Kernel driver i2c-parport

Author: Jean Delvare <khali@linux-fr.org>

This is a unified driver for several i2c-over-parallel-port adapters,
such as the ones made by Philips, Velleman or ELV. This driver is
meant as a replacement for the older, individual drivers:
 * i2c-philips-par
 * i2c-elv
 * i2c-velleman
 * video/i2c-parport (NOT the same as this one, dedicated to home brew
                      teletext adapters)

It currently supports the following devices:
 * (type=0) Philips adapter
 * (type=1) home brew teletext adapter
 * (type=2) Velleman K8000 adapter
 * (type=3) ELV adapter
 * (type=4) Analog Devices ADM1032 evaluation board
 * (type=5) Analog Devices evaluation boards: ADM1025, ADM1030, ADM1031
 * (type=6) Barco LPT->DVI (K5800236) adapter
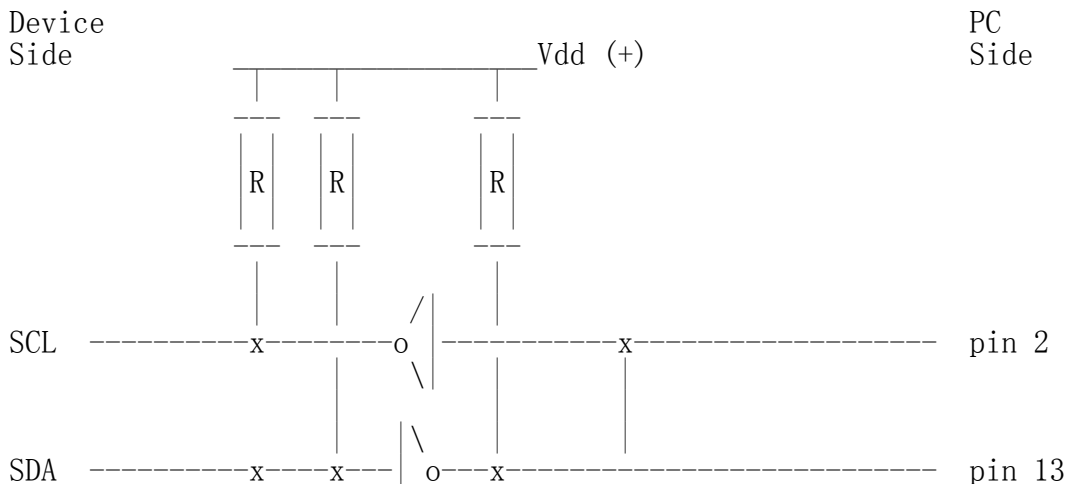 * (type=7) One For All JP1 parallel port adapter

These devices use different pinout configurations, so you have to tell
the driver what you have, using the type module parameter. There is no
way to autodetect the devices. Support for different pinout configurations
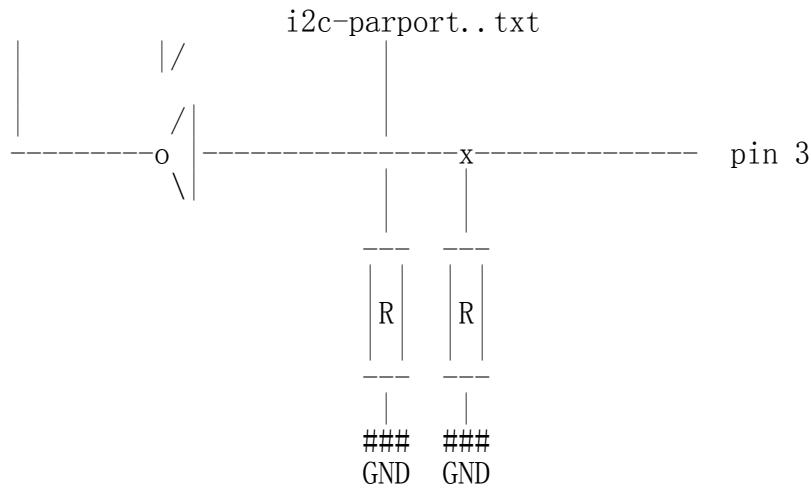can be easily added when needed.

Earlier kernels defaulted to type=0 (Philips). But now, if the type
parameter is missing, the driver will simply fail to initialize.

