UCCP Economy Mode Design

Copyright © Imagination Technologies Limited. All Rights Reserved.

This document is strictly confidential. Neither the whole nor any part of the information contained in, nor the product described in, this document may be adapted or reproduced in any material form except with the written permission of Imagination Technologies Limited. Imagination Technologies, the Imagination logo, PowerVR, MIPS, Meta, Ensigma and Codescape are trademarks or registered trademarks of Imagination Technologies Limited. All other logos, products, trademarks and registered trademarks are the property of their respective owners. This document can only be distributed subject to the terms of a Non Disclosure Agreement or Licence with Imagination Technologies Limited.

Filename : UCCP Economy Mode Design  
Version : 1.0.15 Not Issued - Live Document  
Issue Date : 22 Jan 2014  
Author : Imagination Technologies Limited

Contents

[1. Introduction 3](#_Toc378273285)

[2. WLAN Requirements 3](#_Toc378273286)

[3. Proposed Solution 3](#_Toc378273287)

[4. Changes Required 4](#_Toc378273288)

List of Figures

**No table of figures entries found.**

IF{{Internal}}

Document History

| Issue | Date | Changes/Comments |
| --- | --- | --- |
| 1.0.9 | 12 Dec 2013 | External Issue. |
| 1.0.10 | 16 Dec 2013 | External Issue |
| 1.0.14 | 22 Jan 2014 | External Issue |
| 1.0.15 | 22 Jan 2014 | External Issue |

END{{Internal}}

# Introduction

TSB HOST Linux/Android OS has a power management framework. This power management framework controls HOST system suspend/resume function and supports the standard Linux suspend/resume framework.

There are two modes of system operation.

a) Normal mode

b) Economy mode.

By default, the system operates in "normal mode". Based on certain criteria, the power management framework can put the system in a low power mode also known as "Economy mode".

Before switching to "Economy mode", the PM system invokes the 'suspend' function of all the registered drivers.

Before switching back to "Normal mode", the PM system invokes the 'resume' function of all the registered drivers.

*Note: IMG will implement all the changes listed in the document. TSB is expected to review and provide feedback.*

# WLAN Requirements

During Economy mode, the UCCP:

1. Shall maintain and monitor an existing connection with an access point
2. Shall not access Host DDR until it receives a confirmation from the host driver
3. Shall wake up the Host CPU by raising an interrupt when one of the following conditions is true
4. There is incoming data packet intended for the Host CPU
5. Connection to the Access point is lost
6. Shall NOT wake up the Host CPU if there is no connection with any access point prior to entering economy mode

# Proposed Solution

The Linux MAC80211 stack registers the following functions to support the suspend/resume framework:

ieee80211\_suspend

ieee80211\_resume

ieee80211\_set\_wakeup

When the *ieee80211\_suspend* is invoked mac80211 will quiesce first and stop transmitting and doing any other configuration, and then suspend the device. There are 2 ways in which the device can be suspended depending on whether WoWLAN is enabled or disabled:

1. If WoWLAN is supported by the device, then the device is put in a suspended state wherein it goes into a 802.11 power save mode. The device can also have the capability of waking up the host if one of the WoWLAN triggers is received.
2. If WoWLAN is not supported by the device, then suspend will just deconfigure the device which will tear down any existing connections etc.

When *ieee80211\_resume* is invoked it will bring the device to active state by:

1. Bringing it out of 802.11 powersave if WoWLAN is enabled
2. Reconfiguring the device if WoWLAN is disabled.

ieee80211\_set\_wakeup is used to enable/disable wakeup by the device when WoWLAN configuration is changed.

Given the above design, in order to meet the WLAN requirements stated above, we will have to enable support for the WoWLAN feature in the driver,so that the *ieee80211\_suspend* does not deconfigure the device completely and just puts it into powersave mode.

