

Document: WG14/N1293
Date: 2008/03/14
Project: WG14 TR 24732
References: WG14/N1241
Authors: Rich Peterson, Jim Thomas
Reply to: Rich Peterson <Rich.Peterson@hp.com>

Subject: C99 consistency changes in TR 24732

Problem:

Some details in the TR that to be corrected for consistency with C99 and/or similar sections in the TR.

Suggested changes to the TR:

1. Document page 20 (PDF 23), at the bottom. [C99 consistency - C99 says "positive or unsigned", but 754R does not have unsigned infinity.]
 Change:
 DEC_INFINITY
 expands to a constant expression of type `_Decimal32` representing infinity.
 to:
 DEC_INFINITY
 expands to a constant expression of type `_Decimal32` representing positive infinity.
2. Document page 21 (PDF 24), at the top [needs to say constant expression, self-consistency]:
 Change:
 DEC_NAN
 expands to quiet decimal floating NaN for the type `_Decimal32`.
 to:
 DEC_NAN
 expands to a constant expression of type `_Decimal32` representing a quiet NaN.
3. Document page 23 (PDF 26), [typo]:
 Change:
 modd64
 to:
 modfd64
4. Document page 35 (PDF 38), at the bottom, [important clarification]:
 Change:
 When one of the arguments is a decimal floating type,
 to:
 When one of the type-generic arguments has a decimal floating type,
5. Document page 36 (PDF 39), at the top [C99 consistency] This is needed for consistency with C99 type-generic rules (C99 7.22), which say "if any argument for generic parameters has type double or is of integer type, the type determined is double". These C99 words guarantee that if any argument for a generic parameter has an integer type, the type determined for the function is at least as wide as double. I.e. if all arguments have integer types, or if there is a mix of integer and type float, the type determined is double. The only

way that the type determined can be float is if all arguments are of type float. The existing words determine type `_Decimal32` for a mixture of `_Decimal32` and integer types. The new words determine type `_Decimal32` only when all arguments have type `_Decimal32`, matching the C99 behavior for type float.

Change second bullet:

- Otherwise, if any argument has type `_Decimal64`, the type determined is `_Decimal64`.
- to:
- Otherwise, if any argument has type `_Decimal64`, or if one argument has an integer type and another argument has type `_Decimal32`, the type determined is `_Decimal64`.