C Floating Point Study Group Teleconference

2019-07-17
8 AM PDT / 11 PM EDT / 3 PM UTC

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

 New agenda items:
   None.

 Carry over action items:
   Fred: Give a new version of the SNAN initialization paper (as per CFP1316). - Done.

 Last meeting action items:
   Jim: Create a link to the 250 draft into the references section in the C FP wiki. - Done.
   Rajan: Forward the IEEE article to WG14 once David H sends it out to us. - Not done.
    David: OK to send the draft even though it is not final and has been changed.
   Jim: Draft a slide deck and a proposal based on CFP1331. - Partially done. Slide deck to carry over. Proposal done.
   Jim: Draft a note to warn about CFP1315's rounding of negative constants issue. - Done.

 New action items:
   Jim: Draft a slide deck based on CFP1331.
   Jim: Add wording to the CFP1331 proposal make it clear this is for particular evaluation methods, and submit the document to W14.
   Jim: Add a statement as to why there is a second name for log1p as a footnote as a new proposal.
   Fred: Submit CFP 1340 to WG14.
   Jim: Reword CFP 1337 as per action item discussion in the 2019/07/17 meeting.
   Jim: Send CFP 1349 to the WG14 reflector as the CFP position.
   Fred: Write a paper to make the range error for small nonzero x consistent for expm1, logp1, log10p1's other base functions.
   Jim: Add in the normalized discussion from Fred (CFP 1341, CFP 1342) to the agenda for next meeting.

 Next Meeting(s):
   Wednesday, August 21st, 2019, 11:00 AM EDT, 8:00 AM PDT, 3:00 PM UTC
   Same teleconference number.
   Please notify the group if this time slot does not work.

 Discussion:
   754 revision:
    Should be done. Likely to be published on Monday. Draft 263 seems to be good.
    Likely one more meeting to discuss the background document, repository and maintenance, hand over to any future group for a future revision.
    Mike will send the 263 pdf for posting on the CFP wiki.
    Jim: Specifying identities for math functions was something I wanted to ask the 754 group. Will bring it up there.

C++ Liaison:
  Nothing new.ok

C2X integration:
  Draft with TS 1, 2, and 4a: http://wiki.edg.com/pub/CFP/WebHome/all-20190708.pdf
  Part 3 – as annex, given to Jens: http://wiki.edg.com/pub/CFP/WebHome/n2405.pdf
  Part 4b - Looking as an updated TS.
  Part 5a,b,c,d – Discuss later. Part 5d is a TS update.

 Action item details:
  Fred: Give a new version of the SNAN initialization paper (as per CFP1316). See Fred's CFP 1340.
    *Fred: Submit this paper to WG14

  Jim: Draft a slide deck and a proposal based on CFP1331.
http://wiki.edg.com/pub/CFP/WebHome/C2x_proposal_-_TS_18661-5abc-20190709.pdf
    Mike: Where it says evaluation to _Decimal64 is it greater than or equal to or only _Decimal64?
    Jim: It is about _Decimal{32,64} being _Decimal64 while larger is to the larger ones like _Decimal128 -> _Decimal128.
    Jim: Requires support for these evaluation methods. We could add this.
    Fred: Or just say required to have _Decimal32 evaluated as _Decimal64.
    Jim: But that doesn't say what _Decimal64 is evaluated to.
    Jim: Do we need to make it clearer that we are referring to particular evaluation methods?
     Fred/Mike: Yes.
    Ian: Does the result have to be rounded correctly to _Decimal32?
     Jim/Fred: No.
    Jim: This does not preclude other evaluation methods. They have to have at least this one, but can have others.
    AI: Jim: Add wording to the CFP1331 proposal make it clear this is for particular evaluation methods and submit to WG14.

  Jim: Draft a note to warn about CFP1315's rounding of negative constants issue. See Jim's CFP 1337.
    There is one correction in the last two sentences to have precision to be precision and range.
    Rajan: There could be other rounding modes so we should not list them as the only ways. Perhaps say "rounding modes such as ..." for each of the same result and different result cases. Also the part of exact results should be a separate note or a separate paragraph.
    Fred: We should also not assume correct rounding.

