Power Management Debugging Guide

TCCxxxx-Android-ALL-V1.03E

Rev. 1.03

Feb 25, 2014

Telechips

DISCLAIMER

All information and data contained in this material are without any commitment, are not to be considered as an offer for conclusion of a contract, nor shall they be construed as to create any liability. Any new issue of this material invalidates previous issues. Product availability and delivery are exclusively subject to our respective order confirmation form; the same applies to orders based on development samples delivered. By this publication, Telechips, Inc. does not assume responsibility for patent infringements or other rights of third parties that may result from its use.

Further, Telechips, Inc. reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.
No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of Telechips, Inc.
This product is designed for general purpose, and accordingly customer be responsible for all or any of intellectual property licenses required for actual application. Telechips, Inc. does not provide any indemnification for any intellectual properties owned by third party.
Telechips, Inc. can not ensure that this application is the proper and sufficient one for any other purposes but the one explicitly expressed herein. Telechips, Inc. is not responsible for any special, indirect, incidental or consequential damage or loss whatsoever resulting from the use of this application for other purposes.

COPYRIGHT STATEMENT

Important Notice

For customers who use licensed Codec ICs and/or licensed codec firmware of mp3:
"Supply of this product does not convey a license nor imply any right to distribute content created with this product in revenue-generating broadcast systems (terrestrial. Satellite, cable and/or other distribution channels), streaming applications(via internet, intranets and/or other networks), other content distribution systems(pay-audio or audio-on-demand applications and the like) or on physical media(compact discs, digital versatile discs, semiconductor chips, hard drives, memory cards and the like). An independent license for such use is required. For details, please visit http://mp3licensing.com".

For customers who use other firmware of mp3:
"Supply of this product does not convey a license under the relevant intellectual property of Thomson and/or Fraunhofer Gesellschaft nor imply any right to use this product in any finished end user or ready-to-use final product. An independent license for such use is required. For details, please visit http://mp3licensing.com".

For customers who use Digital Wave DRA solution:
"Supply of this implementation of DRA technology does not convey a license nor imply any right to this implementation in any finished end-user or ready-to-use terminal product. An independent license for such use is required."

For customers who use DTS technology:
"This product made under license to certain U.S. patents and/or foreign counterparts."
"© 1996 – 2011 DTS, Inc. All rights reserved."

For customers who use Dolby technology:
"Supply of this Implementation of Dolby technology does not convey a license nor imply a right under any patent, or any other industrial or intellectual property right of Dolby Laboratories, to use this Implementation in any finished end-user or ready-to-use final product. It is hereby notified that a license for such use is required from Dolby Laboratories."

For customers who use MS technology:
"This product is subject to certain intellectual property rights of Microsoft and cannot be used or distributed further without the appropriate license(s) from Microsoft."

Revision History

Date	Version	Description
2011-03-07	1.00	1st draft.
2012-02-29	1.01	Update to Android 4.0 (Ice Cream Sandwitch)
2013-12-13	1.02	Update to Android 4.2 (Jelly Bean)
2014-02-25	1.03	Update to Android 4.4 (Kitkat)

