Fast CPU DVFS using ARM SCMI firmware interface

Linux Plumber Conference, 2017

Sudeep Holla <sudeep.holla@arm.com>

September 14, 2017
Why is fast switching not possible on ARM platforms?

- Most ARM platforms have external power management ICs (PMIC) to provide CPU DVFS.
- PMICs are generally connected with a slow SPI/I2C communication interface.
- PMICs are also sometimes multi-function device (MFD) providing other functionality requiring serialized access.
- I2C/SPI bus may also be shared amongst other slave devices.
- Difficult to achieve fast switching if Linux drives these I2C/SPI bus or PMIC.
- Recent trend is embedded microcontrollers in systems to abstract various power and other system management tasks.
Standard firmware interface ... really?

- PSCI covers only CPU power management but not CPU performance or peripheral device management.
- SCMI (System Control and Management Interface) is an extensible interface covering performance, power and various other system management functions.
- Builds on strong trend in the industry towards embedded platform microcontroller.
So, can we do fast switching using SCMI?

- Operate in a fire and forget fashion
- Not need to handle interrupts/notifications
- Just poll until remote receives DVFS request
- But... is that loss in accuracy?
- Counters like x86 aperf/mperf may help!
So, can we do fast switching using SCMI?

- Operate in a fire and forget fashion
- Not need to handle interrupts/notifications
- Just poll until remote receives DVFS request
- But... is that loss in accuracy?
- Counters like x86 aperf/mperf may help!

References to patches/discussions
- Frequency invariance support by Dietmar
- SCMI support
- SCMI Fast Switch support
Activity Monitors/Counters

- OK we have some aperf/mperf like counters
  - so what’s the problem?
- Fragmented!
  - we have variety of them
- Counters may not be clocked at same rate
- Not yet architected
- SCMI has some DVFS statistics support
  - the accuracy and timing may not be sufficient