Phase-Aware Web Browser Power Management on HMP Platforms

Nadja Peters[1], Sangyoung Park[1], Daniel Clifford[2], Sami Kyostila[2], Ross McIlroy[2], Benedikt Meurer[2], Hannes Payer[2],

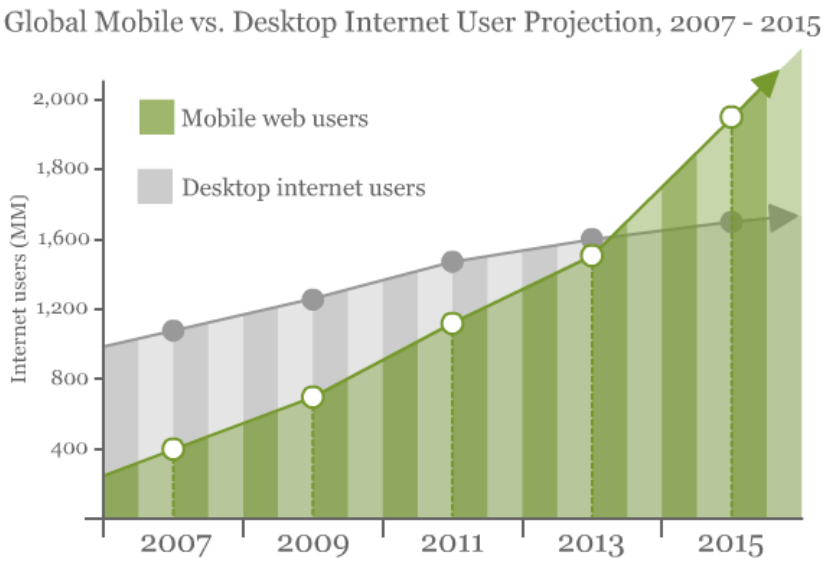
Samarjit Chakraborty[1]

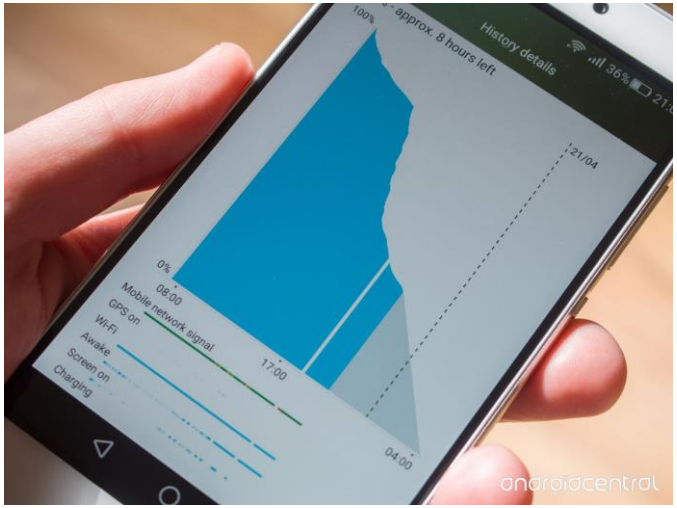
Technical University of Munich, Chair of Real-Time Computer Systems[1]

Google Inc[2]

Introduction

Our goal is to maintain a good web browsing experience while preserving the runtime of the smartphone's battery.

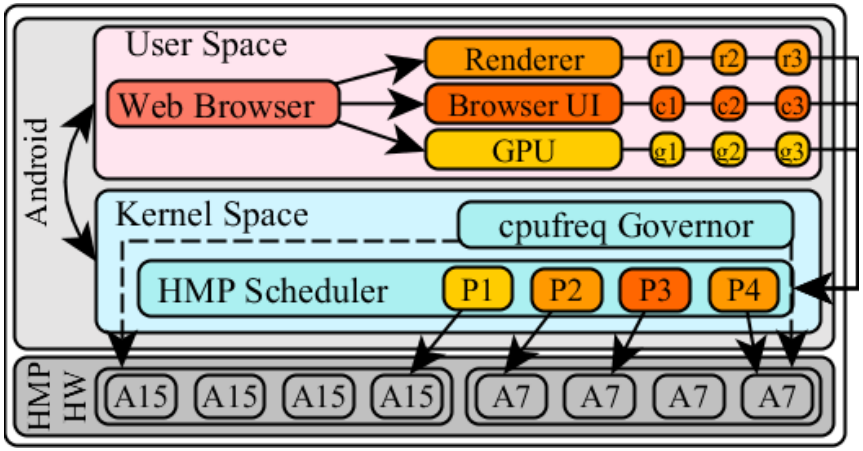




Sources: www.pixabay.com, www.dreamstime.com
www.androidcentral.com, Morgan Stanley.

