

# why tasklet cant sleep

Источник: <https://lists.kernelnewbies.org/pipermail/kernelnewbies/2011-November/003812.html>

Hi santhosh,

On Thu, Nov 10, 2011 at 8:38 AM, santhosh kumars <[karuna.linux@gmail.com](mailto:karuna.linux@gmail.com)> wrote:

```
> hi,  
> can any one please explain why tasklets cant sleep.softirqs and  
> tasklets processing aided by the a set of per processor  
> kernel threads(ksoftirqd) so it means tasklets/softirqs run in process  
> context. it means tasklet can sleep right?
```

Generally speaking, tasklets "borrow" the stack of whatever process happened to be running.

Stepping back a bit, hardware interrupts come along and interrupt whatever task happened to be running. When the hardware IRQ stack gets back down to the task level again (i.e. all nested HW IRQs are processed, then it enabled interrupts and starts any queued tasklets. So tasklets are really still "interrupt" context, but with interrupts enabled.

Since the tasklet is borrowing a stack, that process can't run until the tasklet is finished, so to simplify things no context switches can occur while tasklets are running.

If your kernel was configured with a separate irq stack, then it's conceivable that tasklets could run essentially in a thread context. But because not all architectures implement a separate hw irq stack, and you want drivers and stuff to be portable, you have to cater to the lowest common denominator.

If you want your tasklet to run in process context, then use a kernel thread instead of a tasklet.

--

Dave Hylands  
Shuswap, BC, Canada