**NOTICE REGARDING QUALCOMM ATHEROS, INC.**

Effective June 2016, Qualcomm Atheros, Inc. (QCA) transferred certain of its assets, including substantially all of its products and services, to its parent corporation, Qualcomm Technologies, Inc. Qualcomm Technologies, Inc. is a wholly-owned subsidiary of Qualcomm Incorporated. Accordingly, references in this document to Qualcomm Atheros, Inc., Qualcomm Atheros, Atheros, QCA or similar references, should properly reference, and shall be read to reference, Qualcomm Technologies, Inc.

Qualcomm Atheros, Inc.
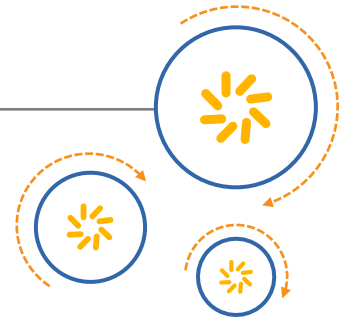
IPQ4019/IPQ4029 AP.DK04 Reference Design

Setup Guide

80-Y9700-6 Rev. F

June 29, 2016

Confidential and Proprietary – Qualcomm Atheros, Inc.

Revision history

Revision	Date	Description
A	September 2015	Initial release
B	November 2015	▪ Updated Table 2-1 AP.DK04 board feature mapping ▪ Added 2.2 Jumper configuration
C	January 2016	▪ Updated section 1.2 Overview ▪ Updated section 2 Connecting AP.DK04 Serial Port, Enclosure Version ▪ Added section 3.1 AP.DK04 assembly guidance
D	May 2016	CPU frequency changed to 716.8 MHz DDR frequency changed to 672 MHz
E	June 2016	▪ Section 1.2 Overview: updated DDR3L ▪ Chapter 2 Connecting AP.DK04 Serial Port, Enclosure Version: added a step to verify serial adapter board orientation ▪ Chapter 4 Configuring the COM Port: updated connection description ▪ Chapter 5 Image Flashing Procedure from U-Boot: added flashing procedures for each memory configuration ▪ Section 6.1.2 DBDC in Open Security: updated code
F	June 2016	Section 1.1, Purpose: Added FCC Notice.

Contents

1 Introduction

1.1 Purpose

This document describes the AP.DK04 reference design configuration and the interface capabilities of IPQ4019 and IPQ4029 processors.

FCC NOTICE: This kit is designed to allow:

(1) Product developers to evaluate electronic components, circuitry, or software associated with the kit to determine whether to incorporate such items in a finished product and

(2) Software developers to write software applications for use with the end product. This kit is not a finished product and when assembled may not be resold or otherwise marketed unless all required FCC equipment authorizations are first obtained. Operation is subject to the condition that this product not cause harmful interference to licensed radio stations and that this product accept harmful interference. Unless the assembled kit is designed to operate under part 15, part 18 or part 95 of the FCC's rules, the operator of the kit must operate under the authority of an FCC license holder or must secure an experimental authorization under part 5 of the FCC's rules.

1.2 Overview

The IPQ4019/IPQ4029 is a highly integrated system-on-chip (SoC) designed for high-performance, power efficient, and cost-effective 2×2, 802.11ac, dual-band concurrent access point applications. The SoC incorporates a quad-core ARM Cortex A7 processor, two dual-band, concurrent 802.11ac Wave-2 Wi-Fi subsystems, and a five-port Gigabit Ethernet Layer2/3/4 multilayer switch supporting line rate Hardware Network Address Translation (HNAT).

It supports one USB3.0, one USB2.0 and one PCIe2.0. It also supports miscellaneous interfaces such as SDIO, I^2S, SPDIF, I^2C, SMI, UART, and JTAG, which can be configured as general purpose I/O pins.

