



**Linaro
connect**
San Francisco 2017

Enabling the runtime PM centric path for ACPI

Ulf Hansson, Linaro





**Linaro
connect**

San Francisco 2017

ENGINEERS
AND DEVICES
WORKING
TOGETHER

Agenda

- Recap: The RPM centric path.
- The ACPI PM domain.
- The direct_complete path.
- Adapting the ACPI PM domain.
- The series got posted.
- Conclusion and next steps.





**Linaro
connect**

San Francisco 2017

Goals of RPM centric path

- Decrease system resume time of devices.
- Avoid wasting power at system resume of devices.
- Make it easy to deploy system sleep support in drivers.
- Avoid open coding in drivers.
- Simplify upstreaming of ARM SoC PM code.

ENGINEERS
AND DEVICES
WORKING
TOGETHER





Linaro
connect

San Francisco 2017

ENGINEERS
AND DEVICES
WORKING
TOGETHER

RPM centric path - how?

- **Observation:** Operations in runtime suspend and system suspend are often very similar and vice versa for resume.
- Allows the runtime PM callbacks to be re-used during system suspend/resume.
- Don't power on the device during system resume, unless really needed. Instead it postpones that to runtime PM!

*Deploy runtime PM support,
get system sleep for “free”!*



Let's deploy the RPM centric path!

mydrv.c:

```
static const struct dev_pm_ops mydrv_dev_pm_ops = {  
    SET_RUNTIME_PM_OPS(mydrv_runtime_suspend,  
                       mydrv_runtime_resume,  
                       NULL)  
    SET_SYSTEM_SLEEP_PM_OPS(pm_runtime_force_suspend,  
                             pm_runtime_force_resume)  
};
```

Yeah, we did it!





**Linaro
connect**

San Francisco 2017

ENGINEERS
AND DEVICES
WORKING
TOGETHER

Need flexibility?

- Use it from any sleep phase?
 - ->suspend(), ->suspend_late(), ->suspend_noirq().
 - ->resume(), ->resume_early(), ->resume_noirq().
- Additional operations required during system sleep?
 - Invoke pm_runtime_force_suspend|resume() from system sleep callbacks.
- Drivers deploying the RPM centric path.
 - 3.15: 1
 - 4.5: 16
 - next: ~60 consolidation/optimization.
- Find examples: “git grep pm_runtime_force_suspend”.





**Linaro
connect**

San Francisco 2017

ENGINEERS
AND DEVICES
WORKING
TOGETHER



Constraints for middle layers

- Buses and PM domains, which devices are being managed by a driver using the RPM centric path, needs to play along.
- Simple buses like, platform, spi, i2c, amba, are completely transparent. No changes needed.
- The generic PM domain easily adopted long time ago.
- Let's adopt this for other subsystems!



**Linaro
connect**

San Francisco 2017

ENGINEERS
AND DEVICES
WORKING
TOGETHER

Why the ACPI PM domain?

- Used by ARM.
 - Do we have HW that supports ACPI PM?
- Cross SoC drivers!
 - The device attached to the ACPI PM domain.
 - The device attached to the generic PM domain.
 - The device have no PM domain.
- Keep system sleep deployment simple in drivers.
- No matter of PM domain - benefit from the optimizations.





**Linaro
connect**

San Francisco 2017

ENGINEERS
AND DEVICES
WORKING
TOGETHER

The ACPI PM domain

- Uses runtime/system PM to manage the power state of a device, through the ACPI FW.
- Relies on ACPI FW, to internally manage the power state of shared resources for a group of devices.
- Supports the direct_complete path for system sleep.
- The direct_complete path?





**Linaro
connect**

San Francisco 2017

Adopting the ACPI PM domain

- Make the ACPI PM domain's runtime PM callback aware of they can be called when runtime PM is disabled.
- Adopt the ACPI PM domain's system sleep callback to trust the driver to deal with system sleep.
 - The driver needs to inform the ACPI PM domain in some way.
- Prove the behaviour in a cross SoC driver.

ENGINEERS
AND DEVICES
WORKING
TOGETHER





**Linaro
connect**

San Francisco 2017

Suggested approach - rejected!

- Converting i2c-designware-platdrv.
 - 1 file changed, 5 insertions(+), 29 deletions(-)
- Series [1] submitted and re-spinned.
 - 7 files changed, 204 insertions(+), 101 deletions(-)

[1]

<https://www.spinics.net/lists/arm-kernel/msg603856.html>

ENGINEERS
AND DEVICES
WORKING
TOGETHER





**Linaro
connect**

San Francisco 2017

ENGINEERS
AND DEVICES
WORKING
TOGETHER

Conclusion and next steps

- The `direct_complete` path isn't good enough.
- It will get worse when more cross SoC drivers hits the same problems.
- We and the PM core maintainer are working on a new method to enable the ACPI PM domain.
- Until then:
 - Non-optimized system sleep support for drivers dealing with devices attached to the ACPI PM domain.





**Linaro
connect**
San Francisco 2017

Thank You

#SFO17

SFO17 keynotes and videos on: connect.linaro.org

For further information: www.linaro.org

