

WG14 N2648
Meeting notes

C Floating Point Study Group Teleconference

2021-01-13
8 AM PST / 11 AM EST / 4 PM UTC

Attendees: Rajan, Jim, Fred, Ian, Mike, David O., Damian, David H.

New agenda items:
None

Carry over action items:
None.

Last meeting action items (all done unless specified otherwise below):

Fred: Redo updates to N2546 with the changes in CFP 1859.

Jim: Give a short summary of differences between N2579 and N2601 (differences between TS part 3 as an Annex updates 2 and 3).

Fred: Update the example in G.5.1 using fmax to use the newer functions as a new proposal.

Rajan: Respond to the WG14 reflector message to say CFP wants equivalence to strtod and hence we don't want to parse digit separators either.

Fred: Write a WG14 editorial informational paper as per CFP 1821.

Fred: Write a CFP paper for the pow(1,NaN) and the compound(NaN, 0) case with respect to quantum exponent of the result.

David H: Look into numerically equivalent vs numerically equal usage in the C standard and revisit CFP 1849. - *Not done

Fred/Jim: Have a statement in the main body of the standard saying opposite signed zeros compare equal.

Fred: Still want to change negative to say less than zero in certain cases in C.

Jim: I will send out something to say negative zero and NaN's with a negative sign bit are not negative values in C.

Jim: Need to reword the signbit description in C.

New action items:

Jim/Fred: Go through all the CFP proposals submitted to ensure they were put in the C standard draft (N2596) correctly.

Fred: Submit CFP 1869 to WG14 with the typo fix of adding a space to the second last change.

Fred: Resend the document relating to updating the example in G.5.1 using max to use the newer functions as a new proposal.

Fred: Submit CFP 1870 to WG14.

Fred: Send out the changed text (CFP 1891) to 754.

Rajan: Test the change in CFP 1891 with IBM's implementation.

Fred: Submit the change proposed in CFP 1891 to WG14.

Jim/Fred: Make CFP 1866 into a draft WG14 proposal with the change "Positive zeros compare equal to negative zeros."

Fred: Change "negative signed value" to "a value with a negative sign" in CFP 1886 and submit it to WG14.

Jim: Make CFP 1881 into a WG14 proposal removing "with a negative sign bit".

Jim: Look at other uses of the phrase "sign bit" in the C standard.

Jim: Submit CFP 1879 to WG14 changing "reports" to "determines" in the footnote.

Rajan: Look into whether "should" can be used in an ISO standard. (Yes, allowed:
See <https://www.iso.org/foreword-supplementary-information.html>)

Jim: Submit CFP 1883 as a WG14 proposal.

Jim: Get permission from WG14 to revise TS 18661-5 along the lines of what is proposed for TS 18661-4 in CFP 1856.

Fred: Follow up with the WG14 editor about the changes in CFP 1874.

Jim: Combine CFP 1880 and CFP 1885, and update CFP 1885 to address the and/or ambiguity in the first suggested change and put it in the form of a proposal.

Jim: Get permission from WG14 to revise TS 18661-5 along the lines of what is proposed for TS 18661-4 in CFP 1856.

Next Meeting(s):

Wednesday, February 17th, 2021, 4PM UTC

ISO Zoom teleconference

Please notify the group if this time slot does not work.

WG14 meeting:

Rajan: Went over the WG14 meeting results for our papers. See CFP1868.

C++ liaison:

David O.: Nothing about IEEE, but there is a C/C++ compatibility study group starting up (Aaron Ballman).

C23 integration

Latest C2X draft: <http://www.open-std.org/jtc1/sc22/wg14/www/docs/n2{596,573,478}.pdf> also as link on CFP wiki

Part 2

Part 3

Part 4ab

Part 5abcd

IEC 60559:2020 support

Action item details

Fred: Redo updates to N2546 with the changes in CFP 1859.

See Cfp-interest 1869: DEC_EVAL_METHOD

Jim: Missing space in second last change (before "and").

Good to submit to WG14.

Jim: Give a short summary of differences between N2579 and N2601 (differences between TS part 3 as an Annex updates 2 and 3).

See Cfp-interest 1867: AI to list latest changes in TS part 3 as annex

No issues.

Fred: Update the example in G.5.1 using fmax to use the newer functions as a new proposal.

AI: Fred to resend the document relating to updating the example in G.5.1 using max to use the newer functions as a new proposal.

Rajan: Respond to the WG14 reflector message to say CFP wants equivalence to strtod and hence we don't want to parse digit separators either.

See WG14 reflector message 18602.

No issues.

Fred: Write a WG14 editorial informational paper as per CFP 1821.

See Cfp-interest 1870: +(x)

No issues (unary + is the copy operation).

AI: Fred: Submit CFP 1870 to WG14.

Fred: Write a CFP paper for the pow(1,NaN) and the compound(NaN, 0) case with respect to quantum exponent of the result.

See Cfp-interest 1872, 1890, 1891: Quantum exponent of NaN

Jim: We generally don't like getting ahead of IEEE, but this is a case where it seems OK.

