IEEE P1003.2 Draft 11.2 – September 1991

Copyright © 1991 by the
Institute of Electrical and Electronics Engineers, Inc.
345 East 47th Street
New York, NY 10017, USA
All rights reserved as an unpublished work.

This is an unapproved and unpublished IEEE Standards Draft, subject to change. The publication, distribution, or copying of this draft, as well as all derivative works based on this draft, is expressly prohibited except as set forth below.

Permission is hereby granted for IEEE Standards Committee participants to reproduce this document for purposes of IEEE standardization activities only, and subject to the restrictions contained herein.

Permission is hereby also granted for member bodies and technical committees of ISO and IEC to reproduce this document for purposes of developing a national position, subject to the restrictions contained herein.

Permission is hereby also granted to the preceding entities to make limited copies of this document in an electronic form only for the stated activities.

The following restrictions apply to reproducing or transmitting the document in any form: 1) all copies or portions thereof must identify the document's IEEE project number and draft number, and must be accompanied by this entire notice in a prominent location; 2) no portion of this document may be redistributed in any modified or abridged form without the prior approval of the IEEE Standards Department.

Other entities seeking permission to reproduce this document, or any portion thereof, for standardization or other activities, must contact the IEEE Standards Department for the appropriate license.

Use of information contained in this unapproved draft is at your own risk.

IEEE Standards Department
Copyright and Permissions
445 Hoes Lane, P.O. Box 1331
Piscataway, NJ 08855-1331, USA
+1 (908) 562-3800
+1 (908) 562-1571 [FAX]

unspecified if these options are specified after a –k option. This will allow implementations to make the options independent of each other when the obsolescent forms are finally dropped (if that ever happens).

Historical documentation indicates that ‘‘setting –n implies –b.’’ The description of –n already states that optional leading <blank>s are tolerated in doing the comparison. If –b is enabled, rather than implied, by –n, this has unusual side effects. When a character offset is used into a column of numbers (e.g., to sort mod 100), that offset will be measured relative to the most significant digit, not to the column. Based upon a recommendation of the author of the original sort utility, the –b implication has been omitted from POSIX.2 and an application wishing to achieve the previously mentioned side effects will have to manually code the –b flag.

4.59 stty — Set the options for a terminal

4.59.1 Synopsis

```
stty [ –a| –g ]
stty operands
```

4.59.2 Description

The stty utility shall set or report on terminal I/O characteristics for the device that is its standard input. Without options or operands specified, it shall report the settings of certain characteristics, usually those that differ from implementation-defined defaults. Otherwise, it shall modify the terminal state according to the specified operands. Detailed information about the modes listed in the first five groups below are described in POSIX.1 {8} Section 7. Operands in the Combination Modes group (see 4.59.4.6) shall be implemented using operands in the previous groups. Some combinations of operands are mutually exclusive on some terminal types; the results of using such combinations are unspecified.

Typical implementations of this utility require a communications line configured to use a POSIX.1 {8} *termios* interface. On systems where none of these lines are available, and on lines not currently configured to support the POSIX.1 {8} termios interface, some of the operands need not affect terminal characteristics.

4.59.3 Options

The stty utility shall conform to the utility argument syntax guidelines described in 2.10.2.

The following options shall be supported by the implementation:

Copyright © 1991 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

12146 `-a` Write to standard output all the current settings for the terminal.

12147 `-g` Write to standard output all the current settings in an
12148 unspecified form that can be used as arguments to another invo-
12149 cation of the `stty` utility on the same system. The form used
12150 shall not contain any characters that would require quoting to
12151 avoid word expansion by the shell; see 3.6.

12152 ### 4.59.4 Operands

12153 The following operands shall be supported by the implementation to set the termi-
12154 nal characteristics:

12155 #### 4.59.4.1 Control Modes

12156 `parenb (-parenb)` Enable (disable) parity generation and detection.
12157 This shall have the effect of setting (not setting)
12158 PARENB in the *termios c_cflag* field, as defined in
12159 POSIX .1 {8}.

