WG14 N2343
Meeting notes

C Floating Point Study Group Teleconference

2019-02-20
8 AM PST / 11 PM EST / 4 PM UTC

Attendees: Rajan, Ian, Jim, 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.
Seems to be incompatible in the older C++ standard. Ex. C++ did not allow evaluation in a greater precision than the type.
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. - Done.
David: Check the augmented* C function specifications to ensure they match what IEEE has. - 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.

Last meeting action items:
Jim: Post the draft of Part 1 integrated into C2x on the CFP site. - Done.
Jim: Make the first change as per CFP1238. - Done.
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. - Done.
Jim: Make the _EPSILON change as per CFP1238. - Done.
Jim: Make the forth change as per CFP1238 to add in the hyperbolic versions of cos/sin/tan. - Done.
Jim: Make the fifth change as per CFP1238 for the list changes. - Done.
All: Take a look at part 3 annex (http://wiki.edg.com/pub/CFP/WebHome/cfp3x-annex-20190119.pdf) draft before next meeting. - Done.
Jim: Update the proposal to try again to integrate part 4a into C2X. - Done.
David: Check with IEEE group to see if there is any implementations for Part 4b functions (hardware or software). - Done.
All: Review the rationale for part 5 a, b, c proposal. - Carry over.
All: Review http://wiki.edg.com/pub/CFP/WebHome/update_for_C2X_payload_functions.pdf. - Done.
Rajan: Say to WG14 that CFP supports removing the WANT macros and leaving the rest as is due to Fred's reasoning. - Carry over. Note: Also mention CFP's appreciation to Jens for the integration editing (done really well).
Rajan: Check with David Keaton to see if we are going to discuss the WANT macro issue on the agenda. - Done. It is.
Fred: Send links to papers of interest to CFP from N2319-N2323. - Done.

New action items:

Jim: Put a list of new items that need to be done like the binding table and the totalorder* carry over action items as well as adding in the min/max and augmented arithmetic to a new part 4 revision.

Jim: Look at the part 3 X.2.6 change where the removal of the float _Imaginary, etc. part of the list may imply those are not valid types anymore.

Jim: Get a document number to submit the Part 3 as an Annex proposal to WG14 for the mailing.

Jim: Correct typo in page 1 of the proposal for part 4a ("As shown in the table below, C already supports 22 *of* the 39...")

David H: Take a closer look at the new proposal for part 4a (http://wiki.edg.com/pub/CFP/WebHome/C2x_proposal_-_TS_18661-4a-2-20190213.pdf) to ensure it is good for submission via the reflector so it can be submitted by Jim to WG14.

All:
Review http://wiki.edg.com/pub/CFP/WebHome/update_for_C2X_payload_functions.pdfwithin a week. If no changes, Jim will get a document number and send this into WG14.

Jim: Look to see if there is a place to put a note to address the alternate exception handling in annex F to clarify how C specifies using default exception handling only.

Jim: Get a document number to submit the Part 2 (http://wiki.edg.com/pub/CFP/WebHome/cfp2x-C2X-20190215.pdf) to WG14 for the mailing.

Fred: Create papers for the SNAN initialization and unary + operation as CFP papers (CFP 1249, 1253, 1247, 1250) for future submission to WG14.

All: Consider why we didn't have wide string from functions and if we should do them.

Next Meeting(s):

Monday, March 18th, 2019, 11:00 EDT, 8:00 PDT, 3PM UTC
Same teleconference number.

Discussion:

754 revision:

Sponsor ballot deadline at midnight today. Need to address 68 comments (most accepted). Another draft (recirculation) will be issued. An asynchronous public review which ends in a month with a current comment count of 5. Not looking at major changes right now. Got enough Yes votes already.

C++ Liaison:

None.

C2X integration:

WG14 meeting (April 29th-May 3rd) 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:

Integrated into a working C2X draft. Posted on the wiki.

Part 2:

A draft is posted on the wiki for review.
Mike: The preferred quantum exponents table was really nice.

Part 3:

A draft as an annex is ready for review today.