# Changes Required

* 1. Driver-Firmware Interface
     1. The following new command will be needed:

CMD\_PS\_ECON\_CFG

to signal the beginning/end of the economy mode (suspend/resume). The command

can be sent to the firmware in the "struct umac\_cmd\_ps\_cfg" (being used for the

regular powersave) itself.

* + 1. The following new events will be needed:

EVENT\_PS\_ECON\_CFG\_DONE

EVENT\_PS\_ECON\_WAKE

EVENT\_PS\_ECON\_CFG\_DONE event is needed to indicate that the firmware has finished processing the CMD\_PS\_ECON\_CFG command and the driver can return control back to mac80211 to complete the suspend/resume. The following new structure will be needed to signal the event:

*struct umac\_event\_ps\_econ\_cfg\_complete {*

*struct umac\_cmd\_ps\_cfg ps\_cfg;*

*unsigned char status; /\* SUCCESS/FAILURE \*/*

*} \_PACKED\_;*

EVENT\_PS\_ECON\_WAKE event is needed to indicate that UCCP has encountered a

wakeup trigger and has requested the host to go into normal mode from economy

mode. The following new structure will be needed to signal the event:

*struct umac\_event\_ps\_econ\_wake {*

*struct umac\_lmac\_msg\_hdr hdr;;*

*enum UMAC\_PS\_ECON\_WAKE\_TRIG trigger;*

*} \_PACKED\_;*

*where*

*enum UMAC\_PS\_ECON\_WAKE\_TRIG {*

*TRIG\_PKT\_RCV,*

*TRIG\_DISCONNECT*

*};*

The driver need not take any special action on receipt of this event because this is only

used to wake up the host from economy mode. The reason for wakeup is included in

the event. This event will result in the resume function being called. The host sends

CMD\_PS\_ECON\_CFG command to the firmware to confirm the wakeup, the firmware

will then send the RX packet or the DISCONNECT event.

* 1. Host Driver
     1. Following callbacks of the ieee80211\_ops will need to be implemented:

suspend

resume

* + 1. The suspend callback will do the following:
       1. Make a defensive check to make sure that there is only 1 active VIF and it is in STA mode.
       2. Send the CMD\_PS\_ECON\_CFG command to the firmware with the powersave\_state element of struct umac\_cmd\_ps\_cfg set to PWRSAVE\_STATE\_DOZE.
       3. Wait for the EVENT\_PS\_ECON\_CFG\_DONE event from the firmware.
       4. Based on the status indicated in the event message return the control back to mac80211 with the appropriate return value.
       5. If a timeout occurs while waiting for the event return a failure indication to mac80211.

* + 1. The resume callback will do the following:
       1. Send the CMD\_PS\_ECON\_CFG command to the firmware with the powersave\_state element of struct umac\_cmd\_ps\_cfg set to PWRSAVE\_STATE\_AWAKE.
       2. Wait for the EVENT\_PS\_ECON\_CFG\_DONE event from the firmware.
       3. Based on the status indicated in the event message return the control back to mac80211 with the appropriate return value.
       4. If a timeout occurs while waiting for the event return a failure indication to mac80211.
    2. The set\_wakeup op need not be implemented since the UCCP will always be set to wakeup the host, whenever the host is in the economy mode and either of the following events occur:
* Disconnect event
* Packet reception
  1. User Space
     1. It is the responsibility of the User Layer to enable WoWLAN using a tool like *iw* or by sending an equivalent ioctl. For e.g. using *iw* the following command can be used:

*iw phy phy0 wowlan enable any*

(*The WoWLAN trigger can be to any value since the UCCP triggers are not configurable and will always be set to:*

* *Disconnect event*
* *Packet reception*

*irrespective of the user configuration. It is suggested that the value "any" be used for the trigger to keep the configuration simple*.)

Enabling WoWLAN is necessary so that mac80211 does not deconfigure the device on suspend, as explained earlier (in the "Proposed Solution" section).

* + 1. WoWLAN needs to be enabled before the system goes into economy mode for the first time.