

QuecOpen

AZUREWAV Adaptation

Manual

LTE Module Series

Rev. QuecOpen_AZUREWAV_Driver_Adaptation_Manual

Date: 2018-06-13

Status: Preliminary

Our aim is to provide customers with timely and comprehensive service. For any assistance, please contact our company headquarters:

Quectel Wireless Solutions Co., Ltd.

7th Floor, Hongye Building, No.1801 Hongmei Road, Xuhui District, Shanghai 200233, China

Tel: +86 21 5108 6236

Email: info@quectel.com

Or our local office. For more information, please visit:

<http://quectel.com/support/sales.htm>

For technical support, or to report documentation errors, please visit:

<http://quectel.com/support/technical.htm>

Or email to: support@quectel.com

GENERAL NOTES

QUECTEL OFFERS THE INFORMATION AS A SERVICE TO ITS CUSTOMERS. THE INFORMATION PROVIDED IS BASED UPON CUSTOMERS' REQUIREMENTS. QUECTEL MAKES EVERY EFFORT TO ENSURE THE QUALITY OF THE INFORMATION IT MAKES AVAILABLE. QUECTEL DOES NOT MAKE ANY WARRANTY AS TO THE INFORMATION CONTAINED HEREIN, AND DOES NOT ACCEPT ANY LIABILITY FOR ANY INJURY, LOSS OR DAMAGE OF ANY KIND INCURRED BY USE OF OR RELIANCE UPON THE INFORMATION. ALL INFORMATION SUPPLIED HEREIN IS SUBJECT TO CHANGE WITHOUT PRIOR NOTICE.

COPYRIGHT

THE INFORMATION CONTAINED HERE IS PROPRIETARY TECHNICAL INFORMATION OF QUECTEL WIRELESS SOLUTIONS CO., LTD. TRANSMITTING, REPRODUCTION, DISSEMINATION AND EDITING OF THIS DOCUMENT AS WELL AS UTILIZATION OF THE CONTENT ARE FORBIDDEN WITHOUT PERMISSION. OFFENDERS WILL BE HELD LIABLE FOR PAYMENT OF DAMAGES. ALL RIGHTS ARE RESERVED IN THE EVENT OF A PATENT GRANT OR REGISTRATION OF A UTILITY MODEL OR DESIGN.

Copyright © Quectel Wireless Solutions Co., Ltd. 2018. All rights reserved.

About the Document

History

Revision	Date	Author	Description
1.0	2018-03-08	Edison YANG	Initial
1.1	2018-03-19	Edison YANG	Fix cannot send beacon issue
1.2	2018-06-13	Edison	Use kernel driver

Contents

About the Document	2
Contents	3
1 CM256 Hardware Connection Figure and GPIO Configuration	5
2 The Method to Add CM256 Driver	7
2.1. Add Kernel Driver Code	7
2.2. Firmware Generation	8
3 WIFI Feature Testing.....	9
3.1. Load WIFI Driver	9
3.2. Enable WIFI AP Mode.....	9
3.3. Enable WiFi STA Mode	10
3.3.1. Modify File wpa_supplicant.....	10
3.3.2. Enable Users' WiFi Hotspot, and Perform Network Construction Testing.....	11

NOTE

This documents only replies to Linux3.18.20 kernel.