12160 `parodd (-parodd)` Select odd (even) parity. This shall have the effect of
12161 setting (not setting) PARODD in the *termios c_cflag*
12162 field, as defined in POSIX .1 {8}.

12163 `cs5 cs6 cs7 cs8` Select character size, if possible. This shall have the
12164 effect of setting CS5, CS6, CS7, and CS8, respectively,
12165 in the *termios c_cflag* field, as defined in POSIX .1 {8}.

12166 *number* Set terminal baud rate to the number given, if possi-
12167 ble. If the baud rate is set to zero, the modem control
12168 lines shall no longer be asserted. This shall have the
12169 effect of setting the input and output *termios* baud
12170 rate values as defined in POSIX .1 {8}.

12171 `ispeed` *number* Set terminal input baud rate to the number given, if
12172 possible. If the input baud rate is set to zero, the
12173 input baud rate shall be specified by the value of the
12174 output baud rate. This shall have the effect of setting
12175 the input *termios* baud rate values as defined in
12176 POSIX .1 {8}.

12177 `ospeed` *number* Set terminal output baud rate to the number given, if
12178 possible. If the output baud rate is set to zero, the
12179 modem control lines shall no longer be asserted. This
12180 shall have the effect of setting the output *termios*
12181 baud rate values as defined in POSIX .1 {8}.

12182 `hupcl (-hupcl)` Stop asserting modem control lines (do not stop
12183 asserting modem control lines) on last close. This
12184 shall have the effect of setting (not setting) HUPCL in
12185 the *termios c_cflag* field, as defined in POSIX .1 {8}.

Copyright © 1991 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

12186 `hup (–hup)` Same as `hupcl` (`–hupcl`).

12187 `cstopb (–cstopb)` Use two (one) stop bits per character. This shall have
12188 the effect of setting (not setting) CSTOPB in the *ter-*
12189 *mios c_cflag* field, as defined in POSIX .1 {8}.

12190 `cread (–cread)` Enable (disable) the receiver. This shall have the
12191 effect of setting (not setting) CREAD in the *termios*
12192 *c_cflag* field, as defined in POSIX .1 {8}.

12193 `clocal (–clocal)` Assume a line without (with) modem control. This
12194 shall have the effect of setting (not setting) CLOCAL
12195 in the *termios c_cflag* field, as defined in POSIX .1 {8}.

12196 It is unspecified whether `stty` shall report an error if an attempt to set a Control
12197 Mode fails.

12198 **4.59.4.2 Input Modes**

12199 `ignbrk (–ignbrk)` Ignore (do not ignore) break on input. This shall have
12200 the effect of setting (not setting) IGNBRK in the *ter-*
12201 *mios c_iflag* field, as defined in POSIX .1 {8}.

12202 `brkint (–brkint)` Signal (do not signal) INTR on break. This shall have
12203 the effect of setting (not setting) BRKINT in the *ter-*
12204 *mios c_iflag* field, as defined in POSIX .1 {8}.

12205 `ignpar (–ignpar)` Ignore (do not ignore) bytes with parity errors. This
12206 shall have the effect of setting (not setting) IGNPAR in
12207 the *termios c_iflag* field, as defined in POSIX .1 {8}.

12208 `parmrk (–parmrk)` Mark (do not mark) parity errors. This shall have the
12209 effect of setting (not setting) PARMRK in the *termios*
12210 *c_iflag* field, as defined in POSIX .1 {8}.

12211 `inpck (–inpck)` Enable (disable) input parity checking. This shall
12212 have the effect of setting (not setting) INPCK in the
12213 *termios c_iflag* field, as defined in POSIX .1 {8}.

