NAME

screen - pc display interface

DESCRIPTION

Access to the *virtual consoles* are obtained through the device files */dev/ttyv0* - */dev/ttyvb*. Each of these files correspond to a separate virtual console. All virtual console devices can be open at once, but only one is active at a time. The active virtual console "owns" the keyboard and display screen.

Output to a virtual console that not currently is on the display is saved in a buffer that holds a "screenfull" (normally 25) lines. Any output written to */dev/console* (the original console device) is echoed to */dev/ttyv0*.

To switch between the virtual consoles one uses the sequence *ALT+Fn*, which means hold down ALT and press one of the function keys. The virtual console with the same number as the function key is then selected as the current virtual console, and given exclusive use of the keyboard and display. This switch sequence can be changed via the keyboard mapping ioctl call (see keyboard(4)).

The console allows entering values that are not physically present on the keyboard via a special keysequence. To use this facility press and hold down ALT, then enter a decimal number from 0-255 via the numerical keypad, then release ALT. The entered value is then used as the ASCII value for one character. This way it is possible to enter any ASCII value. The console driver also includes a history function. It is activated by pressing the scroll-lock key. This holds the display, and enables the cursor arrows for scrolling up and down through the last scrolled out lines.

The console understands a subset of the ANSI x3.64 character sequences. For compatibility with the old pccons, the PC3 character sequences are also supported.

ANSI	Seq	Function	Termcap entry
=======	=======	==	==============
--	E7	Save cursor position	sc
--	E8	Restore saved cursor position	rc
--	Ec	Reset	rs
--	EM	move cursor up 1 line, scroll if at top	--
CUU	E[nA	move cursor up n lines	up/UP (ku)

CUD	E[nB	move cursor down n lines	do/DO (kd)
CUF	E[nC	move cursor right n characters	nd/RI (kr)
CUB	E[nD	move cursor left n characters	--/LE (kl)
HPA	E[n‘	move cursor to character position n	ch
HPR	E[na	move cursor right n characters	--
VPA	E[nd	move cursor to line n	cv
VPR	E[ne	move cursor down n lines	--
CPL	E[nF	move cursor to start of line, n lines up	-- (@7)
CNL	E[nE	move cursor to start of line, n lines down	nw
CUP	E[y;xH	Move cursor to x, y	cm
HVP	E[y;xf	Move cursor to x, y	--
CBT	E[nZ	Move cursor back n tab stops	bt (kB)
IL	E[nL	Insert n blank lines	al/AL
ICH	E[n@	Insert n blank characters	ic/IC
DL	E[nM	Delete n lines	dl/DL
DCH	E[nP	Delete n characters	dc/DC
ED	E[nJ	Erase part or all of display: n=0 from cursor to end of display, n=1 from begin of display to cursor, n=2 entire display.	cd
EL	E[nK	Erase part or all of line:	ce

		n=0 from cursor to end of line, n=1 from begin of line to cursor, n=2 entire line.	
ECH	E[nX	Erase n characters	ec
SU	E[nS	Scroll display n lines up (forward)	sf/SF
SD	E[nT	Scroll display n lines down (reverse)	sr/SR

SGR	E[nm	Set character attributes:
		n= 0 normal attribute (all off)
		n= 1 bold (highlight)
		n= 4 underscore (if supported by HW)
		n= 5 blink (if supported by HW)
		n= 7 reverse
		n= 22 remove bold
		n= 24 remove underscore
		n= 25 remove blink
		n= 27 remove reverse
		n= 3X set ANSI foreground color (see table)
		n= 4X set ANSI background color (see table)
			--

X=0 black	X=1 red
X=2 green	X=3 brown
X=4 blue	X=5 magenta
X=6 cyan	X=7 light grey
X=9 reset to the normal color	

--	E[s	Save cursor position	sc
--	E[u	Restore saved cursor position	rc
--	E[x	Reset normal colors and attributes to their default values	--
--	E[nz	Switch to virtual console n	--

```
--    E[1;nx  Set normal ANSI background color       --
                   to n (see table)

--    E[2;nx  Set normal ANSI foreground color       --
                   to n (see table)

--    E[3;nx  Set normal video attribute directly    --
                   to n (n     from 0 to 255)

--    E[5;nx  Set normal ANSI reverse background     --
                   color to n (see        table)

--    E[6;nx  Set normal ANSI reverse foreground     --
                   color to n (see        table)

                     n= 0    black         n= 8    dark grey
                     n= 1    red           n= 9    light red
                     n= 2    green         n=10    light green
                     n= 3    brown         n=11    yellow
                     n= 4    blue          n=12    light blue
                     n= 5    magenta       n=13    light magenta
                     n= 6    cyan          n=14    light cyan
                     n= 7    light grey n=15       white

--    E[7;nx  Set normal reverse video attribute     --
                   directly to n (n from 0           to 255)

--          E[=p;dB  Set bell pitch (p) and duration (d),       --
                             pitch is in units of 840 nS,
                             duration is units of 0,1 S.

--          E[=tC    Set global cursor type (see table)         --

                     t=0 normal non-blinking
                     t=1 normal blinking
                     t=2 custom non-blinking
                     t=3 custom blinking
                     t=4 reset cursor (resets custom
                        cursor shape and sets current
                        cursor type to 0)
```

		t=5 hide cursor	
--	E[=s;eC	Set custom cursor shape, where s is the starting and e is the ending scanlines of the cursor.	--
--	E[=s;e;dC	Same as above, except d specifies the direction. If 0, scanlines are counted from the top to the bottom. If 1, from the bottom to the top.	--
--	E[=tS	Set local cursor type (see table)	--
		t=0 normal (global)	ve
		t=1 invisible	vi
		t=2 very visible	vs
--	E[=nA	Set the border color to n (see table) (if supported by HW)	--
--	E[=nF	Set normal foreground color to n (see table)	--
--	E[=nG	Set normal background color to n (see table)	--
--	E[=nH	Set normal reverse foreground color to n (see table)	--
--	E[=nI	Set normal reverse background color to n (see table)	--

n= 0	black	n= 8	dark grey
n= 1	blue	n= 9	light blue
n= 2	green	n=10	light green
n= 3	cyan	n=11	light cyan
n= 4	red	n=12	light red
n= 5	magenta	n=13	light magenta
n= 6	brown	n=14	yellow

n= 7	light grey	n=15	white

note: the first E in the sequences stands for ESC (0x1b)

AUTHORS

Søren Schmidt <*sos@FreeBSD.org*>