

---

# Android Power HAL Overview

---



Qualcomm Technologies, Inc.

80-P8754-81 A

Confidential and Proprietary – Qualcomm Technologies, Inc.

**Restricted Distribution:** Not to be distributed to anyone who is not an employee of either Qualcomm Technologies, Inc. or its affiliated companies without the express approval of Qualcomm Configuration Management.



# Confidential and Proprietary – Qualcomm Technologies, Inc.

---

**NO PUBLIC DISCLOSURE PERMITTED:** Please report postings of this document on public servers or websites to: [DocCtrlAgent@qualcomm.com](mailto:DocCtrlAgent@qualcomm.com).

Not to be used, copied, reproduced, or modified in whole or in part, nor its contents revealed in any manner to others without the express written permission of Qualcomm Technologies, Inc.

MSM is a product of Qualcomm Technologies, Inc. Other Qualcomm products referenced herein are products of Qualcomm Technologies, Inc. or its subsidiaries.

Qualcomm and MSM are trademarks of Qualcomm Incorporated, registered in the United States and other countries. Other product and brand names may be trademarks or registered trademarks of their respective owners.

This technical data may be subject to U.S. and international export, re-export, or transfer (“export”) laws. Diversion contrary to U.S. and international law is strictly prohibited.

Qualcomm Technologies, Inc.  
5775 Morehouse Drive  
San Diego, CA 92121  
U.S.A.

© 2017 Qualcomm Technologies, Inc. and/or its affiliated companies. All rights reserved.

# Revision History

---

Revision	Date	Description
A	November 2017	Initial release

Qualcomm  
Confidential  
2019-11-26 00:42:31 PST  
hongwei.di@archermind.com

# Contents

---

- Objectives
- Android Power HAL
- Power Hint
- Android Power HAL Overview
- Sample Call Flows
- Call Flow Details
- Perflock Voting Mechanism
- Supported Parameters on SDM660 and SDM630
- Supported Parameters on MSM8953, MSM8937, MSM8940, MSM8917, and MSM8920
- References
- Questions?

# Objectives

---

- At the end of this presentation, you will understand the Android Power HAL process for the following chipsets:
  - SDM660
  - SDM630
  - MSM8953
  - MSM8937, MSM8940, MSM8917, and MSM8920

# Android Power HAL

---

- Power hardware abstraction layer (HAL)
  - Enable platform developers and OEMs to make chipsets and use case-specific adjustments to system-level parameters to save power.
  - Adjustments should only be limited to subtle use case-specific parameter tuning for power saving where generic governor behavior is suboptimal.
- How is Power HAL invoked by the applications?
  - Like any HAL, Power HAL is a shared library that must be loaded dynamically by applications.
  - It provides functions that are part of an existing contract (function prototypes) that applications already know about in the form of a structure, so when applications need an action performed, they simply invoke the corresponding function and the HAL does the rest.
  - For Power HAL, the contract is exposed in the `power_module` structure, which is defined in the header `hardware/libhardware/include/hardware/power.h`.

# Power Hint

---

- The following hints are defined in the header as part of the `power_hint_t` enumeration:
  - `POWER_HINT_VSYNC`
    - It is sent by SurfaceFlinger to indicate the ongoing progress of VSYNCs.
    - Qualcomm Technologies, Inc. (QTI) currently does not handle this hint.
  - `POWER_HINT_INTERACTION`
    - It is sent by PowerManagerService (notified by InputManagerService) when user activity, such as touch events and button presses, is detected.
    - This hint may be used to tweak the CPUFreq parameters to improve performance. This has been done for at least one of the QTI devices.
  - `POWER_HINT_VIDEO_ENCODE`:
    - It is sent by the camera HAL when video encode starts and then stops.
    - The data parameter is used to provide additional details about the encoding process in the form of a string that follows the "key1:value1;key2:value2; ...;keyN:valueN" format.
    - The state key is required and takes either 1 or 0 as its value to indicate that the encode has started and stopped, respectively.
    - This hint has been added by QTI.

