Embedded Linux Conference
April 29-May 1, 2014, San Jose, CA

Use-Case Power Management Optimization

Identifying & Tracking Key Power Indicators

Patrick Titiano,
System Power Management Expert,
BayLibre co-founder.
www.baylibre.com

Problem Statement

- Wireless Embedded platforms performances keep increasing
  - Multi-core processors (MPU / GPU) up to 2GHz+, H/W accelerators
  - High-Speed RAM (LPDDR3, Wide I/O) & peripheral buses (USB3)
- But power and thermal budgets remain roughly the same
  - Mobile phone: ~5W, case temperature < 45ºC, 1-day of active use
- => Power Management becomes the critical element.

What’s on the menu today?

- No meat, no fish, only power management stuff ☺
  - Starter
    - Critical Key Performance Indicators (KPI)
  - Main dish
    - Use-Case PM Optimization Methodology
      - Stuffed with practical examples & Thermal Management considerations
  - Dessert
    - Final Thoughts & Recommendations

Critical KPI (Key Power Indicators)

Statistics profiling platform activities, relevant to Power Management

Running Clocks

- `# cat /sys/kernel/debug/clock/summary`

```
ocp_abe_iclk                   aess_fclk                   98304000   27
per_abe_nc_fclk                dpll_abe_m2_ck              98304000   0
div_ts_ck                      l4_wkup_clk_mux_ck          1200000    1
l4_wkup_clk_mux_ck             sys_clkin_ck                38400000   6
lp_clk_div_ck                  dpll_abe_m2x2_ck            12288000   0
l4_div_ck                      l3_div_ck                   100000000  62
l3_div_ck                      div_core_ck                 200000000  47
dpll_mpu_ck                    sys_clkin_ck                700000000  1
```

- Tracks running power resources
  - Clocks, DPLL, power switches, voltage regulators, …
- Highlight unnecessary running clocks & resources
  - Root cause of power switch(es) & voltage regulator(s) maintained ON
    - HW dependencies

C-States (Idle States) Statistics

-
  ```
  # cat /sys/devices/system/cpu/cpu0/cpuidle/state*/usage
   208814669
   1124298
   2263801
   # cat /sys/devices/system/cpu/cpu0/cpuidle/state*/time
   133059448774
   3700489912
   9190480361
  ```

- Low-power state = C-State in CPUIdle Linux Framework terminology
  - From ON (C0) to OFF (Cmax) states, through various clock gating/retention states
    - From NO (C0) to MAX (Cmax) power savings
    - From MIN (C0) to MAX (Cmax) sleep/wakeup latencies

- Highlight
  - Cumulated time spent & transitions into each low-power states
  - How much & deep CPU is able to sleep

Operating Point (OPP) Statistics

- `# cat /sys/devices/system/cpu/cpu0/cpufreq/stats/time_in_state`

```
350000 1086631
700000 410910
920000 13505
1200000 401071
# cat /sys/devices/system/cpu/cpu0/cpufreq/stats/total_trans
132618
```

- Highlight
  - Cumulated time spent in each OPP (pre-defined [MHz/V] set)
  - Cumulated number of OPP transitions

- Assess processing requirements (low/medium/high MHz)
- Assess processing profile (bursty vs smooth)
- Monitor thermal management throttling (highest OPP skipped)

CPU & HW Accelerators Loads

CPU: `# cat /proc/stat`

```
cpu  7465 358 8079 3748103 6510 125 3458 0 0 0
cpu0 3713 158 4943 1868346 2681 125 3450 0 0 0
cpu1 3752 200 3135 1879757 3829 0 8 0 0 0
…
```

Other HW acc. (GPU/DSP/ISP/…):

```
proprietary / not standard instrumentation ☹
```

- Highlight
  - Processing scheduling over time
  - Processing requirements (low / medium / high / … MHz)
  - Most demanding applications – services / performance bottleneck
  - Source of lags, low frame rate, unresponsiveness, …

