

NAME

pts - pseudo-terminal driver

DESCRIPTION

The **pts** driver provides support for a device-pair termed a *pseudo-terminal*. A pseudo-terminal is a pair of character devices, a *master* device and a *slave* device. The slave device provides to a process an interface identical to that described in `tty(4)`. However, whereas all other devices which provide the interface described in `tty(4)` have a hardware device of some sort behind them, the slave device has, instead, another process manipulating it through the master half of the pseudo-terminal. That is, anything written on the master device is given to the slave device as input and anything written on the slave device is presented as input on the master device.

The following `ioctl(2)` calls apply only to pseudo-terminals:

TIOCPKT Enable/disable *packet* mode. Packet mode is enabled by specifying (by reference) a nonzero parameter and disabled by specifying (by reference) a zero parameter. When applied to the master side of a pseudo-terminal, each subsequent `read(2)` from the terminal will return data written on the slave part of the pseudo-terminal preceded by a zero byte (symbolically defined as `TIOCPKT_DATA`), or a single byte reflecting control status information. In the latter case, the byte is an inclusive-or of zero or more of the bits:

`TIOCPKT_FLUSHREAD` whenever the read queue for the terminal is flushed.

`TIOCPKT_FLUSHWRITE` whenever the write queue for the terminal is flushed.

`TIOCPKT_STOP` whenever output to the terminal is stopped by a '^S'.

`TIOCPKT_START` whenever output to the terminal is restarted.

`TIOCPKT_DOSTOP` whenever `VSTOP` is '^S' and `VSTART` is '^Q'.

`TIOCPKT_NOSTOP` whenever the start and stop characters are not '^S/^Q'.

While this mode is in use, the presence of control status information to be read from the master side may be detected by a `select(2)` for exceptional conditions.

This mode is used by `rlogin(1)` and `rlogind(8)` to implement a remote-echoed, locally '^S/^Q' flow-controlled remote login with proper back-flushing of output; it can be used by other similar programs.

TIOCGPTN Obtain device unit number, which can be used to generate the filename of the pseudo-terminal slave device. This `ioctl(2)` should not be used directly. Instead, the `ptsname(3)` function should be used.

TIOCPTMASTER Determine whether the file descriptor is pointing to a pseudo-terminal master device. This `ioctl(2)` should not be used directly. It is used to implement routines like `grantpt(3)`.

FILES

The files used by this pseudo-terminals implementation are:

/dev/pts/[num] Pseudo-terminal slave devices.

DIAGNOSTICS

None.

SEE ALSO

`posix_openpt(2)`, `grantpt(3)`, `ptsname(3)`, `pty(4)`, `tty(4)`

HISTORY

A pseudo-terminal driver appeared in 4.2BSD. In FreeBSD 8.0, it was replaced with the **pts** driver.