

# CV180X & CV181X Wi-Fi 使用手册

Version: 2.0.0

Release date: 2023-02-08

©2022 北京晶视智能科技有限公司 本文件所含信息归北京晶视智能科技有限公司所有。 未经授权,严禁全部或部分复制或披露该等信息。 SOPIIGO CVITEK Wi-Fi 使用指南

# 目录

算能科技

1	声明	2
2	概述	3
3	配置说明         3.1       内核配置         3.2       配置 SDIO         3.3       配置 Pinmux         3.4       配置 WIFI GPIO	<b>4</b> 5 5 6
4	Wi-Fi 工具	7
5	Wi-Fi 基本操作         5.1       STA 模式基本操作         5.1.1       加载驱动         5.1.2       启动 Wi-Fi 与连接 AP         5.1.3       关闭 Wi-Fi 与卸载驱动         5.2       SoftAP         英式基本操作       1         5.2.1       加载驱动         5.2.2       hostapd 配置, udhcpd 配置与启动 SoftAP	9 9 0 2 2 3
6	测试       1         6.1       吞吐率测试       1         6.1.1       发送吞吐率测试       1         6.1.2       接收吞吐率测试       1	.4 .4 .4

i



### 修订记录

Revision Date		Description		
1.0.0	2022/06/10	Initial version		
2.0.0	2023/02/08	cv180x/cv181x 文档兼容		
2.0.1	2023/07/19	双系统文档兼容		







### 法律声明

本数据手册包含北京晶视智能科技有限公司(下称"晶视智能")的保密信息。未经授权,禁止使 用或披露本数据手册中包含的信息。如您未经授权披露全部或部分保密信息,导致晶视智能遭受 任何损失或损害,您应对因之产生的损失/损害承担责任。

本文件内信息如有更改, 恕不另行通知。晶视智能不对使用或依赖本文件所含信息承担任何责任。 本数据手册和本文件所含的所有信息均按"原样"提供, 无任何明示、暗示、法定或其他形式的 保证。晶视智能特别声明未做任何适销性、非侵权性和特定用途适用性的默示保证, 亦对本数据 手册所使用、包含或提供的任何第三方的软件不提供任何保证; 用户同意仅向该第三方寻求与此 相关的任何保证索赔。此外, 晶视智能亦不对任何其根据用户规格或符合特定标准或公开讨论而 制作的可交付成果承担责任。

### 联系我们

地址 北京市海淀区丰豪东路 9 号院中关村集成电路设计园 (ICPARK) 1 号楼

深圳市宝安区福海街道展城社区会展湾云岸广场 T10 栋

电话 +86-10-57590723 +86-10-57590724

邮编 100094 (北京) 518100 (深圳)

官方网站 https://www.sophgo.com/

技术论坛 https://developer.sophgo.com/forum/index.html





Wi-Fi 是 Wi-Fi 联盟的商标, 为一种基于 IEEE 802.11 为标准的无线区域网路技术。具备 Wi-Fi 功能的移动终端可以在讯号覆盖范围内连上网际网路,以减少架设电缆的困扰并提升使用的方便 性。

目前已有很多处理器厂家提供各种型号的 Wi-Fi 处理器解决方案,并有各自不同的驱动,然这些驱动并不具备普适性。另外不同驱动版本所支持的功能及性能亦有可能会有差异,需请 Wi-Fi 解决方案供应商提供合适的 Linux Wi-Fi 驱动来进行移植工作。

Linux 平台上对于不同 Wi-Fi 处理器的驱动与操作方式有通用性,本文档会以 CV180X 为例分别介绍在不同接口上 (如 USB 或是 SDIO) 如何使用 Realtek 解决方案进行驱动移植与调适,以及相关的操作。

本文所使用的 Wi-Fi 模组为

· AP6201BM (Broadcom bcm43013c1), 支持 SDIO 接口。







# 3.1 内核配置

修改 build/boards/{processor\_name}/{board\_name}/linux/cvitek\_{board\_name}\_defconfig,
 ex. build/boards/cv1801c\_wevb\_0009a\_spinor/linux/cvitek\_cv1801c\_wevb\_0009a\_spinor\_defconfig
 使能 Wifi 相关 Configuration (标注红色部分为基本必须开启的 Configuration, 其他部分则是需求开启)。