1 CM256 Hardware Connection Figure and GPIO Configuration

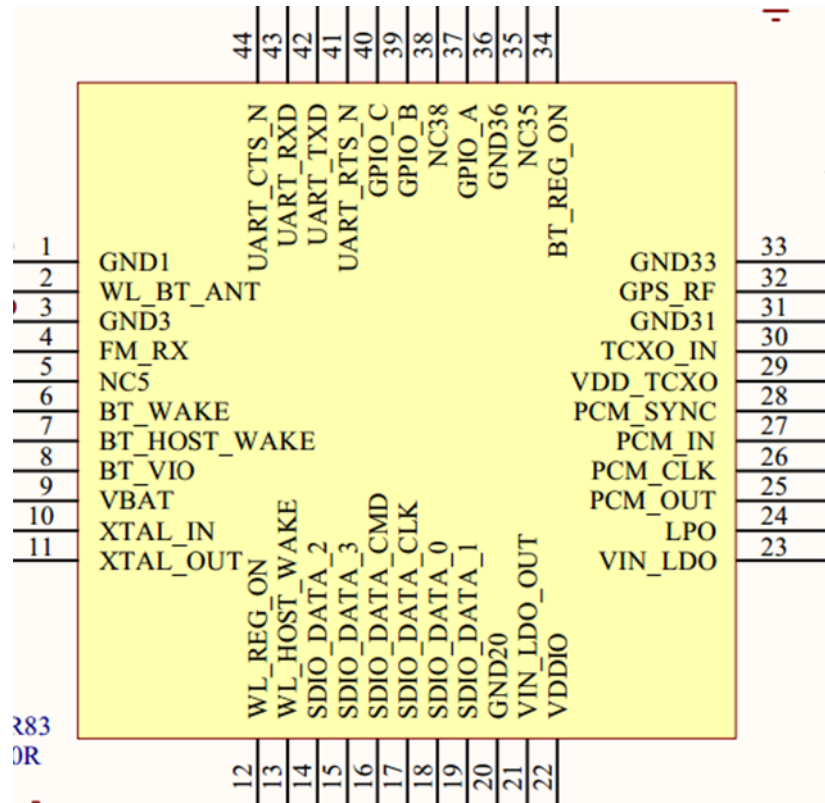


Figure 1: CM256 Hardware Connection Figure

Table 1: GPIO Configuration

CM256		MDM9x07		
PIN NAME	PIN NO.	SDIO Interface	PIN NO.	GPIO
SDIO_DATA_3	15	SDC1_DATA3	129	GPIO_12
SDIO_DATA_2	14	SDC1_DATA2	130	GPIO_13
SDIO_DATA_1	19	SDC1_DATA1	131	GPIO_14

SDIO_DATA_0	18	SDC1_DATA0	132	GPIO_15
SDIO_DATA_CLK	17	SDC1_CLK	133	GPIO_16
SDIO_DATA_CMD	16	SDC1_CMD	134	GPIO_17
WL_REG_ON	12	WLAN_EN	136	GPIO_38
WL_HOST_WAKE	13	WAKE_ON_WIRELESS	135	GPIO_59
VBAT	9	PM_ENABLE	127	GPIO_1020
VDDIO	22			

NOTE

INTEST PM_ENABLE connected to WL_REG_ON

2 The Method to Add CM256 Driver

2.1. Add Kernel Driver Code

(1) Copy the source code of CM256 driver to directory of ql-ol-sdk/ql-ol-kernel/drivers/net/wireless/bcmdhd.

(2) Modify ql-ol-sdk/ql-ol-kernel/drivers/net/wireless/Kconfig, please refer to below figure.

Add source "drivers/net/wireless/bcmdhd/Kconfig"

```

314 source "drivers/net/wireless/ath/Kconfig"
315 source "drivers/net/wireless/b43/Kconfig"
316 source "drivers/net/wireless/b43legacy/Kconfig"
317 source "drivers/net/wireless/brcm80211/Kconfig"
318 source "drivers/net/wireless/hostap/Kconfig"
319 source "drivers/net/wireless/ipw2x00/Kconfig"
320 source "drivers/net/wireless/iwlwifi/Kconfig"
321 source "drivers/net/wireless/iwlegacy/Kconfig"
322 source "drivers/net/wireless/libertas/Kconfig"
323 source "drivers/net/wireless/orinoco/Kconfig"
324 source "drivers/net/wireless/p54/Kconfig"
325 source "drivers/net/wireless/rt2x00/Kconfig"
326 source "drivers/net/wireless/rtlwifi/Kconfig"
327 source "drivers/net/wireless/ti/Kconfig"
328 source "drivers/net/wireless/zd1211rw/Kconfig"
329 source "drivers/net/wireless/mwifiex/Kconfig"
330 source "drivers/net/wireless/cw1200/Kconfig"
331 source "drivers/net/wireless/rsi/Kconfig"
332 source "drivers/net/wireless/cnss/Kconfig"
333 source "drivers/net/wireless/bcmdhd/Kconfig"
334

```

(3) Modify ql-ol-sdk/ql-ol-kernel/drivers/net/wireless/Makefile, please refer to below figure.

