

Fixing USB interrupt problems with ACPI enabled

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2003-08/8106.html>

From: Nakajima, Jun (jun.nakajima_at_intel.com)

Date: 08/30/03

Date: Sat, 30 Aug 2003 08:47:41 -0700

To: "lkml" <linux-kernel@vger.kernel.org>, <acpi-devel@lists.sourceforge.net>

Doing this for Len, who is on vacation. We would like to thank the people who provided debugging info such as `acpidmp`, `dmidecode`, and `dmesg`. This is one of our findings, and we believe this would fix some interrupt problems (with USB, for example) with ACPI enabled, especially when the `dmesg` reads like:

```
ACPI: PCI Interrupt Link [ALKA] enabled at IRQ 0
```

```
ACPI: PCI Interrupt Link [ALKB] enabled at IRQ 0
```

```
ACPI: PCI Interrupt Link [ALKC] enabled at IRQ 0
```

```
ACPI: PCI Interrupt Link [ALKD] enabled at IRQ 0
```

Basically we assumed that `_CRS` returned the one we set with `_SRS`, when setting up a PCI interrupt link device, but that's not the case with some AML codes. Some of them always return 0.

Attached is a patch against 2.4.23-pre1. It should be easy to apply this to 2.6.

Thanks,

Jun

```
diff -ru /build/orig/linux-2.4.23-pre1/drivers/acpi/pci_link.c
```

```
linux-2.4.23-pre1/drivers/acpi/pci_link.c
```

```
--- /build/orig/linux-2.4.23-pre1/drivers/acpi/pci_link.c
```

```
2003-08-25 04:44:41.000000000 -0700
```

```
+++ linux-2.4.23-pre1/drivers/acpi/pci_link.c 2003-08-29
```

```
20:21:13.000000000 -0700
```

```
@@ -216,7 +216,6 @@
```

```
    return AE_CTRL_TERMINATE;
```

```
}
```

```
-
```

```
static int
```

```
acpi_pci_link_get_current (
```

```
    struct acpi_pci_link *link)
```

```
@@ -275,6 +274,26 @@
```

```
    return _VALUE(result);
```

```
}
```

Linux-Kernel: Fixing USB interrupt problems with ACPI enabled

```
+static int
+acpi_pci_link_try_get_current (
+ struct acpi_pci_link *link,
+ int irq)
+{
+ int result;
+
+ result = acpi_pci_link_get_current(link);
+ if (result && link->irq.active) {
+ return_VALUE(result);
+ }
+
+ if (!link->irq.active) {
+ ACPI_DEBUG_PRINT((ACPI_DB_ERROR, "No active IRQ resource
found\n"));
+ printk(KERN_WARNING "_CRS returns NULL! Using IRQ %d for
device (%s [%s]).\n", irq, acpi_device_name(link->device),
acpi_device_bid(link->device));
+ link->irq.active = irq;
+ }
+
+ return 0;
+}

static int
acpi_pci_link_set (
@@ -359,7 +378,7 @@
    }

    /* Make sure the active IRQ is the one we requested. */
- result = acpi_pci_link_get_current(link);
+ result = acpi_pci_link_try_get_current(link, irq);
    if (result) {
        return_VALUE(result);
    }
@@ -573,10 +592,6 @@
        else
            printk(" %d", link->irq.possible[i]);
    }
- if (!link->irq.active)
- printk(", disabled");
- else if (!found)
- printk(", enabled at IRQ %d", link->irq.active);
    printk("\n");

    /* TBD: Acquire/release lock */
```

-
To unsubscribe from this list: send the line "unsubscribe linux-kernel" in
the body of a message to majordomo@vger.kernel.org

Linux–Kernel: Fixing USB interrupt problems with ACPI enabled

More majordomo info at <http://vger.kernel.org/majordomo–info.html>

Please read the FAQ at <http://www.tux.org/lkml/>