

# **Wi-Fi User Guide for Linux**

AMPAK CONFIDENTIAL

## Revision History

Date	Version	Description	Author
2014/11/17	0.1	1. Initial revision of WLAN function and Wi-Fi driver version is 1.28.23.1	Terence Hsieh
2015/10/30	0.2	1. Modify GPIO Configuration in Wi-Fi Driver 2. Modify WLAN MAC Address Configuration 3. Add SoftAP configuration	Terence Hsieh
2016/10/30	0.3	1. Upgrade driver to 1.201.59.7	Terence Hsieh

AMPAK CONFIDENTIAL

# TABLE OF CONTENTS

INTRODUCTION .....	1
WLAN SOFTWARE ARCHITECTURE OVERVIEW .....	2
BROADCOM WLAN DONGLE BASIC CONCEPT .....	2
WLAN DONGLE OVERVIEW .....	2
WLAN SOFTWARE PACKAGE .....	3
Wi-Fi DRIVER AND KERNEL CONFIGURATION.....	4
ENABLE CFG80211 AND REMOVE CONFIG_BRCMFMAC IN LINUX KERNEL ..	4
DRIVER CONFIGURATION .....	4
GPIO CONFIGURATION IN Wi-Fi DRIVER .....	5
STATION MODE OPERATION.....	7
INSTALL DONGLE HOST DRIVER .....	7
ADD WPA SUPPLICANT CONFIGURATION FILE.....	7
INITIAL WPA SUPPLICATION.....	8
SCAN NETWORK .....	9
SOFTAP MODE OPERATION .....	10
INSTALL DONGLE HOST DRIVER .....	10
ADD HOSTAPD CONFIGURATION FILE .....	10
INITIAL HOSTAPD .....	11
WLAN OOB (OUT-OF-BAND) INTERRUPT MODE .....	12
KERNEL CONFIGURATION.....	12
WLAN MAC ADDRESS CONFIGURATION.....	13

# LIST OF FIGURES

Figure 1: Broadcom SDIO WLAN Dongle Concept.....3

Figure 2: Wi-Fi driver configuration in menuconfig.....4

Figure 3: Connect to WLAM AP .....8

Figure 4: Scan WLAN network.....9

Figure 5: OOB configuration in menuconfig ..... 12

AMPAK CONFIDENTIAL

## LIST OF TABLE

AMPAK CONFIDENTIAL

## **INTRODUCTION**

This user guide is intended to give Ampak Wi-Fi module users a general guide of how to enable the WLAN in Linux operating system.

AMPAK CONFIDENTIAL

# **WLAN SOFTWARE ARCHITECTURE OVERVIEW**

## **BROADCOM WLAN DONGLE BASIC CONCEPT**

The WLAN software package contains the dongle host driver for the host, a downloadable binary image for WLAN dongle, and management utilities.

The wireless driver runs on the WLAN dongle. The SDIO host controller passes IEEE 802.3 packets, and the necessary control packets, back and forth over the SDIO bus. A special Broadcom Device Class protocol is used to encapsulate control packets on a separate logical control channel and to add packet information to the data channel.

The advantage of using the dongle concept is that the wireless driver is executed externally from a host device, which means the host device does not have to use CPU or memory resources in order to execute the wireless driver's functionality. The use of the dongle provides the following benefits to the host:

- Power savings
- A reduction in driver size and complexity
- Processor offloading for activities such as checksum calculation and Address Resolution Protocol (ARP) execution

## **WLAN DONGLE OVERVIEW**

The Dongle Host Driver (DHD) is the executable module that provides encapsulated communication between the host device and the Ampak module over the SDIO bus.

The dongle software architecture is based on two major components:

- Dongle Host Driver: A host-based driver used to provide a communication channel with the dongle device firmware.