State-of-the-Art HMP HW Platform

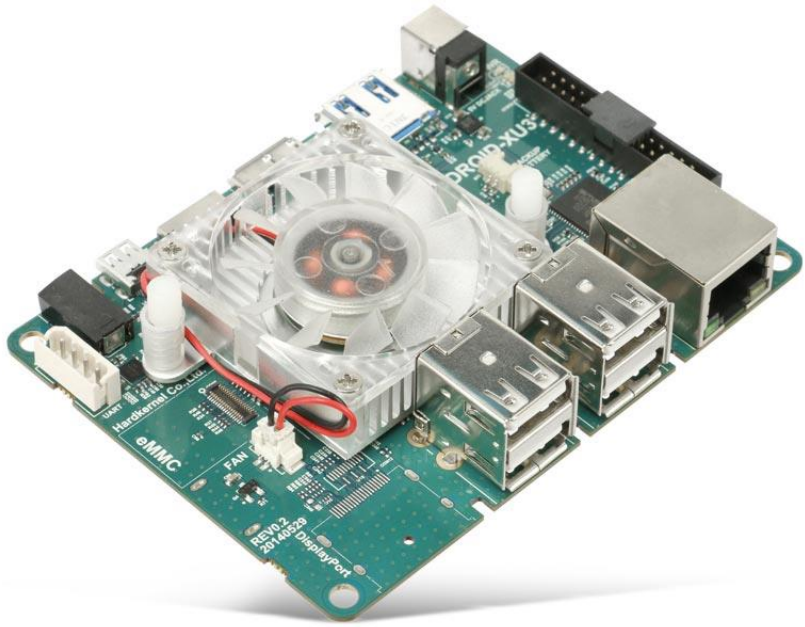
Odroid-XU3 big.LITTLE HMP platform with Exynos5422 SoC (Galaxy S5)



Schematic view of the Odroid-XU3's SoC and power management..

Android power management entities:

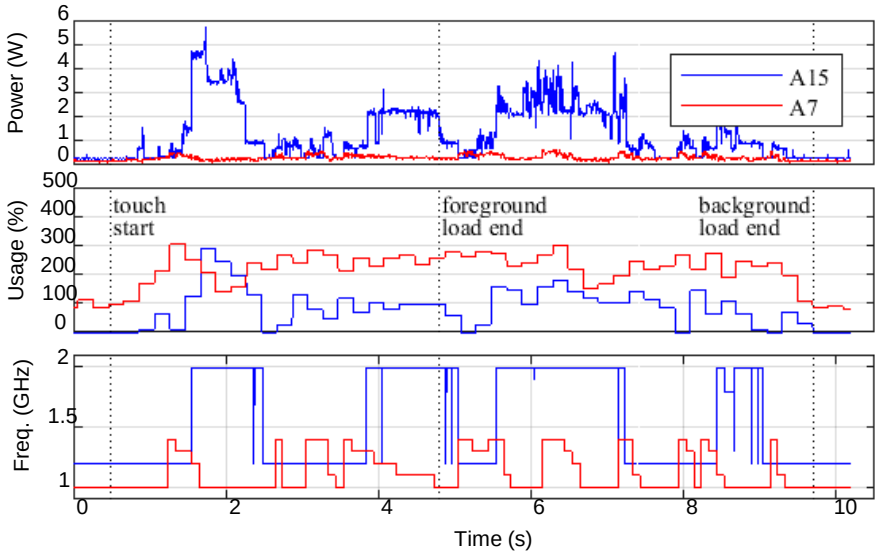
- Per-CPU:
  - Frequency governor
  - Power state control
  - Task scheduler
- Global :
  - HMP scheduler for all CPUs



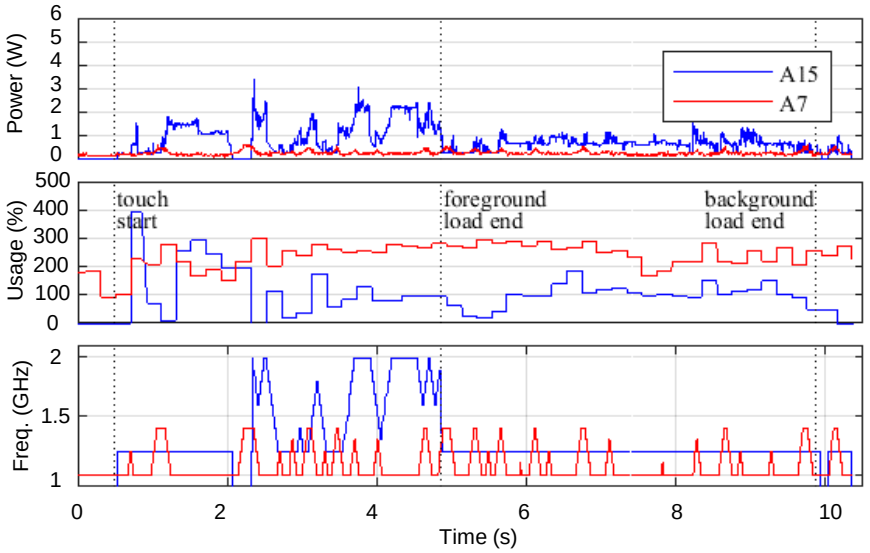
Odroid-XU3 board.
Source: www.hardkernel.com

