

**WG14 N2075**  
**Meeting notes**

**C Floating Point Study Group Teleconference**

2016-08-30  
9 AM PDT / 12 PM EDT

**Attendees:** Rajan, Jim, David H., Fred, Mike, David Chen

**New agenda items:**

None.

**Last meeting action items:**

- Jim: Check one of the files from the EDG backup for testing the off site backup. - Not done.
- Jim: Writing up issue C email reflector message 14283 (evaluation macros) as a DR against C11. - Done (DR1 in the proposed DR's sent out).
- Jim: Add the footnote text as per email on 2016/07/26 to Draft DR 1 ("Is return of same type convertFormat or copy?"). - Done (Draft DR3).
- Jim: Part 2: Page 42: Put (significant) in front of 'number' in "integers (s, c, q), where n is the number of digits" - Done (no parenthesis).
- Rajan: Send out rules for C2X proposals to the group. - Done.

**New action items:**

- Jim: Update the proposal for Part 2 to make it more similar to Part 1's proposal.
- Jim: For all proposals: Change to "This proposal incorporates" as the starting.
- Rajan: Proposal 1: Update the spreadsheet of part 1 features based on Marius' note and send it out to the group for final review.
- Rajan: Proposal 1: Change the prior art text that has "ex." to "Example:" since Ex. could mean excluding.
- Rajan: Proposal 2: Add in prior art based off the spreadsheet of features.
- Jim: Proposal 4b: Mention underflow as well (alongside the existing overflow).
- Jim: Proposal 5a: Look at leaving out the types in the second paragraph.
- Jim: Proposal 5d: Title: alternate expression handling -> alternate exception handling
- Jim: Proposal 5d: Say "portable handling of exceptional cases".
- Jim: Proposal 5d: Simplify the abstract by removing the detail after "Some actions control".
- Jim: Proposal 5d: Say "The pragma allows the program to deal with exceptional cases without having to consider implementation details."
- Jim: DDR7: Integrate the changes proposed for the usual arithmetic conversions into a combined document to make it easier to read/understand.
- Jim: DDR?: Create a new DDR for the typo that Joseph just raised in reflector message 14358.
- Jim: Consult with Mike to discuss the preferred quantum exponent for hypot (in the TS), rsqrt, pow\* (in the TS), etc.

**Next Meeting:**

September 27th, 2016, 12:00 EDT, 9:00 PDT  
Same teleconference number.

Next WG14 meeting (Pittsburgh, 2016/10/17) has a September 19th mailing deadline.

**Discussion:**

IEEE 754 revision:

Had a long discussion on comparisons. Is it mandated to compare different formats but with the same radix?

Ex. How many signals do you get?

If the operations are required, can anything we (WG14) have proposed detect the difference?

C++ liaison:

Ian not present.

Part 5:

Published.

What should be proposed for the C standard (Jim's email sent on 2016/08/28):

All the parts are proposed in the TS as optional features other than Part 1.

Feature macros, want macros, etc.

Can add it to the proposals.

May need changing macros, etc. if proposals are merged or not.

We're currently not proposing them as optional other than the parts with the WANT macros.

Part 1 + Part 2 together or separate?

Mike: Part 1 and 2 should have the same gravity.

Most of part 1 is in Annex F which is an optional feature. We have made changes in the main part of the standard as well (like new functions).

May be easier to get part 1 approved than part 1 + 2 since part 1 is in the standard already.

Mike: May be better to have both or none rather than one and not the other.

These are the required parts of 754.

They should have been a single section.

Jim: The issue is not matching IEEE, it is the language standard which determines what fits for it.

The second proposal does list the tie in of both parts, but perhaps we need to sell decimal better in the proposal with perhaps references to Mike's site.

Mike: Remove the "in recognition" part since it should be part of the IEEE binding in general instead of being apologetic. It could say the same as part 1 that it updates C for decimal changing to 'add' instead of 'update' where appropriate.

Jim: There is no updating of decimal since it's not there in the C standard while binary is.

\*ToDo: Jim: Update the proposal for Part 2 to make it more similar to Part 1's proposal.

Prior art needs to be specified.

If anything does not get in, we can have references to the TS's themselves.

Ex. Extended types don't make it in, we can reference the TS in the standard.

Part 1 proposal:

Mike: Minor: "This proposal incorporates" as the starting.

