WLAN  
Host Loopback Testbench

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 : WLAN.Host Loopback Testbench  
Version : 1.0.16 Not Issued - Live Document  
Issue Date : 22 Jan 2014  
Author : Imagination Technologies Limited

Contents

[1. Introduction 3](#_Toc378273352)

[2. Description 4](#_Toc378273353)

List of Figures

**No table of figures entries found.**

IF{{Internal}}

Document History

| Issue | Date | Changes/Comments |
| --- | --- | --- |
| 1.0.16 | 22 Jan 2014 | External Issue. |

END{{Internal}}

# Introduction

The block diagram below describes the RPU MAC+PHY level loopback scenario to assist Toshiba in their Cadence Palladium Emulator based loopback testing. This document gives a brief description and test procedure for this testing



**Linux Host –** This is an ARM Host processor running linux and the delivered WLAN host driver. For the purpose of this testing, a testbench is implemented in the WLAN host driver that dumps the TX packet data into a local memory and after the loopback, it collects the received packet data, compares the transmitted and received payload and qualifies PASS/FAIL criteria based on this.

The Linux driver uses 4 MB of host RAM starting at physical address specified by HAL\_HOST\_UCCP\_RAM\_START in hal\_hostport.h

*Only for the Loopback test*, the RPU uses an additional 16MB of host RAM starting from HAL\_HOST\_UCCP\_RAM\_START + 4 MB for storing I/Q data.

**RPU –** This primarily comprises of a Meta (RPU-Meta) Processor running the MAC firmware and the MCP processor running the PHY firmware. A small testbench is introduced for the purpose of this testing which does the following –

1. Store the PHY transmit digital IQ samples to Host DDR memory
2. Once the transmit is completed, the same digital IQ samples are read back from the Host DDR memory and fed to the Receive path to complete the loopback.
3. Note that this test expects that the 64MB of DDR memory is accessible from the RPU.

# Description

1. Load the UCCP Firmware – refer to WLAN\_UCCP420/docs/Firmware Download.1.0.55.External.doc
2. Compile driver in MAC+PHY loop back mode

* Edit the Makefile of the driver, uncomment the line containing the BB\_LOOPBACK flag.

#ccflags-y += -DBB\_LOOPBACK

Note: For detailed command dumps uncomment the line

"#ccflags-y += -DCONFIG\_HAL\_DEBUG" in the Makefile.

* Compile the driver.

1. Bring up interface in production mode

# insmod uccp420.ko

# cat /proc/umac/params

# echo "production\_test=1" > /proc/umac/params

# iwconfig wlan0 mode ad-hoc

# iwconfig wlan0 essid prodtest

# iwconfig wlan0 channel 1

# ifconfig wlan0 up

1. Configure the Interface (using iwconfig/proc interface) and modify the “pktgen-work.sh” as per the TestCase parameters.
   1. To Configure data rate,

#echo tx\_fixed\_mcs\_indx=-1 >/proc/umac/params

#echo tx\_fixed\_rate=<datarate> >/proc/umac/params

where, <datarate> = 1,2,5.5,11,6,9,12,18,24,36,48,54

* 1. To Configure the MCS Index

#echo tx\_fixed\_rate=-1 >/proc/umac/params

#echo tx\_fixed\_mcs\_indx=<MCS> >/proc/umac/params

where, <MCS> = 0,1,2,3,4,5,6,7

* 1. To modify payload length in the pktgen script

For E.g., pgset "pkt\_size 1500"

* 1. To modify packet count in the pktgen script

For E.g., pgset "count 5"

1. Run baseband loop back test by running packet gen script provided in wlan/software/uccp/drivers/softmac/linux/scripts/pktgen-work.sh.

#./pktgen-work.sh

1. Check for the below messages for TestCase Result using ‘dmesg’ command on the linux terminal

PASS: “uccp420wlan\_rx\_frame: TX and RX packet contents match, TC PASS”

FAIL: “uccp420wlan\_rx\_frame: TX and RX packet contents \*doesn't match\*, TC FAIL”