The AP.DK04 reference design board supports the following configurations:

- 4-layer board
- On-chip dual-band concurrent (DBDC) 2×2 2 GHz 802.11n (256 QAM) and 2×2 5 GHz 802.11ac
- DC 12 V, 3 A power adapters
- DDR3L 1344 16-bit interface, supports up to 512 MB
- 70 GPIOs multiplex with function interfaces (SPI/I^2C/JTAG/UART/WCI/PCM/I^2S/SPDIF)
- SPI NOR
- 8-bit parallel NAND flash

- eMMC/SD card

- LCD with touch panel

- PCIe-mini

- VoIP

- JTAG

- UART

- One USB3.0 and one USB2.0 interface

- 5 GE Ethernet ports

- Two reset buttons

  - One for hardware reset

  - One for software reset/manufactory default reset

2 Connecting AP.DK04 Serial Port, Enclosure Version

For the demo board with enclosure, attention should be paid to the serial connections (see Figure 2-1 and Figure 2-2).
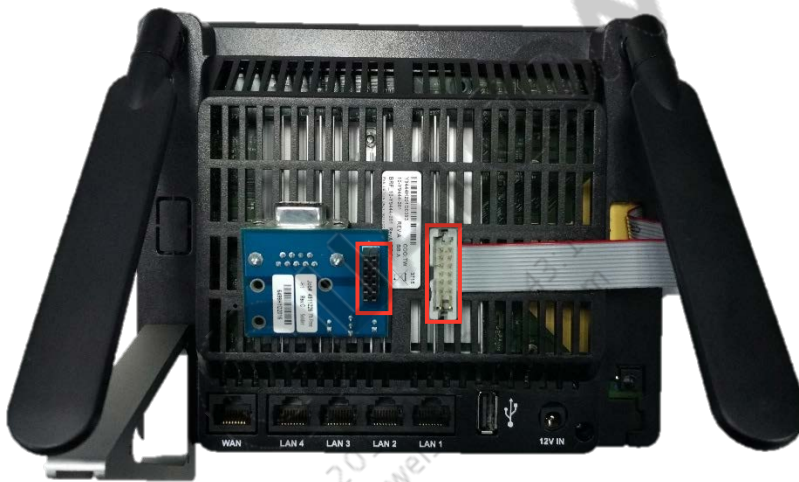
Figure 2-1 Serial connection before connection

Figure 2-2 Serial connection after connection

WARNING: Incorrect orientation can cause damage to the adapter and the AP.DK04 reference design when power is applied.

1. Use the RS232 cable to connect the AP.DK04 to an available COM/USB port on the PC. Connect the other end of the RS232 cable to the RS232 adapter.

NOTE: If the PC has more than one COM port, note the correct COM port number connected to the cable.

2. Before powering up the AP, verify if all the connections are correct.

3. Connect the 12 V DC power adapter to the power port of the reference design.

4. Check the level shifter IC on the serial adapter board during first power on for excessive rapid temperature rise, and power off immediately if this is detected. This indicates a wrong connection orientation of the serial adapter board, which will be damaged by prolonged excessive temperature.

MAY CONTAIN U.S. AND INTERNATIONAL EXPORT CONTROLLED INFORMATION

3 AP.DK04 Full Running Configuration

3.1 AP.DK04 assembly guidance

Figure 3-1 AP.DK04 after assembly

3.2 AP.DK04 reference design configuration

Table 3-1 AP.DK04 running frequency

Blocks	Current Frequency
Cortex A7 *4	716.8 MHz
DDR3L	672 MHz

Table 3-2 AP.DK04 board feature mapping

Functions	Application 1 (Y9761-H100/H110/H111/H200/H210/H211/H220/H213/H112)	Application 2 (Y9761-H201/H221)	Application 3 (Y9761-H202/H222)
2×2 2 GHz + 2×2 5 GHz DBDC	√	√	√
SPI NOR Flash 16 Mbytes	√	√	×
Parallel NAND Flash (1 Gbit, x8 bus)	√(Boot from)	×	√(Boot from)
DDR3L (128 M x 16 bit)	√	√	√
PSGMII (4 Port GE LAN, 1 GE WAN)	√	√	√
MDIO	√	√	√
USB3.0/2.0	√	√	√
USB2.0	√	√	√
MMC plus slot (SD card or special MMC board)	√	√	√
EJTAG	√	√	√
UART0 for BLSP	√	√	√
UART1(HS) for BLSP/WIFI0/WIFI1	√	√	√
5 inch TFT 800x480 LCD with touch panel (LCD+I2C0)	√	×	√
PCIe2.0 standard interface	√	√	√
9 LEDs (7 LEDs for 2 wire mode control, 2 LEDs for PCIe and 3.3 V power)	√	√	√
WCI	√	√	√
Audio interface (including I^{2}S/SPDIF/PWM)	×	√	×
I^{2}C0 for AKD4613 configuration	×	√	×
SPI1+PCM to support VoIP	×	√	√

NOTE: The running configuration of the AP.DK04 may vary with new AP.DK04 software releases or new versions of the AP.DK04 reference design.