David H: 1989 should be 1985 for IEC 60559.

Jim/Fred: 1989 is the correct year for it.

Jim: Don't want to add in the part about signalling NaN's but since it is sizable, had to add it.

\*ToDo: Rajan: Update the spreadsheet of features based on Marius' note and send it out to the group for final review.

\*ToDo: Rajan: Change the prior art from Ex. to Example: since Ex. could mean excluding.

Part 2 proposal:

No comments other than the expected wording changes to mirror part 1's proposal.

Part 4a proposal:

Fred: The "units of pi" seemed weird since shouldn't they be degrees?

Jim: No, it is for the sinpi type functions.

Perhaps say "half revolutions" in parenthesis as well?

Leave as is.

Should formatting for headers, cr, etc. be changed?

No, plain text is better for abstracts.

Part 4b proposal:

Fred: Do the functions avoid intermediate underflow?

David H: Yes they do.

\*ToDo: Jim: Mention underflow as well.

The implementation can do the platform specific work for performance while leaving the code (not results) portable.

Part 5a proposal:

Mike: Should the "The proposed specification requires binary ..." part be dropped since there is a lot of detail there.

Perhaps simplify it?

\*ToDo: Jim: Proposal 5a: Look at leaving out the types in the second paragraph.

Part 5b proposal:

No comments.

Part 5c proposal:

No comments.

Part 5d proposal:

Say portable handling of exceptional cases.

\*Mike: Simplify it by removing the detail after "Some actions control".

David H: The intention is to allow the programmer to look at the highest possible level instead of being concerned with implementation details.

David H: The pragma allows the program to deal with exceptional cases without having to consider implementation details.

David H: The prior art final sentence, should we say what we chose and why?

Jim: That might be a statement for all the proposals (Example: rounding direction) since that is the prior art in C for things.

David H: Should we have an overall introduction?

Jim: It was our goal to minimize the invention. Perhaps say that for the cover letter?

Email comments from Joseph Myers continued (<http://wiki.edg.com/pub/CFP/WebHome/DRs2-20160822.pdf>):

DDR1: (C)

It could have been read to apply to unary operators (widening them) which was not the intent.

Fred: Note that at some point WG14 will stop accepting DR's and then it would have to be a proposal.

Rajan: I thought they only close off DR's for standards that already have newer versions and just have a cut-off where all closed DR's make it in to the next draft of the standard while others don't.

Subject to email discussion before this meeting to add in "In all cases, assignment and cast remove all extra range and precision."

DDR2: (C) DECIMAL\_DIG

No comments:

DDR3:

No comments.

DDR4:

No comments.

DDR5:

No comments.

DDR6:

Removed the parenthesis from Fred's suggestion.

DDR7:

Fred: Does this cover \_Float32x being greater width than long double in the usual arithmetic conversions?

Jim: This only applies to when the formats are the same.

\*Jim: Can integrate it for the usual arithmetic conversions to make it easier to read/understand.

We should add that typo that Joseph just raised this morning.

Other:

Jim: Currently we have preferred quantum exponents for some of the math functions but not all (Example: The math functions in part 4). We should add it for them as well to be in sync with the new revision of 754.

David H: Doesn't happen that often, but for some functions like hypot it can. Mike is the expert but doesn't care so...

If 754 does add it then C should as well. Otherwise don't.

Jim: twoSum/twoProd: If they go in we can try to get them in for C2X as well if the math functions proposal get in, or into the TS if they don't.

They have two inputs and two outputs.

Jim: 'a/A' style formatting for decimal:

If the precision is shorter than required for the input then the conversion is not a true IEEE conversion due to the overflow case not being handled according to IEEE.

Through email, we favored the second option. Wording is in the 2016/08/29 email from Jim.

Perhaps we should keep the type and round to the precision and range of that type.

This would be closer to what is already specified, otherwise we don't really know the effects of the change.

Jim to send out something if he can come up with something.

Fred: Can say rounding with unbounded exponent (assuming just taking the words as is and tweaking).

Jim: Once you do that the rest of the spec says it is treated as if the precision is not specified but this doesn't apply since there is no typed format for the result.

Rajan Bhakta  
z/OS XL C/C++ Compiler Technical Architect  
ISO C Standards Representative for Canada  
C Compiler Development