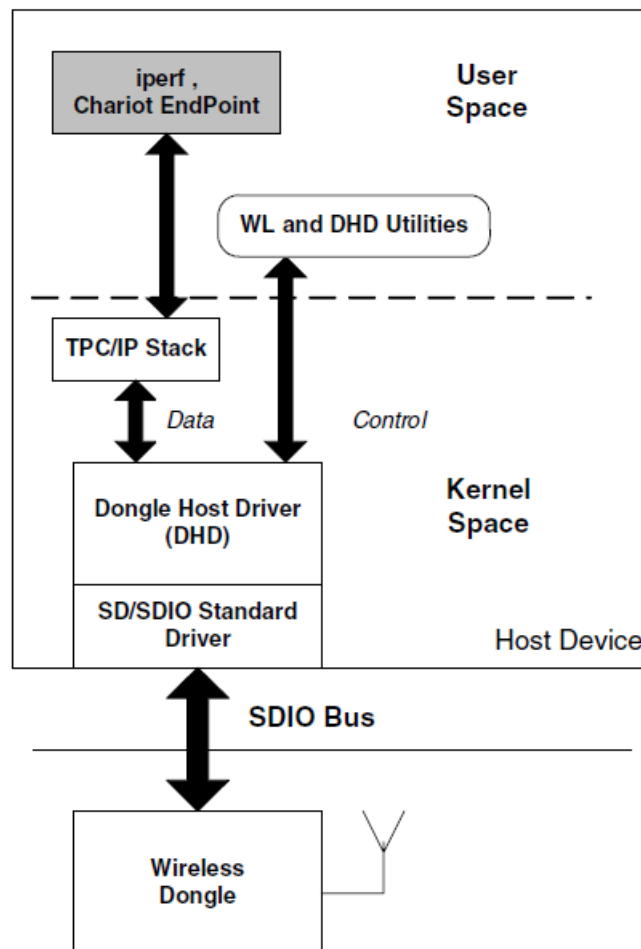


Figure 1: Broadcom SDIO WLAN Dongle Concept

## WLAN SOFTWARE PACKAGE

The provided WLAN software package contains following files:

- Dongle host driver (bcmhdhd.ko)
- Dongle device firmware (fw\_bcmdhd.bin)
- NVRAM (nvram.txt)



# WI-FI DRIVER AND KERNEL CONFIGURATION

## ENABLE CFG80211 AND REMOVE CONFIG\_BRCMFMAC IN LINUX KERNEL

CONFIG\_CFG80211=y

# CONFIG\_BRCMFMAC is not set

## DRIVER CONFIGURATION

Put bcmdhd driver to kernel/drivers/net/wireless/bcmdhd and modify following for building bcmdhd driver

### 1. Add following to kernel/drivers/net/wireless/Kconfig

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

### 2. Add following to kernel/drivers/net/wireless/Makefile

obj-\$(CONFIG\_BCMDHD) += bcmdhd/

### 3. Wi-Fi driver configuration

Edit proper firmware/nvram/config path and choose correct interrupt type for your device and then build kernel image and modules.