12214 `istrip (–istrip)` Strip (do not strip) input characters to seven bits.
12215 This shall have the effect of setting (not setting)
12216 ISTRIP in the *termios c_iflag* field, as defined in
12217 POSIX .1 {8}.

12218 `inlcr (–inlcr)` Map (do not map) NL to CR on input. This shall have
12219 the effect of setting (not setting) INLCR in the *termios*
12220 *c_iflag* field, as defined in POSIX .1 {8}.

12221 `igncr (–igncr)` Ignore (do not ignore) CR on input. This shall have
12222 the effect of setting (not setting) IGNCR in the *termios*
12223 *c_iflag* field, as defined in POSIX .1 {8}.

12224 `icrnl (–icrnl)` Map (do not map) CR to NL on input. This shall have
12225 the effect of setting (not setting) ICRNL in the *termios*

Copyright © 1991 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

c_iflag field, as defined in POSIX.1 {8}.

`ixon (–ixon)` Enable (disable) START/STOP output control. Output from the system is stopped when the system receives STOP and started when the system receives START. This shall have the effect of setting (not setting) IXON in the *termios c_iflag* field, as defined in POSIX.1 {8}.

`ixoff (–ixoff)` Request that the system send (not send) STOP characters when the input queue is nearly full and START characters to resume data transmission. This shall have the effect of setting (not setting) IXOFF in the *termios c_iflag* field, as defined in POSIX.1 {8}.

4.59.4.3 Output Modes

`opost (–opost)` Post-process output (do not post-process output; ignore all other output modes). This shall have the effect of setting (not setting) OPOST in the *termios c_oflag* field, as defined in POSIX.1 {8}.

4.59.4.4 Local Modes

`isig (–isig)` Enable (disable) the checking of characters against the special control characters INTR, QUIT, and SUSP. This shall have the effect of setting (not setting) ISIG in the *termios c_lflag* field, as defined in POSIX.1 {8}.

`icanon (–icanon)` Enable (disable) canonical input (ERASE and KILL processing). This shall have the effect of setting (not setting) ICANON in the *termios c_lflag* field, as defined in POSIX.1 {8}.

`iexten (–iexten)` Enable (disable) any implementation-defined special control characters not currently controlled by `icanon`, `isig`, `ixon`, or `ixoff`. This shall have the effect of setting (not setting) IEXTEN in the *termios c_lflag* field, as defined in POSIX.1 {8}.

`echo (–echo)` Echo back (do not echo back) every character typed. This shall have the effect of setting (not setting) ECHO in the *termios c_lflag* field, as defined in POSIX.1 {8}.

`echoe (–echoe)` The ERASE character shall (shall not) visually erase the last character in the current line from the display, if possible. This shall have the effect of setting (not setting) ECHOE in the *termios c_lflag* field, as defined in POSIX.1 {8}.

Copyright © 1991 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

`echok (–echok)` Echo (do not echo) NL after KILL character. This shall have the effect of setting (not setting) ECHOK in the *termios c_lflag* field, as defined in POSIX.1 {8}.

`echonl (–echonl)` Echo (do not echo) NL, even if `echo` is disabled. This shall have the effect of setting (not setting) ECHONL in the *termios c_lflag* field, as defined in POSIX.1 {8}.

`noflsh (–noflsh)` Disable (enable) flush after INTR, QUIT, SUSP. This shall have the effect of setting (not setting) NOFLSH in the *termios c_lflag* field, as defined in POSIX.1 {8}.

`tostop (–tostop)` Send SIGTTOU for background output. This shall have the effect of setting (not setting) TOSTOP in the *termios c_lflag* field, as defined in POSIX.1 {8}.

NOTE: Setting TOSTOP has no effect on systems not supporting the POSIX.1 {8} job control option.

4.59.4.5 Special Control Character Assignments

control-character string

Set *control-character* to *string*. If *control-character* is one of the character sequences in the first column of Table 4-9, the corresponding POSIX.1 {8} control character from the second column shall be recognized. This shall have the effect of setting the corresponding element of the *termios c_cc* array (see POSIX.1 {8} 7.1.2).

