Reference for WiFi Driver Efuse Buffermode Configuration

Version: 1.0
Release date: 2011-01-11

Document Revision History

Revision	Date	Author	Description
1	*01/11/11*	*Yilin.you*	*Usage of WIFI driver Efuse Buffermode*

Table of Contents

Document Revision History 2
Table of Contents 3
1 Introduction 4
2 Compile option & Related Commands 5
3 Common Case 6
4 Efuse/Buffermode QA 7

1 Introduction

This document mainly describes how to configure WIFI chip with correct Efuse and Buffermode parameters. (Some commands and common cases are introduced as example, which can be configured differently according to the actual situation)

Efuse is the hardware that restore wifi parameters for RT309x ,RT339x and RT539x chip. The store process could be implemented with two methods: one is to burn parameters to the Efuse directly , which is obviously costly and unpractical as the Efuse's burn time limit; and the other method is to read from buffer, for which the parameter is transferred from RT30xxEEPROM.bin. This method is called "buffermode". In condition that user want to configure wifi parameter based on the debugging result, if he/she chooses Efuse method, the configuration will be stored to hardware immediately after set, and if the "buffermode" method is chosen, he/she will have to load the configuration to RT30xxEEPROM.bin and store the bin binary to the flash so as to make sure the correct configuration still exists after system reboot.

2 Compile option & Related Commands

Compile option in WiFi driver: RTMP_EFUSE_SUPPORT
Compile option configured in SDK menuconfig:
1) TCSUPPORT_MTD_ENCHANCEMENT
2) TCSUPPORT_RESERVEAREA_BLOCK support block 1, 2, 3, 4

Related Commands

a. iwpriv ra0 set efuseFreeNumber=1

This command will calculate the number of the efuse's block. Tips: the number of writing effuse will be limited, one efuse has 45 blocks and there're 16 bytes in each block. Every block can be write form 0 to 1 and for cases write from 1 to 0 will call another block, which means attention when writing effuse.

b. iwpriv ra0 set efuseDump=1

dump all contents in efuse

c. iwpriv ra0 set efuseLoadFromBin=/filepath

Transfer binary file(locates in /filepath) into efuse

d. iwpriv ra0 set efuseBufferModeWriteBack=1

Transfer buffer content into etc/RT30xxEEPROM.bin

e.iwpriv ra0 e2p 2=0103

Set EEPROM content 0x0002 with 0x0103

f.iwpriv ra0 e2p 0

Read EEPROM content 0x0000

g.tcapi set WLan_Common EfuseBufferMode 0/1

Set mode with efuse(0) or buffermode(1), tips: the setting will not effect after reboot until tcapi save. You can check current mode with " tcapi get WLan_Common EfuseBufferMode" , 0 means effuse and 1 means buffermode

h.tcapi set WLan_Common WriteBinToFlash 1

Restore /etc/RT30xxEEPROM.bin into flash. Tips: the setting will not take effect until "tcapi commit WLan"

3 Common Case

Generally, the SDK release set Buffer mode as default value and will illustrate if set efuse. You can also check current mode with “tcapi get WLan_Common EfuseBufferMode”. The following cases are targeted to buffer mode:

Case1: Case 1:efuse is empty and none wifi parameters yet, and want to restore keys into flash and take effect after system reboot.

Step1: power on, the CPE will load the default “etc/RT30xxEEPROM.bin” into memory since no wifi paramets found in flash yet.

Step2: RD can use ATE command to debug WiFi parameters in production test, in that case you can input value into buffer using QAtool. Remember to execute “brctl delif br0 ra0” and “ated –bbr0 –ira0 -u” to make the QAtool work fluently.

Step3: Use “iwpriv ra0 set efuseBufferModeWriteBack=1” to load the test pass parameters into “/etc/RT30xxEEPROM.bin” after production test

Step4: “tcapi set WLan_Common WriteBinToFlash 1” and “tcapi commit WLan” to store /etc/RT30xxEEPROM.bin into reserve area of the flash.

Step5: System reboot, and the CPE will find binary file in flash and transmit it into /etc/RT30xxEEPROM.bin, then driver will load /etc/RT30xxEEPROM.bin since the setting is buffer mode. Eventually, the bin file stored in flash last time will take effect.

Case2: CPE status: efuse is empty and none wifi paramets stored in flash. Hope to restore debugging paramets into effuse and use the wifi parameters in efuse

Step1:Continue with step3 in Case1: iwpriv ra0 set efuseLoadFromBin=/etc/RT30xxEEPROM.bin

Step2: tcapi set WLan_Common EfuseBufferMode 0，tcapi save， system reboot

Case3:CPE status: efuse has been written while still want to debug and store wifi parameters again.

Step1:tcapi set WLan_Common EfuseBufferMode 0，tcapi save，reboot

Step2: Write efuse with e2p command

Case 4: CPE status: efuse has been written and none wifi driver in flash. Hope to debug and store wifi parameters into flash.

Step1:tcapi set WLan_Common EfuseBufferMode 1，tcapi save, reboot

Step2: None wifi parameters found in flash and using default configuration of RT30xxEEPRPOM.bin.

Step3:Continue with step2,step3,step4 in Case 1

4 Efuse/Buffermode QA

We have listed several common QA as follow:

Q1: How to know the current mode of wifi driver?

A: tcapi get WLan_Common EfuseBufferMode. 0 means efuse mode and 1 means buffer mode

Q2: I have confirmed the mode set id effuse(0), but still failed writing effuse with e2p command?

A: If your cpe's effuse is empty or information not enough, WIFI driver will ignore the configuration mode and forced to buffer mode so as to load normal wifi parameters for driver.

Q3: I have debugged with some Wifi parameters and loss them after reboot?

A: please confirm that you have execute these commands after debugging:

iwpriv ra0 set efuseBufferModeWriteBack=1

tcapi set WLan_Common WriteBinToFlash 1

tcapi commit WLan

Q4:Why the QAtool can not work ?

A: firstly, confirm that your WinPcap has been installed on you PC(which will be included in QAtool),secondly ,confirm that your PC lan has connected to CPE, thirdly, execute "brctl delif br0 ra0" and "ated –bbr0 –ira0 –u"

Q5: Efuse has been written on my CPE, and I want to store the paramets into flash and using buffer mode in future?

A: Confirm that you have configured with efuse mode and saved EEPROM content into efuse.bin using QAtool. Change mode into buffer mode : buffermode，tcapi set WLan_Common EfuseBufferMode 1，tcapi save, reboot. Then, click on load file in QA tool and load effuse.bin into buffer:

iwpriv ra0 set efuseBufferModeWriteBack=1

tcapi set WLan_Common WriteBinToFlash 1

tcapi commit WLan