Add configuration obj-\$(CONFIG_BCMDHD) += bcmdhd

```

61 obj-$(CONFIG_CW1200) += cw1200/
62 obj-$(CONFIG_RSI_91X) += rsi/
63
64 obj-$(CONFIG_WCNSS_CORE) += wcnss/
65
66 obj-$(CONFIG_CNSS) += cnss/
67 obj-$(CONFIG_WCNSS_MEM_PRE_ALLOC) += cnss_prealloc/
68 obj-$(CONFIG_CNSS_CRYPTO) += cnss_crypto/
69 obj-$(CONFIG_BCMDHD) += bcmdhd/

```

(4) Modify ql-ol-sdk/ql-ol-kernel/arch/arm/configs/mdm9607-perf_defconfig, please refer to below figure.

Add configuration CONFIG_BCMDHD=m


```

400 CONFIG_CRYPTO_CRC32C=y
401 CONFIG_CRYPTO_DEV_QCRYPTO=y
402 CONFIG_CRYPTO_DEV_QCOM_MSM_QCE=y
403 CONFIG_CRYPTO_DEV_QCEDEV=y
404 CONFIG_QMI_ENCDEC=y
405 #will.shao, add wakelocks for sleep
406 CONFIG_PM_WAKELOCKS=y
407 #carl.yin, add for ap6212a1201.c
408 CONFIG_BCMDHD=m
409
410 #
411 # Quectel optional drivers
412 #
413 CONFIG_QUECTEL_DRIVER=y
414 # CONFIG_LOW_POWER_CONSUME is not set
415 # CONFIG_QTZONE is not set
416 CONFIG_QSTART=y
417 # CONFIG_QSMD is not set
418 # CONFIG_QSMEM is not set

```

Rebuild kernel, the steps are as follows.





- 1) #cd ~/ql-ol-sdk
- 2) #make clean
- 3) #make kernel_menuconfig
- 4) #make

2.2. Firmware Generation

(1) Copy WiFi Firmware

Please copy bcm_170607_nvram, bcm43455-7.45.100.9.bin, BCM434545.hcd to directory ~/temp/ql-ol-sdk/ql-ol-rootfs/etc/firmware/cm256/

Firmware List:

 bcm_170607_nvram	2017/8/2 11:11	文本文档	3 KB
 bcm43455-7.45.100.9.bin	2017/11/16 8:36	BIN 文件	521 KB
 BCM434545.hcd	2016/12/30 13:59	HCD 文件	53 KB
 wlan	2018/2/6 14:32	文件	5 KB

(2) Replace WLAN Script

Please replace ~/temp/ql-ol-sdk/ql-ol-rootfs/etc/init.d/wlan with the wlan script in above list.

(3) Delete file ql-ol-rootfs/etc/rc5.d/s91start_shortcut_fe_le in EC2X module

(4) Rebuild rootfs

```

# cd ~/temp/ql-ol-sdk
# make rootfs

```

3 WIFI Feature Testing

There are AT command, API and cmdline 3 methods to test CM256. Here only introduces cmdline.