Motivation

Ondemand Governor



Browser Governor

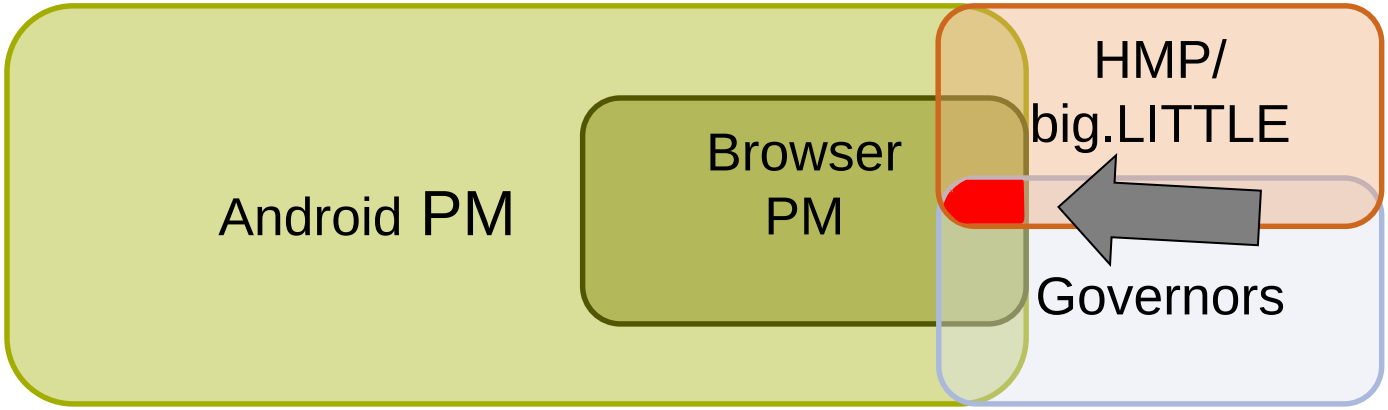


Loading a Reddit web page with the Android default governor (right) versus the proposed Browser governor (right, 40% less energy).

CPU governor frequency regulation:

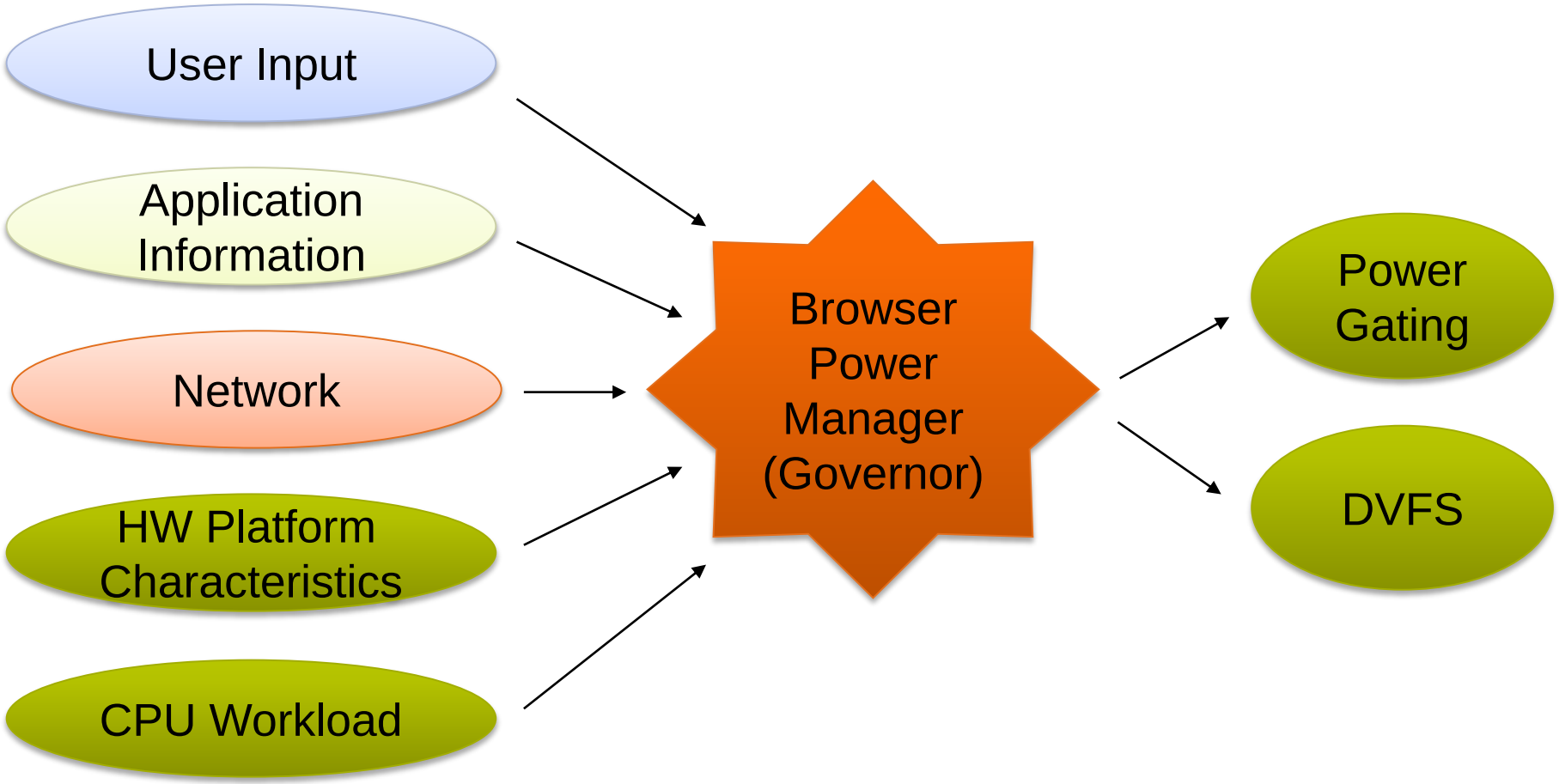
- Ondemand: Workload-based, no power-gating
- Browser: Application-specific information **and** workload-based
  - ➢ Differentiate between high- and low priority workload

