
ERR= parameter 7
error-handling 5
executive function subroutines 4

floating-point support 3
FORTRAN IV comparisons 8
FORTRAN IV realtime subroutine library 4
FORTRAN IV object support library 3

hexadecimal listing 5

I/O statements 11
ISA subroutines 4

language statements 10
libraries 3
list-directed I/O 7
listings 5
logical and relational operations 6
logical operators 10

mathematical and functional subroutine library 5
MFSL 5
minimum system configuration 3
multiple program support 6

naming technique 7

object support library 3
operator station 3
overflow 5

parameters 7
process I/O subroutines 4
program preparation 2

READ statement 7
realtime library 4
relational operators 10
resources 3

single-precision 7
source statement listing 5
specification statements 10
storage estimates 3
storage location map 5
subprograms and statement functions 11
symbol and label index 5
system configuration, minimum 3
system service subroutines 4

time and date subroutines 4

underflow 5

WRITE statement 7

4953 processor 3
4955 processor 3
4962 disk storage unit 3
4964 diskette unit 3
4974 printer 3
4979 display station 3
4982 sensor I/O unit 3

FORTRAN IV:
Introduction
GC34-0132-0

READER'S
COMMENT
FORM

Cut or Fold Along Line

YOUR COMMENTS, PLEASE . . .

Your comments assist us in improving the usefulness of our publications; they are an important part of the input used in preparing updates to the publications. All comments and suggestions become the property of IBM.

Please do not use this form for technical questions about the system or for requests for additional publications; this only delays the response. Instead, direct your inquiries or requests to your IBM representative or to the IBM branch office serving your locality.

Corrections or clarifications needed:

Page Comment

What is your occupation?____________________________________
Number of latest Technical Newsletter (if any) concerning this publication: ______________
Please indicate your name and address in the space below if you wish a reply.

____

____

____

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments.)

GC34-0132-0

Your comments, please . . .

This manual is part of a library that serves as a reference source for IBM systems. Your comments on the other side of this form will be carefully reviewed by the persons responsible for writing and publishing this material. All comments and suggestions become the property of IBM.

Cut Along Line

Fold

Fold

First Class
Permit 40
Armonk
New York

Business Reply Mail
No postage stamp necessary if mailed in the U.S.A.

IBM Corporation
Systems Publications, Dept 27T
P.O. Box 1328
Boca Raton, Florida 33432

Fold

Fold

IBM Series/1 FORTRAN IV: Introduction Printed in U.S.A. GC34-0132-0

IBM®

International Business Machines Corporation
General Systems Division
5775D Glenridge Drive N.E.
P.O. Box 2150, Atlanta, Georgia 30301
(U.S.A. only)

FORTRAN IV:
Introduction
GC34-0132-0

READER'S
COMMENT
FORM

Cut or Fold Along Line

YOUR COMMENTS, PLEASE . . .

Your comments assist us in improving the usefulness of our publications; they are an important part of the input used in preparing updates to the publications. All comments and suggestions become the property of IBM.

Please do not use this form for technical questions about the system or for requests for additional publications; this only delays the response. Instead, direct your inquiries or requests to your IBM representative or to the IBM branch office serving your locality.

Corrections or clarifications needed:

Page Comment

What is your occupation?____
Number of latest Technical Newsletter (if any) concerning this publication: __________________
Please indicate your name and address in the space below if you wish a reply.

____

____

____

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A.
(Elsewhere, an IBM office or representative will be happy to forward your comments.)

Your comments, please . . .

This manual is part of a library that serves as a reference source for IBM systems. Your comments on the other side of this form will be carefully reviewed by the persons responsible for writing and publishing this material. All comments and suggestions become the property of IBM.

Cut Along Line

Fold

Fold

First Class
Permit 40
Armonk
New York

Business Reply Mail
No postage stamp necessary if mailed in the U.S.A.

IBM Corporation
Systems Publications, Dept 27T
P.O. Box 1328
Boca Raton, Florida 33432

Fold

Fold

IBM®
International Business Machines Corporation
General Systems Division
5775D Glenridge Drive N.E.
P.O. Box 2150, Atlanta, Georgia 30301
(U.S.A. only)

International Business Machines Corporation

General Systems Division
5775D Glenridge Drive N.E.
P. O. Box 2150
Atlanta, Georgia 30301
(U.S.A. only)

IBM Series/1 FORTRAN IV: Introduction Printed in U.S.A. GC34-0132-0