The AP.DK04 reference design kit includes:

- The AP.DK04 reference design

- 12 V power adaptor

- RS232 UART adapter
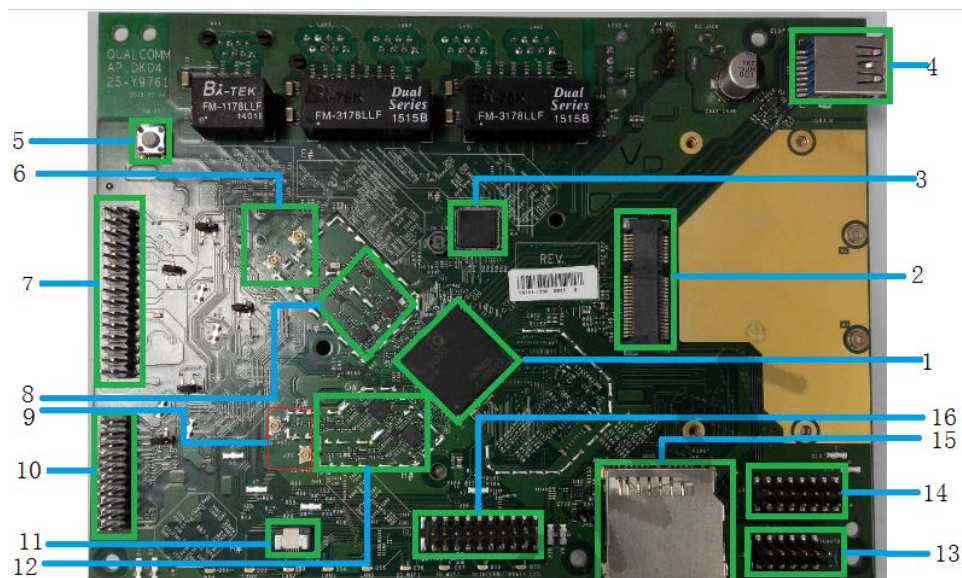
Figure 3-2 AP.DK04 reference design – front view
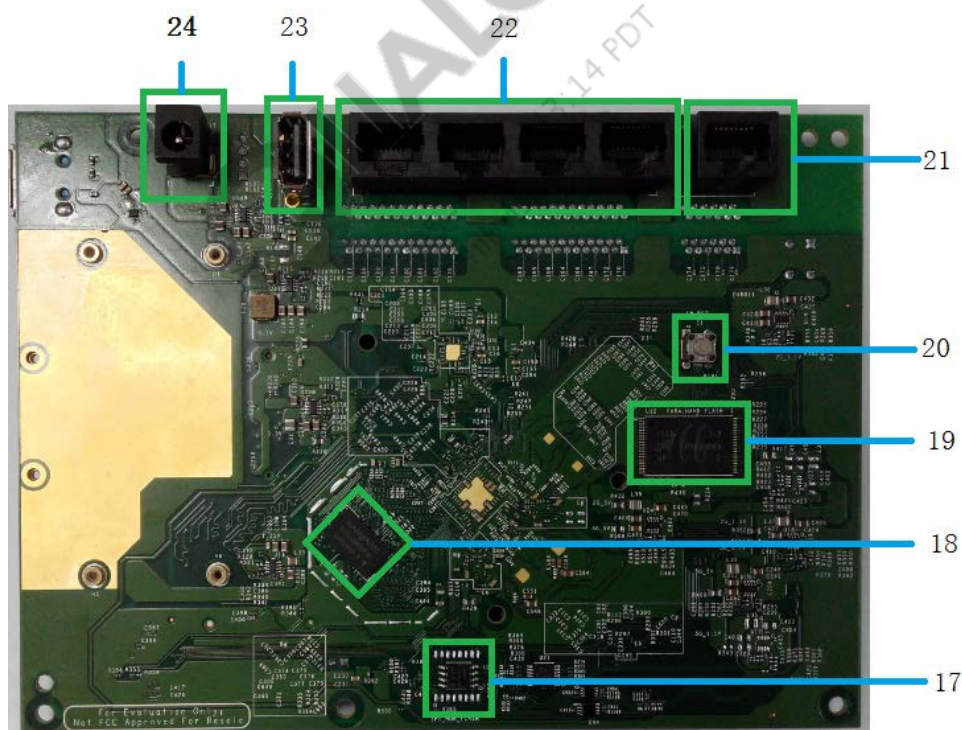
Figure 3-3 AP.DK04 reference design – bottom view