 Other issues
  Fred's WG 14 papers (WG14 email thread "N2380: printf of NaN()") - See Jim's CFP 1349
    *Jim: Send CFP 1349 to the WG14 reflector as the CFP position.

 Issues raised by Jens
  Approach is define prefixes and reserve names under those prefixes.
    Jim/Ian: The different styles with existing C18 names and these new ones with similar functionality is not good.
    WANT macros are not sufficient since the names could already be in the library independently of the user source WANT macro definition.
     Jim: How much is this a real problem? Is it not solved by the compiler and linker people?
    Jim: Is there a namespace issue with the operators and types for your library Mike?
     Mike: Not that I've come across
    Jim: Do language providers have application test suites that would fail if these identifiers were added to math.h?
     David H: It is hard to add 1700 identifiers.

Ian: Ideally you'd want some major customers involved as well.

Should we have a clear statement of the pros and cons of the WANT macros?

Fred: For GCC, I can specify a flag that works on the compiler and linker and automatically links in all the libraries for me.

Jim: Was there consideration of allowing includes with finer granularity.

Rajan: Yes, but no desire for it in WG14 when we discussed it in London.

Naming of correctly rounded math functions.

No real issue with using cr_, we only did it cr to match existing C standard text.

Fred: Prefer the _.

We can support this if Jens proposes it.

Obsolescing log1p - See Jim's CFP 1348

Ian: I got an email complaining about this as it would break their application and they don't have source code that can be recompiled.

Jim: It would be good to add a statement as to why there is a second name as a footnote.

Specifying more special cases for math functions, e.g., periodicity for half-revolution trig functions.

Perhaps as recommended practice.

Issues with +/-0, NaNs, etc. and other identities.

Fred: I would like having the identities being added and requiring sin(30 degrees) being exactly a half.

Jim: How would we come up with a list?

Ian: We could ask Robert Enenkel at IBM.

David H: I don't think we want to copy the entire IEEE standard into the C standard. C still caters to non-IEEE implementations.

Jim: All IEEE says is to correctly round, and if you can't do that, this doesn't help.

Keep on the agenda since this right now is QoI.

Putting the half-revolution trig functions into their own subclause.

No issue with this.

Range error may occur if nonzero x is too small for expm1.

Fred: We should address this.

Rajan: It is covered under the general statement about allowing other range errors, but we can make it clearer or consistent with the other exp*10* functions.

Result: Look into doing something for this to make it more consistent.

Fred: exp10m1 has finite while exp2m1 doesn't. It is only a problem for a positive large number, not just any large number.

Rajan: Related to DR40?

Range error may occur if nonzero x is too small for logp1 and log10p1.

Result: Look into doing something for this to make it more consistent.

*Fred: Write a paper to make the range error for small nonzero x consistent for expm1, logp1, log10p1's other base functions.

Action items from WG14 London meeting:

C FP: Give 18661 part 4a (not reduction functions) for inclusion into C2X (http://wiki.edg.com/pub/CFP/WebHome/n2401.pdf)

Done.

C FP: Put N2309 into TS 18661-4 and C2X (http://wiki.edg.com/pub/CFP/WebHome/n2401.pdf)

Done.

TS DR13: Move to C2X (C FP action item)
  Done.

TS DR16: Move to C2X (C FP action item).
  Done.

TS DR20-25: Move to C2X (C FP action item).
  Done.

CFP compendium - See Jim's CFP 1332
  Plan looks good.

Other items:
CFP 1341: FP_NORMAL.
  Besides normalized finite numbers, FP_NORMAL may have other kinds such as unnormals.
  Fred: DFP has a lot of finite numbers that are not normalized.
  Jim: There is a problem with the term normalized. I may have been better saying normal. Unnormals may resolve to sub-normals if it was attempted to be normalized. There can be unnormalized representations of normal numbers.
  David H: Maybe the footnote needs to say unnormal numbers can be FP_NORMAL or FP_SUBNORMAL depending on their value.
  Fred: Will go back to look at the test around normal numbers and come back here.
  *AI: Add in the normalized discussion from Fred to the agenda for next meeting.