

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 #67

Topic: `tcflow()` **Relevant Sections:** 7.2.2.2 **Classification:** Duplicate of PASC 2003.1 #17

The `tcflow()` function shall suspend transmission or reception of data on the object referred to by `fildev`, depending on the value of `action`:

- (1) If `action` is `TCOOFF`, it shall suspend output.
- (2) If `action` is `TCOON`, it shall restart suspended output.
- (3) If `action` is `TCIOFF`, the system shall transmit a STOP character, which is intended to cause the terminal device to stop transmitting data to the system. (See the description of `IXOFF` in 7.1.2.2.)
- (4) If `action` is `TCION`, the system shall transmit a START character, which is intended to cause the terminal device to start transmitting data to the system. (See the description of `IXOFF` in 7.1.2.2.)

The problem found is items (3) and (4) could be interpreted two ways.

Interpretation One: (3) If `action` is `TCIOFF`, WHETHER OUTPUT IS SUSPENDED OR NOT, the system shall transmit a STOP character, which is intended to cause the terminal device to stop transmitting data to the system. (See the description of `IXOFF` in 7.1.2.2.) (4) If `action` is `TCION`, WHETHER OUTPUT IS SUSPENDED OR NOT, the system shall transmit a START character, which is intended to cause the terminal device to start transmitting data to the system. (See the description of `IXOFF` in 7.1.2.2.)

Interpretation Two: (3) If `action` is `TCIOFF`, AND OUTPUT IS NOT SUSPENDED, the system shall transmit a STOP character, which is intended to cause the terminal device to stop transmitting data to the system. (See the description of `IXOFF` in 7.1.2.2.) (4) If `action` is `TCION`, AND OUTPUT IS NOT SUSPENDED, the system shall transmit a START

character, which is intended to cause the terminal device to start transmitting data to the system. (See the description of IXOFF in 7.1.2.2.) The IEEE Std 2003.1-1992 section 7.2.2.3.2 test assertions [07,08] assume "Interpretation One."

This interpretation expects a tty device which has had its output suspended by the action of TCOOFF, item (1), to transmit the STOP/START character so that it is made available to a read() on another tty device (even though output is suspended). If "Interpretation Two" was assumed, a tty device which has had its output suspended by the action of TCOOFF, item (1), would accept but not transmit the STOP/START character until suspended output had been restarted by the action of TCOON, item (2). Only then would it be transmitted and made available to a read() on another tty device. The POSIX Std 1003.1-1990 does not clearly state the behavior of items (3) and (4) when item (1) is in effect. Which interpretation is intended by POSIX IEEE Std 1003.1-1990?

Interpretation Response

This is a duplicate of 2003.1 #17, see attachment below:

PASC Interpretation reference 2003.1-92 #17 Classification: No change This response will be incorporated in an IEEE interpretations publication, and will be also made available online

Interpretation Number: (to be assigned by the IEEE)

Topic: tcflow **Relevant Sections:** 7.2.2.3.2

The IEEE Std 2003.1-1993 section 7.2.2.3.2 has the following assertions. 07 A call to tcflow(fildes,TCIOFF) causes the system to transmit a STOP character, and the return value is zero. Testing Requirement(s): Test when the data transmission on the line is suspended and is not suspended. 08 A call to tcflow(fildes,TCION) causes the system to transmit a START character to restart suspended input, and the return value is zero. Testing Requirement(s): Test when the data transmission on the line is suspended and is not suspended. The problem lies in the testing requirements, for sending a STOP character when the line is already suspended, and for sending a START character when the line is not suspended. The testing requirement makes additional implementation restrictions beyond that specified in IEEE Std POSIX 1003.1:1990. We would request that these assertions be reworded to: 07 A call to tcflow(fildes,TCIOFF) causes the system to transmit a STOP character, and the return value is zero. Testing Requirement(s): Test when the data transmission on the line is not suspended. 08 A call to tcflow(fildes,TCION) causes the system to transmit a START character to restart suspended input, and the return value is zero. Testing Requirement(s): Test when the data transmission on the line is suspended.

Interpretation Response 2

The testing requirements are correct as they are now written. They refer to suspension of output on the line. IEEE Std 1003.1-1990 provides for programmatic flow control on terminals that use asynchronous serial data transmission through the tcflow() inter-

face (7.2.2.2: page 146, lines 697-706). The specifications for output control (requests TCOOFF and TCOON) define a persistent state of “suspended output”, such that a call to `tcflow(fildes, TCOOFF)` causes output to be suspended and the output stays suspended until a call is made to `tcflow(fildes, TCOON)`. The specifications for input flow control say simply that a STOP character or a START character be sent for `tcflow(fildes, TCIOFF)` and `tcflow(fildes, TCION)`, respectively. These STOP and START characters are intended to be sent to the terminal at the remote end of the line. This means that STOP or START must be transmitted whether or not output is suspended. The text of the testing requirements would be easier to understand if they referred specifically to suspension of output.

Rationale for Interpretation

The output to a terminal is produced by processes on the local system. Therefore, `tcflow()` can control suspension or resumption of output unconditionally. The case is different for input, which is generated by a remote device that is ordinarily not under the direct control of the system. For input, `tcflow()` sends the STOP and START characters to request that the remote device suspend or resume transmission. Note that in the descriptions of the four possible actions for `tcflow()` in IEEE Std 1003.1-1990 697-706) neither the TCION nor the TCIOFF action is conditional on whether output is suspended. This means that the START and STOP characters are treated as special characters, and are not considered to be output. There is also a definite advantage to users in requiring unconditional sending of START and STOP because this is what makes it possible for an application to regain control of a terminal connection that has become confused because of flow control problems.