Contributions of this Paper



- Present the effectiveness of using **application-specific information** available directly from the browser
- **Establish a communication channel** between kernel and application layer
- **Implement** the *Browser Governor* for Google Chrome and **evaluate** on a real hardware platform

Browser Governor Overview



Performance Requirements: RAIL

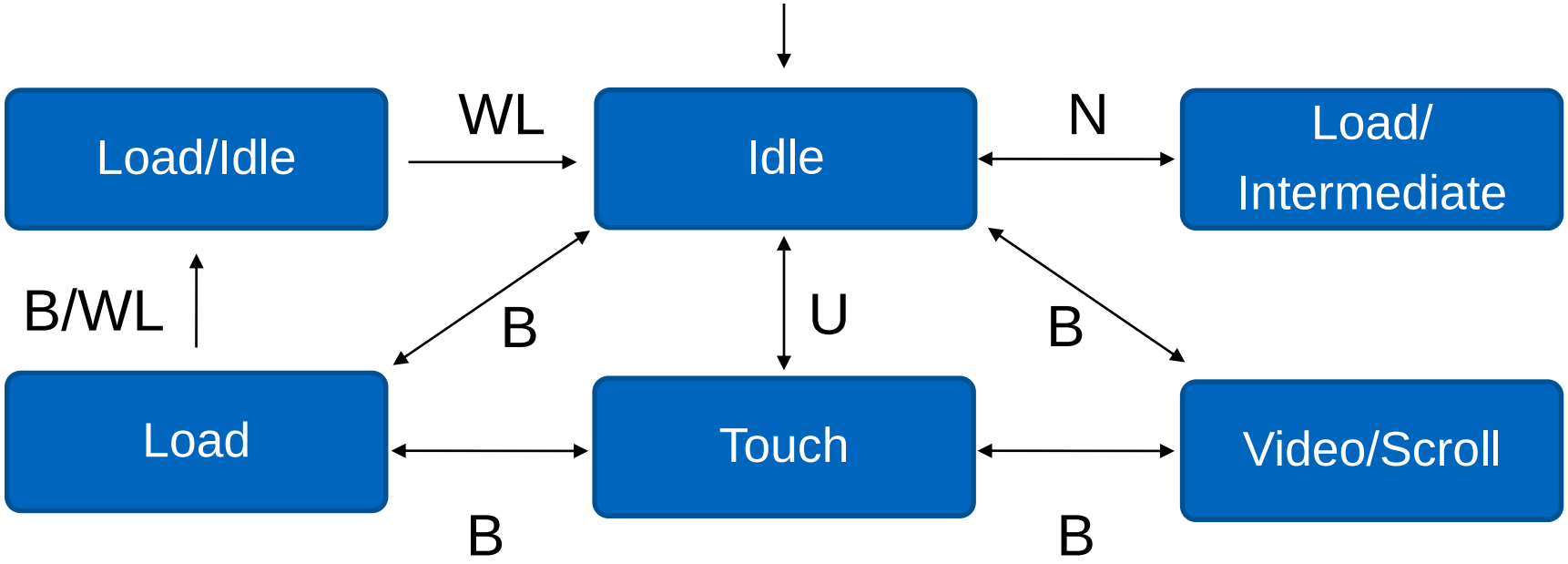
User-centric performance model RAIL.

RAIL Step	Latency	(User) Actions
Response, Animation	< 16ms	User drags finger and app's response is bound to finger position, ongoing page scroll/animation
Response, Animation	< 100ms	User taps an icon/button, initiates page scroll, animation begins
Idle	-	Background activities
Load	< 5s	Page ready to use on mobiles (fore-ground load only)

