WG14 N3552

Meeting notes

C Floating Point Study Group Teleconference

2025/04/16: 8 AM PDT / 11 AM DST / 3 PM UTC

Attendees

Jim Thomas, Rajan Bhakta, Jerome Coonen, Joshua Cranmer, David Hough, Fred Tydeman, Damian McGuckin

Updated agenda plus new items

https://wiki.edg.com/pub/CFP/WebHome/CFP%20meeting%20agenda-20250416-update.pdf – These minutes should be read alongside the agenda, with its many reference links.

Previous meeting notes

https://wiki.edg.com/pub/CFP/WebHome/n3527.pdf

Study group logistics

Next meeting: 14 May 2025, 8:00 AM PDT
ISO Zoom teleconference
Please notify the group if this time slot does not work.

C documents

The latest C2Y draft is N3467 Feb 2025. https://www.open-std.org/jtc1/sc22/wg14/www/docs/n3467.pdf
C23 has been published ISO/IEC 9899, available for purchase. https://www.iso.org/standard/82075.html

IEEE 754 liaison

David: Very slow progress.

Jerome: Much discussion of topics like the fate of the extended formats (think 1980s coprocessors) and roundTowardOdd.

C++ liaison

Joshua: Next meeting is in June, so a few quiet months coming.

WG14 update

None

TS-4 and TS-5 revisions

Jim: Published and available for purchase (see agenda link) for a bit over $100US.
Rajan: Will investigate copies for standards development within the CFP group.

Carry-over action items from last meeting

Damian: complex.h intro, cproj, and editorial suggestions.
Jim: Cproj and complex.h done. Keep editorial suggestions as a carryover.

Action items from last meeting

Jim: Draft a paper with suggestion for frexp and double double to address the WG14 objections.
Done

Rajan: Send a note to WG14 about the N3447 issue.
Rajan: Done and have follow-up later in the agenda.
Damian & Jim: Tune up the cproj proposal suitable for WG14.
Done

Ly: Send email to CFP about the next____() issues pertaining to the several functions inspired by nextafter().

Status unknown

Jerome and Rajan: There are perhaps two threads of next___ discussion on distinct issues, to be resolved when all input is received. Historically, 754-1985 perhaps over-cautiously signaled underflow and inexact in the subnormal domain, leading to the less noisy nextUp and nextDown functions that return their exact (in the number system) result. Aside from whether to raise exceptions, there is the cost, which Ly is expected to address further.

Discussion of issues:

Action item from WG 14 re. frexp and double-double

Jim: Proposes a footnote to clarify the meaning of "floating point number" for purposes of the possible unspecified resits. Takes new action to submit same.

UB for translation-time expression evaluation (N3447 issue)

Rajan: CFP 3410 has the issue thread. The subtle change is to, "...is evaluated with arithmetic range and precision at least as great..."

Jim: New wording avoids "shall" and hence any UB, and also emphasizes that the range and precision apply to the evaluation.

Rajan: Will send a note to WG14 with the group's approval. Wording is in 3413. Takes an action.

cproj cleanup

Damian: Proposal is ready to send.

Jim: Should say "return the value of the argument" vs. "return the argument".

Rajan: The extra parentheses noted in the expression "cimag((z))" protect macro substitution.

Fred: But the extra parens are not in the working draft, and seem not to be the convention.

Damian: Will bring into conformance with the current draft. Takes action to revise with corrections made in the discussion and submit.

TS-4 issue #1014 about error handling

Jim: TS-4 has functions outside of math .h that refer to domain errors and range errors. Such errors are specific to math.h, not to the general library functions, so a cross-reference is needed.

Rajan: This is a reasonable response. Suggest keeping a list of such proposals for a future TS revision. Until we can submit to the issue tracker, we can follow up in email.

Jim: Takes action to draft a response for a future revision.

complex.h introduction

Damian: Changes accommodate disappearance of the imaginary types.