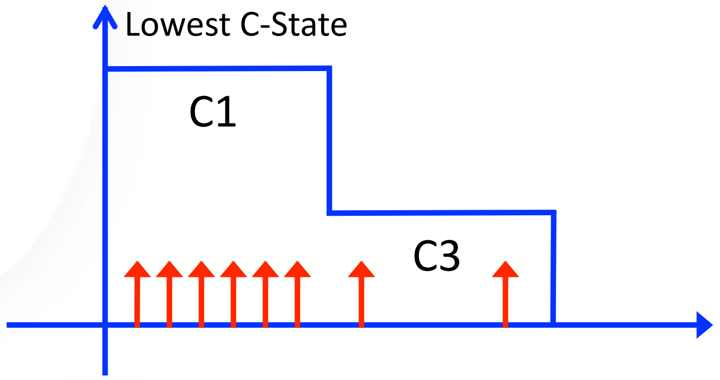
Memory Bandwidth

- Usually HW / Proprietary non-standard instrumentation ☹
- Track memory / bus occupancy
- Data bus load (MB/s)
- Memory / Bus latencies
- Highlight
  - Potential root cause of lags, low frame rate, unresponsiveness, …

Interrupts

```
# cat /proc/interrupts
       CPU0       CPU1
  39:       6          0      GIC  TWL6030-PIH
 213:   22218          0     GPIO  wl1271
 393:       0          1  twl6040  twl6040_irq_ready
IPI1:   22086      94686  Rescheduling interrupts
IPI3:   68462      59269  Single function call interrupts
 LOC:  816383     411488  Local timer interrupts
```

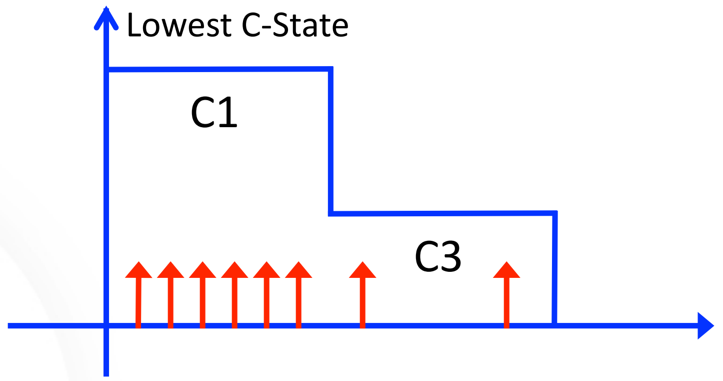
- Track peripheral activities over time



- Highlight
  - Unexpected interrupt sources / rates
  - Potential root cause of reduced usage of CPU low-power states
  - Potential root cause of High latency / performance degradation

Timers

```
# cat /proc/timer_stats
# cat /proc/timer_list
```



- Track CPU wakeup sources and rates
- Highlight
  - Unnecessary CPU wakeup sources
  - Potential root cause of reduced usage of CPU low-power states
  - Potential root cause of High latency / performance degradation

Temperatures

- `# cat /sys/devices/platform/omap/omap_temp_sensor.0/temperature`

  NB: platform-dependent!

- Track various temperature sensors
- CPU, GPU, PCB, SDRAM, case, ...

- Highlight power and performance degradation due to over-heating / thermal management throttling
- Power consumption increases a lot (explodes?) with temperatures
  - Thermal runaway

Use-Case PM Optimization: Proposed Methodology

Use-case: interactions within platform to accomplish a goal.

Modelize

- Define critical use-cases for your platform
  - MP3, AV-Payback, 3D Gaming, Capture, Idle, Voice-call, Web Browsing …
- Create a power model of your platform
  - MPU / GPU / Bus / Memory / Peripherals power consumption
    - Static (leakage), Dynamic (= f(MHz)), temperature
- Create a power model of targeted use-cases
  - Split use-case into simple atomic functions (slices)
    - Required peripherals, processing loads and profiles, memory / bus bandwidth, data transfer sequence diagram, …
  - Must be measurable onboard
- Define power targets and thermal budget per use-case
  - Generated from power model

Instrument

- SW
  - Kernel
  - Power Management Frameworks
  - Scripts to reproduce use-cases
  - User-space tools to collect and process power data
  - See omapconf example

- HW
  - Lab equipment with high-resolution current probes
  - Sense resistors to measure current &voltage *simultaneously*
  - Temperature sensors (embedded, external)
  - HW trace
  - Embedded power measurement capabilities is a plus