SMBus alert support is available on adapters which have this line properly
connected to the parallel port's interrupt pin.

Building your own adapter

If you want to build you own i2c-over-parallel-port adapter, here is
a sample electronics schema (credits go to Sylvain Munaut):

```
Device                                                        PC
Side                   _____Vdd (+)            Side
                      |         |          |
                     ___       ___        ___
                    |   |     |   |      |   |
                    | R |     | R |      | R |
                    |   |     |   |      |   |
                     ---       ---        ---
                      |         |   /|     |
                      |         |  / |     |
SCL  -----------x---------o | ------------x------------------ pin 2
                      |         | \ |     |
                      |         |  \|     |
                      |      |  \      |
SDA  -----------x---x---  | o---x------------------------ pin 13
```

```
                                  i2c-parport..txt
    |               |/                      |
    |                /                       |
    |               /|                       |
    ----------o |---------------------x------------- pin 3
    |          \|                      |   |
    |                                  |   |
                                      ---  ---
                                     | |  | |
                                     |R|  |R|
                                     | |  | |
                                      ---  ---
                                       |    |
                                      ###  ###
                                      GND  GND
```
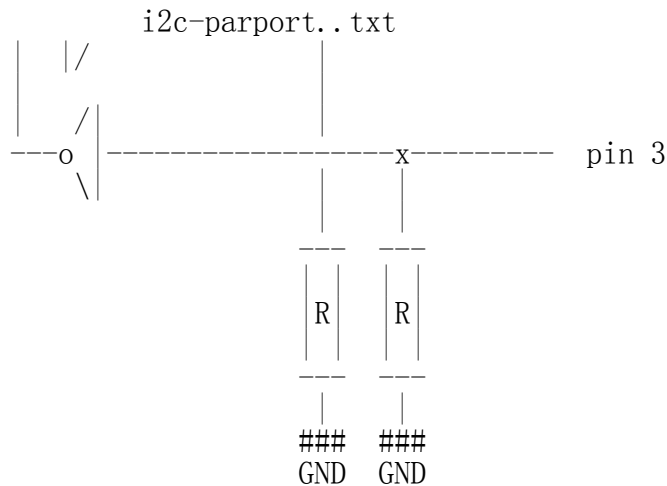
Remarks:
 - This is the exact pinout and electronics used on the Analog Devices
   evaluation boards.
                  /
 - All inverters -o |- must be 74HC05, they must be open collector output.
                  \|
 - All resitors are 10k.
 - Pins 18-25 of the parallel port connected to GND.
 - Pins 4-9 (D2-D7) could be used as VDD is the driver drives them high.
   The ADM1032 evaluation board uses D4-D7. Beware that the amount of
   current you can draw from the parallel port is limited. Also note that
   all connected lines MUST BE driven at the same state, else you'll short
   circuit the output buffers! So plugging the I2C adapter after loading
   the i2c-parport module might be a good safety since data line state
   prior to init may be unknown.
 - This is 5V!
 - Obviously you cannot read SCL (so it's not really standard-compliant).
   Pretty easy to add, just copy the SDA part and use another input pin.
   That would give (ELV compatible pinout):


```
Device                                                              PC
Side           _____Vdd (+)         Side
                |   |             |   |
               ---  ---          ---  ---
              | |  | |          | |  | |
              |R|  |R|          |R|  |R|
              | |  | |          | |  | |
               ---  ---          ---  ---
                |    |    \       |    |
SCL ----------x----x--| o---x------------------- pin 15
                     |    /       |    |
                     |            |    |
                |     /|          |    |
               --o |---------------x----------- pin 2
                |     \|          |    |
                |               |    |
                |    \         |    |
SDA -----------x---x--| o-------------x-------- pin 10
```