Table 4-9 – `stty` Control Character Names

control-character	POSIX.1 {8} Subscript	Description
`eof`	VEOF	EOF character
`eol`	VEOL	EOL character
`erase`	VERASE	ERASE character
`intr`	VINTR	INTR character
`kill`	VKILL	KILL character
`quit`	VQUIT	QUIT character
`susp`	VSUSP	SUSP character
`start`	VSTART	START character
`stop`	VSTOP	STOP character

If *string* is a single character, the control character shall be set to that character. If *string* is the two-character sequence `"^–"` or the string `"undef"`, the control character shall be set to {_POSIX_VDISABLE}, if it is in effect for the device; if {_POSIX_VDISABLE} is

Copyright © 1991 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

not in effect for the device, it shall be treated as an error. In the POSIX Locale, if *string* is a two-character sequence beginning with circumflex (^), and the second character is one of those listed in the ^*c* column of Table 4-10, the control character shall be set to the corresponding character value in the Value column of the table.

Table 4-10 – `stty` Circumflex Control Characters

^*c*	Value	^*c*	Value	^*c*	Value
a, A	<SOH>	l, L	<FF>	w, W	<ETB>
b, B	<STX>	m, M	<CR>	x, X	<CAN>
c, C	<ETX>	n, N	<SO>	y, Y	<EM>
d, D	<EOT>	o, O	<SI>	z, Z	<SUB>
e, E	<ENQ>	p, P	<DLE>	[	<ESC>
f, F	<ACK>	q, Q	<DC1>	\	<FS>
g, G	<BEL>	r, R	<DC2>	]	<GS>
h, H	<BS>	s, S	<DC3>	^	<RS>
i, I	<HT>	t, T	<DC4>	_	<US>
j, J	<LF>	u, U	<NAK>	?	<DEL>
k, K	<VT>	v, V	<SYN>		

`min` *number*
`time` *number*
: Set the value of `min` or `time` to *number* . MIN and TIME are used in noncanonical mode input processing (–`icanon`).

4.59.4.6 Combination Modes

saved settings
: Set the current terminal characteristics to the saved settings produced by the –`g` option.

`evenp` or `parity`
: Enable `parenb` and `cs7`; disable `parodd`.

`oddp`
: Enable `parenb`, `cs7`, and `parodd`.

–`parity`, –`evenp`, or –`oddp`
: Disable `parenb`, and set `cs8`.

`nl` (–`nl`)
: Enable (disable) `icrnl`. In addition, –`nl` unsets `inlcr` and `igncr`.

`ek`
: Reset ERASE and KILL characters back to system defaults.

`sane`
: Reset all modes to some reasonable, unspecified, values.

Copyright © 1991 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

12345 ### 4.59.5 External Influences

12346 #### 4.59.5.1 Standard Input

12347 Although no input is read from standard input, standard input is used to get the
12348 current terminal I/O characteristics and to set new terminal I/O characteristics.

12349 #### 4.59.5.2 Input Files

12350 None.

12351 #### 4.59.5.3 Environment Variables

12352 The following environment variables shall affect the execution of `stty`:

12353 **LANG** This variable shall determine the locale to use for the
12354 locale categories when both **LC_ALL** and the correspond-
12355 ing environment variable (beginning with **LC_**) do not
12356 specify a locale. See 2.6.

12357 **LC_ALL** This variable shall determine the locale to be used to over-
12358 ride any values for locale categories specified by the set-
12359 tings of **LANG** or any environment variables beginning
12360 with **LC_**.

12361 **LC_CTYPE** This variable shall determine the locale for the interpreta-
12362 tion of sequences of bytes of text data as characters (e.g.,
12363 single- versus multibyte characters in arguments) and
12364 which characters are in the class `print`.