Automate

- Apples must be compared to apples
  - Power, voltages, currents are analog variables
    - Inherent variations in measurements
  - Measurements should be repeated and averaged before analysis

- Long, annoying, **approximate & source of error** if not automated!
  - Bad practices (real-life ☹☹☹) examples:
    - Power consumption of 10 different rails for 10 different use-cases reported by hand for measurement equipment to test report
    - Boot time measured with a simple watch

Characterize Silicon raw performances

- Raw Leakage current & dynamic consumption (mA / MHz / V)
  - I/O
  - Low-power Retention states
  - CPU (Dhrystone, …), GPU (GLBench, …), other processing unit(s)
  - Bus
  - Cache, RAM
  - Peripherals
  - Temperatures

- To assess power model and power targets
  - Based on estimated Silicon power performances
  - Consider process corners / worst-case

Assess Power Model

- Compare raw Silicon power performances to estimates
- Refine power model with raw Silicon power performances measurements until converged
- Re-generate power targets accordingly

Measure use-cases

- Take multiple measurements of a same use-case
- Check that all measurements are in a same ballpark
  - Not exceeding ±5%
  - Example: 3 samples of a same use-case showing 50% to 100% variation between measures
    - Bad practice: report the average value (real!)
    - Good practice: report issue with the measurement setup
- Collect and save all useful KPI statistics, for further analysis.

Analyze KPI for Leakage

- Power Consumption is made of static (a.k.a. leakage) and dynamic power consumption

- Analyze Static Power Consumption (a.k.a. leakage) always first

- Ensure no power is wasted
  - Supplied Voltages
  - Miss-configured I/O
    - Unused I/O not in high-impedance state, short-circuit
    - Bad pull-up /pull-down configuration:
      - Dual (at each end), combined up + down, unnecessary
  - Running clocks / DPLL instead of idle
  - Unused logic powered ON / not retained
  - Unused Voltage regulators left ON
  - Low-power states usage / Idle policies
  - SDRAM: self-refresh / power-down / other IP-specific power features

Analyze KPI for Dynamic Consumption

- Once leakage is under control, chase for extra processing / bottlenecks

- CPU / GPU / HW Accelerators
  - Supplied Voltages
  - Processes, timers, interrupts, sleep durations & levels
  - Processing loads (and profiles) vs estimations
  - CPU IPC performances (latencies, rates)
  - OPP statistics / DVFS and idle policies efficiencies
  - Cache efficiency

Analyze KPI for Dynamic Consumption

- Bus / SDRAM
  - Supplied Voltage
  - Assess loads vs estimations
  - Assess latencies
  - Assess idle duration
  - SDRAM: refresh cycle rates, …

Analyze Temperature

- Keep temperature within expected limits for a given use-case
  - Fine-tune DVFS policies
  - Shutdown unnecessary logic

  - Heating increases power consumption

- Heating degrades performances
  - CPU/GPU throttling

Fix!

- Code
- Power Model
- Iterate until targets and measurements converge
  - Discuss (negotiate ;-)) with architects and developers
    - Implementation? Power Estimations? Both?
  - Set an acceptable limit
    - Usually power targets cannot be reached or exceeded
    - Define when to stop optimization

Track

- Do not let power diverge again!
  - Monitor power consumption over new releases until the end of the development life-cycle
  - Be strong, reject patches hurting power
    - The same way patches hurting performances and stability are.
- Yes, you’re never done! ☺
  - Tracking phase should be fully automated, ultimately.

Use-Case Analysis

3D Wallpaper Example

Omapconf

- Public Linux user-space standalone application
  https://github.com/omapconf/omapconf

- Designed to provide a quick 'n easy power/performance runtime diagnostics
  - => KPI analysis

- Omapconf use-case KPI automated audits used to illustrate low-power 3D use-case example:
  - ‘Water’ Android Live Wallpaper on Panda AOSP 4.2.2 platform

