

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 #3

Topic: EIO missing from tc* functions **Relevant Sections:** 7.1.1.4

POSIX 1003.1-1990 section 7.1.1.4 (page 131, lines 88-90) states:

“Certain calls that set terminal parameters are treated in the same fashion as write, except that TOSTOP is ignored; that is, the effect is identical to that of terminal writes when TOSTOP is set.”

In the preceding paragraph the final sentence says: “.... If TOSTOP is set, and the process group of the writing process is orphaned, and the writing process is not ignored or blocking SIGTTOU, the write() returns -1 with errno set to [EIO], and no signal is sent.”

Routines that fit this description are:

tcdrain(), page 145, section 7.2.2
tcflow(), page 145, section 7.2.2
tcflush(), page 145, section 7.2.2
tcsendbreak(), page 145, section 7.2.2
tcsetattr(), page 143, section 7.2.1

Why is [EIO] not included in the error section of any of these interfaces? Must they return -1 with errno set to [EIO] in the given circumstances? May they?

Interpretation Response

The requirement in section 7.1.1.4 “Terminal Access Control” lines 88-90 refers to “calls that set terminal parameters”. The list of functions for which this requirement is valid is limited to tcsetattr(). For the case of tcsetattr(), the implementation must return [EIO]

in the circumstances described as a consequence of lines 88-90 and the description of [EIO] on page 121 lines 280-285.

Rationale for Interpretation

Lines 84-87 of the same section refers only to the `write()` system call, and not explicitly to the `tcdrain()`, `tcflow()`, `tcflush()`, `tcsendbreak()` functions. Although they appear to be implementable in terms of `write()` this is not required by the standard. Were this to be required, language similar to that in section 8.2.3 would be included, i.e. "the underlying function is...". Therefore they are not required to return [EIO] by the current standard. They may however return [EIO] under the conditions laid out in section 2.4, lines 521 to 526.