Table 3-3 AP.DK04 interface details

Item	Description
1	IPQ4019/ IPQ4029
2	PCIe mini connector
3	QCA8075 5-port GE PHY
4	USB 3.0
5	Button: Hardware reset to chip power on reset
6	2 GHz antenna connector Down J27, 2 GHz Chain 0 Top J26, 2 GHz Chain 1
7	VoIP connector
8	2 GHz XLNA/XPA
9	5 GHz antenna connector Down J30, 5 GHz Chain 0 Top J28, 5 GHz Chain 1
10	EJTAG connector
11	LCD touch panel controller connector
12	5 GHz XLNA/XPA
13	System UART0
14	UART1, reserved for Bluetooth
15	eMMC/SD card slot
16	Audio /LCD connector
17	SPI NOR flash
18	DDR3 memory
19	Parallel NAND flash
20	Software reset button
21	WAN port
22	4 LAN port
23	USB 2.0 connector
24	12 V power connector

3.3 Jumper configuration

There are 5 jumpers on AP.DK04 for bootstrapping.

Table 3-4 Jumper settings

Pin ID	Pin name boot_config[n]	Alternate function	Jumper	Function description
K3 V25	1 7	GPIO14 GPIO51	J36 J37	▪ 00 <GPIO51:GPIO14> = Boot interface is SPI ▪ 01 <GPIO51:GPIO14> = Boot interface is EMMC ▪ 10 <GPIO51:GPIO14> = Boot interface is Parallel NAND ▪ 11 <GPIO51:GPIO14> = Boot from USB
A18	0	GPIO3	J40	Apps authentication Enable. Enables authentication for various AP code segments. ▪ 0 = No authentication ▪ 1 = Enable authentication
AB24	3	GPIO36	J38	▪ 0 = Boot from code RAM ▪ 1 = Boot from ROM
U25	9	GPIO55	J39	▪ 0 = Normal boot ▪ 1 = Force boot from USB

Figure 3-4 Jumper settings

MAY CONTAIN U.S. AND INTERNATIONAL EXPORT CONTROLLED INFORMATION
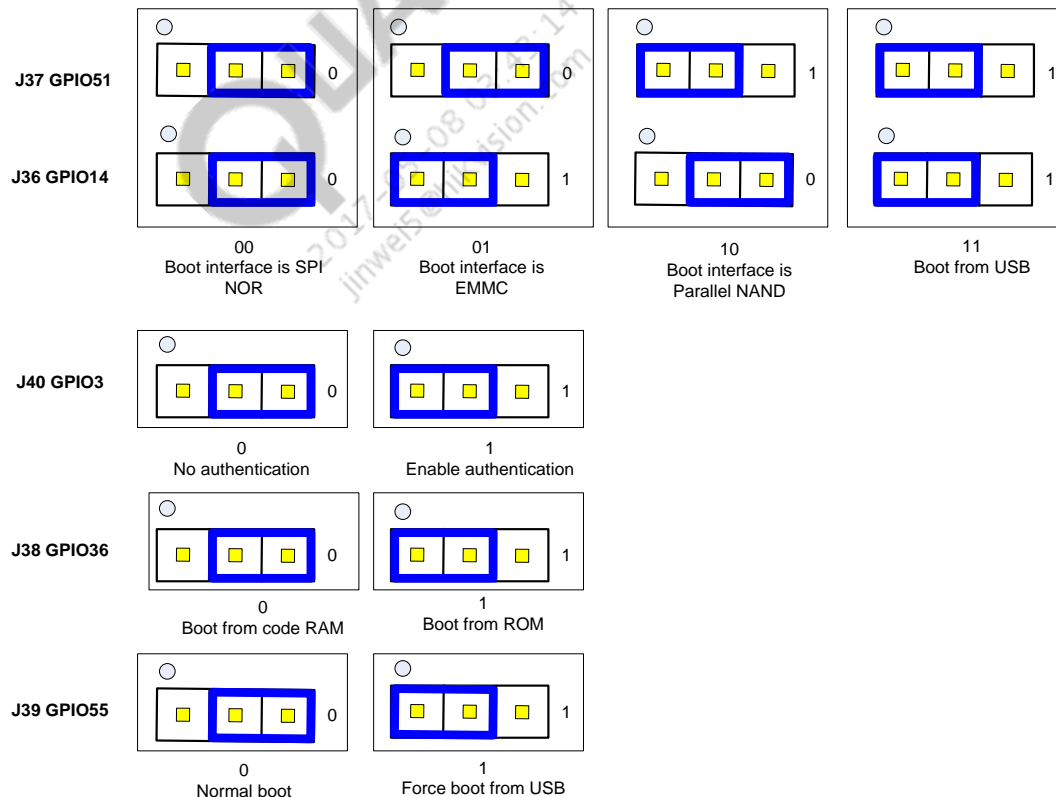
On PCB, the white circle near each jumper indicates pin 1; see Figure 3-5. If jumper connects pin 1 and 2, pull up. If jumper connects pin 2 and 3, pull down.
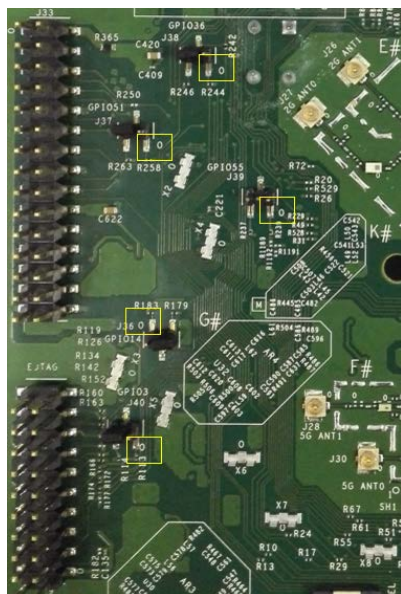
Figure 3-5 Jumper pin 1 location