Use-case Analysis: 3D Wallpaper

```
root@panda:/ # omapconf audit lp3d

|---------------------------------------------------------------------------|
| 3D Low-Power Power Settings Audit | Current      | Expected (POR) | STATUS |
|---------------------------------------------------------------------------|
| CPUFreq Governor                  | interactive  | interactive    | Pass   |
| OPPs                              |              |                |        |
|   VDD_MPU                         | OPP_NITRO    | OPP50          | FAIL   |
|   VDD_IVA                         | OPP50        | OPP50          | Pass   |
|   VDD_CORE                        | OPP100       | OPP50          | FAIL   |
| Voltages                          |              |                |        |
|   VDD_MPU                         | 1.200000 V   | <= 1.025000 V  | FAIL   |
|   VDD_IVA                         | 0.835600 V   | <= 0.962200 V  | Pass   |
|   VDD_CORE                        | 1.063480 V   | <= 0.962200 V  | FAIL   |
| RETENTION Voltages                |              |                |        |
|   VDD_MPU                         | 0.830000 V   | 0.750000 V     | FAIL   |
|   VDD_IVA                         | 0.835600 V   | 0.759640 V     | FAIL   |
|   VDD_CORE                        | 0.835600 V   | 0.759640 V     | FAIL   |
| Clock Speeds (4)                  |              |                | FAIL   |
|---------------------------------------------------------------------------|
```

Use-case Analysis: 3D Wallpaper

```
|----------------------------------------------------------------------------|
| 3D Low-Power Power Settings Audit      | Current       | Expected (POR) | STATUS |
|----------------------------------------------------------------------------|
| Lowest C-State entered                 | C1            | C1             | Pass   |
| Power & Clock Domains State (2)        |               |                |        |
|   DSP (DSP)                            | RET (Gated)   | OFF (Gated)    | FAIL   |
|   GFX                                  | ON            | ON             | Pass   |
| SYSCONFIG Settings (4)                 |               |                | FAIL   |
| DPLLs Status (4)                       |               |                |        |
|   ABE                                  | Stopped       | Locked         | FAIL   |
|   CORE                                 | Locked        | Locked         | Pass   |
|----------------------------------------------------------------------------|

|-------------------------------------------------------------|
| 3D Low-Power Audit Metrics              | Count | Breakout (%) |
|-------------------------------------------------------------|
| Number of tests run                     | 51    | 100%         |
| Number of tests passed                  | 35    | 68.63%       |
| Number of tests failed                  | 16    | 31.37%       |
|-------------------------------------------------------------|
```

Use-case Analysis: 3D Wallpaper

```
|-----------------------------------------------------------------------------|
| CLOCK SPEED AUDIT       |             |        Clock Rate (MHz)   |          |
| Module | Source Clock | OPP          | Current        | Expected     | STATUS   |
|-----------------------------------------------------------------------------|
| MPU    | MPU_DPLL_CLK | OPP_TURBO    | 920.000        | 920.000      | pass     |
| GFX    | GFX_FCLK     | OPP100       | 307.200        | 307.200      | pass     |
| DISPC  | DSS_FCLK     | OPP100       | 153.600        | 170.667      | FAIL     |
| EMIF1  | EMIF_L3_ICLK | OPP100       | 200.000        | 200.000      | pass     |
|-----------------------------------------------------------------------------|

|-----------------------------------------------------------------------------|
| MODULE SYSCONFIG AUDIT | AUTOIDLE | IDLE                 | STANDBY           |
|-----------------------------------------------------------------------------|
| DSI1                   | Pass     | Pass                 |                   |
| HDMI                   |          | Warning (Smart-Idle) |                   |
| RFBI                   | Pass     | Warning (Force-Idle) |                   |
| GFX                    |          | Pass                 | FAIL (Reserved)   |
| HSI                    | NA       | NA                   | NA                |
|-----------------------------------------------------------------------------|
```

Use-case Analysis: 3D Wallpaper

```
root@panda:/ # omapconf audit perf 15 -d 3
|--------------------------------------------------------------------------|
| C-State        | Entered? | Hit Number | Time Spent (s) | Time Spent (%) |
|--------------------------------------------------------------------------|
| C0             | Yes      |            | 9s 181ms 413us | 61.2%          |
| C1             | Yes      | 5764       | 5s 818ms 587us | 38.8%          |
| C2             | No       |            |                |                |
| C3             | No       |            |                |                |
|--------------------------------------------------------------------------|
CPUFreq Governor: interactive
Total number of OPP transitions: 164
|--------------------------------|
| MPU OPP    | Time Spent in OPP |
|--------------------------------|
| OPP50      | 3s230ms           |
| OPP100     | 5s730ms           |
| OPP_TURBO  | 2s610ms           |
| OPP_NITRO  | 3s430ms           |
|--------------------------------|
```

