TS 18661-1
FOR C2X

WG 14 - Pittsburg
October 17-21, 2016

C FP group
TS 18661-1 for C2x

TS 18661 background
TS 18661-1
  - Overview
  - Contents
  - Summary
Floating-point and C standards

- IEEE 754 1985
- IEC 60559:1989
- C90 (1990)
- C99 (2000)
- TR 24732
  - 60559 decimal
- IEEE 754 2008
- IEC 60559:2011
- C11 (2010)
- TS 18661
  - Full 60559 support
Background

Specify a C binding for IEEE 754-2008

- Work began 2009
- Under direction of ISO/IEC JTC1/SC22/WG14 – C
- Expertise in floating-point and language standards, compilers, libraries
Principles

• Support all of the current FP standard, as-is
• Specify as changes to C11
• Use existing C mechanisms, minimize language invention
• Develop specification in parts, to pipeline process
• Supersede TR 24732 (decimal)
• Allow support by free-standing C implementations
• Deliver an ISO/IEC Technical Specification
Status

• In five parts
  Required features in IEC 60559
  1. Binary floating-point arithmetic
  2. Decimal floating-point arithmetic
  Recommended features in IEC 60559
  3. Interchange and extended types
  4. Supplementary functions
  5. Supplementary attributes

• All parts published 2014-2016
Publications


- ISO/IEC TS 18661-4:2015, Information Technology — Programming languages, their environments, and system software interfaces — Floating-point extensions for C — Part 4: Supplementary functions

- ISO/IEC TS 18661-5:2016, Information Technology — Programming languages, their environments, and system software interfaces — Floating-point extensions for C — Part 5: Supplementary attributes
TS 18661-1 overview

- C currently supports the 1989 version of the IEC 60559 floating-point standard
- ISO/IEC TS 18661-1:2014 updates this support to the required features for binary floating-point arithmetic in the current IEC 60559 standard (2011)
- TS 18661-1 includes specific changes to C11
1-4: Boilerplate, introduction, terminology, etc.
5.1: Allows conformance for freestanding implementations.
5.2: Conformance macro __STDC_IEC_60559_BFP__ obsoletes __STDC_IEC_559__ as declaration of support for annex F.
__STDC_IEC_60559_COMPLEX__ obsoletes __STDC_IEC_559_COMPLEX__.
5.3: Lists all identifiers defined or declared by user-defined __STDC_WANT_IEC_60559_BFP_EXT__.
6: Updates introduction to annex F.
7.1: Updates terminology to match IEC 60559.
7.2: Adds concept of canonical encoding (in 5.2.4.2.2). Cleans up use of term *encoding*.
8: Updates C binding for IEC 60559, in annex F.
9: Updates footnote to clarify IEC 60559 recommendation on implicit floating-to-integer conversions and the “inexact” exception.
10.1: Updates annex F conversions to decimal character sequences to support new stricter requirements. Adds new `<float.h>` macro `CR_DECIMAL_DIG`.
10.2: Adds new *strto* functions in `<stdlib.h>`, to enable support by freestanding implementations.
11: Adds support for new IEC 60559 requirement for constant rounding modes:

- Changes to distinguish *constant vs dynamic* control modes (mostly just inserting “dynamic” where appropriate).
- New rounding control pragma

```
#pragma STDC FENV_ROUND direction
```
- Identifies library functions affected by the pragma (when macro expansion has not been disabled). User functions not affected.
- Example shows how to implement constant rounding modes with dynamic ones.
12: Cleans up NaN specifications so an implementation could support signaling NaNs within the C standard:

- Implementation may define \texttt{FE\_SNANS\_ALWAYS\_SIGNAL} to declare support for signaling NaNs.
- Clarifies that F.10 latitude for “underflow” and “inexact” applies to functions not covered by IEC 60559.

13: Adds “width” macros, e.g., \texttt{LONG\_WIDTH}, for all integer types, for use with \texttt{roundfp} functions and for general use.
14: Adds new required `<math.h>` functions and macros:

- **fromfp**, etc. convert all floating types to integers, of all widths, signed and unsigned, w/ and w/o “inexact” exceptions.