<i>#↓</i>
# WiFie
#~
CONFIG_WLAN=y.↓
CONFIG_CFG80211=y+
CONFIG_CFG80211_DEFAULT_PS=y+
CONFIG_CFG80211_CRDA_SUPPORT=y.↓
# CONFIG_CFG80211_WEXT is not set
# CONFIG_MAC80211 is not set.
# CONFIG_MAC80211_HAS_RC is not set⊷
# CONFIG_MAC80211_RC_MINSTRE is not set
# CONFIG_MAC80211_RC_MINSTREL_HT is not set↩
# CONFIG_MAC80211_RC_DEFAULT_MINSTREL is not set.
# CONFIG_MAC80211_RC_DEFAULT="minstrel_ht"
CONFIG_CVI_WIFI_PIN=y.↓
CONFIG_WIRELESS=y↓
CONFIG_WLAN_VENDOR_REALTEK=y
CONFIG_RTL8189FS=m #此选项根据所用 wifi 芯片选择对应的驱动(需要做适配)。
# CONFIG_WEXT_CORE is not set
# CONFIG WEXT PROC is not set?

### 由于 Wi-Fi 接口为 SDIO,因此需开启

Build/boards/cv180x/cv1801c\_wevb\_0009a\_spinor/dts\_riscv/{board\_name}.dtsi 确认 wifisd 节点配置如下:

wifisd:wifi-sd@4320000 {
 compatible = "cvitek,cv181x-sdio";
 bus-width = <4>;
 reg = <0x0 0x4320000 0x0 0x1000>;

(下页继续)

SOPHGO 算能科技 CVITEK Wi-Fi 使用指南

(续上页)

```
reg_names = "core_mem";
src-frequency = <37500000>;
min-frequency = <400000>;
max-frequency = <50000000>;
64_addressing;
reset_tx_rx_phy;
non-removable;
pll_index = <0x7>;
pll_reg = <0x300207C>;
no-mmc;
no-sd;
};
```

另编辑 build/boards/default/dts/cv180x/{board\_name}\_{bga or qfn}.dtsi, ex. build/boards/default/dts/cv180x/cv180x\_asic\_bga.dtsi 或相对应板型的 dtsi 文件确认无 删除 wifi-sd@5000000 节点的配置, 示例如下:

```
/* /delete-node/ wifi-sd@5000000; */ /* 将这行注释或者删除 */
```

/delete-node/ i2c@04010000; /delete-node/ i2c@04020000; /delete-node/ ethernet@04520000; /delete-node/ i2s@04120000; ...

## 3.2 配置 SDIO

请参考《CVITEK 外围设备驱动操作指南》中与 SDIO 相关章节。SDIO IO 电压为 3.3V, 需确 认 Wi-Fi 模块 IO 电压和 SDIO 电压一致。

## 3.3 配置 Pinmux

若 Wi-Fi 模块使用的接口为 SDIO 时, 需配置 SDIO 的管脚复用。CV180X/1X 可以通过在 build/boards/{processor\_name}/{board\_name}/u-boot/cvi\_board\_init.c 文件中添加以下 pin-mux 设置来配置 SDIO 的 pinmux (这里是 181h 的 evb 配置, 具体配置哪根引脚需要查看电路 图中 soc 到 wifi 模组 processor\_en 的引脚是哪根):

```
int cvi_board_init(void)
{
    ...
    //#######WIFI
    pinmux_config(PINMUX_SDIO1);
    PINMUX_CONFIG(JTAG_CPU_TCK, XGPIOA_18);
    ...
    return 0;
}
```

相关管脚配置细节,请参考 u-boot-2021.10/board/cvitek/cv181x/board.c



# 3.4 配置 WIFI GPIO

由于 Wi-Fi 模块的 processor\_en 引脚是由 SOC 上的一根 GPIO 控制,为了操作这只 GPIO,我 们专门做了一个简单的模块,在 wifi 驱动中使用该模块提供的接口对 wifi 进行上下电。可以通 过设备树指定 wifi 使用的 gpio: (其中 wakeup 的功能没有使用,可以去除; poweron 引脚和上 一小节设置的 pinmux 对应)

wifi_pin {	
→ compatible	e = "cvitek,wifi-pin";
→ poweron-g	pio = < <b>&amp;porta</b> 18 GPIO_ACTIVE_HIGH>;
→ wakeup-gpt	io = < <b>&amp;porte</b> 7 GPIO_ACTIVE_HIGH>;
<b>};</b>	