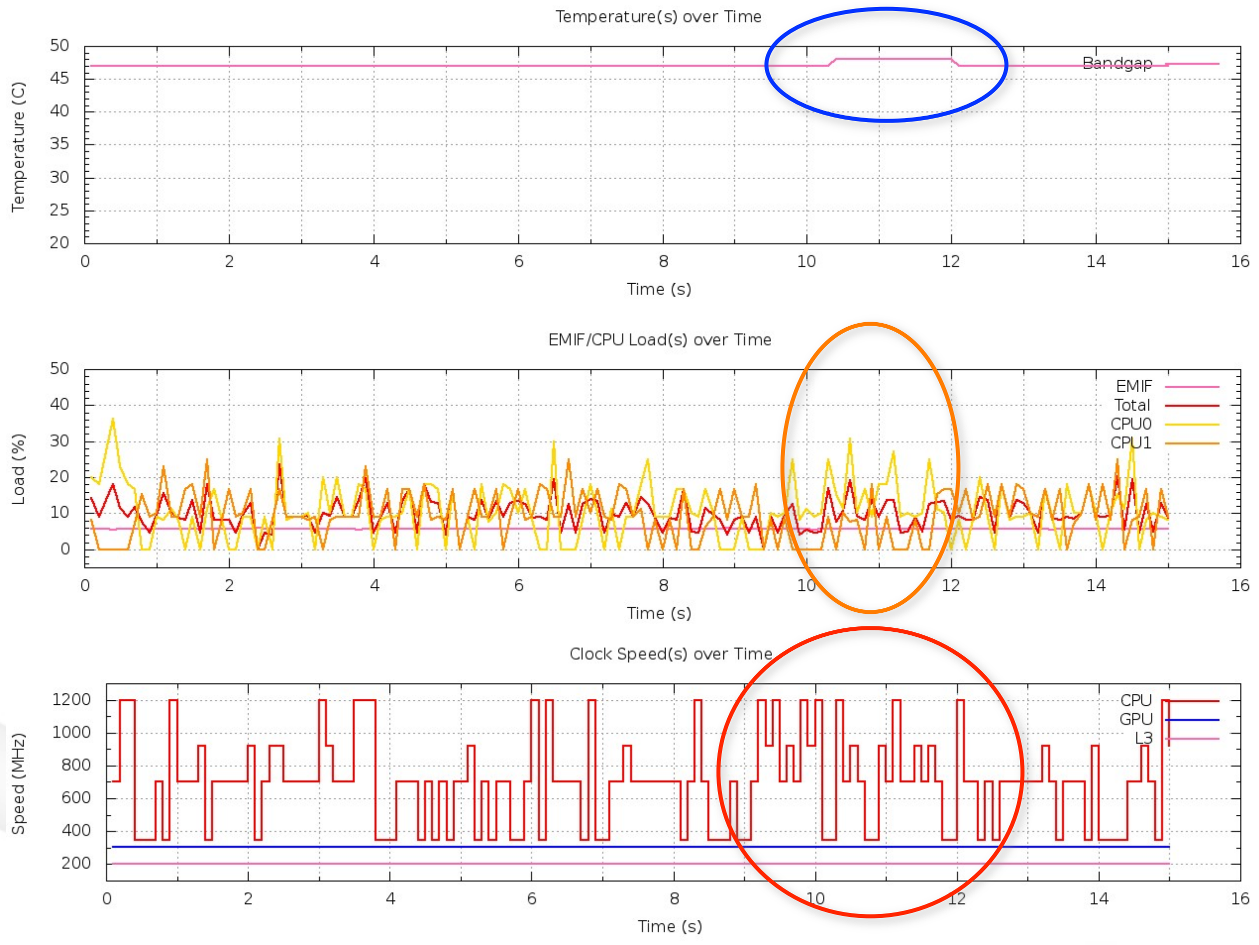
Use-case Analysis: 3D Wallpaper

```
|---------------------------|
| CPU    | Average Load (*) |  (*) CANNOT be converted to Mhz. OPP may have changed during the audit.
|---------------------------|
| CPU0   | 6.34%            |
| CPU1   | 2.35%            |
| Total  | 4.34%            |
|---------------------------|

CPU was interrupted 5281 times by the following 7 sources:
|------------------------------------------------------------|
| IRQ # | Device Name | Occurrence | Proportion | Rate        |
|------------------------------------------------------------|
| 69    | gp timer    | 1922       | 36.4%      | 128.1/sec   |
| 53    | SGX ISR     | 1837       | 34.8%      | 122.5/sec   |
| 57    | OMAP DISPC  | 1060       | 20.1%      | 70.7/sec    |
| 91    | mmc1        | 211        | 4.0%       | 14.1/sec    |
| 44    | DMA         | 193        | 3.7%       | 12.9/sec    |
| 115   | mmc0        | 40         | 0.8%       | 2.7/sec     |
| 213   | wl1271      | 18         | 0.3%       | 1.2/sec     |
|------------------------------------------------------------|
```