TABLE OF CONTENTS

Contents

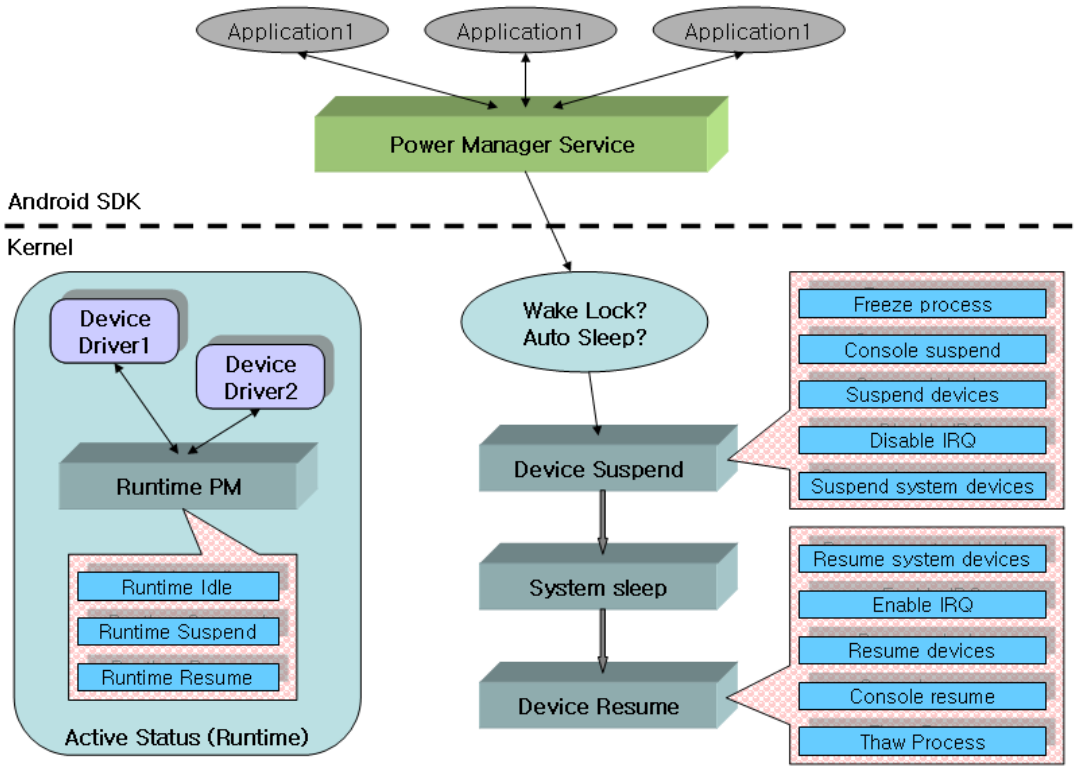
1 Introduction

This guide shows Android power management – especially Suspend and Resume feature which is based on Linux “Suspend to RAM” and how to setup kernel options to debug Android PM system.
This guide is made for Telechips Android Platform based on Kitkat version.

2 Android Power Management

This chapter shows how to suspend system and resume from suspend mode.
Android suspends system based on Linux “Suspend to RAM” and Runtime-PM.

2.1 State Transition



2.2 Runtime PM

Turn off (stop clock or remove power) hardware components that aren’t going to be used in the near future, transparently from the user space’s viewpoint.

Runtime PM framework uses abstract states of devices

- ACTIVE
  Device can do I/O (presumably in the full-power state)
- SUSPENDED
  Device cannot do I/O (presumably in a low-power state)
- SUSPENDING
  Device state is changing from ACTIVE to SUSPENDED
- RESUMING
  Device state is changing from ACTIVE to SUSPENDED

2.3 Device Suspend

If there is no wake-locked applications, the system enters into suspend mode.
Before entering suspend mode, all device drivers have to be suspended and it is the main process to enter sleep mode.
To enter sleep mode, following operations are performed.

① Freeze Process
  User space process and other tasks are stopped.

② Console Suspend
Disabling UART console.
This can be skipped by changing kernel option for debugging purpose.

③ Suspend Devices
Performs call back function to suspend all non-syscore device including platform devices.

④ Disable IRQ

⑤ Suspend System Devices
Performs call back function to suspend syscore device.

2.4 System Sleep

System enters into shut-down mode. If there is hardware reset or RTC wake-up, wake-up operation will be started.
The detailed operation will be different according to machine type and board.

2.5 Device Resume

After wake-up, the required operation for each driver will be done to resume pervious status.
Operation procedure is as follows.

① Resuem System devices
Performs call back function to resume syscore device.

② Enable IRQ

③ Resume devices
Performs call back for all non-syscore device.