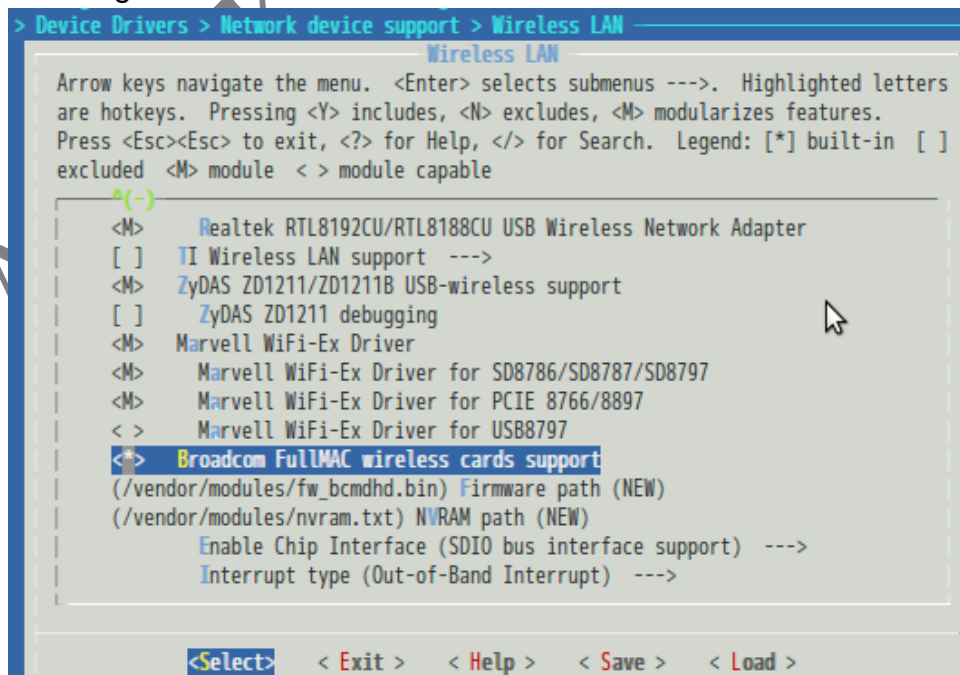


Figure 2: Wi-Fi driver configuration in menuconfig

## GPIO CONFIGURATION IN WI-FI DRIVER

### 1. WL\_REG\_ON and WL\_HOST\_WAKE Configuration

Please check your schematic and fill right GPIO number which connected to WL\_REG\_ON and WL\_HOST\_WAKE.

```
int dhd_wlan_init_gpio(void)
{
    uint host_oob_irq_flags = 0;

    gpio_wl_reg_on = 330; // WL_REG_ON is the input pin of WLAN module
    gpio_wl_host_wake = 324; // WL_HOST_WAKE is output pin of WLAN module
    ...
}
```

### 2. Add CUSTOMER\_HW\_PLATFORM into Makefile and implement card detection in dhd\_gpio.c

```
DHDOFILES += dhd_gpio.o
```

```
DHDCFLAGS += -DCUSTOMER_HW -DDHD_OF_SUPPORT
```

```
DHDCFLAGS += -DCUSTOMER_HW_PLATFORM
```

```
static int dhd_wlan_set_carddetect(bool present)
{
    int err = 0;

    #if !defined(BUS_POWER_RESTORE)
        if (present) {
            printf("==== Card detection to detect SDIO card! =====\n");
        }
    #endif
    #ifdef CUSTOMER_HW_PLATFORM
        err = sdhci_force_presence_change(&sdmmc_channel, 1);
    #endif /* CUSTOMER_HW_PLATFORM */
    } else {
        printf("==== Card detection to remove SDIO card! =====\n");
    }
    #ifdef CUSTOMER_HW_PLATFORM
        err = sdhci_force_presence_change(&sdmmc_channel, 0);
    #endif /* CUSTOMER_HW_PLATFORM */
}
```

```
#endif
```

```
    return err;
```

```
}
```

```
/* where sdhci_force_presence_change is the card detection function */
```

AMPAK CONFIDENTIAL

## **STATION MODE OPERATION**

This is a quick example for Wi-Fi connection and you can refer to the website of WPA Supplicant for more detail.

### **INSTALL DONGLE HOST DRIVER**

1. If bcmdhd driver is built a kernel module, you will need to insert bcmdhd.ko by yourself.

```
# insmod /lib/modules/bcmdhd.ko "firmware_path=/etc/firmware/fw_bcmdhd.bin  
nvram_path=/etc/firmware/nvram.txt"  
# ifconfig wlan0 up
```

2. If bcmdhd driver is built in kernel image, you will need to configure firmware path.

```
# echo -n "/etc/firmware/fw_bcmdhd.bin" > /sys/module/bcmdhd/parameters/firmware_path  
# ifconfig wlan0 up
```

### **ADD WPA SUPPLICANT CONFIGURATION FILE**

Please create wpa\_supplicant.conf file with following content:

```
ctrl_interface=/var/run/wpa_supplicant
```

#### **# Open system without encryption**

```
network={  
    ssid="tttb"  
    key_mgmt=NONE  
}
```

#### **# Open/Shared authentication with WEP encryption**

```
network={  
    ssid="tttb"  
    key_mgmt=NONE  
    auth_alg=OPEN SHARED  
    wep_key0=1234567890  
}
```