3.4 Lauterbach connection

Connect the Lauterbach connector into the JTAG connector (J29) on the AP.DK04 board.

Figure 3-6 Correct Lauterbach orientation

3.5 Serial port connection

Connect the RS232 adapter to the AP.DK04 UART connector (J35).

Figure 3-7 Correct UART orientation
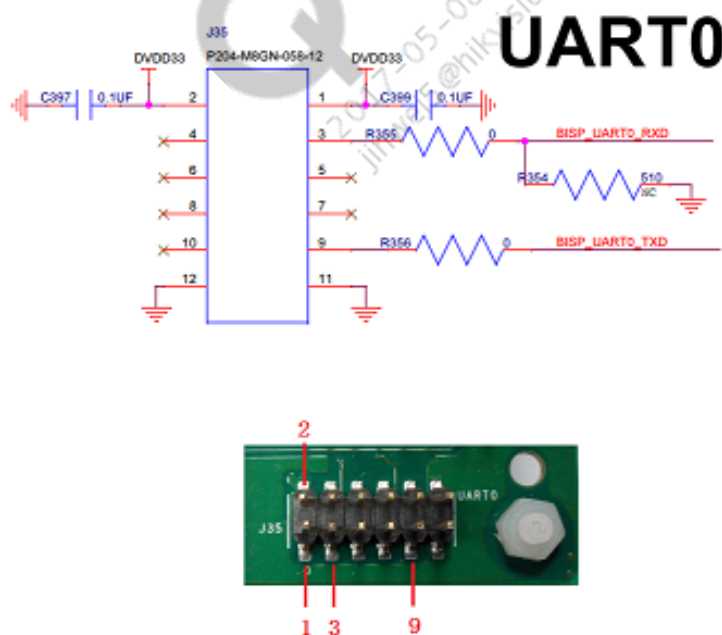
Figure 3-8 AP.DK04 UART connection pin definition

4 Configuring the COM Port

Establish a console session to the AP with either of the following:

- A Telnet connection (not possible unless a complete image has been flashed and booted)
- A direct console connection (preferred)

Direct console via serial port is required for initial programming. Serial port settings are:

- Speed: 115200 bits per second
- 8 data bits
- 1 stop bit
- 0 parity bit
- No flow control

5 Image Flashing Procedure from U-Boot

The AP.DK04 reference design supports multiple flash devices. This chapter describes the flashing procedures for each memory configuration. In this release, the AP.DK04 reference designs use different configuration files as shown below.