→ Performance requirements for the browser governor.
→ Metrics: Load time, FPS

Browsing Phases

Individual power management strategy for each phase

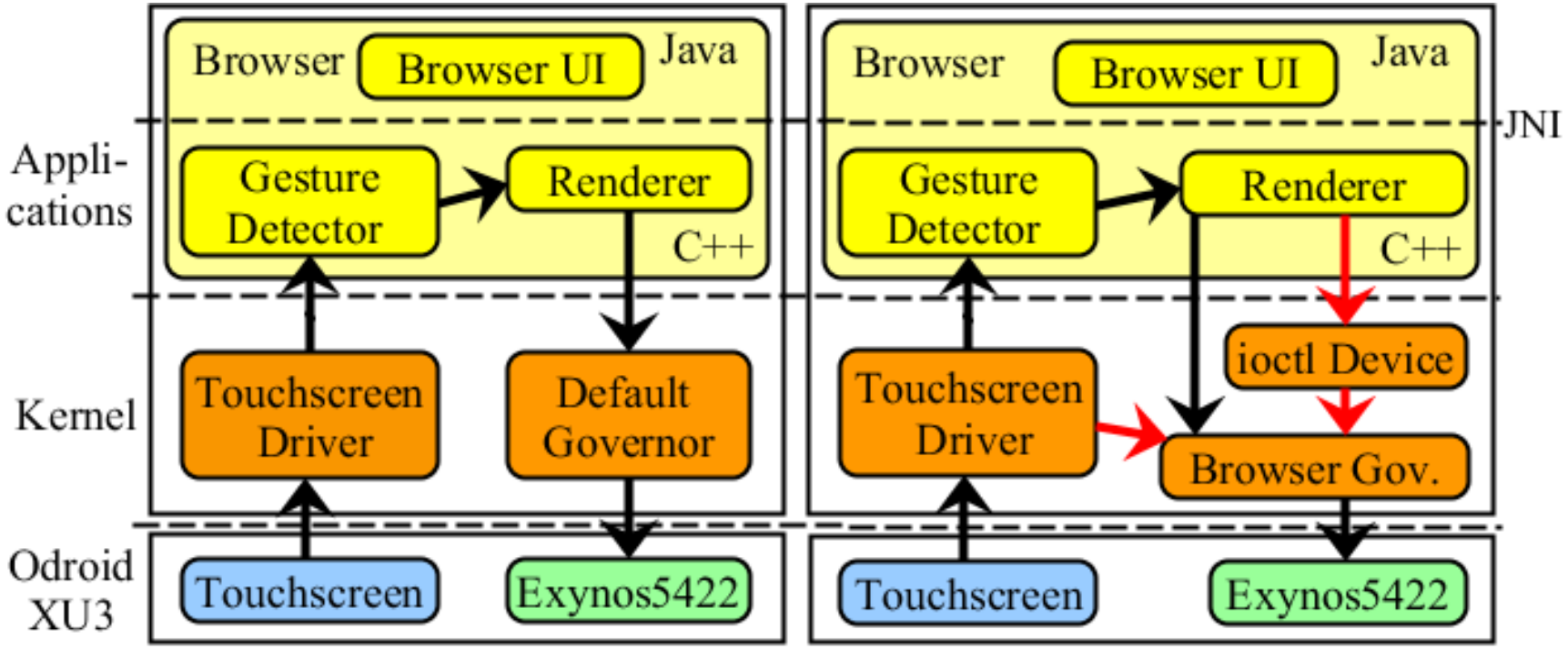


Phase state transition diagram:

- U: User inputs
- B: Browser state changes
- N: Network traffic
- WL: CPU workload

Governor Implementation

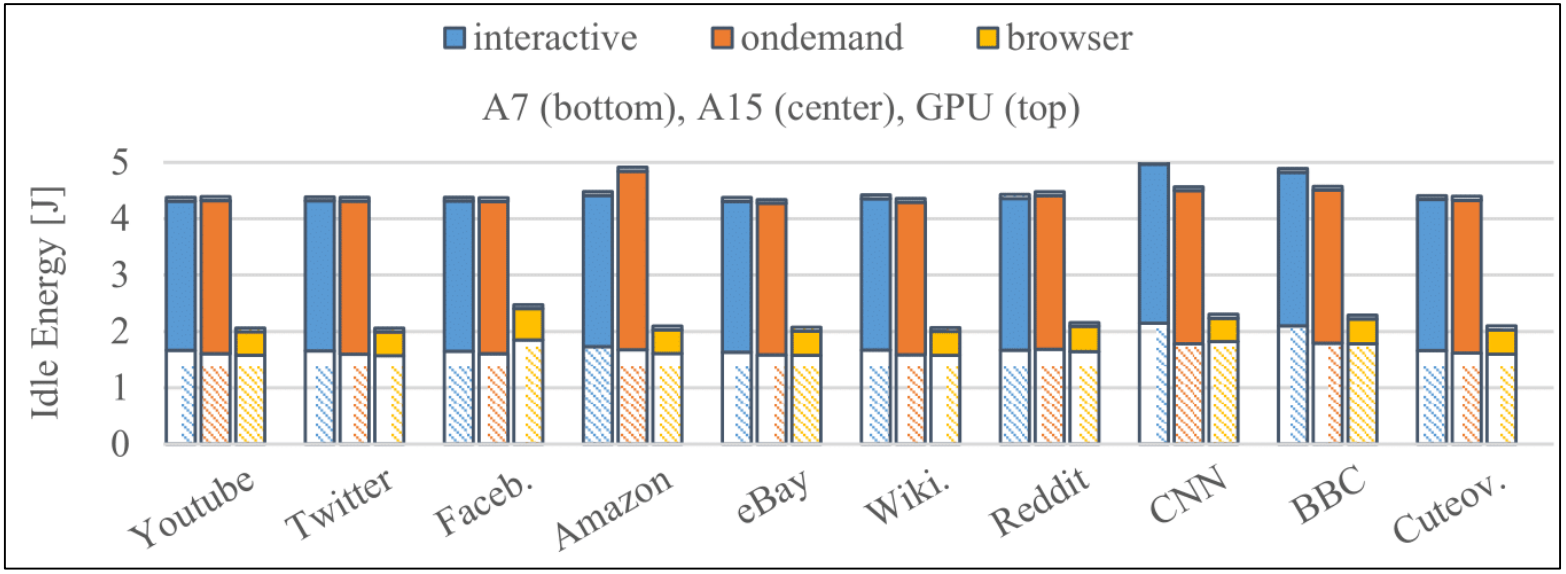
Android System Design, touch information propagation