## Power Hint (cont.)

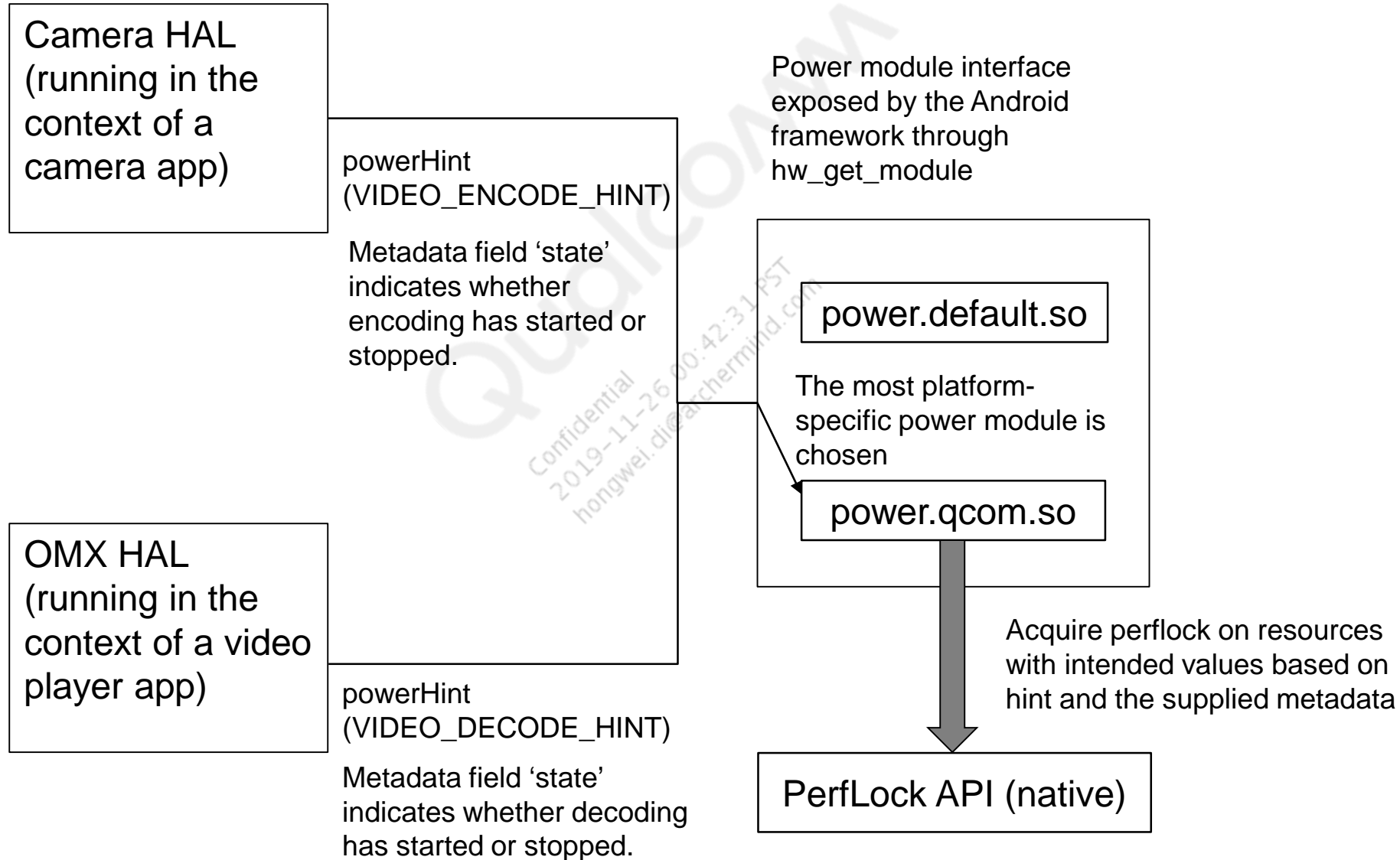
---

- **POWER\_HINT\_VIDEO\_DECODE**

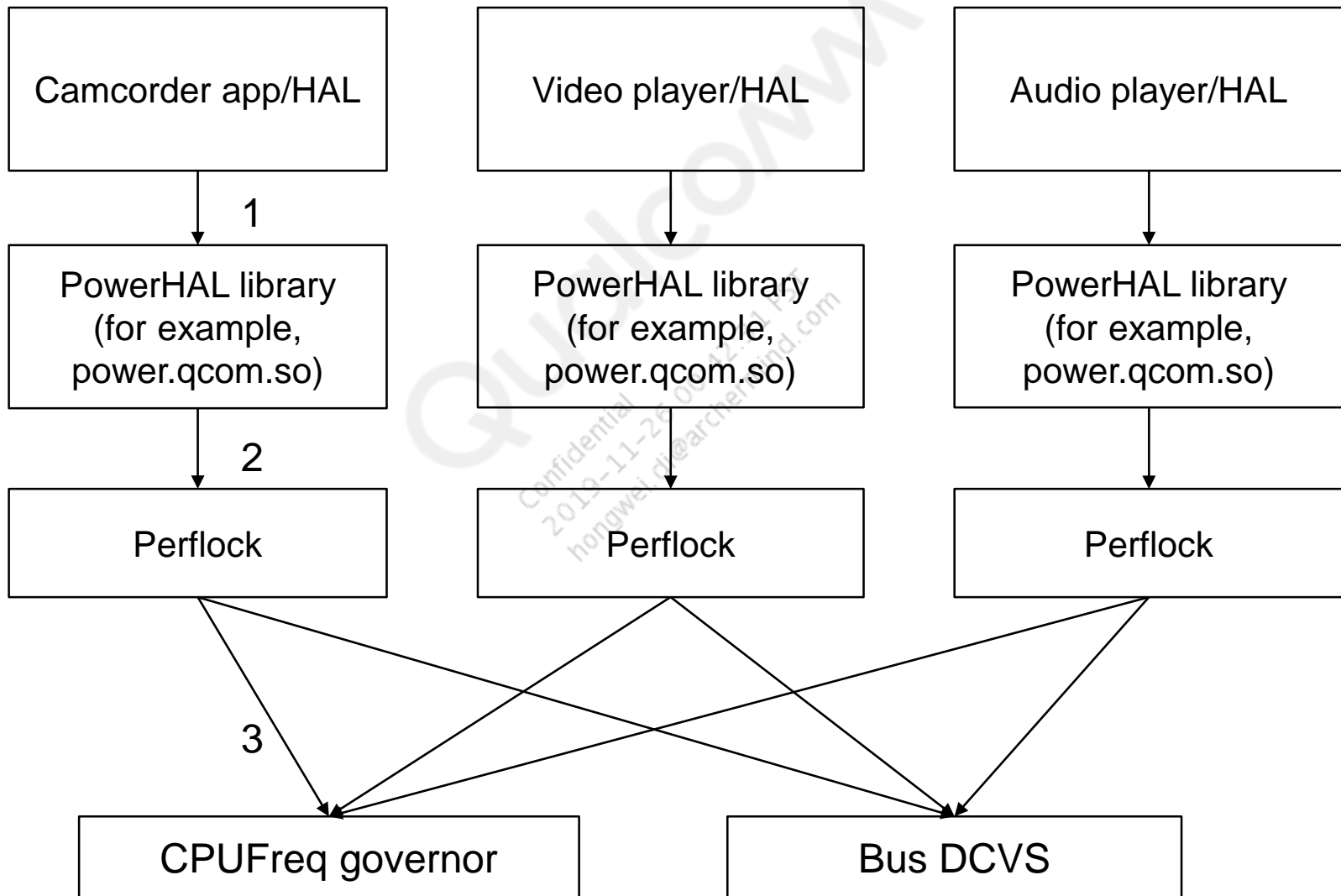
- It is sent by the OpenMAX (OMX) video decoder HAL when video decode starts and then stops.
- The data parameter is used to provide additional details about the decoding process in the form of a string that follows a "key1:value1;key2:value2;...;keyN:valueN" format.
- The state key is required and takes either 1 or 0 as its value to indicate that the encode has started and stopped respectively.
- This hint has been introduced by QTI.



