TECHNICAL SPECIFICATION

ISO/IEC TS 18661-1

First edition
2014-07-15

Information technology — Programming languages, their environments, and system software interfaces — Floating-point extensions for C —

Part 1: Binary floating-point arithmetic

Technologies de l'information — Langages de programmation, leurs environnements et interfaces du logiciel système — Extensions à virgule flottante pour C —

Partie 1: Arithmétique binaire en virgule flottante

Reference number
ISO/IEC TS 18661-1:2014(E)

© ISO/IEC 2014



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2014

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Introduction v
Background v
IEC 60559 floating-point standard v
C support for IEC 60559 vi
Purpose vii

1 Scope 1

2 Conformance 1

3 Normative references 1

4 Terms and definitions 2

5 C standard conformance 2
5.1 Freestanding implementations 2
5.2 Predefined macros 2
5.3 Standard headers 3

6 Revised floating-point standard 5

7 Types 6
7.1 Terminology 6
7.2 Canonical representation 7

8 Operation binding 8

9 Floating to integer conversion 13

10 Conversions between floating types and character sequences 13
10.1 Conversions with decimal character sequences 13
10.2 Conversions to character sequences 14

11 Constant rounding directions 15

12 NaN support 22

13 Integer width macros 27

14 Mathematics <math.h> 29
14.1 Nearest integer functions 29
14.1.1 Round to integer value in floating type 29
14.1.2 Convert to integer type 31
14.2 The llogb functions 34
14.3 Max-min magnitude functions 35
14.4 The nextup and nextdown functions 36
14.5 Functions that round result to narrower type 37
14.6 Comparison macros 40
14.7 Classification macros 41
14.8 Total order functions 43
14.9 Canonicalize functions 44
14.10 NaN functions 45

15 The floating-point environment <fenv.h> 47
15.1 The fesetexcept function 47
15.2 The fetestexceptflag function 48
15.3 Control modes 48

16 Type-generic math <tgmath.h> 50

Bibliography 52

© ISO/IEC 2014 – All rights reserved

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the WTO principles in the Technical Barriers to Trade (TBT) see the following URL: Foreword - Supplementary information

The committee responsible for this document is ISO/IEC JTC 1, *Information technology*, SC 22, *Programming languages, their environments, and system software interfaces*.

ISO/IEC TS 18661 consists of the following parts, under the general title *Information technology — Programming languages, their environments, and system software interfaces — Floating-point extensions for C*:

— *Part 1: Binary floating-point arithmetic*

— *Part 2: Decimal floating-point arithmetic*

— *Part 3: Interchange and extended types*

— *Part 4: Supplementary functions*

— *Part 5: Supplementary attributes*

Part 1 updates ISO/IEC 9899:2011, *Information technology — Programming languages — C*, Annex F in particular, to support all required features of ISO/IEC/IEEE 60559:2011, *Information technology — Microprocessor Systems — Floating-point arithmetic*.

Part 2 supersedes ISO/IEC TR 24732:2009, *Information technology — Programming languages, their environments and system software interfaces — Extension for the programming language C to support decimal floating-point arithmetic*.

Parts 3-5 specify extensions to ISO/IEC 9899:2011 for features recommended in ISO/IEC/IEEE 60559:2011.

 © ISO/IEC 2014 – All rights reserved

Introduction

Background

IEC 60559 floating-point standard

The IEEE 754-1985 standard for binary floating-point arithmetic was motivated by an expanding diversity in floating-point data representation and arithmetic, which made writing robust programs, debugging, and moving programs between systems exceedingly difficult. Now the great majority of systems provide data formats and arithmetic operations according to this standard. The IEC 60559:1989 international standard was equivalent to the IEEE 754-1985 standard. Its stated goals were:

1 Facilitate movement of existing programs from diverse computers to those that adhere to this standard.

2 Enhance the capabilities and safety available to programmers who, though not expert in numerical methods, may well be attempting to produce numerically sophisticated programs. However, we recognize that utility and safety are sometimes antagonists.

3 Encourage experts to develop and distribute robust and efficient numerical programs that are portable, by way of minor editing and recompilation, onto any computer that conforms to this standard and possesses adequate capacity. When restricted to a declared subset of the standard, these programs should produce identical results on all conforming systems.