3.1. Load WIFI Driver

```
# cd /etc/init.d
# ./wlan start
# iw dev wlan0 set 4addr on
# brctl addif bridge0 wlan0
```

```
root@mdm9607-perf:~# cd /etc/init.d/
root@mdm9607-perf:/etc/init.d# ./wlan start
root@mdm9607-perf:/etc/init.d# ifconfig
bridge0    Link encap:Ethernet  HWaddr 82:c0:09:70:c1:92
            inet addr:192.168.225.1  Bcast:192.168.225.255  Mask:255.255.255.0
            inet6 addr: fe80::60c0:9ff:fe7a:be8f/64  Scope:Link
            UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
            RX packets:0 errors:0 dropped:0 overruns:0 frame:0
            TX packets:6 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
            RX bytes:0 (0.0 B)  TX bytes:440 (440.0 B)

lo         Link encap:Local Loopback
            inet addr:127.0.0.1  Mask:255.0.0.0
            inet6 addr: ::1/128  Scope:Host
            UP LOOPBACK RUNNING  MTU:65536  Metric:1
            RX packets:0 errors:0 dropped:0 overruns:0 frame:0
            TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
            RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

rmnet0     Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
            UP RUNNING  MTU:2000  Metric:1
            RX packets:0 errors:0 dropped:0 overruns:0 frame:0
            TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

rndis0     Link encap:Ethernet  HWaddr 1A:2E:DA:D9:D4:F5
            inet addr:169.254.3.1  Bcast:169.254.3.255  Mask:255.255.255.0
            inet6 addr: fe80::182e:daff:fad9:d4f5/64  Scope:Link
            UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
            RX packets:0 errors:0 dropped:0 overruns:0 frame:0
            TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

wlan0      Link encap:Ethernet  HWaddr 00:11:22:33:06:13
            UP BROADCAST MULTICAST  MTU:1500  Metric:1
            RX packets:0 errors:0 dropped:0 overruns:0 frame:0
            TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:3000
            RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

Load WIFI driver successfully

```
root@mdm9607-perf:/etc/init.d#
root@mdm9607-perf:/etc/init.d# iw dev wlan0 set 4addr on
root@mdm9607-perf:/etc/init.d# brctl addif bridge0 wlan0
root@mdm9607-perf:/etc/init.d#
```

Bridging Operation

3.2. Enable WIFI AP Mode

For EC2X Module:

hostapd /etc/hostapd.conf -B (At this time, mobile terminal can be connected with WIFI module,

ssid=QSoftAP, password: 1234567890)
hostapd_cli (hostapd cmdline mode, could configure and query operation)

For AG35 Module:

hostapd /data/misc/wifi/hostapd.conf -B
hostapd_cli

If mobile terminal still cannot search hotspot after operating above steps normally, please query whether channel in file /etc/hostapd.conf set to 0, if yes, please modify it to 1, 6 or 13.

The image shows a terminal window with the following content:

```

root@mdm9607-perf:/etc# hostapd hostapd.conf -B
Configuration file: hostapd.conf
wlan0: interface state UNINITIALIZED->COUNTRY_UPDATE
root@mdm9607-perf:/etc#
root@mdm9607-perf:/etc# hostapd_cli
hostapd_cli v2.7
Copyright (c) 2004-2014, Jouni Malinen <j@w1.fi> and contributors

This software may be distributed under the terms of the BSD license.
See README for more details.

Selected interface 'wlan0'
Interactive mode
> help
Commands:
  mib                get MIB variables (dot1x, dot11, radius)
  sta <addr>         get MIB variables for one station
  all_sta            get MIB variables for all stations
  new_sta <addr>     add a new station
  deauthenticate <addr> deauthenticate a station
  disassociate <addr> disassociate a station
  sa_query <addr>    send SA Query to a station
  wps_pin <uuid> <pin> [timeout] [addr] add WPS Enrollee PIN
  wps_check_pin <PIN> verify PIN checksum
  wps_pbc            indicate button pushed to initiate PBC
  wps_cancel        cancel the pending WPS operation
  wps_ap_pin <cmd> [params...] enable/disable AP PIN
  wps_config <SSID> <auth> <encr> <key> configure AP
  wps_get_status    show current WPS status
  get_config        show current configuration
  help             show this usage help
  interface [ifname] show interfaces/select interface
  level <debug level> change debug level
  license           show full hostapd_cli license
  quit             exit hostapd_cli
> get_con
AAAAAAAA[[3~A[[3~
Unknown command 'get_con[3~[3~'
> get_conA[[DAHAH
Unknown command 'get_conn[n'
> get_config
bssid=00:11:22:33:06:13
ssid=QSoftAP
wps_state=disabled
key_mgmt=WPA-PSK
group_cipher=CCMP
rsn_pairwise_cipher=CCMP

```

Annotations in the image:

- A red arrow points from the text "Enable hostapd WiFi AP mode, at this time, mobile terminal can be connected with WiFi module, ssid=QSoftAP, password: 1234567890" to the command `hostapd hostapd.conf -B`.
- A red arrow points from the text "hostapd_cli command, users can configure and query configuration and information, such as get_config" to the `hostapd_cli` command and the `get_config` command in the help list.
- A yellow box highlights the output of the `get_config` command, showing the current configuration.

3.3. Enable WiFi STA Mode

3.3.1. Modify File wpa_supplicant

For EC2X module, please modify configuration file /etc/wpa_supplicant.conf, for AG35 module, please modify configuration file /data/misc/wifi/wpa_supplicant.conf, the example is shown as follows.