该配置位于 build/boards/default/dts/{processor\_name}/{processor\_name}\_base.dtsi 文件中。





# **4** Wi-Fi 工具

操作 Wi-Fi 需要使用到 wpa\_supplicant, wpa\_cli, hostapd 等开源工具。

· 透过选单模式选择 Rootfs packages → Target package wifi,再打开 wireless 选项,存档离 开后,开始编译

(To	p) → Rootfs pac	kages				
	*****	CVi	Tek MediaSDK	Configuration		
[ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ]	Target package Target package	ntp secure_im libiw python3.7 ncurses libz uhubon htop ota serve procrank er cvitracer lame libmad nanomsg wifi	age r			
5.0		1 / 1	[50.6]		[0] 0	
[Spa [0] [F]	ace/Enter] Togg Load Toggle show-be	le/enter	[ESC] Leave m [?] Symbol in [C] Toggle st	nenu 1fo 1ow-name mode	[5] Save [/] Jump to sy [A] Toggle sho	ymbol ow-all mode
[Q]	Quit (prompts	for save)	[D] Save mini	imal config (ad	lvanced)	on acc mode

SOPIHGO 算能科技

### (Top) → SDK options

	CViTek MediaSDK Configuration
[]	C library (musl library for user mode application on riscv64)> Build static binary (no shared libs) Build SDK with debug config
Ϊi	Enable SDK sanitizer
[*]	Do not install sample and self test application
[*]	Install the osdrv/extdrv/wireless/*.ko
[]	Do not compile frame buffer drivers
ίi	Select CONFIG_NO_TP to build osdrv without Touchscreen driver(extdrv/tp)
ίi	Select CONFIG_USB_OSDRV_CVITEK_GADGET to build osdrv with usb gadget cvg
[*]	Make the boot image only have one dtb
[]	Compile 64MB DDR size project
10	

或修改 build/boards/{processor\_name}/{board\_name}/{board\_name}\_defconfig 使能如下选项,再透过命令模式,执行 defconfig \$CHIP\_\$BOARD,以自动做好配置

#
# Rootfs packages
#
...
CONFIG\_TARGET\_PACKAGE\_WIFI=y
CONFIG\_CP\_EXT\_WIRELESS=y
# end of Rootfs packages

若用户欲更新至最新版本,可至 http://w1.fi/releases 或是 http://www.linuxfromscratch.org/ blfs/view/svn/basicnet/wireless\_tools.html 获取,并自行安装至 rootfs。

CHAPTER 5. WI-FI 基本操作



# **5** Wi-Fi 基本操作

# 5.1 STA 模式基本操作

### 5.1.1 加载驱动

步骤 1. 载入驱动

需检查/mnt/system/ko/3rd 下是否已具备下图红框处的三份文件:

[root@cvitek]/mnt/system/ko/3rd# ls
8188fu.ko 8189fs.ko \_\_\_\_\_

注意,由于 wifi 驱动并没有放在 linux 源码目录树中,所以无法 build-in,只能编译成 ko。

insmod /mnt/system/ko/3rd/8189fs.ko

步骤 2. 查看驱动是否载入成功

执行 shell 指令:

ifconfig -a

若载入成功,可在执行 shell 指令后看到 wlan0 网口

CHAPTER 5. WI-FI 基本操作

CVITEK Wi-Fi 使用指南

/ # ifconfig -a					
eth0	Link encap:Ethernet HWaddr 00:00:00:00:00:00				
	BRUADCASI MULIICASI MIU:1500 Metric:1 BX packets:0 errors:0 dropped:0 overrups:0 frame:0				
	TX packets:0 errors:0 dropped:0 overruns:0 carrier:0				
	collisions:0 txqueuelen:1000				
	RX bytes:0 (0.0 B) IX bytes:0 (0.0 B) Interrupt:16				
lo	Link encap:Local Loopback				
	LOOPBACK MTU:65536 Metric:1				
	RX packets:0 errors:0 dropped:0 overruns:0 frame:0				
	collisions:0 txqueuelen:1				
	RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)				
sit0	Link encap:IPv6-in-IPv4				
	NOARP MTU:1480 Metric:1				
	RX packets:0 errors:0 dropped:0 overruns:0 frame:0				
	collisions:0 txqueuelen:1				
	RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)				
wlan⊖	Link encap:Ethernet HWaddr FC:6B:F0:7B:D1:29				
	BROADCAST MULTICAST MTU:1500 Metric:1				
	RX packets:0 errors:0 dropped:0 overruns:0 trame:0				
	collisions:0 txqueuelen:1000				
	RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)				