12365 **LC_MESSAGES** This variable shall determine the language in which mes-
12366 sages should be written.

12367 #### 4.59.5.4 Asynchronous Events

12368 Default.

12369 ### 4.59.6 External Effects

12370 #### 4.59.6.1 Standard Output

12371 If operands are specified, no output shall be produced.

12372 If the `-g` option is specified, `stty` shall write to standard output the current set-
12373 tings in a form that can be used as arguments to another instance of `stty` on the
12374 same system.

12375 If the `-a` option is specified, all of the information as described in 4.59.4 shall be
12376 written to standard output. Unless otherwise specified, this information shall be
12377 written as `<space>`-separated tokens in an unspecified format, on one or more

Copyright © 1991 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

lines, with an unspecified number of tokens per line. Additional information may be written.

If no options or operands are specified, an unspecified subset of the information written for the `-a` option shall be written.

If speed information is written as part of the default output, or if the `-a` option is specified and if the terminal input speed and output speed are the same, the speed information shall be written as follows:

`"speed %d baud;"`, *<speed>*

Otherwise, speeds shall be written as:

`"ispeed %d baud; ospeed %d baud;"`, *<ispeed>*, *<ospeed>*

In locales other than the POSIX Locale, the word `baud` may be changed to something more appropriate in those locales.

If control characters are written as part of the default output, or if the `-a` option is specified, control characters shall be written as:

`"%s = %s;"`, *<control-character name>*, *<value>*

where *value* is either the character, or some visual representation of the character if it is nonprintable, or the string `<undef>` if the character is disabled.

4.59.6.2 Standard Error

Used only for diagnostic messages.

4.59.6.3 Output Files

None.

4.59.7 Extended Description

None.

4.59.8 Exit Status

The `stty` utility shall exit with one of the following values:

0 The terminal options were read or set successfully.

>0 An error occurred.

Copyright © 1991 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

12405 ### 4.59.9 Consequences of Errors

12406 Default.

12407 ### 4.59.10 Rationale. *(This subclause is not a part of P1003.2)*

12408 **Examples, Usage**

12409 Since POSIX.1 {8} doesn't specify any output modes, they are not specified in this
12410 standard either. Implementations are expected to provide `stty` operands
12411 corresponding to all of the output modes they support.

12412 In many ways outside the scope of POSIX.2, `stty` is primarily used to tailor the
12413 user interface of the terminal, such as selecting the preferred ERASE and KILL
12414 characters. As an application programming utility, `stty` can be used within shell
12415 scripts to alter the terminal settings for the duration of the script. The `-g` flag is
12416 designed to facilitate the saving and restoring of terminal state from the shell
12417 level. For example, a program may:

```
12418     saveterm="$(stty -g)"   # save terminal state
12419     stty (new settings)     # set new state
12420     ...                     # ...
12421     stty $saveterm          # restore terminal state
```

12422 Since the format is unspecified, the saved value is not portable across systems.

12423 Since the `-a` format is so loosely specified, scripts that save and restore terminal
12424 settings should use the `-g` option.

12425 **History of Decisions Made**

12426 The original `stty` manual page was taken directly from System V and reflected
12427 the System V terminal driver *termio*. It has been modified to correspond to the
12428 POSIX.1 {8} terminal driver *termios*.

12429 The *termios* section states that individual disabling of control characters is an
12430 option {_POSIX_VDISABLE}. If enabled, two conventions currently exist for speci-
12431 fying this: System V uses `"^-"`, and BSD uses `undef`. Both are accepted by
12432 POSIX.2 `stty`. The other BSD convention of using the letter `u` was rejected
12433 because it conflicts with the actual letter `u`, which is an acceptable value for a con-
12434 trol character.

