WG14 N2339
Meeting notes

C Floating Point Study Group Teleconference

2019-01-24
8 AM PST / 11 PM EST / 4 PM UTC

Attendees: Rajan, Jim, Ian, Fred, Mike, David H

New agenda items:
None.

Carry over action items:
Ian: See if there is an incompatibility between C and C++ for constants being evaluated to a wider format (Ex. FLT_EVAL_METHOD affects constants in C++, and wider return values) - Keep open.
Jim: Update the binding table in parts 1 and 2 to handle the new IEEE-754:2018 functions when published. - Keep open.
David: Check the min/max C specification to ensure it matches what IEEE has. - Not done.
David: Check the augmented* C function specifications to ensure they match what IEEE has. - Not done.
All: totalorder* differ for NaN payloads: Note that we don't have approval to move up to 754 201x yet. - Keep open: Revisit after we move up to the 754 draft.
Fred: Ensure that the items for P4_CR_for_rootn.pdf match IEEE. - Done.

Last meeting action items:
Ian: Find the C++ standard reference and macro name for their handling of floating point literals. - See CFP1236, Done.
Jim: Let the WG14 editors know that we are waiting for the Part 1 integrated draft before putting in Part 2. - Done.
Fred: Provide words for a macro for printf n-char-sequence maximum length that the implementations have to define. - Done.
Jim: Provide the NaN payload specification editorial updates (positive signed floating point integers) to the WG14 editors. - Done. Discussed below.

New action items:
Jim: Post the draft of Part 1 integrated into C2x on the CFP site.
Jim: Make the first change as per CFP1238.
Jim: Rework Part 3 as an Annex to add a note saying there is no _Decimal32x since IEC60559 doesn't have _Decimal32 as a basic type.
Jim: Make the _EPSILON change as per CFP1238.
Jim: Make the fourth change as per CFP1238 to add in the hyperbolic versions of cos/sin/tan.
Jim: Make the fifth change as per CFP1238 for the list changes.
All: Take a look at part 3 annex (http://wiki.edg.com/pub/CFP/WebHome/cfp3x-annex-20190119.pdf) draft before next meeting.
Jim: Update the proposal to try again to integrate part 4a into C2X.
David: Check with IEEE group to see if there is any implementations for Part 4b functions (hardware or software).
All: Review the rationale for part 5 a, b, c proposal.
All: Review http://wiki.edg.com/pub/CFP/WebHome/update_for_C2X_payload_functions.pdf.
Rajan: Say to WG14 that CFP supports removing the WANT macros and leaving the rest as is

due to Fred's reasoning above.

Rajan: Check with David to see if we are going to discuss the WANT macro issue on the agenda.

Fred: Send links to papers of interest to CFP from N2319-N2323.

Next Meeting(s):

Wednesday, February 20th, 2019, 11:00 EST, 8:00 PST, 4PM UTC
Same teleconference number.

Discussion:

754 revision:

Moved to sponsor ballot until February 20th.
Got 4 votes, need 33 more for a valid ballot. Need 3/4 for approval.
Set up ballot review committee.
Still a number of open suggestions to deal with.
Have a document listing new features and describing older ones that are often misunderstood. See http://754r.ucbtest.org/background/

C++ Liaison:

None.

C2X integration:

WG14 meeting pre-meeting mailing deadline: March 18th, 2019.

Meeting information: http://www.open-std.org/jtc1/sc22/wg14/www/docs/n2318.htm, Venue information N2308 (linked to in the agenda).

Part 1:

We have a draft of this integrated in already. It is being reviewed already by Fred.
*Jim: Post the draft of Part 1 integrated into C2x on the CFP site.

Part 2:

Need to rebase on Part 1 + C2x draft and then send it out to the WG14 editors for Part 2 integration.

Part 3: Draft annex for review (http://wiki.edg.com/pub/CFP/WebHome/cfp3x-annex-20190119.pdf)

Also see Fred’s email [Cfp-interest 1231]

From CFP1238:

First change: OK.

*Jim: Make the first change as per CFP1238.

Second change: IEEE doesn't specify it (no _Decimal32 basic format to extend). Perhaps say that there is no basic format?

Mike: Not much empirical use of _Decimal64 either. Perhaps make the statement stronger.