Android Default

Browser Governor

Results: Idle Phase (1/5)



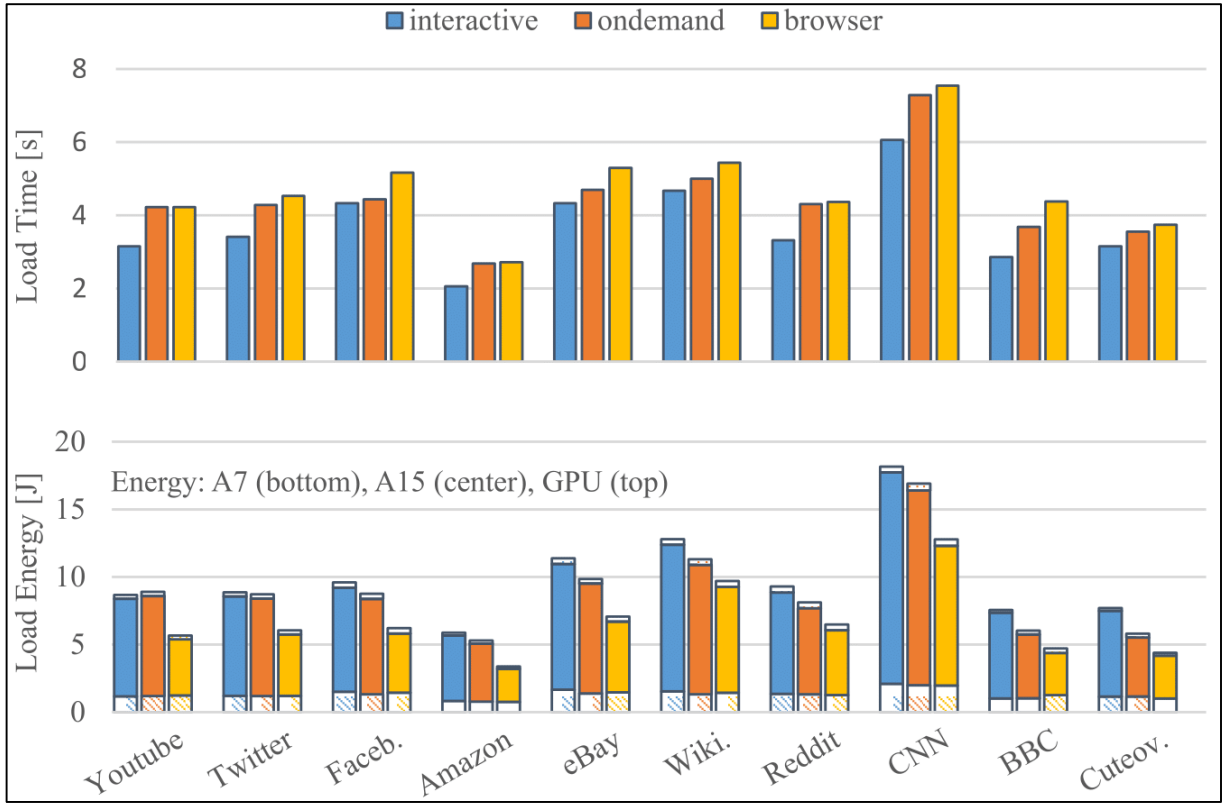
Power Management Strategy:

- Turn off A15 when system is idle (no user interaction, network, etc.)

Energy savings:

- 52.0% over interactive
- 51.5% over ondemand

Results: Load Phase (2/5)



Performance Metric:

