A Brief Introduction to RTL8192C driver Power Saving

1.) RTL8192C consists of two power modes:

A). Minimum Power Saving mode.

In this PS mode, RTL8192C wakes up every beacon interval. Figure 1 shows the RTL8192C power consumption under Minimum Power Saving mode.
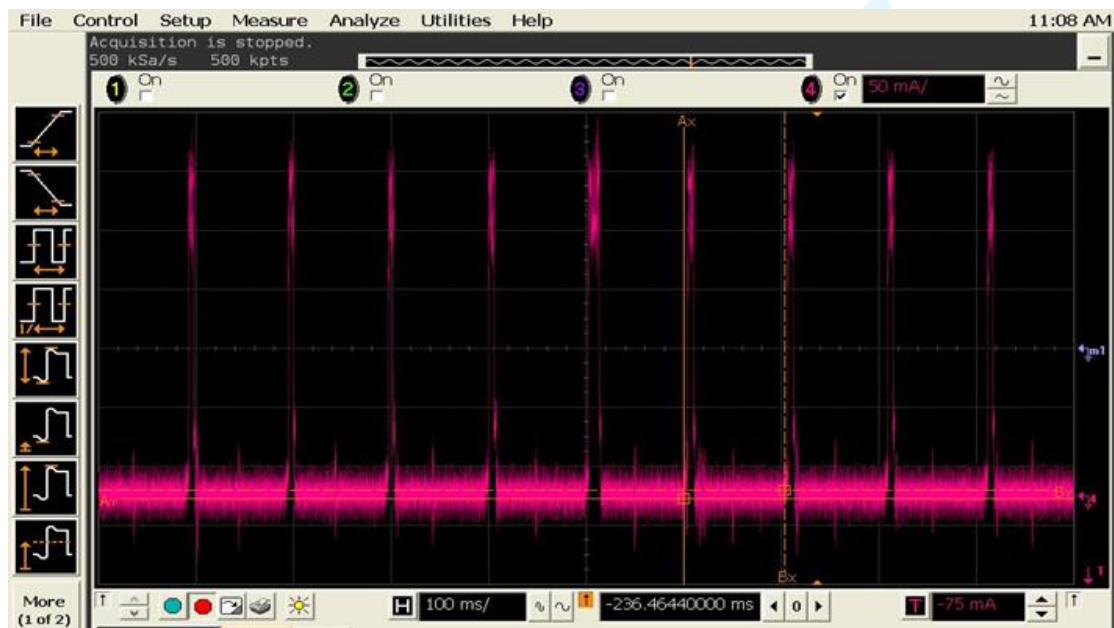

Figure 1. Minimum Power Saving mode (Associated idle)

B). Maximum Power Saving mode:

When driver be set to maximum power saving mode, which means RTL8192C wake up to receive beacon when DTIM count of beacon is zero.

If DTIM period of beacon is **3** RTL8192C would wake up every **3** bacon interval. Figure 2 illustrates RTL8192C power consumption under Maximum Power Saving mode:
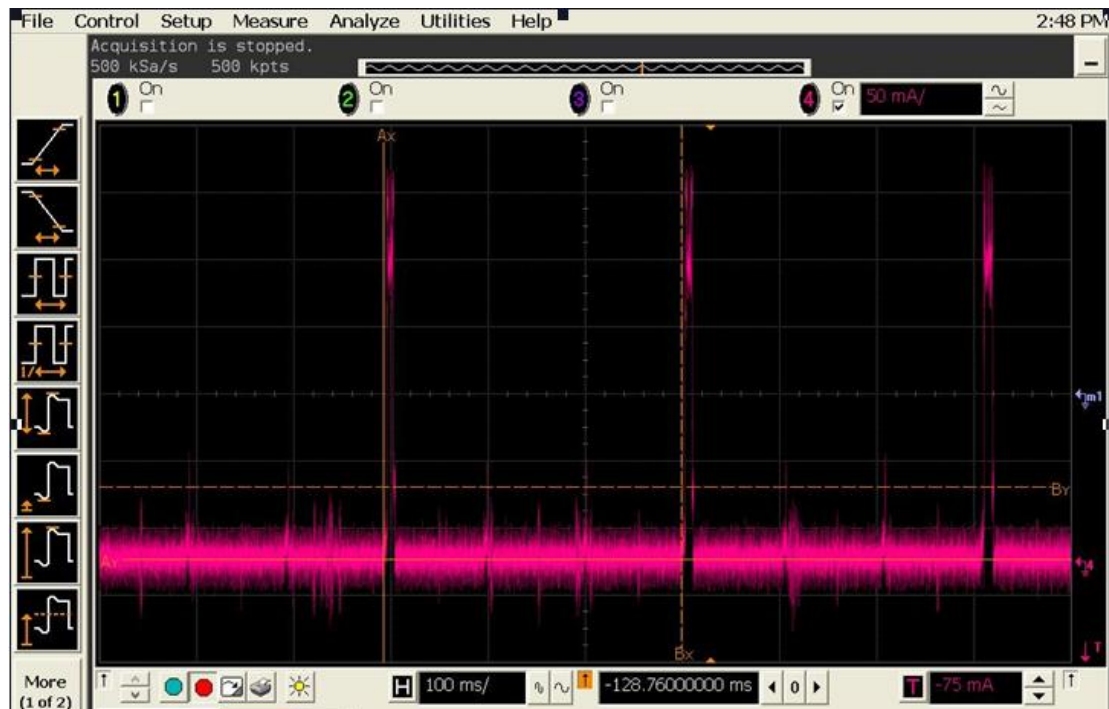

