Why spinlocks are used in interrupt handlers

I would like to know why spin locks are used instead of semaphores inside an interrupt handler.

c operating-system linux-kernel

edited Jan 7 '13 at 7:29
Bijoy Thangaraj
2,846 2 16 41

asked Aug 12 '10 at 4:06
mousey
2,198 7 23 40

2 Answers

Semaphores cause tasks to sleep on contention, which is unacceptable for interrupt handlers. Basically, for such a short and fast task (interrupt handling) the work carried out by the semaphore is overkill. Also, spinlocks can't be held by more than one task.

answered Aug 12 '10 at 4:14
Michael Foukarakis
16k 2 38 65

@mfukar I know the reason but I didnt understand why double acquire deadlocks can occur because of sleeping and not while using spin locks – mousey Aug 12 '10 at 4:19

Also I would like to know why spin_trylock() is needed ? – mousey Aug 12 '10 at 4:33

@mousey: double-acquire deadlocks cannot occur with spinlocks, because if a spinlock is shared with an interrupt handler all lockers of the spinlock must use the interrupt-disabling variant of the lock function. – caf Aug 12 '10 at 4:55

@caf who shares spin lock with an interrupt handler ? Can you please explain. Similarly mutexes can be done in the same way right. we can disable all the interrupts. – mousey Aug 12 '10 at 5:00

1 @mousey: Someone shares a spin lock with an interrupt handler if they access the same shared data that the interrupt handler does. Mutexes cannot disable interrupts, since if you disable interrupts and sleep, you will never wake up! – caf Aug 12 '10 at 5:18

The problem is that interrupt handlers (IH) are triggered asynchronously and in unpredictable way, out of the scope of any other activities running in the system. In fact, IHs run out of the scope of concept of the threads and scheduling at all. Due to this all mutual exclusion primitives which rely to the scheduler are unacceptable. Because they usage in the IH can dramatically increases the interrupt handling latencies (in case of IH running in the context of low priority thread) and is able to produce deadlocks (in case of IH running in the context of thread which hold the lock).

You can look at nice and detailed description of spinlocks at http://www.makelinux.net/ldd3/chp-5-sect-5.

edited Sep 15 '13 at 19:41

answered Sep 15 '13 at 18:13
ZarathustrA
549 3 6