- **llogb** -- since **logbl** doesn’t meet IEC 60559 spec if **int** is 16 bits and **long double** is binary128

- **fmaxmag**, **fminmag**

- **nextup**, **nextdown**
14: New `<math.h>` functions and macros (cont.):

- **fadd, faddl, daddl** round result to narrower type, and “fast” macros for them. Similar functions for subtract, multiply, divide, FMA, and square root.
- **iseqsig** -- `==` but with “invalid” on quiet NaN arguments.
- **iscanonical, issignaling, issubnormal, iszero. iszero** avoids triggering signaling NaNs.
- **totalorder, totalordermag** – total order for canonical numbers in the type.
- **canonicalize** – the IEC 60559 `convertFormat` operation for same-type conversions.
- **getpayload, setpayload, setpayloadsig** for NaN significand bits.
15: Adds new `<fenv.h>` features:

- **fesetexcept** sets exceptions without signaling.
- **fetestexceptflag** tests saved exceptions, for fewer accesses to dynamic modes.
- **fegetmode**, **fesetmode**, **femode_t**, **FE_DFL_MODE** manage dynamic modes collectively.

16: `<tgmath.h>` update for new functions.
Summary

• C currently supports IEC 60559:1989. TS 18661-1 updates this support to the required features for binary FP in the current IEC 60559:2011.
• Changes are primarily in the library.
• Most features already implemented.

Overview

- C11 supports the 1989 version of the IEC 60559 standard.
- TS 18661-2 adds the required features for decimal floating-point arithmetic in the current IEC 60559 standard (2011).
- TS 18661-2 gives specific changes to C11 + TS 18661-1.
- Implementation may support IEC 60559 for binary or decimal FP, or both, or neither.
1-4: Boilerplate, introduction, terminology, etc.
5.1: Allows conformance for freestanding decimal implementations.
5.2: Conformance macro \texttt{__STDC_IEC_60559_DFP__}. Updates F.1 so that annex F applies to decimal (and/or binary).
5.3: Lists all identifiers defined or declared by user-defined \texttt{__STDC_WANT_IEC_60559_DFP_EXT__}.
6: Adds \textit{decimal floating types} \texttt{_Decimal32}, \texttt{_Decimal64}, \texttt{_Decimal128}. Complex and imaginary types not defined for decimal. Defines \textit{standard floating types} to be \texttt{float}, \texttt{double}, and \texttt{long double}. 
Contents

7: Adds characteristics for decimal floating types in `<float.h>`, analogous to binary. Defines view of FP model as triples (s, c, q). Specifies preferred quantum exponents.

8: Expands binding in annex F to include decimal.

9: Specifies conversions involving decimal floating types. Enhances usual arithmetic conversions to handle decimal.

10: Defines suffixes for decimal floating constants. Not for hexadecimal.

11: Prohibits operations mixing decimal with standard, complex, or imaginary floating types. Gives overview of details to come. Updates F.9.2 with expression-transformation issues specific to decimal.
12.2: Defines macros for decimal rounding modes, like binary, but including **FE_DEC_TONEARESTFROMZERO**. Binary and decimal use same exception flags. Adds constant rounding mode pragma for decimal.

12.3: Adds decimal analogs for binary features in `<math.h>`.

12.4: Adds decimal specific functions:

- **quantizedN** adjust quantum exponent.
- **samequantumdN** tests for same quantum exponents.
- **quantumdN** computes argument’s quantum.
- **llquantexpdN** computes argument’s quantum exponent.
- encode and decode functions for external data in the two allowed IEC 60559 decimal encodings.
12.5: Adds I/O specifiers for decimal. Style a formatting for decimal that preserves quantum exponents.

12.6-8: Adds \texttt{strto} and \texttt{strfrom} functions for decimal, similar to binary. \texttt{strto} functions preserves quantum exponent. Wide string versions too.

13: Adds \texttt{<tgmath.h>} support for decimal. Examples.
Summary

• C currently supports IEC 60559:1989. TS 18661-2 updates this support to the required features for decimal FP in the current IEC 60559:2011.
• Most features correspond to features for binary.
• Most features already implemented.