# Android Power HAL Overview



# Sample Call Flows



# Call Flow Details

---

1. The application, or its corresponding HAL, sends a hint to the Power HAL when a power-impacting activity (such as video encoding) is triggered. For this, the powerHint function is used with the type of the hint; for example, POWER\_HINT\_VIDEO\_ENCODE as one argument and additional or optional metadata as the other argument.
2. The QTI-specific Power HAL, power.qcom.so, internally uses Perflock API methods to modify the CPUFreq and Bus dynamic clock and voltage scaling (DCVS) nodes.
3. Perflock changes the sysfs nodes to use values requested by applications. It changes the nodes' values based on votes from the applications. Slide 12, Perflock Voting Mechanism contains more information on the voting mechanism.

# Perflock Voting Mechanism

---

- Perflock is programmed to pick the vote that works best for performance.
- Example:
  1. An application raises a request to the Power HAL that hispeed\_freq should be updated to 902 MHz.
  2. However, there is already a vote to keep hispeed\_freq at 1.4 GHz.
  3. The request to be executed is the one (latest request) that asks for hispeed\_freq to be at 902 MHz.

# Supported Parameters on SDM660 and SDM630

Use case	Governor, bus DCVS, and scheduler parameters requested
Video encode and preview	<ul style="list-style-type: none"> <li>■ CPUFreq parameters <ul style="list-style-type: none"> <li>▪ hispeed freq – 902 MHz (default – 1401 MHz)</li> <li>▪ go hispeed load – 95 (default – 90 ms)</li> <li>▪ above_hispeed_delay – 40 ms (default – 19000 1401600:39000)</li> <li>▪ target_loads – 95 (default – 85 1344000:80)*</li> </ul> </li> <li>■ Bus DCVS v2 parameter <ul style="list-style-type: none"> <li>▪ Sample_ms of 10 ms (default – 4 ms)</li> </ul> </li> <li>■ Scheduler parameter <ul style="list-style-type: none"> <li>▪ nr_run – 5 (default – 3)*</li> </ul> </li> </ul>
Display off	<ul style="list-style-type: none"> <li>■ CPUFreq parameters <ul style="list-style-type: none"> <li>▪ hispeed freq – 902 MHz (default – 1401 MHz)</li> <li>▪ go hispeed load – 95 (default – 90 ms)</li> <li>▪ above_hispeed_delay – 40 ms (default – 19000 1401600:39000)</li> </ul> </li> <li>■ Bus DCVS v2 parameter <ul style="list-style-type: none"> <li>▪ Sample_ms of 10 ms (default – 4 ms)</li> </ul> </li> <li>■ Scheduler parameter <ul style="list-style-type: none"> <li>▪ Sched group upmigrate – 500 (default – 140)**</li> </ul> </li> </ul>

\* Only applicable for SDM630

\*\* Not applicable for SDM630

# Supported Parameters on MSM8953, MSM8937, MSM8940, MSM8917, and MSM8920

Use case	Governor parameters requested
Video encode and preview	CPUFreq parameters <ul style="list-style-type: none"><li>▪ timer_rate – 40 (default – 20 ms)</li><li>▪ use_sched_load – 1 (default – 0)</li><li>▪ use_migration_notif – 1 (default – 0)</li></ul>
Display off	CPUFreq parameters <ul style="list-style-type: none"><li>▪ timer_rate – 40 (default – 20 ms)</li><li>▪ timer_rate – 50 (default – 20 ms)*</li><li>▪ notify_on_migrate – 0 (default – 1)*</li></ul>

\* Applies to only the MSM8937, MSM8940, MSM8917, and MSM8920 chipset family

# References

---

Acronym or term	Definition
DCVS	Dynamic clock and voltage scaling
HAL	Hardware abstraction layer
OMX	OpenMAX

Qualcomm  
Confidential  
2019-11-26 00:42:31 PST  
hongwei.di@archarmind.com

## Questions?

<https://createpoint.qti.qualcomm.com>