Figure 2. Maximum Power Saving mode (Associated idle)

2.) The following document will describe the way to let this driver to enable the power saving functionality.

==
A-1.) Power Saving Mode – by modify Makefile
==

You can easy set the power saving to **Min-PS** (rtw_power_mgnt=1) by modify the **CONFIG_POWER_SAVING** to **y**, at the line 20 of Makefile file.

Ex:

CONFIG_POWER_SAVING   =   y

==
A-2.) Power Saving Mode – by change load-time parameters
==

(1) In order to enter PS Mode, you need to add the parameter of "rtw_power_mgnt=1" when executing "insmod 8192cu.ko" :

        $>insmod 8192cu.ko rtw_power_mgnt=1
        or
        $>insmod 8192cu.ko rtw_power_mgnt=2

(2)Parameter Notes:

        rtw_power_mgnt=1 ;//enable PS, MIN_PS Mode
        rtw_power_mgnt=2 ;//enable PS, MAX_PS Mode

==
B-1.) USB Autosuspend - by modify Makefile
==

You can turn on USB autosuspend function by modify the **CONFIG_USB_AUTOSUSPEND** to **y**, at the line 21 of Makefile file.

Ex:

CONFIG_USB_AUTOSUSPEND   =   y

==
B-1.) USB Autosuspend - by change load-time parameters
==

(1) In PS mode, you can turn on or turn off the USB autosuspend functionality by adding the following parameter.

    $>insmod 8192cu.ko rtw_power_mgnt=1 rtw_enusbss = 1
      or

```
$>insmod 8192cu.ko rtw_power_mgnt=1 rtw_enusbss = 0
  or
$>insmod 8192cu.ko rtw_power_mgnt=2 rtw_enusbss = 1
  or
$>insmod 8192cu.ko rtw_power_mgnt=2 rtw_enusbss = 0
```

(2)Parameter Notes:

rtw_enusbss = 1; // enable USB autosuspend

rtw_enusbss = 0; // **default**, disable USB autosuspend

==
C-1.) RF on/off detection – by modify Makefile
==

You can turn on this function by modify the
CONFIG_HW_PWRP_DETECTION to **y**, at the line 22 of Makefile file.
Ex:
CONFIG_HW_PWRP_DETECTION   =   y

==
C-2.) RF on/off detection – by change load-time parameters
==

(1) If you want turn on this function in load time, you need to add the parameter
of " hwpwrp_detect =1" when executing "insmod 8192cu.ko" :

```
$>insmod 8192cu.ko rtw_hwpwrp_detect =1
```

(2)Parameter Notes:

rtw_hwpwrp_detect =1 ;//enable hw power pin detection

rtw_hwpwrp_detect =0 ;//disable hw power pin detection

==
D.) Two Levels of Unassociated Idle
==

In unassociated idle state, we can choose 2 levels of power saving as the
default setting: IPS_NORMAL and IPS_LEVEL_2. IPS_NORMAL has lower
power consumption, but needs to take more time than IPS_LEVEL_2 to wake
up, while the power consumption of IPS_LEVEL_2 is higher than
IPS_NORMAL about 10 mA but is faster to be woken up. IPS_LEVEL_2 is
suitable for users who care the transition speed more than power consumption

especially when using wifi on low speed IO interface such as USB1.1.

By default, the unassociated idle is IPS_NORMAL, You can change the default setting to IPS_LEVEL_2 with two ways:
1. Compilation time:
   Enabling flag "CONFIG_IPS_LEVEL2" in Include/autoconf.h

```
#ifdef CONFIG_IPS
#define CONFIG_IPS_LEVEL2
#endif
```

2. Driver module insertion time: insmod
   Insert module with "rtw_ips_mode=1"
```
$>insmod 8192cu.ko rtw_power_mgnt=1 rtw_ips_mode=1
```

3.) Power consumption example

Scenario (unit: mA) @5V	RTL8188CUS
Associated Idle	85.6
Unassociated Idle	80.2
Associated Idle (enable power save) - LPS	34.5
Unassociated Idle (enable power save) – IPS-NORMAL	18.9
Unassociated Idle (enable power save) – IPS-LEVEL_2	29
Unassociated Idle (enable selective suspend)	0.72