4 Provide direct support for

   a. Execution-time diagnosis of anomalies

   b. Smoother handling of exceptions

   c. Interval arithmetic at a reasonable cost

5 Provide for development of

   a. Standard elementary functions such as exp and cos

   b. Very high precision (multiword) arithmetic

   c. Coupling of numerical and symbolic algebraic computation

6 Enable rather than preclude further refinements and extensions.

To these ends, the standard specified a floating-point model comprising:

formats – for binary floating-point data, including representations for Not-a-Number (NaN) and signed infinities and zeros

operations – basic arithmetic operations (addition, multiplication, etc.) on the format data to compose a well-defined, closed arithmetic system; also conversions between floating-point formats and decimal character sequences, and a few auxiliary operations

context – status flags for detecting exceptional conditions (invalid operation, division by zero, overflow, underflow, and inexact) and controls for choosing different rounding methods

The ISO/IEC/IEEE 60559:2011 international standard is equivalent to the IEEE 754-2008 standard for floating-point arithmetic, which is a major revision to IEEE 754-1985.

The revised standard specifies more formats, including decimal as well as binary. It adds a 128-bit binary format to its basic formats. It defines extended formats for all of its basic formats. It specifies data interchange

© ISO/IEC 2014 – All rights reserved

formats (which may or may not be arithmetic), including a 16-bit binary format and an unbounded tower of wider formats. To conform to the floating-point standard, an implementation must provide at least one of the basic formats, along with the required operations.

The revised standard specifies more operations. New requirements include – among others – arithmetic operations that round their result to a narrower format than the operands (with just one rounding), more conversions with integer types, more classifications and comparisons, and more operations for managing flags and modes. New recommendations include an extensive set of mathematical functions and seven reduction functions for sums and scaled products.

The revised standard places more emphasis on reproducible results, which is reflected in its standardization of more operations. For the most part, behaviors are completely specified. The standard requires conversions between floating-point formats and decimal character sequences to be correctly rounded for at least three more decimal digits than is required to distinguish all numbers in the widest supported binary format; it fully specifies conversions involving any number of decimal digits. It recommends that transcendental functions be correctly rounded.

The revised standard requires a way to specify a constant rounding direction for a static portion of code, with details left to programming language standards. This feature potentially allows rounding control without incurring the overhead of runtime access to a global (or thread) rounding mode.

Other features recommended by the revised standard include alternate methods for exception handling, controls for expression evaluation (allowing or disallowing various optimizations), support for fully reproducible results, and support for program debugging.

The revised standard, like its predecessor, defines its model of floating-point arithmetic in the abstract. It neither defines the way in which operations are expressed (which might vary depending on the computer language or other interface being used), nor does it define the concrete representation (specific layout in storage, or in a processor's register, for example) of data or context, except that it does define specific encodings that are to be used for data that may be exchanged between different implementations that conform to the specification.

IEC 60559 does not include bindings of its floating-point model for particular programming languages. However, the revised standard does include guidance for programming language standards, in recognition of the fact that features of the floating-point standard, even if well supported in the hardware, are not available to users unless the programming language provides a commensurate level of support. The implementation’s combination of both hardware and software determines conformance to the floating-point standard.

C support for IEC 60559

The C standard specifies floating-point arithmetic using an abstract model. The representation of a floating-point number is specified in an abstract form where the constituent components (sign, exponent, significand) of the representation are defined but not the internals of these components. In particular, the exponent range, significand size, and the base (or radix) are implementation-defined. This allows flexibility for an implementation to take advantage of its underlying hardware architecture. Furthermore, certain behaviors of operations are also implementation-defined, for example in the area of handling of special numbers and in exceptions.

The reason for this approach is historical. At the time when C was first standardized, before the floating-point standard was established, there were various hardware implementations of floating-point arithmetic in common use. Specifying the exact details of a representation would have made most of the existing implementations at the time not conforming.

Beginning with ISO/IEC 9899:1999 (C99), C has included an optional second level of specification for implementations supporting the floating-point standard. C99, in conditionally normative Annex F, introduced nearly complete support for the IEC 60559:1989 standard for binary floating-point arithmetic. Also, C99’s informative Annex G offered a specification of complex arithmetic that is compatible with IEC 60559:1989.

 © ISO/IEC 2014 – All rights reserved

ISO/IEC 9899:2011 (C11) includes refinements to the C99 floating-point specification, though is still based on IEC 60559:1989. C11 upgrades Annex G from “informative” to “conditionally normative”.

ISO/IEC TR 24732:2009 introduced partial C support for the decimal floating-point arithmetic in ISO/IEC/IEEE 60559:2011. ISO/IEC TR 24732, for which technical content was completed while IEEE 754-2008 was still in the later stages of development, specifies decimal types based on ISO/IEC/IEEE 60559:2011 decimal formats, though it does not include all of the operations required by ISO/IEC/IEEE 60559:2011.

Purpose

The purpose of ISO/IEC TS 18661 is to provide a C language binding for ISO/IEC/IEEE 60559:2011, based on the C11 standard, that delivers the goals of ISO/IEC/IEEE 60559 to users and is feasible to implement. It is organized into five Parts.