- Loading time
- 8.1% slower than ondemand
- 28.2% slower than interactive

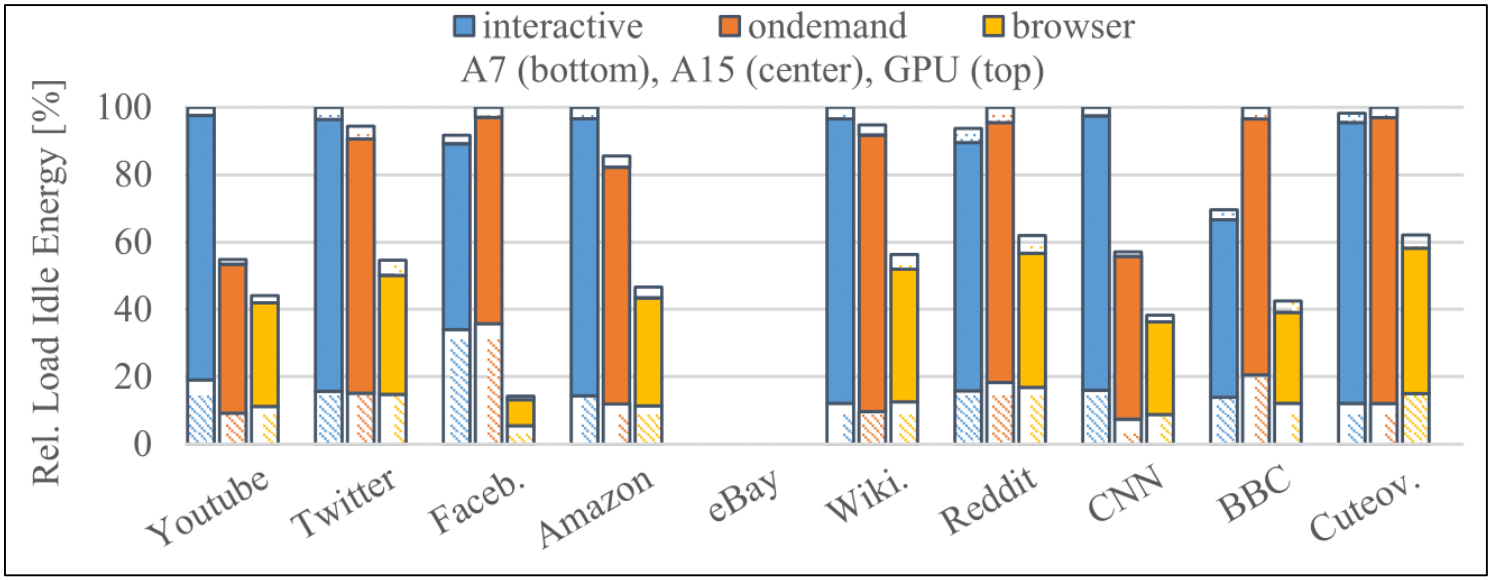
Power Management Strategy:

- A7 + A15 DVFS, no A15 power gating
- Occurs when load state available from the browser (foreground load)

Energy savings:

- 33.4% over interactive
- 25.3% over ondemand

Results: Load/Idle Phase (3/5)



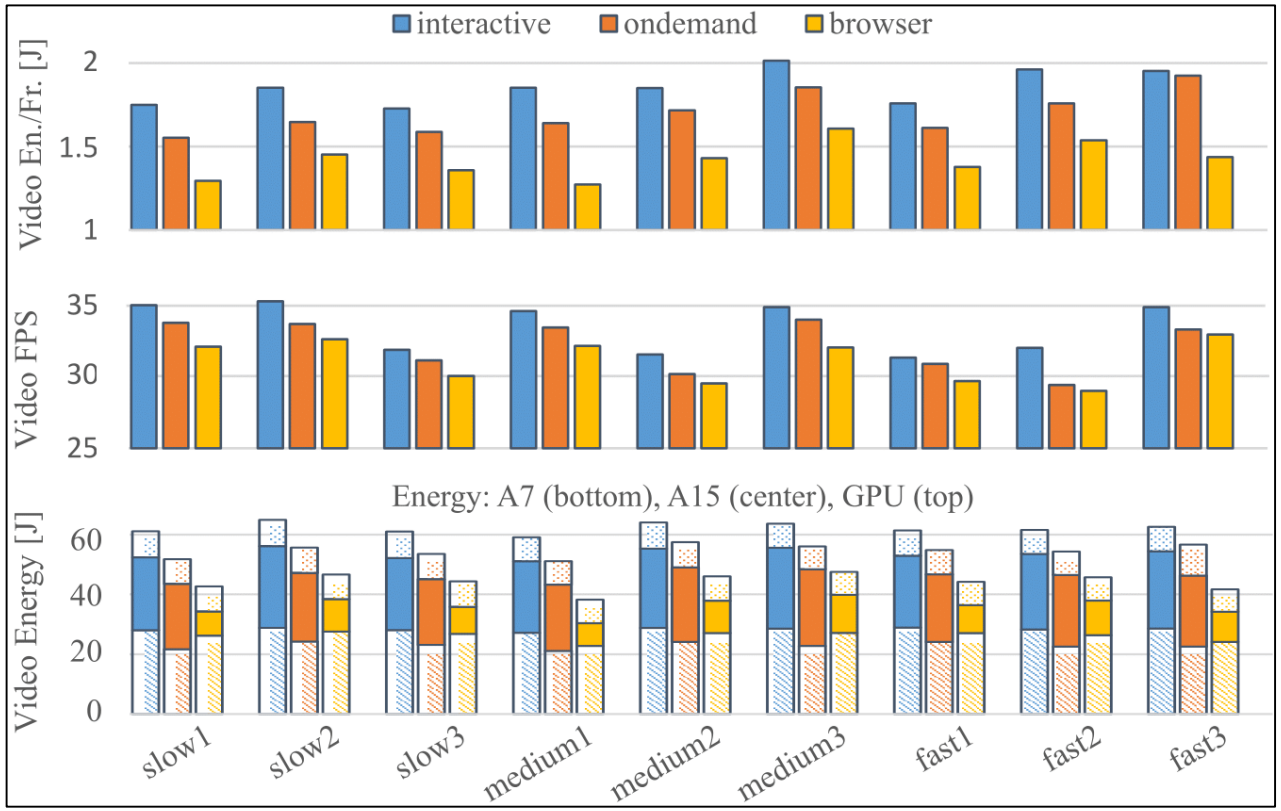
Power Management Strategy:

- A7 DVFS + A15 at lowest frequency
- Occurs when load state is over but workload is still high (background load)

Energy savings:

- 50.5% over interactive
- 44.4% over ondemand

Results: Video Phase (4/5)



Performance Metrics:

- FPS
  - Interactive: 33.5
  - Ondemand: 32.2
  - Browser: 31.2
- Energy/Frame
  - 23.6% over interactive
  - 16.6% over ondemand

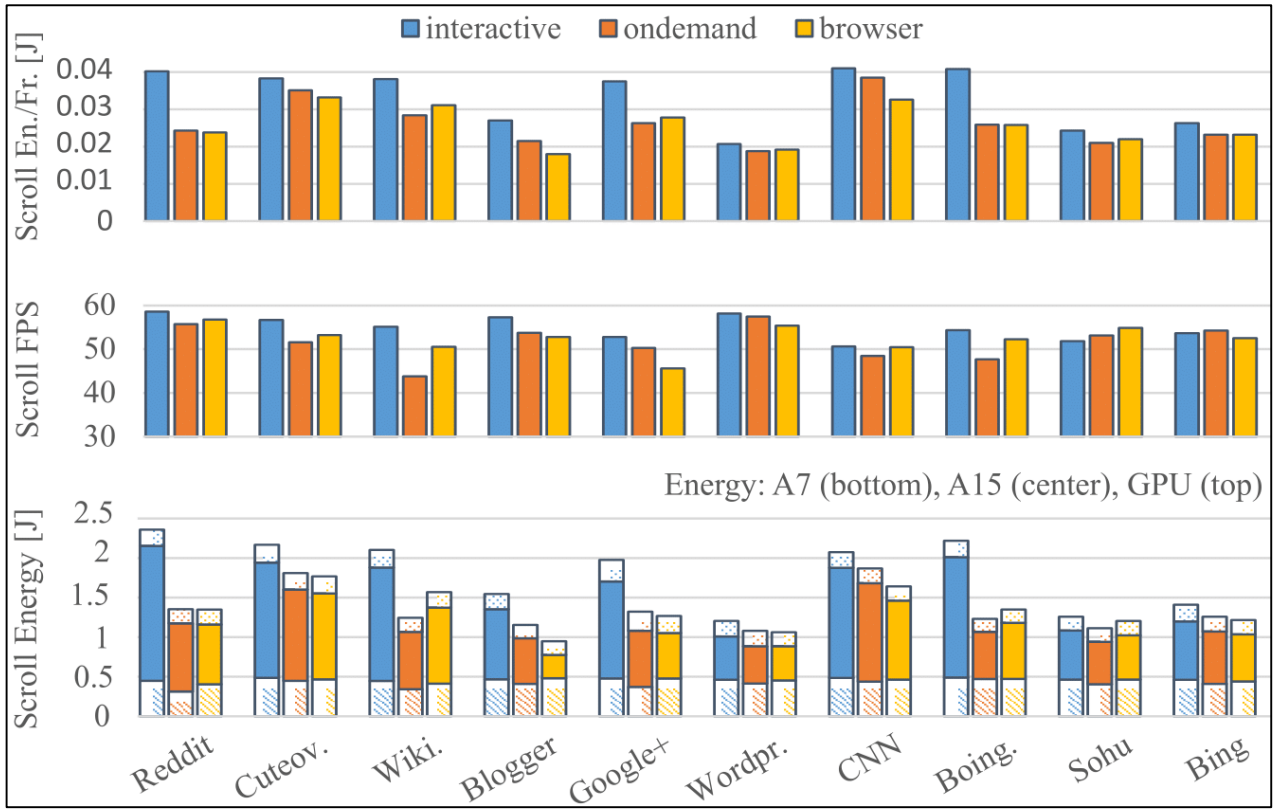
Power Management Strategy:

- A7 + A15 DVFS, A15 power gating
- Fixed FPS target value of 30
- Occurs when video state available from the browser

Energy savings:

- 29.0% over interactive
- 19.2% over ondemand

Results: Scroll Phase (5/5)



Performance Metrics:

- FPS
  - Interactive: 54.9
  - Ondemand: 51.6
  - Browser: 52.5
- Energy/Frame
  - 21.0% over interactive
  - 1.7% over ondemand

Power Management Strategy:

- A7 + A15 DVFS, A15 power gating
- Occurs when scroll state available from the browser
- Target 55 FPS

Energy savings:

- 25.1% over interactive
- Same as ondemand

Summary

- Application information can be exploited for (CPU) power management
- Phase information can be used for scenario specific power management, such as loading or scrolling
- Implementation and verification of the approach using Google Chrome and the Odroid-XU3 board
- Large energy saving potentials especially in idle phases
- Generalization of approach for other applications/browsers possible