```

ctrl_interface=/var/run/wpa_supplicant

network={
    ssid="Fill in WiFi name, such as iphone 7"
    key_mgmt=WPA-PSK
    psk="Fill in WiFi password"
}

```

Fill in WiFi name in double quotes of `ssid=""`, it is recommended to use the English character ssid.
Fill in WiFi password in double quotes of `psk=""`.

After the modification as shown below.

```
root@mdm9607-perf:/etc# cat wpa_supplicant.conf
ctrl_interface=/var/run/wpa_supplicant

network={
    ssid="Quectel_TEST"
    key_mgmt=WPA-PSK
    psk="1234567890"
}
```

3.3.2. Enable Users' WiFi Hotspot, and Perform Network Construction Testing.

The first step:

```
# /etc/init.d/wlan start
```

The second step:

For EC2X module:

```
# wpa_supplicant -Dnl80211 -iwlan0 -c/etc/wpa_supplicant.conf
```

For AG35 module:

```
# wpa_supplicant -Dnl80211 -iwlan0 -c/data/misc/wifi/wpa_supplicant.conf
```

STA network construction successful, If ping each other between AP and STA is required, need get IP operation via ahcp. Specific command is as follows.

```
# udhcpc -i wlan0 -s /etc/udhcpc.d/50default
```

```
root@mdm9607-perf:/etc# wpa_supplicant -Dnl80211 -iwlan0 -c/etc/wpa_supplicant.conf
Successfully initialized wpa_supplicant
eap_proxy:eap_proxy_get_imsi: Not initialized

eap_proxy: eap_proxy initializing for DUAL SIM build 2
eap_proxy: QMI uim service client initialized with success 0x1 0

eap_proxy: QMI_UIM_EVENT_REG_REQ_V01, qmi_err_code: 0x0 wpa_uim[0].qmi_uim_svc_client_ptr =0x1Error=0x0
eap_proxy: event_resp_msg.event=1,

eap_proxy: reading card 1 values
eap_proxy: QMI UIM service is not initialized for sim = 0
eap_proxy: Error while reading SIM card status
eap_proxy: QMI auth service client initialized with success 0x2 eapol_proxy=0xd98e8
eap_proxy: QMI uim service client initialized with success 0x3 0

eap_proxy: QMI_UIM_EVENT_REG_REQ_V01, qmi_err_code: 0x0 wpa_uim[1].qmi_uim_svc_client_ptr =0x3Error=0x0
eap_proxy: event_resp_msg.event=1,

eap_proxy: reading card 2 values
eap_proxy: QMI UIM service is not initialized for sim = 1
eap_proxy: Error while reading SIM card status
eap_proxy: QMI auth service client initialized with success 0x4 eapol_proxy=0xd98e8
eap_proxy: Eap_proxy initialized successfully
wlan0: Trying to associate with SSID 'Quectel_TEST'
eap_proxy: eap_proxy_notify_config

eap_proxy: eap_proxy_allowed_method
wlan0: Associated with ba:d7:a
wlan0: WPA: Key negotiation completed with ba:d7:a [PTK=CCMP GTK=CCMP]
wlan0: CTRL-EVENT-CONNECTED - Connection to ba:d7:a completed [id=0 id_str=]
```

Network construct
successfully