Power Management System in OSes for IoT

Xiaoyu MIN
EIT ICT Labs Embedded System
Aalto University, School of Science
Contents

• Importance of EPM in IoT
• Application-level EPM
  – Android wake lock for Applications
• OS-level EPM
  – ACPI
  – Features in Mobile OS
• Hardware EPM
  – PMIC (Power Management Integrated Chip)
Need for improved power management

- Capabilities of mobile handheld devices improving rapidly:
  - Wireless connectivity
  - Storage capacity

- Battery capacity improving slowly
  - I/O devices decrease handheld battery lifetime by 60%
  - OEMs provide techniques to reduce battery consumption by severely compromising the performance
  - Careful power management needed
Power Management is essential in IoT Network

• Sensor network are often designed for reliable real-time services in IoT
  – Limited resource and strict power consumption
  – Additional limitation towards some characteristics of conventional OS

• Nodes are designed to operate with limited resources
  – Power: WSN based use batteries as a power supply
  – Memory and operational capabilities: sensing is less resource demanding than computation in conventional OS

• Power management is very important in IoT
  – Various OS designed for IoT devices: TinyOS, Contiki OS, Android etc.
Applications

Application A
Application B
Application C

WI = newWakeLock(...);
WI.acquire();
WI.release();

Applications Framework

PowerManager
Android.os.PowerManager

Power
Android.os.Power

PowerManagerService
Android.server.PowerManagerService

Libraries (user space)

Core Libraries

Power
/lib/hardware/power.c

Linux Kernel

Linux Drivers

Android Power Management
/drivers/android/power.c

Android_register_early_suspend()
Android_register_early_resume()

Linux Power Management

JNI
Android Wake lock Mechanism

• Android implements an application framework on top of the kernel called Android Power Management Applications Framework

• The Android PM Framework is like a driver. It is written in Java which connects to Android power driver through JNI

• Currently Android only supports screen, keyboard, buttons backlight, and the brightness of screen

• Through the framework, user space applications can use ‘PowerManger’ class to control the power state of the device
Android Wake lock Mechanism
ACPI Specifications

• Advanced Configuration and Power Interface (ACPI) specification provides an open standard for device configuration and power management by the operating system.

• Power States
  – Global states
  – Device states

• Processor states

• Performance states
ACPI Specifications

- ACPI specification defines the following four Global ‘Gx’ states and six Sleep ‘Sx’ states for an ACPI-compliant computer-system:

- Legacy State: The state on an operating system which does not support ACPI. In this state, the hardware and power are not managed via ACPI, effectively disabling ACPI.

- G0 (S0)
  - Working
  - ‘Awaymode’ is a subset of S0, where monitor is off but background tasks are running

- G2 (S5), Soft Off

- G3, Mechanical Off
  - The computer's power has been totally removed via a mechanical switch
ACPI Specifications

• G1, Sleeping, subdivides into the four states S1 through S4:
  – S1: All processor caches are flushed, and the CPU(s) stop executing instructions. Power to the CPU(s) and RAM is maintained; devices that do not indicate they must remain on may be powered down
  – S2: CPU powered off. Dirty cache is flushed to RAM
  – S3(mem): Commonly referred to as Standby, Sleep, or Suspend to RAM. RAM remains powered
  – S4: Hibernation/Suspend-to-Disk - All content of main memory is saved to non-volatile memory such as a hard drive, and is powered down
Power management for TinyOS & Contiki

- **TinyOS**
  - Scheduler responsible for power management
  - Power management
    - Radio power management
    - Managing power of sensor nodes
  - Power-Save mode
    - Low-power mode (e.g. TinyOS Timer)

- **Contiki OS**
  - No standard mechanisms for managing the power state of peripheral devices
  - Power optimizations
    - Micro-controller in a sleep mode
    - Power estimation as additional feature
Power management integrated circuit

- Power management integrated circuits (power management ICs or PMICs) are integrated circuits (or a system block in a system-on-a-chip device) for managing power requirements of the host system.
- A PMIC is often included in battery-operated devices such as mobile phones and portable media players.
- A PMIC may include the following parts:
  - Battery management
  - Voltage regulations
  - DC to DC conversion (dynamic voltage scaling)
  - Power sequencing
  - Miscellaneous functions
Dynamic frequency scaling

- Also known as **CPU throttling**, whereby the frequency of a microprocessor can be automatically adjusted “on the fly”, either to conserve power or to reduce the amount of heat generated from the chip.

- **Dynamic voltage scaling** is another power conservation technique that is often used in conjunction with frequency scaling, as the frequency that a chip may run at is related to the operating voltage.

- According to the ACPI Specs, the C0 working state of a modern-day CPU can be divided into the so-called "P"-states (performance states) which allow clock rate reduction and "T"-states (throttling states) which will further throttle down a CPU (but not the actual clock rate) by inserting STPCLK (stop clock) signals and thus omitting duty cycles.
Power management integrated circuit

- Some models feature a low-dropout regulator (LDO), and a real-time clock (RTC) co-operating with a backup battery.
- A PMIC can use pulse-frequency modulation (PFM) and pulse-width modulation (PWM). It can use switching amplifier (Class-D electronic amplifier).
- Most of the PMIC are integrated with the processor design and they are provided by the chip manufacturers.
Further References

- Chameleon: Application Level Power Management with Performance Isolation, Xiaotao Liu, Prashant Shenoy, Mark Corner
- Ghosts in the Machine: Interfaces for Better Power Management, Manish anand, Edmund B. Nightngale
- Embedded Linux Optimizations, Michael Opdenacker, Thomas Petazzoni, Free Electrons
- Android Power Management, Jerrin Shaji George
- The Case for Higher-Level Power Management, Carla Schlatter Ellis
- Hierarchical Power Management with Application to Scheduling, Peng Rong, Massoud Pedram
- Power Management in Mobile Devices, Findlay Shearer
- Etc..
Thank you!