## # WPA/WPA2-PSK authentication with TKIP/AES encryption

```
network={
    ssid="tttb"
    psk="12345678"
}
```

## INITIAL WPA SUPPLICANT

```
#wpa_supplicant -Dnl80211 -i wlan0 -c wpa_supplicant.conf -d&
```

```
D/MediaScannerService( 248): done scanning volume external
D/wpa_supplicant( 360): Initializing interface 'wlan0' conf '/data/wifi/wpa-psk.conf'
D/wpa_supplicant( 360): Configuration file '/data/wifi/wpa-psk.conf' -> '/data/wifi/wpa-psk.conf'
D/wpa_supplicant( 360): Reading configuration file '/data/wifi/wpa-psk.conf'
D/wpa_supplicant( 360): ctrl_interface='/data/misc/wifi/sockets'
D/wpa_supplicant( 360): Priority group 0
D/wpa_supplicant( 360): id=0 ssid='tttb'
D/wpa_supplicant( 360): Initializing interface (2) 'wlan0'
D/wpa_supplicant( 360): Interface wlan0 set UP - waiting a second for the driver to c
D/wpa_supplicant( 360): SIOCGIWRange: WE(compiled)=22 WE(source)=19 enc_capa=0xf
D/wpa_supplicant( 360): capabilities: key_mgmt 0xf enc 0xf flags 0x0
D/wpa_supplicant( 360): WEXT: Operstate: linkmode=1, operstate=5
D/wpa_supplicant( 360): Own MAC address: 00:22:f4:02:90:c5
D/wpa_supplicant( 360): WPA: Installing PTK to the driver.
D/wpa_supplicant( 360): wpa driver wext set key: alg=3 key_idx=0 set_tx=1 seq_len=6 key_len=16
D/wpa_supplicant( 360): EAPOL: External notification - portValid=1
D/wpa_supplicant( 360): State: 4WAY HANDSHAKE -> GROUP HANDSHAKE
I/wpa_supplicant( 360): CTRL-Event-STATE-CHANGE id=0 state=6 BSSID=00:00:00:00:00:00
D/wpa_supplicant( 360): WPA: Installing GTK to the driver (keyidx=1 tx=0 len=32).
D/wpa_supplicant( 360): wpa driver wext set key: alg=2 key_idx=1 set_tx=0 seq_len=6 key_len=32
I/wpa_supplicant( 360): WPA: Key negotiation completed with 00:0a:79:bf:ee:d0 [PTK=CCMP GTK=TKIP]
D/wpa_supplicant( 360): Cancelling authentication timeout
D/wpa_supplicant( 360): State: GROUP HANDSHAKE -> COMPLETED
I/wpa_supplicant( 360): CTRL-Event-STATE-CHANGE id=0 state=7 BSSID=00:00:00:00:00:00
I/wpa_supplicant( 360): CTRL-Event-CONNECTED - Connection to 00:0a:79:bf:ee:d0 completed (reauth) [id=0 id_str=
D/wpa_supplicant( 360): wpa driver wext set operstate: operstate 0->1 (UP)
D/wpa_supplicant( 360): WEXT: Operstate: linkmode=-1, operstate=6
D/wpa_supplicant( 360): EAPOL: External notification - portValid=1
D/wpa_supplicant( 360): EAPOL: External notification - EAP success=1
D/wpa_supplicant( 360): EAPOL: SUPP_PAE entering state AUTHENTICATING
D/wpa_supplicant( 360): EAPOL: SUPP_BE entering state SUCCESS
D/wpa_supplicant( 360): EAP: EAP entering state DISABLED
D/wpa_supplicant( 360): EAPOL: SUPP_PAE entering state AUTHENTICATED
D/wpa_supplicant( 360): EAPOL: SUPP_BE entering state IDLE
D/wpa_supplicant( 360): EAPOL authentication completed successfully
D/wpa_supplicant( 360): RTM_NEWLINK: operstate=1 ifi flags=0x11043 ([UP][RUNNING][LOWER UP])
```

```
# wl ssid
Current SSID: "tttb"
# wl bssid
00:0A:79:BF:EE:D0
# ifconfig wlan0 192.168.1.99
# netcfg
lo        UP        127.0.0.1        255.0.0.0        0x00000049
ifb0      DOWN    0.0.0.0          0.0.0.0          0x00000082
ifb1      DOWN    0.0.0.0          0.0.0.0          0x00000082
usb0      DOWN    0.0.0.0          0.0.0.0          0x00001002
sit0      DOWN    0.0.0.0          0.0.0.0          0x00000080
ip6tnl0   DOWN    0.0.0.0          0.0.0.0          0x00000080
wlan0     UP        192.168.1.99     255.255.255.0    0x00001043
# ping -c 3 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=1057 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=109 ms

--- 192.168.1.1 ping statistics ---
3 packets transmitted, 2 received, 33% packet loss, time 2008ms
rtt min/avg/max/mdev = 109.942/583.635/1057.328/473.693 ms, pipe 2
```

