

## Re: [PATCH] input: extend EV\_LED

---

*Source:* <http://linux.derkeiler.com/Mailing-Lists/Kernel/2007-02/msg06406.html>

---

- *From:* Németh Márton <[nm127@xxxxxxxxxxxxx](mailto:nm127@xxxxxxxxxxxxx)>
  - *Date:* Sun, 18 Feb 2007 08:45:00 +0100 (CET)
- 

On Fri, 16 Feb 2007, Henrique de Moraes Holschuh wrote:

On Thu, 15 Feb 2007, Richard Purdie wrote:

This has been discussed in several places several times.

The problem

with hardware accelerated flashing is that you're are often limited to certain constraints (this case being no exception) and indicating what these are to userspace in a generic fashion is difficult.

The hability to blinking at one rate is *\*very\** common on laptops. Blinking

at a few discrete rates is also common enough. They should be supported in a generic way.

[...]

Here's a suggestion for a simple, non-overengineered interface: a "blink"

attribute (on/off) for leds which can hardware-blink.

Only one blink

frequency is common enough that this attribute by itself

is very useful

(e.g. it is all a ThinkPad and most WiFi/network card leds need).

For hardware-blink leds with various frequencies, there is the typical way

to provide such things: give us a RO blink\_available\_frequencies attribute

which says which discrete frequencies are allowed (space separated), and a

RW blink\_frequency attribute to set the frequency. If instead of

blink\_available\_frequencies, the driver provides RO blink\_frequency\_min and

\_max attributes, then it means it can blink on that range of freqs.

That is simple enough to implement and use, and generic enough. You just

need to set in stone if you want the freq in Hz, or a submultiple. You can

even implement an optional "blink" software emulation that drivers can hook

into for systems where the driver \*knows\* that led access is fast, but there

is no hardware blinking emulation.

Re: [PATCH] input: extend EV\_LED

A blinking led is basically a PWM (Pulse Width Modulation) signal. A PWM signal has three different attribute. The first one is the amplitude, this attribute is already provided by the led subsystem as "brightness". There are two more attributes, which are the frequency [Hz] and the duty cycle [%], or the on-time [ms] and off-time [ms].

The frequency [Hz] and duty cycle [%] parameters has the problem, that if we are limited to integers, it is not possible to express slower blinks than 1Hz. We could also use [mHz] (milli-Hertz), but it is not very common unit.

The on-time [ms] and off-time [ms] seems to be easier to handle (and this is also easier to simulate from software if ever needed). An RO attribute could be introduced: 'pwm\_available', which can contain parameter pairs separated by space, and the new parameter pair is in new line. An RW attribute 'pwm' could accept a parameter pair separated by space.

A range could be also given in 'pwm\_available', like this: '500-1000 500-1000'. This has the limitation that if there is a hardware which can blink a LED in differet frequencies, but only with fixed 50% duty cycle, the on-time off-time pair cannot express this range easily.

My real-world example would be my Clevo D4J (D410J) notebook's mail led. It has three known state: off, PWM at 1Hz (50%), PWM at 0.5Hz (50%). In case the on-time [ms] off-time [ms] parameter pairs are used:

```
$ cat pwm_available
500 500
1000 1000
```

What is your opinion?

Is there more real-world hardware accelerated blinking LED parameter examples?

NMarci

---

KedvezQ tandíjak, különleges kedvezmények és magas szintű szolgáltatások.  
Ez az Educomm online nyelviskola és az egyedülálló EDUCATOR. Kattints ide!  
<http://ad.adverticum.net/b/cl.1.6022.131839.203066/click.prm>

-

Re: [PATCH] input: extend EV\_LED

Re: [PATCH] input: extend EV\_LED

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@xxxxxxxxxxxxxxxxx  
More majordomo info at <http://vger.kernel.org/majordomo-info.html>  
Please read the FAQ at <http://www.tux.org/lkml/>