

IEEE Standards Interpretation for IEEE Std 1003.1™-1990 IEEE Standard for Information Technology--Portable Operating System Interfaces (POSIX®)

Copyright © 2001 by the Institute of Electrical and Electronics Engineers, Inc. 3 Park Avenue New York, New York 10016-5997 USA All Rights Reserved.

Interpretations are issued to explain and clarify the intent of a standard and do not constitute an alteration to the original standard. In addition, interpretations are not intended to supply consulting information. Permission is hereby granted to download and print one copy of this document. Individuals seeking permission to reproduce and/or distribute this document in its entirety or portions of this document must contact the IEEE Standards Department for the appropriate license. Use of the information contained in this document is at your own risk.

IEEE Standards Department Copyrights and Permissions 445 Hoes Lane, Piscataway, New Jersey 08855-1331, USA

Interpretation Request #110

Topic: raise vs asynchronous signal **Relevant Sections:** 2.2.2.10 , p14 I 120 PASC

Is the result of raise () (or kill() of self) a synchronous or asynchronous signal?

This is to bring work in scope for the Austin Group revision. The problem is that synchronous and asynchronous signals are not quite well defined, particularly in the case of raise(). The current wording is "a signal that is not attributable to a specific thread". Since all signals are generated by SOME thread, they are all attributable to a specific thread (just not necessarily the one handling the signal). I believe the thrust of it is "attributable to the thread handling the signal". Right now, "synchronously generated signal" is being treated as a synonym for "execution fault signal", which it isn't, at least according to the definition of the word "synchronously", because raise() would also be synchronously generated. Suggested alternative wording: Asynchronously generated signal: "A signal that is not attributable to the actions of the thread handling the signal." and Synchronously generated signal: "A signal that is attributable to the thread which is handling it, and which is handled immediately upon its generation; that is, is not blocked or ignored. Note: timer and I/O completion signals are considered attributable to the timer or I/O system." Note... there are two intertwined issues here; please address both of them: 1) "attributable to a process" is trivially true; it's "the same thread" that's critical to fixing this. 2) "handled immediately". This certainly is the intent, but it could be circumvented. Doing a raise() of a signal known to be blocked is not an unreasonable programming technique, but it wouldn't be a synchronous signal. On the other hand, doing a raise() of an unblocked signal is synchronous with the program. (Or... rename the concept to "error signal" or something like that.) --

Interpretation Response

The answer to the question in the problem statement is "synchronous". The problem

raised in the “solution” section is being referred to the sponsor.

Rationale for Interpretation

New definitions are required, those proposed will not work, this is referred to the sponsor. Forwarded to Interpretations group: 19 June 2000 Proposed resolution: 25 July 2000 Finalized Interpretation: 29 August 2000