Figure 3: Connect to WLAM AP

## SCAN NETWORK

```
#wpa_cli -i wlan0 -p /var/run/wpa_supplicant/ scan
```

```
#wpa_cli -i wlan0 -p /var/run/wpa_supplicant/ scan_results
```

```
# wpa_cli -i wlan0 -p /data/misc/wifi/sockets scan
OK
#
# wpa_cli -i wlan0 -p /data/misc/wifi/sockets scan_results
bssid / frequency / signal level / flags / ssid
00:0a:79:bf:ee:d0      2452      241      [WPA-PSK-TKIP+CCMP][WPA2-PSK-TKIP+CCMP][WPS]      ttth
00:0a:79:c8:d1:48      2462      228      [WPA2-PSK-CCMP][WPS]      ttth
00:1c:0f:a3:22:1c      2412      204      [WPA-PSK-TKIP]      Navigator-SMB
4c:e6:76:71:aa:06      2437      204      [WPA-PSK-TKIP+CCMP][WPA2-PSK-TKIP+CCMP][WPS]      4CE67671AA06
00:90:4b:11:88:80      2417      203      [WPA-PSK-CCMP][WPA2-PSK-CCMP][WPS]      BUFFALO-118880-1
00:0a:79:ed:fd:b0      2442      203      [WPA-PSK-TKIP+CCMP][WPA2-PSK-TKIP+CCMP][WPS]      000A79EDFDB1_ngb
00:94:56:32:13:21      2412      202      [WPA2-PSK-CCMP][WPS]      ASUS 2G
00:0d:0b:5e:e5:3c      2462      202      [WPA-PSK-TKIP]      buffalo bruce
00:0c:43:45:44:80      2412      197      [WPA-PSK-CCMP][WPA2-PSK-CCMP][WPS]      belkin.3489
00:94:63:25:32:22      2462      193      [WPA-PSK-CCMP][WPS]      VIZIO_xwr200
00:94:56:33:14:56      2412      192      [WPA-PSK-CCMP][WPS]      VIZIO1206
00:90:4b:3d:6f:3a      2417      188      [WPA-PSK-TKIP]      tony_router
84:c9:b2:59:a7:58      2417      180      [WPA-PSK-TKIP][WPS]      beauty
00:90:4b:11:88:81      2412      198      [WEP]      BUFFALO-118880-3
00:12:17:a6:af:e2      2462      197      [WEP]      ---888---
00:16:16:22:c7:30      2412      190      [WEP]      MIS
00:1d:7e:a3:04:98      2437      183      [WEP]      valentim
00:0a:79:bf:ee:d1      2452      241      CG-Guest
```

Figure 4: Scan WLAN network

## **SOFTAP MODE OPERATION**

This is a quick example for SoftAP setup and you can refer to the website of HOSTAPD for more detail.

### **INSTALL DONGLE HOST DRIVER**

3. If bcmhdhd driver is built a kernel module, you will need to insert bcmhdhd.ko by yourself.  
# insmod /lib/modules/bcmhdhd.ko "firmware\_path=/etc/firmware/fw\_bcmhdhd\_ **apsta**.bin  
nvram\_path=/etc/firmware/nvram.txt"

4. If bcmhdhd driver is built in kernel image, you will need to configure firmware path.  
# echo -n "/etc/firmware/fw\_bcmhdhd\_ **apsta**.bin" >  
/sys/module/bcmhdhd/parameters/firmware\_path

### **ADD HOSTAPD CONFIGURATION FILE**

Please create hostapd.conf file with following content:

#### **# Open system without encryption**

```
interface=wlan0
driver=nl80211
ctrl_interface=/var/run/hostapd
ssid=AndroidAP
channel=6
ieee80211n=1
hw_mode=g
ignore_broadcast_ssid=0
```

#### **# or WPA2-PSK authentication with AES encryption**

```
interface=wlan0
driver=nl80211
ctrl_interface=/data/misc/wifi/hostapd
ssid=AndroidAP
channel=6
```

ieee80211n=1  
hw\_mode=g  
ignore\_broadcast\_ssid=0  
wpa=2  
rsn\_pairwise=CCMP  
wpa\_passphrase=12345678

## **INITIAL HOSTAPD**

#hostapd hostapd.conf &

AMPAK CONFIDENTIAL



# WLAN OOB (OUT-OF-BAND) INTERRUPT MODE

Host can enter sleep mode, but keep Ampak module alive. Once Ampak module receives packets, it can wake up host through a pre-defined GPIO pin.