12435 Early drafts did not specify the mapping of ^*c* to control characters because the
12436 control characters were not specified in the POSIX Locale character set description
12437 file requirements. The control character set is now specified in 2.4.1, so the tradi-
12438 tional mapping is specified. Note that although the mapping corresponds to
12439 control-character key assignments on many terminals that use ISO/IEC 646 {1} (or
12440 ASCII) character encodings, the mapping specified here is to the control charac-
12441 ters, not their keyboard encodings.

12442 The combination options `raw` and `cooked` (`-raw`) were dropped from the standard
12443 because the exact values that should be set are not well understood or commonly

Copyright © 1991 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

agreed on. In particular, *termios* has no explicit RAW bit, and the options that should be re-enabled (`–raw`) are not clear. General programming practice is to save the terminal state, change the settings for the duration of the program, and then reset the state. This is easy to do within a C program, however it is not possible for a single invocation of `stty` to restore the terminal state (`–raw`) without knowledge of the prior settings. Using the `–g` option and two calls to `stty`, a shell application could do this as described above. However, it is impossible to implement this as a single option. Also, it is not clear that changing word size and parity is appropriate. For example, requiring that `cooked` set `cs7` and `parenb` would be disastrous for users working with 8–bit international character sets. In general, these options are too ill-defined to be of any use.

Since *termios* supports separate speeds for input and output, two new options were added to specify each distinctly.

The `ixany` input mode was removed from Draft 8 on the basis that it could not be implemented on a POSIX.1 {8} system without extensions.

Some historical implementations use standard input to get and set terminal characteristics; others use standard output. Since input from a login TTY is usually restricted to the owner while output to a TTY is frequently open to the world, using standard input provides fewer chances of accidentally (or mischievously) altering the terminal settings of other users. Using standard input also allows `stty –a` and `stty –g` output to be redirected for later use. Therefore, usage of standard input is required by this standard.

The `tostop` option was omitted from early drafts through an oversight. It is the only option that requires job control to be effective, and thus could have gone into the UPE as a modification to `stty`, but since all other terminal control features are in the base standard, `tostop` was included as well.

Copyright © 1991 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.

12470 # 4.60 `tail` — Copy the last part of a file

12471 ## 4.60.1 Synopsis

12472 `tail [-f][ -c` *number* `| -n` *number* `][`*file*`]`

12473 *Obsolescent versions:*

12474 `tail -[`*number*`][c|l][f] [`*file*`]`

12475 `tail +[`*number*`][c|l][f] [`*file*`]`

12476 ## 4.60.2 Description

12477 The `tail` utility shall copy its input file to the standard output beginning at a
12478 designated place.

12479 Copying shall begin at the point in the file indicated by the `-c` *number* or
12480 `-n` *number* options (or the ±*number* portion of the argument to the obsolescent
12481 version). The option-argument *number* shall be counted in units of lines or bytes,
12482 according to the options `-n` and `-c` (or, in the obsolescent version, the appended
12483 option suffixes `l` or `c`).

12484 Tails relative to the end of the file may be saved in an internal buffer, and thus
12485 may be limited in length. Implementations shall ensure that such a buffer, if any,
12486 is no smaller than {LINE_MAX}*10 bytes.

12487 ## 4.60.3 Options

12488 The `tail` utility shall conform to the utility argument syntax guidelines
12489 described in standard described in 2.10.2, except that the obsolescent version
12490 accepts multicharacter options that can preceded by a plus sign.

12491 The following options shall be supported by the implementation in the nonob-
12492 solescent version:

12493 `-c` *number* The *number* option-argument shall be a decimal integer whose
12494 sign affects the location in the file, measured in bytes, to begin
12495 the copying:

12496

Sign	Copying Starts
+	Relative to the beginning of the file.
-	Relative to the end of the file.
none	Relative to the end of the file.

12497
12498
12499

12500 The origin for counting shall be 1; i.e., `-c +1` represents the first 1
12501 byte of the file, `-c -1` the last. 1

Copyright © 1991 IEEE. All rights reserved.
This is an unapproved IEEE Standards Draft, subject to change.