Part 1, this document, provides changes to C11 that cover all the requirements, plus some basic recommendations, of ISO/IEC/IEEE 60559:2011 for binary floating-point arithmetic. C implementations intending to support ISO/IEC/IEEE 60559:2011 are expected to conform to conditionally normative Annex F as enhanced by the changes in Part 1.

Part 2 enhances ISO/IEC TR 24732 to cover all the requirements, plus some basic recommendations, of ISO/IEC/IEEE 60559:2011 for decimal floating-point arithmetic. C implementations intending to provide an extension for decimal floating-point arithmetic supporting ISO/IEC/IEEE 60559:2011 are expected to conform to Part 2.

Part 3 (Interchange and extended types), Part 4 (Supplementary functions), and Part 5 (Supplementary attributes) cover recommended features of ISO/IEC/IEEE 60559:2011. C implementations intending to provide extensions for these features are expected to conform to the corresponding Parts.

© ISO/IEC 2014 – All rights reserved

Information technology — Programming languages, their environments, and system software interfaces — Floating-point extensions for C —

Part 1:
Binary floating-point arithmetic

1 Scope

This part of ISO/IEC TS 18661 extends programming language C to support binary floating-point arithmetic conforming to ISO/IEC/IEEE 60559:2011. It covers all requirements of IEC 60559 as they pertain to C floating types that use IEC 60559 binary formats.

This part of ISO/IEC TS 18661 does not cover decimal floating-point arithmetic, nor does it cover most optional features of IEC 60559.

This part of ISO/IEC TS 18661 is primarily an update to IEC 9899:2011 (C11), normative Annex F (IEC 60559 floating-point arithmetic). However, it proposes that the new interfaces that are suitable for general implementations be added in the Library clauses of C11. Also it includes a few auxiliary changes in C11 where the specification is problematic for IEC 60559 support.

2 Conformance

An implementation conforms to this part of ISO/IEC TS 18661 if

a) It meets the requirements for a conforming implementation of C11 with all the changes to C11 specified in this part of ISO/IEC TS 18661; and

b) It defines **__STDC_IEC_60559_BFP__** to **201404L.**

3 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 9899:2011, *Information technology — Programming languages — C*

ISO/IEC 9899:2011/Cor.1:2012, *Technical Corrigendum 1*

ISO/IEC/IEEE 60559:2011, *Information technology — Microprocessor Systems — Floating-point arithmetic* (with identical content to IEEE 754-2008, *IEEE Standard for Floating-Point Arithmetic*. The Institute of Electrical and Electronic Engineers, Inc., New York, 2008)

© ISO/IEC 2014 – All rights reserved

4 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 9899:2011, ISO/IEC/IEEE 60559:2011 and the following apply.

4.1
C11
standard ISO/IEC 9899:2011, *Information technology — Programming Languages — C*, including *Technical Corrigendum 1* (ISO/IEC 9899:2011/Cor.1:2012)

5 C standard conformance

5.1 Freestanding implementations

The following change to C11 expands the conformance requirements for freestanding implementations so that they might conform to this part of ISO/IEC TS 18661.

Change to C11:

Insert after the third sentence of 4#6:

> The strictly conforming programs that shall be accepted by a conforming freestanding implementation that defines **__STDC_IEC_60559_BFP__** may also use features in the contents of the standard headers **<fenv.h>** and **<math.h>** and the numeric conversion functions (7.22.1) of the standard header **<stdlib.h>**. All identifiers that are reserved when **<stdlib.h>** is included in a hosted implementation are reserved when it is included in a freestanding implementation.

5.2 Predefined macros

The following changes to C11 obsolesce **__STDC_IEC_559__**, the current conformance macro for Annex F, in favor of **__STDC_IEC_60559_BFP__**, for consistency with other conformance macros and to distinguish its application to binary floating-point arithmetic. The macro **__STDC_IEC_559__** is retained as obsolescent, for compatibility with existing programs.

Changes to C11:

In 6.10.8.3#1, before:

> **__STDC_IEC_559__** The integer constant **1**, intended to indicate conformance to Annex F (IEC 60559 binary floating-point arithmetic).

insert:

> **__STDC_IEC_60559_BFP__** The integer constant **201404L**, intended to indicate conformance to Annex F (IEC 60559 binary floating-point arithmetic).

In 6.10.8.3#1, append to the **__STDC_IEC_559__** item:

> Use of this macro is an obsolescent feature.

The following changes to C11 obsolesce **__STDC_IEC_559_COMPLEX__**, the current conformance macro for Annex G, in favor of **__STDC_IEC_60559_COMPLEX__**, for consistency with other conformance macros.

© ISO/IEC 2014 – All rights reserved