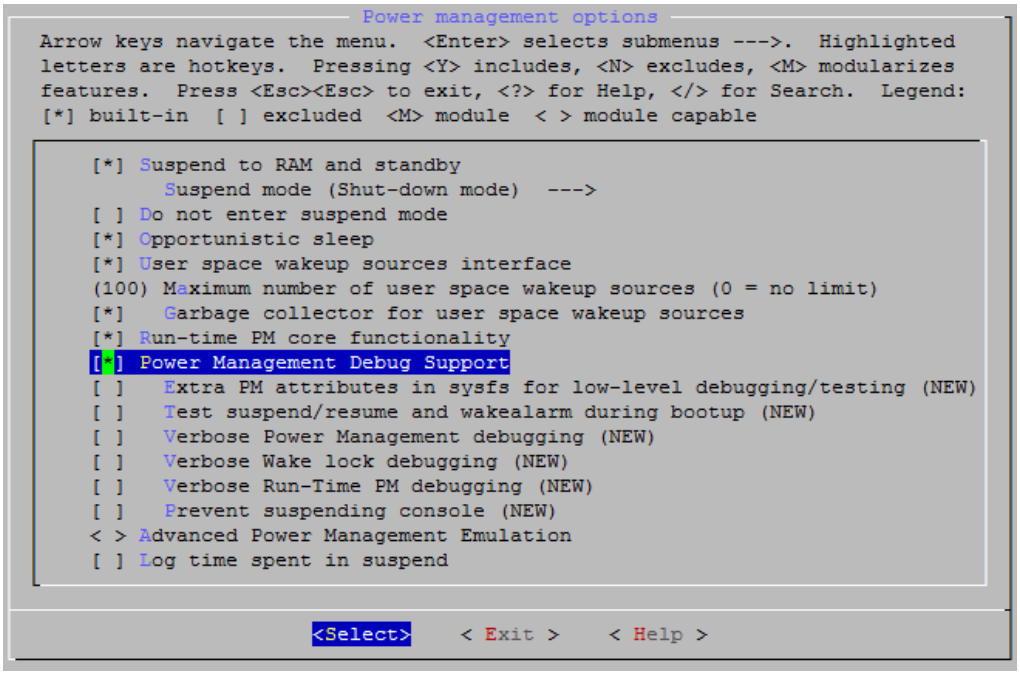
④ Console resume
Activate Console.

⑤ Thaw process
Activate user process and all tasks.

3 Setting Power Management Debugging Option

There is several options to debug power management in kernel menu configuration.
By selecting this option, you can show various events which are happened during state transition.
This chapter shows those options in detail.

After executing kernel menuconfig, select Power management option → Power Management Debug Support.
Various sub-menu will be displayed.



Options to be used to debug is as follows.

- Verbose Power Management debugging
- Verbose Wake lock debugging
- Verbose Run-Time PM debugging
- Prevent suspending console
- Test suspend/resume and wakealarm during bootup

3.1 Verbose Power Management debugging

By enabling this option, the detailed operating status during Device Suspend and Device Resume will be out.
You can check suspend/resume call back function of most device driver include platform device.

```
[    71.922329] tcc-adc tcc-adc: bus suspend
[    71.922373] tcc-i2c tcc-i2c.3: bus suspend
[    71.922410] tcc-i2c tcc-i2c.2: bus suspend
[    71.922444] tcc-i2c tcc-i2c.1: bus suspend
[    71.922479] tcc-i2c tcc-i2c.0: bus suspend
[    71.922513] tcc-i2c tcc-i2c.4: bus suspend
[    71.922544] tcc-uart tcc-uart.4: bus suspend
[    71.922592] tcc-uart tcc-uart.1: bus suspend
[    71.922635] tcc-uart tcc-uart.0: bus suspend
[    71.947736] platform tcc_mtd: bus suspend
[    71.947773] tcc-ts tcc-ts: bus suspend
[    71.947812] android-battery android-battery: bus suspend
[    71.947859] reg-dummy reg-dummy: bus suspend
[    71.947959] PM: suspend of devices complete after 155.387 msecs
[    71.948544] PM: late suspend of devices complete after 0.554 msecs
[    71.949254] PM: noirq suspend of devices complete after 0.681 msecs
[    71.949274] Disabling non-boot CPUs ...
[    71.950005] PM: noirq resume of devices complete after 0.447 msecs
[    71.950407] PM: early resume of devices complete after 0.265 msecs
```

```
[    71.950751] reg-dummy reg-dummy: bus resume
[    71.950792] android-battery android-battery: bus resume
[    71.950817] tcc-ts tcc-ts: bus resume
[    71.950844] platform tcc_mtd: bus resume
[    71.950867] tcc-uart tcc-uart.0: bus resume
[    71.950919] [UART00] setting ulcon: 00000003, umcon: 00000002, brddiv to 26, baud 115200, uart_clk 48000000
[    71.950991] [UART00] setting ulcon: 00000003, umcon: 00000002, brddiv to 26, baud 115200, uart_clk 48000000
[    71.951052] tcc-uart tcc-uart.1: bus resume
[    71.951093] tcc-uart tcc-uart.4: bus resume
[    71.951127] tcc-i2c tcc-i2c.4: bus resume
[    71.951153] tcc-i2c tcc-i2c.0: bus resume
[    71.951236] tcc-i2c tcc-i2c.1: bus resume
[    71.951277] tcc-i2c tcc-i2c.2: bus resume
[    71.951312] tcc-i2c tcc-i2c.3: bus resume
[    71.951346] tcc-adc tcc-adc: bus resume
[    71.951385] tcc-rtc tcc-rtc: bus resume
```

