Connect 2025

# Multiple system-wide low power-states

LIS25-129 Ulf Hansson, Linaro KWG ulf.hansson@linaro.org

### Agenda

- Background
- Use cases
- A possible solution
- Next phase





## System wide sleep

- Triggered from userspace
  - Closing the lid on your laptop
  - The power-button on your TV-remote
- Triggered internally by the kernel
  - *Autosleep* used by Android to enforce system sleep

## System wide sleep - sysfs

- echo [state] > /sys/power/state
  - **freeze** == Suspend-to-Idle == ACPI SO
  - standby == ACPI S1
  - **disk** == Hibernation
  - o mem == Based on /sys/power/mem\_sleep
- echo [state] > /sys/power/mem\_sleep
  - **s2idle** == Suspend-to-Idle
  - **deep** == Suspend-to-RAM
  - **shallow** == "standby"
- echo [state] /sys/power/autosleep
  - State available in /sys/power/state
  - **off** == Disable Autosleep

Documentation/admin-guide/pm/sleep-states.rst

Extend to support multiple platform-specific low power states?

No, doesn't scale!

linaro Connect **Typical use cases** 





S2I - Suspend-to-Idle S2R - Suspend-to-RAM

### Summary - S2R vs S2I

- S2R requires the boot CPU to be the last/first adds latency
- S2R with PSCI is limited to support only one low power state
- S2I has less latency than S2R
- S2I with PSCI allows multiple low power states



## **#1 PM co-processor for peripheral devices**



- Controls power-domains (SCMI power) for peripheral devices
- A generic PM domain provider (genpd) with multiple low power states

Problem:

• The genpd selects the deepest state for S2I/S2R - breaking the system wakeup latency constraint

#### **#2 PSCI on Automotive**



- Multiple low power states are supported by PSCI
- Both of the PSCI CPU suspend modes (OSI/PC).

#### **Problems:**

• Genpd/CPUIdle selects the deepest state for S2I - breaking the system wakeup latency constraint

## A possible solution

- A new genpd governor for S2I/S2R to select the state
- A new CPUIdle governor for S2I to select the state
- A system wide latency constraint cpu\_dma\_latency exists
- Extend DT bindings for idlestates to describe system-wide states



#### The next steps

- Other use cases
- Implementation continues
- Platform deployment and testing
- TI (Baylibre) and QC collaborates with us

Reach out to us to get involved!



# **Thank You!**