## 5.1.2 启动 Wi-Fi 与连接 AP

步骤 1. 启动 Wlan0 网口

sc 执行 shell 指令:

SOPIIGO 算能科技

ifconfig wlan0 up

步骤 2. 启动 wpa\_supplication

执行 shell 指令:

echo "ctrl\_interface=/var/run/wpa\_supplicant" >/tmp/wpa\_supplicant.conf wpa\_supplicant -iwlan0 -Dnl80211 -c/tmp/wpa\_supplicant.conf &

· -iwlan0 表示使用 wlan0 接口

· -Dnl80211 表示使用 cfg80211 接口

步骤 3. 启动 wpa\_cli

执行 shell 指令:

wpa\_cli -i wlan0

执行成功会出现">"提示符号

CVITEK Wi-Fi 使用指南

/ # wpa\_cli wpa\_cli v2.6 Copyright (c) 2004-2016, Jouni Malinen <j@wl.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 >

步骤 4. 扫描附近 AP

SOPIIGO 算能科技

在">"提示符号后执行如下指令:

scan

在出现"CTRL-EVENT-SCAN-RESULTS"后,再执行

scan results

即可得到扫描结果



步骤 5. 连接 AP

- · 连接配置为 WPA-PSK/WPA2-PSK 认证与加密类型方式的 AP
- 1. 在 ">" 提示符号后执行如下指令,以取得网路 ID (此示例中为 0):

 $add\_network$ 

2. 配置网路的 SSID (此示例中的 SSID 为 SW-test, 由步骤 4 取得)

set network 0 ssid "SW-test"

3. 配置网路加密方式与密码 (假设 SW-test 密码为 012345678)

set network 0 psk "012345678"

4. 启动网路

select network 0

5. 观察是否有收到 CTRL-EVENT-CONNECTED, 若有,则表示连接成功。或可执行"status" 指令,查询连线状态

> scan OK	
[ 1206.695367] [0] RTW: wlan0- hw port(0) mac_addr =fc:6b:f0:7b:d1:29	
[ 1206.704508] [0] RTW: nolinked power save leave	
<3>CTRL-EVENT-SCAN-STARTED	
<3>CTRL-EVENT-SCAN-RESULTS	
<3>CTRL-EVENT-NETWORK-NOT-[ 1208.308629] [1] RTW: nolinked power save enter	
FOUND	
scan_results	
> bssid / frequency / signal level / flags / ssid	
ac:9e:17:5b:e7:8c 2462 -39 [WPA-PSK-CCMP+TKIP][WPA2-PSK-CCMP+TKIP][ESS]	SW-test
d8:fe:e3:9f:d8:d8 2427 -58 [WPA-PSK-CCMP+TKIP][WPA2-PSK-CCMP+TKIP][ESS]	avant

6. 输入 "quit" 退出 wpa\_cli, 执行 shell 指令如下以取得动态 IP 地址

udhcpc -b -i wlan0 -R &

7. 执行 ping 指令, 以观察网路是否正常运作, ex.

Ping 8.8.8.8

SOPIIGO 算能科技

· 连接配置为 open system 的 AP

步骤与配置为 WPA-PSK/WPA2-PSK 认证与加密类型方式相同, 唯在配置网路 加密方式需输入如下指令:

set\_network 0 key\_mgmt NONE

### 5.1.3 关闭 Wi-Fi 与卸载驱动

步骤 1. 执行 shell 指令如下:

ifconfig wlan0 down

步骤 2. 执行如下 shell 指令:

rmmod 8189fs.ko

# 5.2 SoftAP 模式基本操作

### 5.2.1 加载驱动

载入驱动方式与 STA 模式相同,请参考章节5.1.1. 加载驱动

## 5.2.2 hostapd 配置, udhcpd 配置与启动 SoftAP

启动 SoftAP 需先启动 hostapd。与 wpa\_supplicant 类似, hostpad 可以用来配置 AP 端的各种 认证协议,以及连接流程。

步骤 1. 启动 hostpad。执行 shell 命令

ifconfig wlan0 192.168.1.1 up hostapd /etc/network/hostapd.conf -B -i wlan0

步骤 2. 启动 udhcpd 以分配动态 IP 给连接装置,执行 shell 命令

udhcpd /etc/network/udhcpd.conf

备注:

