

Specification for

Binary floating point arithmetic for microprocessor systems

Committees responsible for this British Standard

The preparation of this British Standard was entrusted by the Information Systems Technology Standards Policy Committee (IST/-) to Technical Committee IST/6, upon which the following bodies were represented:

- Association for Payment Clearing Services
- British Computer Society
- British Telecommunications plc
- Business Equipment and Information Technology Association
- Department of Trade and Industry (Information Technology Division)
- Department of Trade and Industry (National Physical Laboratory)
- Electricity Supply Industry in England and Wales
- Electronic Engineering Association
- HM Treasury (Central Computer and Telecommunications Agency)
- Information Technology Users' Standards Association
- Inter-Universities Computing Committee
- Joint Network Team
- LAMSAC
- National Computing Centre Ltd.
- National Health Service
- OFTEL (Office of Telecommunications)
- Post Office
- Telecommunication Engineering and Manufacturing Association
- Telecommunications Managers' Association
- User Standards Forum for Information Technology (Institute of Data Processing Management)

This British Standard, having been prepared under the direction of the Information Systems Technology Standards Policy Committee, was published under the authority of the Board of BSI and comes into effect on 31 July 1990

© BSI 07-1999

The following BSI references relate to the work on this standard:
Committee reference IST/6
Draft for comment 89/65509 DC

ISBN 0 580 18114 6

Amendments issued since publication

Amd. No.	Date	Comments

Contents

	Page
Committees responsible	Inside front cover
National foreword	ii
1 Scope	1
1.1 Implementation objectives	1
1.2 Inclusions	1
1.3 Exclusions	1
2 Definitions	1
3 Formats	2
3.1 Sets of values	2
3.2 Basic formats	3
3.3 Extended formats	4
3.4 Combinations of formats	4
4 Rounding	5
4.1 Round to nearest	5
4.2 Directed roundings	5
4.3 Rounding precision	5
5 Operations	5
5.1 Arithmetic	5
5.2 Square root	6
5.3 Floating-point conversions	6
5.4 Conversions between floating point and integer	6
5.5 Round floating-point number to integral value	6
5.6 Binary $\leftrightarrow$ decimal conversion	6
5.7 Comparison	7
6 Infinity, NaNs and signed zero	8
6.1 Infinity arithmetic	8
6.2 Operations with NaNs	8
6.3 The sign bit	9
7 Exceptions	9
7.1 Invalid operations	9
7.2 Division by zero	9
7.3 Overflow	10
7.4 Underflow	10
7.5 Inexact	11
8 Traps	11
8.1 Trap handler	11
8.2 Precedence	11
Appendix A Recommended functions and predicates	12
Figure 1 — Single format	4
Figure 2 — Double format	4
Table 1 — Summary of format parameters	3
Table 2 — Decimal conversion ranges	7
Table 3 — Correctly rounded decimal conversion range	7
Table 4 — Predicates and relations	8

National foreword

This British Standard, prepared under the direction of the Information Systems Technology Standards Policy Committee, is identical with IEC 559:1989 “*Binary floating-point arithmetic for microprocessor systems*”, published by the International Electrotechnical Commission (IEC).

A British Standard does not purport to include all the necessary provisions of a contract. Users of British Standards are responsible for their correct application.

Compliance with a British Standard does not of itself confer immunity from legal obligations.

Summary of pages

This document comprises a front cover, an inside front cover, pages i and ii, pages 1 to 12 and a back cover.

This standard has been updated (see copyright date) and may have had amendments incorporated. This will be indicated in the amendment table on the inside front cover.

1 Scope

1.1 Implementation objectives

It is intended that an implementation of a floating-point system conforming to this standard can be realized entirely in software, entirely in hardware, or in any combination of software and hardware. It is the environment that the programmer or user of the system sees that conforms or fails to conform to this standard. Hardware components that require software support to conform shall not be said to conform apart from such software.

