New I2C Slave Framework (plus Runtime IP Core Switching)

Wolfram Sang / Renesas Kernel Team + Consultant

What is demonstrated

Two I2C controllers, both simultaneously master & slave on the same bus. The slave devices show:

a) register based access
(like EEPROM, data via sysfs)
b) byte read only access
(like sensor, counts seconds)

Switching at runtime between:

a) I2C (*i2c-rcar* driver)
b) IIC (*i2c-sh_mobile* driver)
c) GPIO (*i2c-gpio* driver)

IP cores are muxed to the same pins and keep stable bus numbers.

What was improved

Linux could previously not be an I2C slave device. The I2C core has been extended to pass slave events from a driver to a HW independent backend. RCar driver support, EEPROM like backend, and documentation were mainlined.

Now it is possible to chose at runtime the proper I2C core depending on current needs (e.g. DMA vs. slave). i2c-gpio can be used as fallback in case of unexpected problems.

Hardware Information

Renesas Lager board (RCar H2 SoC), two I2C busses wired together

Source code or detail technical information availability

I2C Slave: all upstream (needs driver support!)

IP core switch: RFC sent to mailing lists