Use-case Analysis: 3D Wallpaper

```
root@panda:/ # omapconf trace perf -d 3 -t 15

|----------------------------------------------------------------------|
| Performance Statistics     | Min     | Max       | Average | Online Time |
|----------------------------------------------------------------------|
| CPU Frequency              | 350MHz  | 1200MHz   |         |             |
| CPU0 Load                  | 0.00%   | 36.36%    | 10.57%  |             |
| CPU1 Load                  | 0.00%   | 25.00%    | 9.44%   |             |
| CPU1 Online Time           |         |           |         | 100.00%     |
| Total CPU Load             | 0.00%   | 23.72%    | 10.01%  |             |
|                            |         |           |         |             |
| GPU Frequency              | 307MHz  | 307MHz    |         |             |
|                            |         |           |         |             |
| L3 Frequency               | 200MHz  | 200MHz    |         |             |
| Total EMIF Data Bus Load   | 5.53%   | 5.92%     | 5.70%   |             |
|                            |         |           |         |             |
| Bandgap Temperature        | 47C     | 48C       | 47.11C  |             |
| PCB Temperature            | NA      | NA        | NA      |             |
|----------------------------------------------------------------------|
```

Use-case Analysis: 3D Wallpaper



Conclusion

Final Recommendations

Anticipate

- Chips and boards shall be designed for power measurement
  - Accessible probe points on voltage rails
  - Use 0-ohm resistor as placeholders to be replaced by sense resistors
  - Design power companion chip with
    - Embedded power sensors
  - HW debug logic to trace power states & transitions
    - Ultimately synchronized with SW markers
- SW shall be instrumented for tracing power management decisions

Partition HW for Power

- Do not build house with a single light switch
  - Use Dedicated clock switch for every peripheral
  - Peripherals grouped per use-case under power switch(es)
  - Avoid sharing scalable voltage regulator(s)
  - Use retention techniques to reduce sleep/wakeup latencies
- Voltage is *KEY*
  - Power is proportional to the square of V
    - $P = a * C * V^2 * f$

Fine-Tune Policies

- The perfect policy does not exist
  - Default policies cannot perform nicely for all use-cases
- Default Linux upstream policies made for desktops & servers, not embedded devices
  - Fine-tune parameters for critical use-cases
  - Develop your own policies
- Do not hesitate to detect use-case & switch policies on the fly

Keep Temperatures Down

- "Easier" to waste less power than find mechanical solutions to dissipate more power
  - Embedded devices are not desktop PC or servers
    - No fan, only a case … and your skin …

- Power consumption increases with temperatures

- Minimize use of performance throttling

Battery is what really matters

- Final goal is to optimize power consumption at battery level
  - Focus attention on main contributors
    - No need to save 30% of power on a rail that only accounts for 2% total
  - Think system-wide, pay attention to side-effects
    - Doing a power optimization on one end may degrade it at another end
    - E.g.: reducing clock rates may lengthen active time and increase DPLL lock time

Q & A

Thank you!