1.2 Inclusions

This standard specifies:

- 1) basic and extended floating-point number formats;
- 2) add, subtract, multiply, divide, square root, remainder and compare operations;
- 3) conversions between integer and floating-point numbers;
- 4) conversions between different floating-point formats;
- 5) conversions between basic format floating-point numbers and decimal strings, and
- 6) floating-point exceptions and their handling, including non-numbers (NaNs).

1.3 Exclusions

This standard does not specify:

- 1) formats of decimal strings and integers;
- 2) interpretation of the signs and significant fields of NaNs, or
- 3) binary $\leftrightarrow$ decimal conversions to and from extended formats.

2 Definitions

biased exponent

the sum of the exponent and a constant (bias) chosen to make the biased exponent's range non-negative

binary floating-point number

a bit-string characterized by three components: a sign, a signed exponent, and a significand. Its numerical value, if any, is the signed product of its significand and two raised to the power of its exponent. In this standard a bit-string is not always distinguished from a number it may represent

denormalized number

a nonzero floating-point number, the exponent of which has a reserved value, usually the format's minimum, and the explicit or implicit leading significant bit of which is zero

destination

the location for the result of a binary or unary operation. The destination may be either explicitly designated by the user or implicitly supplied by the system (e.g. intermediate results in sub-expressions or arguments for procedures). Some languages place the results of intermediate calculations in destinations beyond the user's control. Nonetheless, this standard defines the result of an operation in terms of that destination's format as well as the operands' values

exponent

the component of a binary floating-point number that normally signifies the integer power to which two is raised in determining the value of the represented number. Occasionally the exponent is called the signed or unbiased exponent

fraction

the field of the significand that lies to the right of its implied binary point

mode

a variable that a user may set, sense, save and restore, to control the execution of subsequent arithmetic operations. The default mode is the mode that a program can assume to be in effect unless an explicitly contrary statement is included either in the program or in its specification

the following modes shall be implemented:

- 1) rounding, to control the direction of rounding errors, and in certain implementations.
- 2) rounding precision, to shorten the precision of results. The implementor may, at his option, implement the following modes:
- 3) traps disabled/enabled, to handle exceptions.

NaN

not a number; a symbolic entity encoded in floating-point format. There are two types of NaNs (see 6.2). Signalling NaNs signal the invalid operation exception (see 7.1) whenever they appear as operands. Quiet NaNs propagate through almost every arithmetic operation without signalling exceptions

result

the bit-string (usually representing a number) that is delivered to the destination

significant

the component of a binary floating-point number which consists of an explicit or implicit leading bit to the left of its implied binary point and a fraction field to the right

shall

the word “shall” signifies that which is obligatory in any conforming implementation

should

the word “should” signifies that which is strongly recommended as being in keeping with the intent of the standard, although architectural or other constraints beyond the scope of this standard may, on occasion, render the recommendations impractical

status flag

a variable that may take two states, set and clear. A user may clear a flag, copy it, or restore it to a previous state. When set, a status flag may contain additional system-dependent information, possibly inaccessible to some users. The operations of this standard may, as a side-effect, set some of the following flags: inexact result, underflow, overflow, divide by zero and invalid operation

user

any person, hardware, or program not itself specified by this standard, having access to and controlling those operations of the programming environment specified in this standard

3 Formats

This standard defines four floating-point formats in two groups, basic and extended, each having two widths, single and double. The standard levels of implementation are distinguished by the combinations of formats supported.

3.1 Sets of values

This sub-clause concerns only the numerical values representable within a format, not the encodings which are the subject of the following sub-clauses. The only values representable in a chosen format are those specified via the following three integer parameters:

- | | | |
|------------|---|--|
| P | = | number of significant bits (precision) |
| $E_{\max}$ | = | maximum exponent, and |
| $E_{\min}$ | = | minimum exponent |

Each format's parameters are displayed in Table 1. Within each format just the following entities shall be provided:

Numbers of the form $(-1)^s 2^E (b_0 b_1 b_2 \dots b_{p-1})$

where:

s is 0 or 1;

E is any integer between $E_{\min}$ and $E_{\max}$ inclusive, and each b_i is 0 or 1.