Board/ Configuration	Machine ID	Flash	Image Name
AP.DK04.1-C1	8010001	NOR 16 MB	nor-ipq40xx-single.img
AP.DK04.1-C1	8010001	QPIC NAND (128 MB)	nand-ipq40xx-single.img
AP.DK04.1-C1	8010001	eMMC	emmc-ipq40xx-single.img
AP.DK04.1-C1	8010001	NOR 16 MB + QPIC NAND(128 MB)	nornand-ipq40xx-single.img
AP.DK04.1-C2	8010101	NOR 32 MB - Audio	nor-ipq40xx-single.img
AP.DK04.1-C3	8010201	NOR 16 MB + eMMC	norplusemmc-ipq40xx-single.img

Commands for setting machine ID

```
set machid 0xxxxx
```

Example for AP.DK04.1-C1

```
set machid 0x8010001
```

NOTE: If flashing image return failure after changing machid in U-Boot, reset board and redo image flashing procedure.

Follow 3.3Jumper configurationto check boot interface before flashing new image. Programming an incorrect image will result in a non-bootable board. Check the current boot memory type by using the *smem* command at the U-Boot command line.

- flash_type: 0x6   SPI NOR

- flash_type: 0x2   QPIC NAND

- flash_type: 0x5   eMMC

5.1 Flashing image to SPI NOR

1. On the opened console window, enter U-Boot to upload the flash image.
   ```
   set ipaddr 192.168.1.1
   set serverip 192.168.1.xx (tftpserver- Ipaddress)
   tftpb 0x84000000 nor-ipq40xx-single.img
   imgaddr=0x84000000
   Sf probe
   source $imgaddr:script
   ```

2. After image flashing has completed and console prompt is re-printed, power cycle the board.

5.2 Flashing image to QPIC NAND

1. On the opened console window, enter U-Boot to upload the flash image.
```
set ipaddr 192.168.1.1
set serverip 192.168.1.xx (tftpserver- Ipaddress)
tftpb 0x84000000 nand-ipq40xx-single.img
imgaddr=0x84000000
source $imgaddr:script
```

2. After image flashing has completed and console prompt is re-printed, power cycle the board.

5.3 Flashing image to eMMC

1. On the opened console window, enter U-Boot to upload the flash image.
```
set ipaddr 192.168.1.1
set serverip 192.168.1.xx (tftpserver- Ipaddress)
tftpb 0x84000000 emmc-ipq40xx-single.img
imgaddr=0x84000000
source $imgaddr:script
```

2. After image flashing has completed and console prompt is re-printed, power cycle the board.

5.4 Flashing image to SPI NOR + QPIC NAND

1. On the opened console window, enter U-Boot to upload the flash image.
```
set ipaddr 192.168.1.1
set serverip 192.168.1.xx (tftpserver- Ipaddress)
tftpb 0x84000000 nornand-ipq40xx-single.img
sf probe
imgaddr=0x84000000
source $imgaddr:script
```

2. After image flashing has completed and console prompt is re-printed, power cycle the board.

5.5 Flashing image to SPI NOR + eMMC

1. On the opened console window, enter U-Boot to upload the flash image.
```
set ipaddr 192.168.1.1
set serverip 192.168.1.xx (tftpserver- Ipaddress)
tftpb 0x84000000 norplusemmc-ipq40xx-single.img
sf probe
imgaddr=0x84000000
source $imgaddr:script
```

2. After image flashing has completed and console prompt is re-printed, power cycle the board.

MAY CONTAIN U.S. AND INTERNATIONAL EXPORT CONTROLLED INFORMATION

6 Installing and Configuring the IPQ4019/IPQ4029 Interfaces

6.1 Wi-Fi

NOTE: If Wi-Fi modules are not insmod'ed during the kernel boot-up, use this procedure.

```
cat > ins.sh << EOF
cd /lib/modules/3.14.43
insmod adf.ko && echo == adf done ==
insmod asf.ko && echo == asf done ==
insmod ath_hal.ko && echo == ath_hal done == && sleep 5
insmod ath_rate_atheros.ko && echo == ath_rate_atheros done ==
insmod ath_spectral.ko
insmod ath_dfs.ko
insmod hst_tx99.ko
insmod ath_dev.ko && echo == ath_dev done ==
insmod umac.ko && echo == umac done ==
insmod ath_pktlog.ko
EOF
sh ins.sh
```

