

Linux-Kernel: [PATCH 502] MVME167 serial: Replace bottom half handler with task queue handler

## [PATCH 502] MVME167 serial: Replace bottom half handler with task queue handler

*Source:* <http://linux.derkeiler.com/Mailing-Lists/Kernel/2004-10/10226.html>

---

*From:* Geert Uytterhoeven ([geert\\_at\\_linux-m68k.org](mailto:geert_at_linux-m68k.org))

*Date:* 10/31/04

Date: Sun, 31 Oct 2004 11:03:39 +0100

To: Linus Torvalds <[torvalds@osdl.org](mailto:torvalds@osdl.org)>, Andrew Morton <[akpm@osdl.org](mailto:akpm@osdl.org)>

MVME167 serial: Fix compilation by replacing the bottom half handler with a task queue handler, based on the Cyclades driver (from Kars de Jong)

Signed-off-by: Geert Uytterhoeven <[geert@linux-m68k.org](mailto:geert@linux-m68k.org)>

--- linux-2.6.10-rc1/drivers/char/serial167.c 2004-10-23 10:33:01.000000000 +0200  
+++ linux-m68k-2.6.10-rc1/drivers/char/serial167.c 2004-10-23 15:42:26.000000000 +0200

@@ -39,6 +39,9 @@

\* - don't use the panic function in serial167\_init

\* - do resource release on failure on serial167\_init

\* - include missing restore\_flags in mvme167\_serial\_console\_setup

+ \*

+ \* Kars de Jong <[jongk@linux-m68k.org](mailto:jongk@linux-m68k.org)> - 2004/09/06

+ \* - replace bottom half handler with task queue handler

\*/

#include <linux/config.h>

@@ -89,8 +92,6 @@

#define SERIAL\_TYPE\_NORMAL 1

-DECLARE\_TASK\_QUEUE(tq\_cyclades);

-

static struct tty\_driver \*cy\_serial\_driver;

extern int serial\_console;

static struct cyclades\_port \*serial\_console\_info = NULL;

@@ -373,8 +374,7 @@ static inline void

cy\_sched\_event(struct cyclades\_port \*info, int event)

{

    info->event |= 1 << event; /\* remember what kind of event and who \*/

- queue\_task(&info->tqueue, &tq\_cyclades); /\* it belongs to \*/

- mark\_bh(CYCLADES\_BH); /\* then trigger event \*/

+ schedule\_work(&info->tqueue);

} /\* cy\_sched\_event \*/

## Linux–Kernel: [PATCH 502] MVME167 serial: Replace bottom half handler with task queue handler

```
@@ -467,7 +467,7 @@ cd2401_rxerr_interrupt(int irq, void *de
    and nothing could be done about it!!! */
    }
}
- queue_task(&tty->flip.tqueue, &tq_timer);
+ schedule_delayed_work(&tty->flip.work, 1);
    /* end of service */
    base_addr[CyREOIR] = rfoc ? 0 : CyNOTTRANS;
    return IRQ_HANDLED;
@@ -702,7 +702,7 @@ cd2401_rx_interrupt(int irq, void *dev_i
    udelay(10L);
#endif
}
- queue_task(&tty->flip.tqueue, &tq_timer);
+ schedule_delayed_work(&tty->flip.work, 1);
}
/* end of service */
base_addr[CyREOIR] = save_cnt ? 0 : CyNOTTRANS;
@@ -713,7 +713,7 @@ cd2401_rx_interrupt(int irq, void *dev_i
* This routine is used to handle the "bottom half" processing for the
* serial driver, known also the "software interrupt" processing.
* This processing is done at the kernel interrupt level, after the
- * cy_interrupt() has returned, BUT WITH INTERRUPTS TURNED ON. This
+ * cy#/_interrupt() has returned, BUT WITH INTERRUPTS TURNED ON. This
* is where time-consuming activities which can not be done in the
* interrupt driver proper are done; the interrupt driver schedules
* them using cy_sched_event(), and they get done here.
@@ -721,9 +721,7 @@ cd2401_rx_interrupt(int irq, void *dev_i
* This is done through one level of indirection--the task queue.
* When a hardware interrupt service routine wants service by the
* driver's bottom half, it enqueues the appropriate tq_struct (one
- * per port) to the tq_cyclades work queue and sets a request flag
- * via mark_bh for processing that queue. When the time is right,
- * do_cyclades_bh is called (because of the mark_bh) and it requests
+ * per port) to the keventd work queue and sets a request flag
* that the work queue be processed.
*
* Although this may seem unwieldy, it gives the system a way to
@@ -732,12 +730,6 @@ cd2401_rx_interrupt(int irq, void *dev_i
* had to poll every port to see if that port needed servicing.
*/
static void
-do_cyclades_bh(void)
- {
- run_task_queue(&tq_cyclades);
- } /* do_cyclades_bh */
-
-static void
do_softint(void *private_)
{
```

Linux-Kernel: [PATCH 502] MVME167 serial: Replace bottom half handler with task queue handler

```
    struct cyclades_port *info = (struct cyclades_port *) private_;  
@@ -2278,8 +2270,6 @@ scrn[1] = '\0';  
        return ret;  
    }
```

```
- init_bh(CYCLADES_BH, do_cyclades_bh);  
-
```

```
    port_num = 0;  
    info = cy_port;  
    for (index = 0; index < 1; index++) {  
@@ -2317,8 +2307,7 @@ scrn[1] = '\0';  
        info->blocked_open = 0;  
        info->default_threshold = 0;  
        info->default_timeout = 0;  
- info->tqueue.routine = do_softint;  
- info->tqueue.data = info;  
+ INIT_WORK(&info->tqueue, do_softint, info);  
        init_waitqueue_head(&info->open_wait);  
        init_waitqueue_head(&info->close_wait);  
        /* info->session */
```

Gr{oetje,eeting}s,

Geert

--

Geert Uytterhoeven -- There's lots of Linux beyond ia32 -- geert@linux-m68k.org  
In personal conversations with technical people, I call myself a hacker. But  
when I'm talking to journalists I just say "programmer" or something like that.  
-- Linus Torvalds

-

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