Mike: Yes, this is a 754 errata that will be updated.

Fred: Sent to 754's list but no response.

Mike: Took the no response as assent.

AI: Fred: Send out the changed text (CFP 1891) to 754.

AI: Rajan: Test this change (CFP 1891) with IBM's implementation.

AI: Fred: Submit the change proposed in CFP 1891 to WG14.

David H: Look into 'numerically equivalent' vs 'numerically equal' usage in the C standard and revisit CFP 1849.

Not done.

Carry over

Fred/Jim: Have a statement in the main body of the standard saying opposite signed zeros compare equal.

See Cfp-interest 1866: AI for unsigned zeros to compare equal

Mike: Should say "Positive zeros compare equal to negative zeros."

Jim: Used "compare equal" since that is what was used elsewhere.

AI: Jim/Fred: Make CFP 1866 into a draft WG14 proposal with the change "Positive zeros compare equal to negative zeros."

Fred: Change 'negative' to say 'less than zero' in certain cases in C.

See Cfp-interest 1871, 1886, 1887: Negative

Fred: In the POSIX email list, they said -0 is a negative number. We don't mean that so this change. It is fine for most of the math functions, but printf has an issue.

David said you could say "a value with a negative sign" vs the "negative signed value".

Fred: My understanding is at that level of the hierarchy NaN's don't have a sign.

Mike: At the representation level they can have a sign bit, but that's a different level.

AI: Fred: Change "negative signed value" to "a value with a negative sign" in CFP 1886 and submit it to WG14.

Jim: Send out something to say negative zero and NaN's with a negative sign bit are not negative values in C.

See Cfp-interest 1881: say negative zero and NaN's with a negative sign bit are not negative values

Mike: You could remove the second sentence there. As David said, the fact that NaNs can have negative sign bits depends on the encoding.

Jim: I don't want to use the words "negative NaN".

Mike: Translating a NaN to a string can have a negative sign in front, even though it doesn't have to have a sign bit. Sign bit only applies when talking about an encoding.

Perhaps say "Negative zeros and NaNs are never negative values."

Jim: So remove "with a negative sign bit" from CFP 1881.

AI: Jim: Make CFP 1881 into a WG14 proposal removing "with a negative sign bit".

AI: Jim: Look at other uses of the phrase "sign bit" in the C standard.

Jim: Reword the signbit description in C.
See Cfp-interest 1879: AI to reword signbit description

Mike: Same issue for a negative zero not being a negative value. We should add in "or it is a -0" into the returns section.

Jim: None of the cases here describe -0. In the floating point model, all finite values have a sign. This includes -0.

Jim: For the footnote, should it be changed so reports->determines?
Seems good.

Jim: F.3 seems an awkward place to place the specification of sigbit. Maybe better to put it in F.10?

AI: Jim: Submit CFP 1879 to WG14 changing "reports" to "determines" in the footnote.

Other issues

fabs and copysign cleanup
See Cfp-interest 1883: fabs and copysign cleanup

Mike: For fabs, why not just say floating point values instead of floating point numbers?

Jim: Most descriptions for the functions talk about 'x'.

Mike: The special thing about fabs vs abs is the 'f' meaning floating point values.

Mike: Editorial: The change for F.10.8#1 copysign may have an extra space before the y in the "copysign(x, y)".

AI: Rajan: Look into whether "should" can be used in an ISO standard. (Yes, allowed:
See <https://www.iso.org/foreword-supplementary-information.html>)

AI: Jim: Submit CFP 1883 as a WG14 proposal.

Signaling NaNs
See Cfp-interest 1874, 1875, 1876: SNAN issues

AI: Fred: Follow up with the WG14 editor about the changes in CFP 1874.

5.2.4.2.2 clarification
See Cfp-interest 1880, 1882, 1884, 1885: 5.2.4.2.2 clarification

CFP 1880 seems like an improvement.

Rajan: Suggested change 1 has a problem with the and/or binding leaving it ambiguous.

Jim: Can be changed with a colon and semicolons? Seems heavyweight.

AI: Jim: Combine CFP 1880 and CFP 1885, and update CFP 1885 to address the and/or ambiguity in the first suggested change and put it in the form of a proposal.

Range errors
See Cfp-interest 1841-1843, 1873: C math errors

Ian: For Issue 5, is that for subnormals?

Jim: It is for ease of use. We thought it may be too difficult to know if an operation underflowed or not, so we allowed them to signal underflow.

Jim: At least three separable proposals here: 3,7,4 and 1,8, and the remainder. I'll try to break it down for next time.

Parameterization of interfaces TS 18661 revisions
See Cfp-interest 1856: TS 18661-4 revision

Jim: For part 5, it is hard to read now with the integration of parts 1-3 and parts of 4 into the C standard. It seems to be better as an optional extension rather than a stand alone document. Something like an annex.

David H: We need to get buy in to the big idea before we can get it for the details. We should go ahead and try to get permission for that type of update.

Al: Jim: Get permission from WG14 to revise TS 18661-5 along the lines of what is proposed for TS 18661-4 in CFP 1856.