Software cursor for VGA
=======================

by Pavel Machek <pavel@atrey.karlin.mff.cuni.cz>
and Martin Mares <mj@atrey.karlin.mff.cuni.cz>

Linux now has some ability to manipulate cursor appearance. Normally, you can set the size of hardware cursor (and also work around some ugly bugs in those miserable Trident cards--see #define TRIDENT_GLITCH in drivers/video/vgacon.c). You can now play a few new tricks: you can make your cursor look like a non-blinking red block, make it inverse background of the character it's over or to highlight that character and still choose whether the original hardware cursor should remain visible or not. There may be other things I have never thought of.

The cursor appearance is controlled by a "<ESC>[?1;2;3c" escape sequence where 1, 2 and 3 are parameters described below. If you omit any of them, they will default to zeroes.

Parameter 1 specifies cursor size (0=default, 1=invisible, 2=underline, ..., 8=full block) + 16 if you want the software cursor to be applied + 32 if you want to always change the background color + 64 if you dislike having the background the same as the foreground. Highlights are ignored for the last two flags.

The second parameter selects character attribute bits you want to change (by simply XORing them with the value of this parameter). On standard VGA, the high four bits specify background and the low four the foreground. In both groups, low three bits set color (as in normal color codes used by the console) and the most significant one turns on highlight (or sometimes blinking--it depends on the configuration of your VGA).

The third parameter consists of character attribute bits you want to set. Bit setting takes place before bit toggling, so you can simply clear a bit by including it in both the set mask and the toggle mask.

Examples:
=========

```
To get normal blinking underline, use: echo -e '\033[?2c'
To get blinking block, use:            echo -e '\033[?6c'
To get red non-blinking block, use:    echo -e '\033[?17;0;64c'
```