3.2 Verbose Wake Lock debugging

Android PM uses wake lock to control power state. By checking wake-lock messge, you can investigate overall operation of suspend system.

```
[  125.346919] wakeup_source_deactivate: name:eventpoll
[  125.355607] wakeup_source_deactivate: name:PowerManagerService.WakeLocks
[  125.892864] wakeup_source_deactivate: name:mmc1_detect
[  125.897842] PM: suspend entry 2014-02-13 11:58:03.074893833 UTC
[  125.903782] PM: Syncing filesystems ... done.
[  125.940290] TCC cpufreq suspend: setting frequency to 342000 kHz
[  125.947010] Freezing user space processes ... (elapsed 0.001 seconds) done.
[  125.955636] Freezing remaining freezable tasks ... (elapsed 0.001 seconds) done.
```

You can check wake lock/unlock of each device and PowerManagerService. You can also find when suspend operation begins.

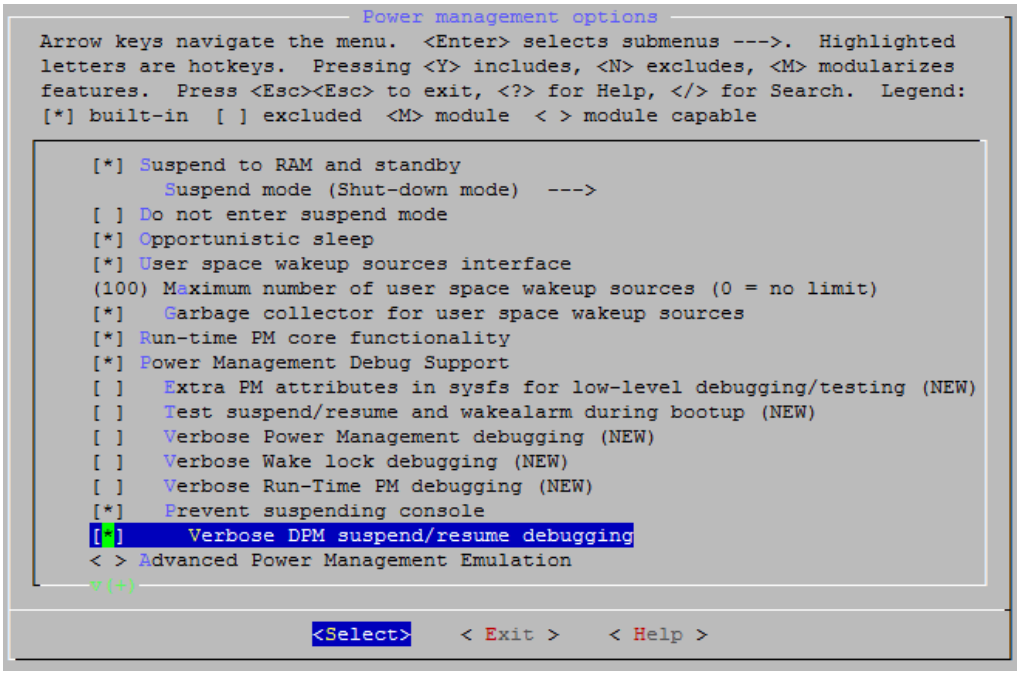
3.3 Verbose PM Runtime debugging

By enabling this option, you can know which device enter the suspend or resume or idle on runtime state.

```
[    43.076524] rpm_idle: rpmflags:0x5
[    43.079883] rpm_resume: dev:mali-utgard, rpmflags:0x4
[    43.087851] rpm_suspend: dev:mali-utgard, rpmflags:0xd
[    43.112400] rpm_resume: dev:mali-utgard, rpmflags:0x4
[    43.117323] rpm_resume: dev:mali-utgard, rpmflags:0x4
[    43.125188] rpm_suspend: dev:mali-utgard, rpmflags:0xd
[    43.385677] rpm_suspend: dev:mali-utgard, rpmflags:0xd
[    44.012696] dwc3_tcc_probe
[    44.064475] time delay = 1
[    44.068938] rpm_resume: dev:dwc3, rpmflags:0x4