可透过修改 hostapd.conf 来配置 SoftAP 的 ssid, channel, 加密认证方式等, 文档位置在 SDK 包里的/ramdisk/rootfs/overlay/{processor\_name}/etc/network 或是平台上的/etc/network。例如可修改 ssid 以及 wpa\_passphrase 来配置 AP 名称以及登入密码

interface=wlan0 ctrl\_interface=/var/run/hostapd ssid=CV180X\_EVB channel=6 wpa=3 wpa\_passphrase=012345678

其他参数意义可参考: http://manpages.ubuntu.com/manpages/bionic/man5/udhcpd.conf.5. html

· 可透过修改 udhcpd.conf 来配置 SoftAP 所分配的 IP 范围。udhcpd.conf 文档位置在 SDK 包 里的/ramdisk/rootfs/overlay/{processor\_name}/etc/network 或是平台上的/etc/network。

# The start and end of the IP lease block start 192.168.1.10 #default: 192.168.0.20 end 192.168.1.254 #default: 192.168.0.254





## 6.1 吞吐率测试

Wifi 的性能可以透过吞吐率测试来观察与调校。一般吞吐率测试最常使用的工具为 iperf3。测试 环境架设如下:



PC 通过有线的以太网路与无线路由器(wireless AP)连接, CVITEK 平台则透过 Wi-Fi 与无 线路由器连接。假设在此示例中, PC 的 IP 地址为 192.168.0.11, CVITEK 平台的 IP 地址为 192.168.0.112。PC 与 CVITEK 平台皆有 iperf3 工具。

#### 发送吞吐率测试 6.1.1

步骤 1. 在 PC 上进入 iperf3 工具目录,执行如下指令:

iperf3 -s

步骤 2. 平台执行 shell 指令如下:

· 测试 TCP 协定

iperf3 -c 192.168.0.11 -t 10

• 测试 UDP 协定

iperf3 -c 192.168.0.11 -t 10 -u -b 100M -l 32k

/ # iperf3 -c 192.168.0.11 -t 10							
С	Connecting to host 192.168.0.11, port 5201						
[	5]	local 192.168	.0.11	2 port 50194	connected to 192	.168.0	.11 port 5201
[	ID]	Interval		Transfer	Bitrate	Retr	Cwnd
[	5]	0.00-1.00	sec	1.42 MBytes	11.9 Mbits/sec	Θ	138 KBytes
[	5]	1.00-2.00	sec	941 KBytes	7.71 Mbits/sec	75	114 KBytes
[	5]	2.00-3.00	sec	627 KBytes	5.14 Mbits/sec	Θ	130 KBytes
[	5]	3.00-4.00	sec	941 KBytes	7.71 Mbits/sec	Θ	137 KBytes
[	5]	4.00-5.00	sec	941 KBytes	7.71 Mbits/sec	Θ	138 KBytes
[	5]	5.00-6.00	sec	1.23 MBytes	10.3 Mbits/sec	Θ	138 KBytes
[	5]	6.00-7.00	sec	941 KBytes	7.71 Mbits/sec	Θ	140 KBytes
[	5]	7.00-8.00	sec	1.53 MBytes	12.8 Mbits/sec	Θ	147 KBytes
[	5]	8.00-9.00	sec	314 KBytes	2.57 Mbits/sec	1	158 KBytes
[	5]	9.00-10.00	sec	753 KBytes	6.17 Mbits/sec	Θ	178 KBytes
[	ID]	Interval		Transfer	Bitrate	Retr	
[	5]	0.00-10.00	sec	9.51 MBytes	7.98 Mbits/sec	76	sender
]	5]	0.00-10.00	sec	8.89 MBytes	7.46 Mbits/sec		receiver

透过 iperf3 发送测试可以取得结果如上图。各参数意义可以透过 iperf3 -h 取得说明。 由上图可以观察到 10 秒平均吞吐率为 7.98Mbps。

## 6.1.2 接收吞吐率测试

步骤 1. 在平台上执行 shell 指令如下:

iperf3 -s

步骤 2. PC 上进入 iperf3 工具目录执行指令如下:

· 测试 TCP 协定

iperf3 -c 192.168.0.112 -t 10

• 测试 UDP 协定

iperf3 -c 192.168.0.112 -t 10 -u -b 100M -l 32k