

## **WG 14 N1837**

### **WG14 CFP meeting minutes for the meeting of 2014/03/18**

2014/04/15, 9:00 AM PDT, 12:00 PM EDT:

**Attendees:** Jim, Rajan, David, Ian

#### **New agenda items:**

None.

#### **Old action items:**

Jim: Part 4: Page 6: Talk to Fred to see if he is OK with the errors - Done

Rajan: Ask WG14 why pole errors are specified the way they are in the C meeting - Done

Everyone: Send Rajan available times for April 7-11th and contact information - Done

David: Making the first draft of part 5 - Done, keep open

Jim: Send David the C floating point pragma scope/effects - Done

#### **Next Meeting:**

May 13th, 2014, 12:00 EST, 9:00 PDT

Same teleconference number.

#### **New action items:**

Jim: Work with John and David Keaton to get the parts 2, 3 and 4 issued for ballot.

Jim: Part 4: Divide it up and assign (or open to assign) reviewers to parts of the document.

Jim: Part 4: Give the document to Joseph and ask if he has any issues with it.

David: Part 5: Complete exception specification with the full syntax dealing with scope and sub-exceptions. Include a discussion document with reasons choices and alternatives.

#### **Discussion:**

WG14 meeting summary:

Pole errors: Not required to support them (optional feature).

Part 1 sent out for publish.

No comments on part 2. Ready for DTS ballot.

No comments on part 3.

WG14 is good with it going out for PDTS ballot without requiring a 90 day review.

Comments on part 4:

The new tanpi, asinpi, acospi should not be a part of the spec and if IEEE does not specify it then we should not either.

Blaine: Call out the missing functions as additions but not part of the IEEE spec

John Benito: Don't add the functions if not required by IEEE

Missing functions straw poll: Should we keep the new functions added (tanpi, asinpi, acospi).

For/Against/Abstain: 11/4/0

WG14 is good with it going out for PDTS ballot without requiring a 90 day review.

Comments on part 5:

Question from committee: Are we still having a part 5?

Answer given: Yes. We will. We are working on it now.

DR441: n1730/n1804: Floating point issues in C11: Weak agree. No Forward p8. Change proposed committee response to F.2 instead of F.3 (second last bullet). Leave in open.

DR442: n1730/n1804: Floating point issues in C11: Weak agree. The larger issue given in n1804 is handled by DR441. i.e. Annex F takes precedence and there is no issue. Leave in open.

DR443: n1730/n1804: Floating point issues in C11: What is the effect of this? Not a defect. Leave in open with this new response.

Conversion from GROFF to LaTeX for the C standard source is under discussion as being open for bidding to do the conversion.

Jim: Cut'n'paste to Word of C11 and has put in part 1. Working on part 2 now. Problems with footnotes and tables in general.

An issue with a footnote indicator disappearing but the footnote staying present.

For part 2 there is a small editorial change needed.

Would like a few days to put in Part 3 to ensure that there is not anything missing.

Bullets get out of order sometimes.

Part 1: Waiting for publication.

No comments.

Sent to ITTF for publication.

Part 2: Will be out for second ISO ballot (DTS).

One comment on PDTS has been handled.

Part 3: (<http://wiki.edg.com/twiki/pub/CFP/WebHome/n1796.pdf>)

PDTS ballot expected to be issued.

Jim needs a few days to check his merge into C11 to make sure everything is good.

Part 4: (<http://wiki.edg.com/twiki/pub/CFP/WebHome/n1797.pdf>)

PDTS ballot expected to be issued.

We should look at it before submitting it for PDTS.

Was posted in the IEEE reflector.

\*Jim: Part 4: Divide it up and assign (or open to assign) reviewers to parts of the document.

\*Jim: Part 4: Give the document to Joseph and ask if he has any issues with it.

Part 5: (Email from David sent 2014/04/14)

Using a pragma to affect a block is a good idea since it follows what we've already done.

Note: We could generalize the structured block to allow exits in between (not only at the end).

Block: Compound statements is what is described in C and can be used.

The other pragmas can be file scope and block scope. We should allow that for these as well.

Ian: We might want to see how this would be handled in C++ as well in a way compatible with C.

C++ is out of scope, so we will not consider it now.

Does underflow belong in this list?

It is an exception.

Note: abrupt only applies to underflow so the parameters are not orthogonal.

We need controls for value changing optimizations (like FP\_CONTRACT).

Difference/tension between programmer intent and debugging intents.

#pragma syntax:

Change "fpe" to "FENV\_EXCEPT".

Change ";" to ",".

Inlining must not effect the specified behaviour. It should work the same way whether inlined or not.

Library functions:

Do the same as FENV\_ROUND pragma: List the functions that are effected if there is no macro suppression.

Since we implement some of the IEEE operations as library functions we have to do the same here.

Without dynamic modes it is hard to do.

Large number of combinations here (different from the rounding modes).

Dynamic traps are available at the system level and the compiler deals with it.

Link time inlining may be an issue for some implementations.

Immediate and delayed: ASAP instead of Immediate?

Could have abort\_asap and abort\_delayed with defaults perhaps.

Computed goto, missing label issues.

Can use a pragma to define a label in a different namespace than normal labels.

Can also make the missing labels undefined.

Note scope issues apply to which label type we choose.

Use the C labels for now and make a note that you can have a separate namespace.

Jim: Can we use function names instead?

David: Where would it return to? Don't want to reinvent setjmp/longjmp.

Ignore functions for now. IEEE talks about goto and abort but not functions.

We could have a \_Noreturn function.

\*David: Part 5: Complete spec with the full syntax dealing with scope and sub-exceptions. Include a discussion document with reasons choices and alternatives.

Resuming alternate exception handling should fit in as well.

Optimization stuff should fit in the same framework.

David: Preferred width: Comes under the category of optimizations.

Jim: Wasn't intended to. It should be totally predictable.

David will handle exceptions now, and get to the other ones later.

Which parts will be optional for part 5 (beyond the entire part being optional)?

Regards,

Rajan Bhakta