第 2 页

```
                      i2c-parport..txt
          |  | /                      |
          |  | /                      |
          |  |/                       |
          |  /|                       |
          |-/ |                       |
          ---o|------------------x--------- pin 3
             \|                  |    |
              |                  |    |
                                 |    |
                                ---  ---
                               |   ||   |
                               | R || R |
                               |   ||   |
                                ---  ---
                                 |    |
                                 |    |
                                ### ###
                                GND  GND
```
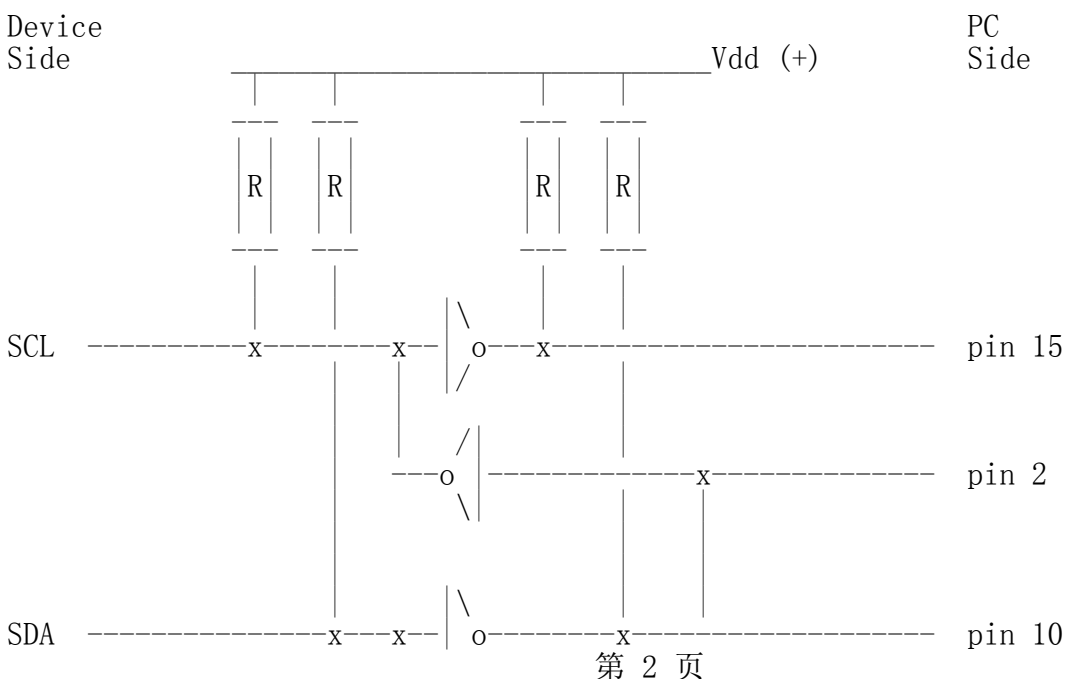
If possible, you should use the same pinout configuration as existing
adapters do, so you won't even have to change the code.

Similar (but different) drivers

This driver is NOT the same as the i2c-pport driver found in the i2c
package. The i2c-pport driver makes use of modern parallel port features so
that you don't need additional electronics. It has other restrictions
however, and was not ported to Linux 2.6 (yet).

This driver is also NOT the same as the i2c-pcf-epp driver found in the
lm_sensors package. The i2c-pcf-epp driver doesn't use the parallel port as
an I2C bus directly. Instead, it uses it to control an external I2C bus
master. That driver was not ported to Linux 2.6 (yet) either.

Legacy documentation for Velleman adapter

Useful links:
Velleman   http://www.velleman.be/
Velleman K8000 Howto   http://howto.htlw16.ac.at/k8000-howto.html

The project has lead to new libs for the Velleman K8000 and K8005:
  LIBK8000 v1.99.1 and LIBK8005 v0.21
With these libs, you can control the K8000 interface card and the K8005
stepper motor card with the simple commands which are in the original
Velleman software, like SetIOchannel, ReadADchannel, SendStepCCWFull and
many more, using /dev/velleman.
  http://home.wanadoo.nl/hihihi/libk8000.htm
  http://home.wanadoo.nl/hihihi/libk8005.htm
  http://struyve.mine.nu:8080/index.php?block=k8000
  http://sourceforge.net/projects/libk8005/

One For All JP1 parallel port adapter

The JP1 project revolves around a set of remote controls which expose
the I2C bus their internal configuration EEPROM lives on via a 6 pin
jumper in the battery compartment. More details can be found at:

http://www.hifi-remote.com/jp1/

Details of the simple parallel port hardware can be found at:

http://www.hifi-remote.com/jp1/hardware.shtml