Rajan: Agree that expanding the wording to say, "...its real part being positive or unsigned zero and it imaginary part being one," is longer but more readable.

Jim: Program elements in red strikethru need program font for clarity (even though they are being removed!). The documents N3460 etc. should be in the references. The explanation of what's wrong in #4 needs to be expanded.

Rajan: Be sure to return a "value" not a "type". Also, the rationale should come first.

Damian (post-meeting): Note that the language of 7.3.1 and 7.12.1 need to be consistent re. the use of value vs. type.

Jerome & Raan violently agree that there is an argument for leading with the punchline (the changes) and following with the rationale, but WG14 prefers that we lead with the rationale.

Jim: If no objection, we propose submitting with the changes proposed in the discussion. Damian takes an action.

powr(negative, qNaN)

Jim: Issue has been raised. Joseph Meyers (whose interpretation of 754 and C23 is correct) suggests an editorial change to bring the language into agreement with at least 3 similar cases of "all y (including NaN)", emphasizing that NaN is included.

Jerome: Takes action to write a proposal with the change given in CFP 3415.

Imaginary I macro and i suffix issues, N3390 update

Jim: Have not seen an update on the removal of the I macro.

Rajan: The work is ongoing in WG14, so look for news to come.

Sign bit of a NaN

Damian: Notice a difference between compilers in the sign of the result of 0/0 or (x-x)/(x-x).

Jim: The sign of a NaN is not specified.

Joshua: There is inconsistency among hardware regarding the sign of NaN. Some compilers prefer positive sign.

Joshua: If the input is unspecified (sign bit of NaN), the output will be.

Jim: No action.

Constant expressions and floating-point exceptions

Jim: Vincent raised this issue, noting that status flags cannot be raised at translation time. So the evaluation cannot be the same in the two contexts.

Fred: The standard also specifies that the flags are cleared at the start of execution.

Jim: There may also be extra range and precision at translation time.

Joshua and Rajan: Some compilers attempt to provide software emulation of the target arithmetic. Even hexadecimal.

Jim: Will carry over for further discussion.

nexttoward exceptions

Jim: Awaiting email input from Ly.

Removal of “(also known as a singularity or infinitary)” in post-N3467 draft

Jim: Takes an action to keep this on the agenda until the removal is confirmed.

More complex functions, e.g. ccbrt, csinpi

Jim: Issue raised by Ly. Awaiting email about it.

CFP list of post C23 issues

Jim: Great that we're getting issues for C2Y, but need to return to the list, Actions to follow.

Range error conditions for atan2 and atan2pi

Carry over

[[reproducible]] example in 6.7.13.8 ignores exceptions

Carry over

Other issues:

None

Adjournment

10:04 AM PDT

Action items to be carried over:

Damian: Update the proposal re. the wording of cimag() and creal(). (These are just some of the editorial suggestions.)

New action items:

Jim: Submit updated proposal re. frexp clarification.

Rajan: Propagate 3413 with wording just approved up to WG14.

Damian: Submit the cproj proposal with minor amendments discussed. Jim to get a document number and help submit.

Jim: Submit email about TS response re. domain and range errors. Rajan to work with issue tracker when possible.

Jim & Damian: Jim to send to CFP an email with suggested wording changes for complex.h. Damian to update the proposal based on these and other suggestions. Jim to get a document number for Damian, then submit the updated proposal.

Jerome: Write a proposal for powr() addressing Joseph's suggestion.

Damian: Will check the pwr() language against his ongoing changes.

Jim: Keep an agenda item re. the removal of “(also known as a singularity or infinitary)” in post-N3467 draft of C2Y.

Discussion issues to be carried over:

Constant expressions and floating-point exceptions

More complex functions, e.g. ccbrt, csinpi

Range error conditions for atan2 and atan2pi

[[reproducible]] example in 6.7.13.8 ignores exceptions

Respectfully submitted.

-Jerome Coonen
650.996.4738
jcoonen@gmail.com