6.1.1 Finding the available wireless interface

Execute *cat /etc/config/wireless* on the AP.DK04 platform to show the state of all Wi-Fi interfaces.

```
config wifi-device 'wifi0'
        option type 'qcawifi'
        option channel 'auto'
        option macaddr '00:03:7f:12:13:13'
        option hwmode '11ng'
        option disabled '0'

config wifi-iface
        option device 'wifi0'
        option network 'lan'
        option mode 'ap'
        option ssid 'OpenWrt'
        option encryption 'none'

config wifi-device 'wifi1'
        option type 'qcawifi'
```

```
        option channel 'auto'
        option macaddr '00:34:56:78:83:83'
        option hwmode '11ac'
        option disabled '0'

config wifi-iface
        option device 'wifi1'
        option network 'lan'
        option mode 'ap'
        option ssid 'OpenWrt'
        option encryption 'none'
```

6.1.2 DBDC in Open Security

```
uci set wireless.@wifi-device[0].disabled=0
uci set wireless.@wifi-device[1].disabled=0
uci commit wireless; wifi
```

6.1.3 DBDC in WPA2-PSK Security

```
uci set wireless.@wifi-device[0].disabled=0
uci set wireless.@wifi-device[0].htmode=VHT40
uci set wireless.@wifi-iface[0].ssid=sara2
uci set wireless.@wifi-iface[0].encryption=psk2
uci set wireless.@wifi-iface[0].key='123456789'
uci commit wireless; wifi

uci set wireless.@wifi-device[1].disabled=0
uci set wireless.@wifi-device[1].htmode=HT80
uci set wireless.@wifi-iface[1].ssid=sara1
uci set wireless.@wifi-iface[1].encryption=psk2
uci set wireless.@wifi-iface[1].key='123456789'
uci commit wireless; wifi
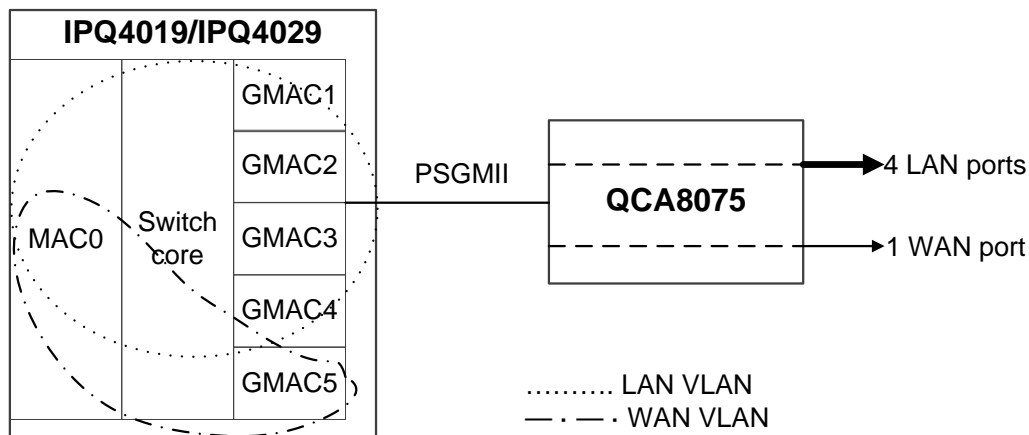```

# 6.2 Ethernet interface



**Figure 6-1 VLAN setting default**

```
root@OpenWrt:cat /etc/config/network
config interface 'loopback'
        option ifname 'lo'
        option proto 'static'
        option ipaddr '127.0.0.1'
        option netmask '255.0.0.0'
config globals 'globals'
        option ula_prefix 'auto'
config interface 'lan'
        option ifname 'eth1'
        option type 'bridge'
        option proto 'static'
        option ipaddr '192.168.1.1'
        option netmask '255.255.255.0'
        option ip6assign '60'
config interface 'wan'
        option ifname 'eth0'
        option proto 'dhcp'
config interface 'wan6'
        option ifname '@wan'
        option proto 'dhcpv6'
config switch
        option name 'switch0'
        option reset '1'
        option enable_vlan '1'
config switch_vlan
        option device 'switch0'
        option vlan '1'
        option ports '0t 1 2 3 4'
config switch_vlan
        option device 'switch0'
        option vlan '2'
        option ports '0t 5'


root@OpenWrt:/# ifconfig
br-lan    Link encap:Ethernet  HWaddr 72:32:65:6E:52:B6
          inet addr:192.168.1.1  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::7032:65ff:fe6e:52b6/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:35 errors:0 dropped:0 overruns:0 frame:0
          TX packets:117 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:2989 (2.9 KiB)  TX bytes:14751 (14.4 KiB)

eth0      Link encap:Ethernet  HWaddr 6A:A6:C9:A8:28:68
          inet6 addr: fe80::68a6:c9ff:fea8:2868/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
```