Part 4ab:

4a has an updated proposal.
For 4b, we should leave as a TS and add in the augmented arithmetic.

Part 5abcd:
Leave to after the Spring 2019 WG14 meeting.

Action item details:
Jim: Post the draft of Part 1 integrated into C2x on the CFP site.
(http://wiki.edg.com/pub/CFP/WebHome/all-diffC2x-20190211.pdf,
http://wiki.edg.com/pub/CFP/WebHome/all-20190211.pdf)
No comments

Regarding Part 3 as annex (http://wiki.edg.com/pub/CFP/WebHome/cfp3x-annex-20190205.pdf):
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: Some references may need to be updated due to the moving nature of the C standard.
Looks good to submit to WG14.

Jim: Update the proposal to try again to integrate part 4a into C2X.
http://wiki.edg.com/pub/CFP/WebHome/C2x_proposal_-_TS_18661-4a-2-20190213.pdf
Rajan: Typo on first page (or the -> of the).

David: Check with IEEE group to see if there is any implementations for Part 4b functions (hardware or software).
David: Don't see any C or Fortran implementations.
Keep as a TS.

All: Review the rationale for part 5 a, b, c proposal.
Wait until after current WG14 meeting.

All: Review http://wiki.edg.com/pub/CFP/WebHome/update_for_C2X_payload_functions.pdf.
Look to close this one off via email.

Rajan: Say to WG14 that CFP supports removing the WANT macros and leaving the rest as is due to Fred's reasoning above.
See Rajan and Fred's email CFP 1242, 1243, 1244. Also Jim's CFP 1261.
See WG14 email 15982 – 15993, 15995.
Related:
Jens idea to rename many of the math functions
Jens idea to "add a version macro (__STDC_MATH_VERSION__ ...) for the clauses where we add interfaces that are not reserved, yet"

Jim: Versioning is not really a floating point issue, it is more of a C standard issue.
Jens seemed to want the WANT macros for the Annex F interfaces. We are OK with keeping it.
Not in favor of any name changes.

Fred: Send links to papers of interest to CFP from N2319-N2323. See Fred's email CFP 1246.
Discuss when Fred is present.

David: min/max review/comparison to IEEE:
Did not see anything that was inconsistent.

David: augmented arithmetic review/comparison to IEEE:
Seemed specified for the default exception handling. Do we want to make the distinction about overflow/underflow and inexact with the exception handling in the C document?
Jim: C doesn't have alternate exception handling so I don't think so.
*Jim: Look to see if there is a place to put a note to address the alternate exception handling in annex F.

Other issues:
Draft for C2X integration posted to wiki for review (http://wiki.edg.com/pub/CFP/WebHome/cfp2x-C2X-20190215.pdf)
Mike: Looks good! Especially the preferred quantum exponents table.
Jim: Note that removing the WANT macros will remove the header lists in the specs as well.

Unary operators. See Fred's email CFP 1249 and 1253.
Changes look to be good.

Initialization with signaling NaN macros. See Fred and Jim's email CFP 1247, 1250.
Changes look to be good.

Edge cases for fegetexceptflag. See Fred and Jim's email CFP 1256, 1260.
Seems good as it is already. No need to change.

Why no wide character string-from functions?
Consider why we didn't have these already and if we should do them.
Ian: Probably because you could always convert to regular strings then convert the regular string to the wide character string.

Why no tgmath macros for functions with pointer parameters? Question from Jens.
Jim: modf was the one exclusion originally due to the pointer.
Rajan: The reason was the specification for tgmath rules was based on float, double, long double and so we just extended it to handle the new types we added. The specification is complicated enough as is and adding in pointer specification would make it much more complicated.
Rajan: The meta question is do we need to do this? If people want to add pointer argument based functions to the tg list then we can pursue this.
Leave it as is for now.

Regards,

Rajan Bhakta
z/OS XL C/C++ Compiler Technical Architect
ISO C Standards Representative for Canada
C Compiler Development
Contact: rbhakta@us.ibm.com, Rajan Bhakta/Houston/IBM