## KERNEL CONFIGURATION

Please select interrupt type to Out-of-Band Interrupt in kernel configuration.

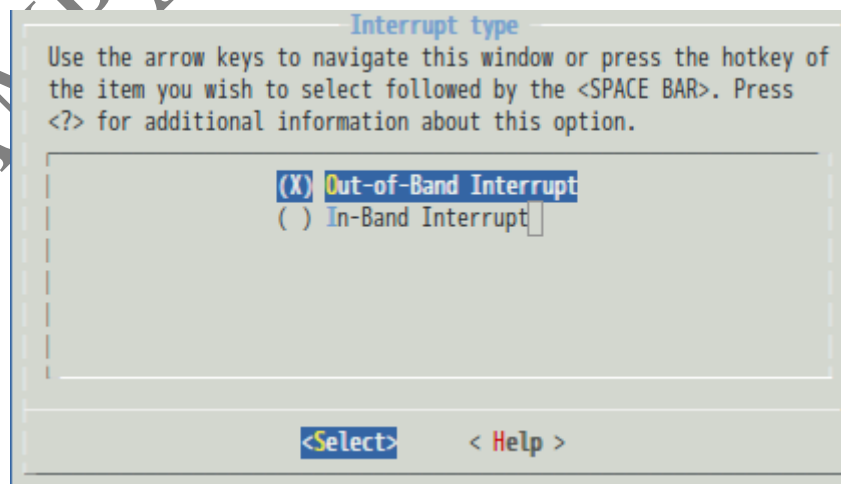
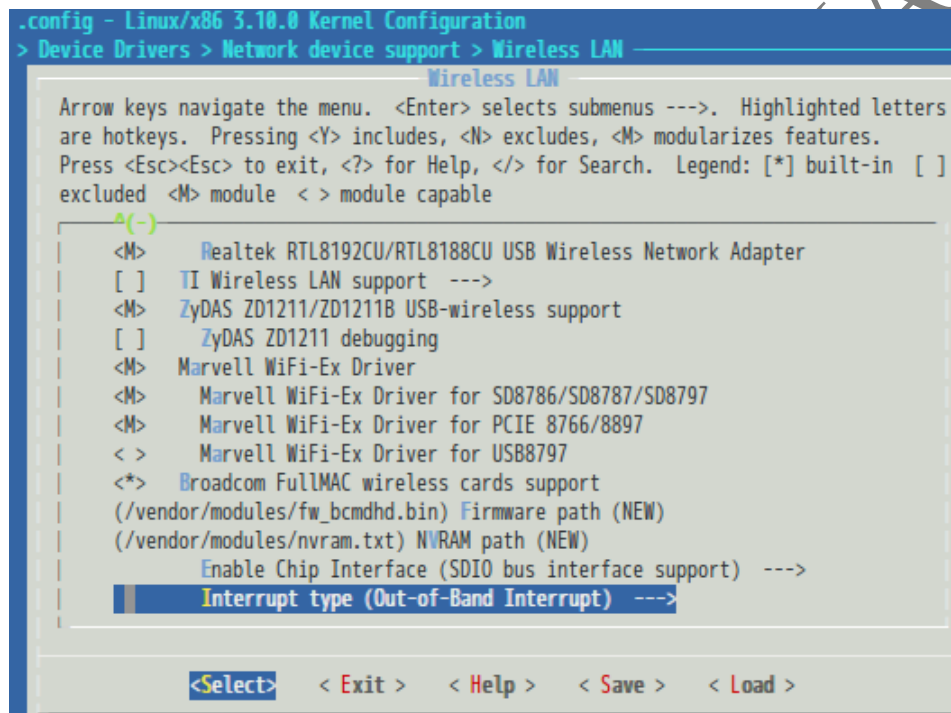


Figure 5: OOB configuration in menuconfig

## **WLAN MAC ADDRESS CONFIGURATION**

If you would like to configure Wi-Fi MAC address, you need to modify driver to get it work.

1. Add -DGET\_CUSTOM\_MAC\_ENABLE in driver Makefile
2. Modify dhd\_wlan\_get\_mac\_addr function in dhd\_gpio.c to read your MAC address where located in your system.
3. Then Wi-Fi driver will change firmware MAC address during initialization.

AMPAK CONFIDENTIAL