

# Re: [PATCH] libata: skip reset on bus not a device

---

*Source:* <http://linux.derkeiler.com/Mailing-Lists/Kernel/2006-09/msg09074.html>

---

- *From:* Tejun Heo <[htejun@xxxxxxxxx](mailto:htejun@xxxxxxxxx)>
  - *Date:* Sat, 30 Sep 2006 23:14:42 +0900
- 

Hello, Joe Jin.

Joe Jin wrote:

On my pc(Dell OptiPlex GX620), while boot up it by 2.6.18, the next info always report:

```
=====
ata2: port is slow to respond, please be patient
ata2: port failed to respond (30 secs)
ata2: SRST failed (status 0xFF)
ata2: SRST failed (err_mask=0x100)
ata2: softreset failed, retrying in 5 secs
ata2: SRST failed (status 0xFF)
ata2: SRST failed (err_mask=0x100)
ata2: softreset failed, retrying in 5 secs
ata2: SRST failed (status 0xFF)
ata2: SRST failed (err_mask=0x100)
ata2: reset failed, giving up
=====
```

I see.

```
00:1f.1 IDE interface: Intel Corporation 82801G (ICH7 Family) IDE
Controller (rev 01) (prog-if 8a [Master SecP PriP])
Subsystem: Dell: Unknown device 01ad
Flags: bus master, medium devsel, latency 0, IRQ 169
I/O ports at <ignored>
I/O ports at <ignored>
I/O ports at <ignored>
I/O ports at <ignored>
I/O ports at ffa0 [size=16]
00: 86 80 df 27 05 00 88 02 01 8a 01 01 00 00 00 00
10: f1 01 00 00 f5 03 00 00 71 01 00 00 75 03 00 00
20: a1 ff 00 00 00 00 00 00 00 00 00 00 28 10 ad 01
30: 00 00 00 00 00 00 00 00 00 00 00 00 0b 01 00 00
```

```
00:1f.2 IDE interface: Intel Corporation 82801GB/GR/GH (ICH7 Family)
```

Re: [PATCH] libata: skip reset on bus not a device

```
Serial ATA Storage Controller IDE (rev 01) (prog-if 8f [Master SecP
SecO PriP PriO])
Subsystem: Dell: Unknown device 01ad
Flags: bus master, 66Mhz, medium devsel, latency 0, IRQ 217
I/O ports at fe00 [size=8]
I/O ports at fe10 [size=4]
I/O ports at fe20 [size=8]
I/O ports at fe30 [size=4]
I/O ports at fea0 [size=16]
Capabilities: <available only to root>
00: 86 80 c0 27 07 00 b0 02 01 8f 01 01 00 00 00 00
10: 01 fe 00 00 11 fe 00 00 21 fe 00 00 31 fe 00 00
20: a1 fe 00 00 00 00 00 00 00 00 00 28 10 ad 01
30: 00 00 00 00 70 00 00 00 00 00 00 00 05 03 00 00
```

Can you post the result of the following command.

```
# lspci -nvvvxxx -s 00:1f.
```

through traced the code, and found which caused it:  
at scsi\_eh\_[port\_no] kernel thread, it should reset the bus, before reset it,  
it never check it if have a device. if it should skip it?  
thanks

Hmmm... Not really. The controller shouldn't report BSY for empty channel. There's a notable exception where PATA pins aren't properly pulled resulting in 0xff status on empty channels. IDE handles the case specially but libata doesn't yet. Can you try the attached patch?

---  
tejun

```
diff --git a/drivers/scsi/libata-core.c b/drivers/scsi/libata-core.c
index 427b73a..c8a1bc6 100644
--- a/drivers/scsi/libata-core.c
+++ b/drivers/scsi/libata-core.c
@@ -2271,11 +2271,14 @@ static inline void ata_tf_to_host(struct
 * Sleep until ATA Status register bit BSY clears,
 * or a timeout occurs.
 *
- * LOCKING: None.
+ * LOCKING:
+ * Kernel thread context (may sleep).
+ *
+ * RETURNS:
+ * 0 on success, -errno otherwise.
 */
-
-unsigned int ata_busy_sleep (struct ata_port *ap,
```

Re: [PATCH] libata: skip reset on bus not a device

Re: [PATCH] libata: skip reset on bus not a device

```
- unsigned long tmout_pat, unsigned long tmout)
+int ata_busy_sleep(struct ata_port *ap,
+ unsigned long tmout_pat, unsigned long tmout)
{
unsigned long timer_start, timeout;
u8 status;
@@ -2283,25 +2286,30 @@ unsigned int ata_busy_sleep (struct ata_
status = ata_busy_wait(ap, ATA_BUSY, 300);
timer_start = jiffies;
timeout = timer_start + tmout_pat;
- while ((status & ATA_BUSY) && (time_before(jiffies, timeout))) {
+ while (status != 0xff && (status & ATA_BUSY) &&
+ time_before(jiffies, timeout)) {
msleep(50);
status = ata_busy_wait(ap, ATA_BUSY, 3);
}

- if (status & ATA_BUSY)
+ if (status != 0xff && (status & ATA_BUSY))
ata_port_printk(ap, KERN_WARNING,
"port is slow to respond, please be patient\n");

timeout = timer_start + tmout;
- while ((status & ATA_BUSY) && (time_before(jiffies, timeout))) {
+ while (status != 0xff && (status & ATA_BUSY) &&
+ time_before(jiffies, timeout)) {
msleep(50);
status = ata_chk_status(ap);
}

+ if (status == 0xff)
+ return -ENODEV;
+
if (status & ATA_BUSY) {
ata_port_printk(ap, KERN_ERR, "port failed to respond "
"%lu secs)\n", tmout / HZ);
- return 1;
+ return -EBUSY;
}

return 0;
diff --git a/include/linux/libata.h b/include/linux/libata.h
index 66c3100..eb7d90f 100644
--- a/include/linux/libata.h
+++ b/include/linux/libata.h
@@ -702,9 +702,8 @@ extern int ata_host_set_suspend(struct a
pm_message_t mesg);
extern void ata_host_set_resume(struct ata_host_set *host_set);
extern int ata_ratelimit(void);
-extern unsigned int ata_busy_sleep(struct ata_port *ap,
- unsigned long timeout_pat,
```

Re: [PATCH] libata: skip reset on bus not a device

```
- unsigned long timeout);
+extern int ata_busy_sleep(struct ata_port *ap,
+ unsigned long timeout_pat, unsigned long timeout);
extern void ata_port_queue_task(struct ata_port *ap, void (*fn)(void *),
void *data, unsigned long delay);
extern u32 ata_wait_register(void __iomem *reg, u32 mask, u32 val,
@@ -1019,7 +1018,7 @@ static inline u8 ata_busy_wait(struct at
udelay(10);
status = ata_chk_status(ap);
max--;
- } while ((status & bits) && (max > 0));
+ } while (status != 0xff && (status & bits) && (max > 0));

return status;
}
@@ -1040,7 +1039,7 @@ static inline u8 ata_wait_idle(struct at
{
u8 status = ata_busy_wait(ap, ATA_BUSY | ATA_DRQ, 1000);

- if (status & (ATA_BUSY | ATA_DRQ)) {
+ if (status != 0xff && (status & (ATA_BUSY | ATA_DRQ))) {
unsigned long l = ap->ioaddr.status_addr;
if (ata_msg_warn(ap))
printk(KERN_WARNING "ATA: abnormal status 0x%X on port 0x%IX\n",
```