[    44.073203] rpm_idle: dev:dwc3, rpmflags:0x1
[    44.077721] rpm_resume: dev:dwc3, rpmflags:0x0
[    44.168855] rpm_idle: rpmflags:0x4
[    44.173172] rpm_idle: dev:tcc-dwc3, rpmflags:0x4
[    44.213797] file system registered
[    44.226225]   gadget: android_usb ready
[    44.238841] android_usb: already disabled
[    44.242861] android_usb: already disabled
[    44.252920] adb_open
[    44.256585] mtp_bind_config
[    44.259232] adb_bind_config
[    44.334503] rpm_suspend: dev:mali-utgard, rpmflags:0x9
[    44.339539] rpm_suspend: dev:mali-utgard, rpmflags:0xa
[    44.362679] rpm_idle: rpmflags:0x1
[    47.027382] rpm_resume: dev:mali-utgard, rpmflags:0x4
```

3.4 Prevent suspending console

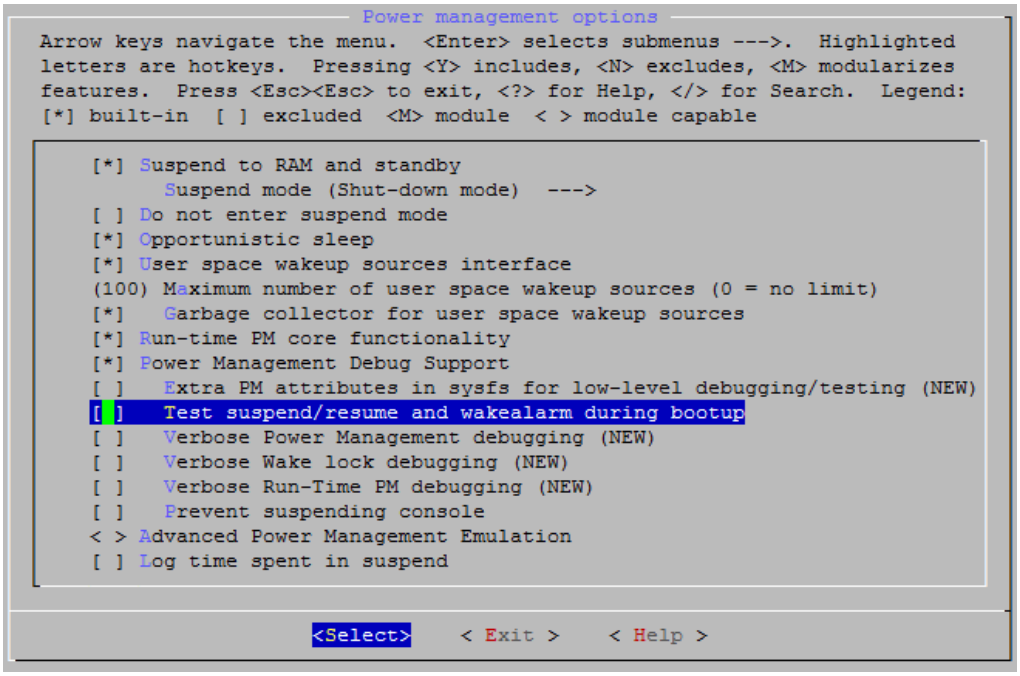
To debug the problem which is happened during suspend/resume, debug message has to bo shown until that point.
However, console will be disabled during the beginning of suspend, the message is not shown.
By enabling this option, PM driver maintains UART console until entering actual suspend state.
During resume, log message will be out from PM driver.



By enabling this option (Verbose DPM suspend/resume debugging), the detailed operating status during DPM Suspend and DPM Resume will be out. (DPM: Dynamic Power Management)

3.5 Time Measurement option

Power management options → Power Management Debug Support → Test suspend/resume and wakealarm during bootup



By enabling this option, we can check elapsed time about suspend and resume.