```
              TX packets:288 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:1000
              RX bytes:0 (0.0 B)  TX bytes:111251 (108.6 KiB)

eth1          Link encap:Ethernet  HWaddr 72:32:65:6E:52:B6
              UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
              RX packets:0 errors:0 dropped:0 overruns:0 frame:0
              TX packets:123 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:1000
              RX bytes:0 (0.0 B)  TX bytes:14850 (14.5 KiB)

root@OpenWrt:/# brctl show
bridge name  bridge id       STP enabled    interfaces
br-lan       7fff.7232656e52b6  no          eth1
```

6.3 USB

The AP.DK04 supports one USB 3.0 and one USB2.0 host ports.

1. Plug a USB device into the USB port to see whether the USB device is recognized.
    ```
    usb 2-1: new SuperSpeed USB device number 8 using xhci-hcd
    usb 2-1: Parent hub missing LPM exit latency info.  Power management
    will be impacted.
    usb-storage 2-1:1.0: USB Mass Storage device detected
    scsi10 : usb-storage 2-1:1.0
    scsi 10:0:0:0:Direct-Access Kingston DataTraveler 3.0 PMAP PQ: 0 ANSI: 6
    sd 10:0:0:0: [sda] 30712320 512-byte logical blocks: (15.7 GB/14.6 GiB)
    sd 10:0:0:0: [sda] Write Protect is off
    sd 10:0:0:0: [sda] Write cache: disabled, read cache: enabled, doesn't
    support DPO or FUA
    sda: sda1
    sd 10:0:0:0: [sda] Attached SCSI removable disk
    ```

    Or type the command `dmesg` to see whether the USB device is recognized. The `dmesg`
    command responds with whether the USB Flash drive is at sda/sdb.

2. When the USB device is mounted at sda1 or another block device, create a new directory
    under the **/** directory, use the `mount` command to mount the USB device to the newly created
    directory:
    ```
    mkdir /tmp/usb1
    mount /dev/sda1 /tmp/usb1
    ```

6.4 eMMc/SD

The AP.DK04 supports one eMMc/SD interface.

1. Plug a SD card into the SD slot to see whether the SD card is recognized.

   Use `fdisk -l` command to identify the device name:

   ```
   root@OpenWrt:/mnt# fdisk -l
   Disk /dev/mmcblk0: 59.5 GiB, 63864569856 bytes, 124735488 sectors
   Units: sectors of 1 * 512 = 512 bytes
   Sector size (logical/physical): 512 bytes / 512 bytes
   I/O size (minimum/optimal): 512 bytes / 512 bytes
   Disklabel type: dos
   Disk identifier: 0x66c8418c


   Device         Boot Start       End  Blocks  Id System
   /dev/mmcblk0p1      2048 124735487 62366720  83 Linux
   ```

2. Create mount point:

   ```
   root@OpenWrt:/# cd /mnt
   root@OpenWrt:/mnt# mkdir sd
   ```

3. Mount:

   ```
   root@OpenWrt:/mnt# mount /dev/mmcblk0p1 /mnt/sd
   NTFS signature is missing.
   Failed to mount '/dev/mmcblk0p1': Invalid argument
   The device '/dev/mmcblk0p1' doesn't seem to have a valid NTFS.
   Maybe the wrong device is used? Or the whole disk instead of a[
   3097.151264] EXT4-fs (mmcblk0p1): couldn't mount as ext3 due to feature
   incompatibilities partition (e.g. /dev/sda, not /dev/sda1)? Or the other
   way around?
   ```

4. Make sure the mount is successful:

   ```
   root@OpenWrt:/mnt# mount

   /dev/mmcblk0p1 on /mnt/sd type ext4 (rw,relatime,data=ordered)
   ```

6.5 VoIP

The AP.DK04 supports VoIP solution. There is one connector J33 on AP.DK04, which can connect with TB724 to achieve VoIP function.

6.6 PCIe

The AP.DK04 supports PCIe 2.0 and there is one PCIe-mini slot J25 which can connect with CUS239 or CUS240 or similar PCIe-mini card to extended more feature.