David: Is there a corresponding issue for binary16?

Yes, but we don't want to change anything here non-editorially.

Perhaps have a note saying there is no _Decimal32x since IEC60559 doesn't have _Decimal32 as a basic type.

*Jim: Rework Part 3 as an Annex to add a note saying there is no _Decimal32x since IEC60559 doesn't have _Decimal32 as a basic type.

Jim: Already changed the f32xsqrt(n) example on the last page that used to be d32x (in the pre-annex form of part 3).

Third change (*_EPSILON):

Fred: Already been done as a CR and accepted by WG14.

*Jim: Make the _EPSILON change as per CFP1238.

Fourth change (_FloatN_t): OK

*Jim: Make the forth change as per CFP1238 to add in the hyperbolic versions of cos/sin/tan.

Fifth change: OK

*Jim: Make the fifth change as per CFP1238 for the list changes.

*All: Take a look at part 3 annex (http://wiki.edg.com/pub/CFP/WebHome/cfp3x-annex-20190119.pdf) draft before next meeting.

Part 4ab:

Currently up to us to update the TS since WG14 doesn't want to add them in.

Jim: I think they should be a part of the standard as they are completion functions not esoteric ones. Ex. The exp functions.

David: They are also widely implemented.

Jim: For reciprocal square root, rootn, pown, etc. there is rational to have them. Performance impacts with those and can't replace them with optimizations due to different results and single rounding. The compound functions have growth and decay applications.

Jim: The half revolution trig functions avoid roundoff error in arguments, and gives exact results for multiples of PI. Also a performance benefit.

David: There is a 754 background document for the power functions if that is helpful.

*Jim: Update the proposal to try again to integrate part 4a into C2X.

Jim: For the reduction functions (4b), the rationale is portable performance. How strong is this given the parallel world?

David: It's not for portable reproducibility. It's a hypothetical performance advantage since I don't think it's been implemented in a portable way.

Can wait for further implementations so keep it as a TS for these functions.

*David: Check with IEEE group to see if there is any implementations for Part 4b functions (hardware or software).

Part 5abcd:

Jim: There is prior art for a lot of this like reproducibility, just different ways of doing it per implementation. I would like to re-propose a, b, c with new rationale since it seems ripe for standardization given the different ways of spelling or mechanisms of doing it.

Fred: My experience in testing is that most compilers don't even do the C99 standard pragmas.

*All: Review the rationale for part 5 a, b, c proposal.

Action item details:

C++ standard reference and macro name for their handling of floating point literals (CFP1236).

Ian to talk to the IBM C++ standards rep again.

Macro for printf n-char-sequence maximum length that the implementations have to define.

See Fred’s [Cfp-interest 1223] and Jim’s [Cfp-interest 1234].

Looks good.

Ensure that the items for P4_CR_for_rootn.pdf match IEEE.

See Fred’s email [Cfp-interest 1222]

Looks good.

NaN payload specification editorial updates (positive signed floating point integers).

See Jim’s 1/9 email “AI about update to NaN payload spec”

http://wiki.edg.com/pub/CFP/WebHome/update_for_C2X_payload_functions.pdf

*All: Review http://wiki.edg.com/pub/CFP/WebHome/update_for_C2X_payload_functions.pdf.

Other issues:

Optional features in C: WG14 email thread “optional features for IEC 60559 integration”.

Some proposals were a different header, function prefixes, etc.

Fred: In a TS it needs a WANT macro since it is not a part of the standard. It is not needed once it is a part of the standard.

Rajan: Can always use the STDC_VERSION macro to avoid collisions, but still has problems with getting the old functions.

*Rajan: Say to WG14 that CFP supports removing the WANT macros and leaving the rest as is due to Fred's reasoning above.

Jim: Perhaps we should propose what should be removed from what we have there now. We can follow the _Complex model for _Decimal.

*Rajan: Check with David to see if we are going to discuss the WANT macro issue on the agenda.

WG14 papers N2319-N2323 by Tydeman: See (SC22WG14.16023), New documents on the WG 14 website.

*Fred: Send links to papers of interest to CFP from N2319-N2323.

Consider C17 footnote 232: Is it ok?

P3 has tgmath example d32xsqrt(n) -> d32xsqrtd64

But there is no _Decimal32x type.

Changed already to f32.