

Griffis, Brad

From: Kevin Hilman [khilman@deeprootssystems.com]
Sent: Tuesday, April 06, 2010 12:42 PM
To: linux-i2c@vger.kernel.org; Ben Dooks
Cc: Dirk Behme; Rajashekhara, Sudhakar; Alexander Vasiliev; Griffis, Brad; Dirk Behme; Jean Delvare (PC drivers, core); Ben Dooks (embedded platforms); Kevin Hilman; Subrahmanya, Chaitrika; Philby John; Srinivasan, Nageswari; linux-kernel@vger.kernel.org
Subject: [PATCH 1/6] i2c: davinci: Fix smbus Oops with AIC33 usage

From: Dirk Behme <dirk.behme@googlemail.com>

This fixes Oops at kernel startup while "scanning" for TLV320AIC23IDx addresses.

Additional fix from Sudhakar Rajashekhara: I think 'first byte set' should come after the write because an I2C transaction is being carried out before configuring the I2C mode register (which has bits to configure Master, Start condition etc), which causes undefined behavior.

Signed-off-by: Sudhakar Rajashekhara <sudhakar.raj@ti.com>

Signed-off-by: Alexander Vasiliev <alexvasiljev@gmail.com>

Signed-off-by: Brad Griffis <bgriffis@ti.com>

Signed-off-by: Dirk Behme <dirk.behme@gmail.com>

Acked-by: Kevin Hilman <khilman@deeprootssystems.com>

```
---
drivers/i2c/busses/i2c-davinci.c | 31 ++++++-----
1 files changed, 27 insertions(+), 4 deletions(-)

diff --git a/drivers/i2c/busses/i2c-davinci.c b/drivers/i2c/busses/i2c-davinci.c
index c89687a..3b9ab23 100644
--- a/drivers/i2c/busses/i2c-davinci.c
+++ b/drivers/i2c/busses/i2c-davinci.c
@@ -112,6 +112,7 @@ struct davinci_i2c_dev {
     u8          *buf;
     size_t      buf_len;
     int         irq;
+    int         stop;
     u8          terminate;
     struct i2c_adapter adapter;
 };
@@ -249,9 +250,6 @@ int i2c_davinci_xfer_msg(struct i2c_adapter *adap, struct i2c_msg *msg,
int stop)
{
    u16 w;
    int r;

-    if (msg->len == 0)
-        return -EINVAL;
-
    if (!pdata)
        pdata = &davinci_i2c_platform_data_default;
    /* Introduce a delay, required for some boards (e.g Davinci EVM) */
@@ -263,6 +261,7 @@ int i2c_davinci_xfer_msg(struct i2c_adapter *adap, struct i2c_msg *msg, int stop)
{
    dev->buf = msg->buf;
    dev->buf_len = msg->len;
+    dev->stop = stop;

    davinci_i2c_write_reg(dev, DAVINCI_I2C_CNT_REG, dev->buf_len);

@@ -280,6 +279,10 @@ int i2c_davinci_xfer_msg(struct i2c_adapter *adap, struct i2c_msg *msg,
int stop)
{
    flag |= DAVINCI_I2C_MDR_TRX;
    if (stop)
        flag |= DAVINCI_I2C_MDR_STP;
```

```

+     if (msg->len == 0) {
+         flag |= DAVINCI_I2C_MDR_RM;
+         flag &= ~DAVINCI_I2C_MDR_STP;
+     }

    /* Enable receive or transmit interrupts */
    w = davinci_i2c_read_reg(dev, DAVINCI_I2C_IMR_REG); @@ -290,9 +293,21 @@
i2c_davinci_xfer_msg(struct i2c_adapter *adap, struct i2c_msg *msg, int stop)
    davinci_i2c_write_reg(dev, DAVINCI_I2C_IMR_REG, w);

    dev->terminate = 0;

+
    /* write the data into mode register */
    davinci_i2c_write_reg(dev, DAVINCI_I2C_MDR_REG, flag);

+
    /*
+   * First byte should be set here, not after interrupt,
+   * because transmit-data-ready interrupt can come before
+   * NACK-interrupt during sending of previous message and
+   * ICDXR may have wrong data
+   */
    if ((! (msg->flags & I2C_M_RD)) && dev->buf_len) {
+        davinci_i2c_write_reg(dev, DAVINCI_I2C_DXR_REG, *dev->buf++);
+        dev->buf_len--;
+    }
+
    r = wait_for_completion_interruptible_timeout(&dev->cmd_complete,
                                                dev->adapter.timeout);

    if (r == 0) {
@@ -371,7 +386,7 @@ i2c_davinci_xfer(struct i2c_adapter *adap, struct i2c_msg msgs[], int
num)

    static u32 i2c_davinci_func(struct i2c_adapter *adap) {
-        return I2C_FUNC_I2C | (I2C_FUNC_SMBUS_EMUL & ~I2C_FUNC_SMBUS_QUICK);
+        return I2C_FUNC_I2C | I2C_FUNC_SMBUS_EMUL;
    }

    static void terminate_read(struct davinci_i2c_dev *dev) @@ -430,6 +445,14 @@ static
irqreturn_t i2c_davinci_isr(int this_irq, void *dev_id)
    case DAVINCI_I2C_IVR_ARDY:
        davinci_i2c_write_reg(dev,
            DAVINCI_I2C_STR_REG, DAVINCI_I2C_STR_ARDY);
+
+        if (((dev->buf_len == 0) && (dev->stop != 0)) ||
+            (dev->cmd_err & DAVINCI_I2C_STR_NACK)) {
+            w = davinci_i2c_read_reg(dev,
+                DAVINCI_I2C_MDR_REG);
+            w |= DAVINCI_I2C_MDR_STP;
+            davinci_i2c_write_reg(dev,
+                DAVINCI_I2C_MDR_REG, w);
+        }
        complete(&dev->cmd_complete);
        break;

--
1.7.0.2

```