

**Proposal for C23  
WG14 N2650**

**Title:** signbit cleanup  
**Author, affiliation:** C FP group  
**Date:** 2021-01-22  
**Proposal category:** Technical  
**Reference:** N2596

The current specification for the **signbit** macro (7.12.3.7) doesn't cover cases where the argument value is unsigned, except for unsigned zeros which are mentioned in a (non-normative) footnote. The suggested changes below move the unsigned zero case from the footnote to the Description, and say that other unsigned cases are implementation defined. Though moving the specification for unsigned zero is technically a substantive change, we don't expect it to affect any implementations.

Current Annex F doesn't explicitly state the IEC 60559 requirement for bit-level determination of the sign. The following suggested changes add this specification, which is not a substantive change since Annex F adopts IEC 60559 by reference.

Also included below is an update regarding **signbit**, to sync with current IEC 60559.

**Suggested changes:**

Changes for 7.12.3.7:

**Description**

The **signbit** macro determines whether the sign of its argument value is negative.<sup>253</sup>) **If the argument value is an unsigned zero, its sign is regarded as positive. Otherwise, if the argument value is unsigned, the result value (zero or non-zero) is implementation defined.**

**Returns**

The **signbit** macro returns a nonzero value if and only if the sign of its argument value is **determined to be** negative.

<sup>253</sup>)The **signbit** macro ~~reports~~**determines** the sign of all values, including infinities, zeros, and NaNs. ~~If zero is unsigned, it is treated as positive.~~

In F.3, after #6, add the paragraph:

[6a] The **signbit** macro, providing the IEC 60559 isSignBit operation, determines the sign of its argument value as the sign bit of the value's representation. This applies to all values, including NaNs whose sign bit is not generally interpreted by IEC 60559.

Change F.3 #6:

The C classification macros fpclassify, iscanonical, isfinite, isinf, isnan, isnormal, issignaling, issubnormal, ~~and~~ **iszero**, **and** **signbit** provide the IEC 60559 operations indicated in the table above provided their arguments are in the format of their semantic type.