

Tina

WIFI 和 Bluetooth 模组移植指南 V1. 1

文档履历

版本号	日期	制/修订人	制/修订记录
V1.0	2016/11/4		Initial version, support for AP6212 and RTL8188EU
V1.1	2017/8/25		add support for XR819



目 录

1. 引言.....	3
1.1. 编写目的.....	3
1.2. 适用范围.....	4
1.3. 相关人员.....	4
2. 模块概述.....	5
3. 模组移植.....	6
3.1. AP6212/AP6212A 等.....	6
3.1.1. 内核配置.....	6
3.1.2. Tina 配置.....	6
3.1.3. sys_config.fex.....	7
3.1.4. AP6212 模组移植相关文件.....	13
3.2. RTL8188EU.....	14
3.2.1. 内核配置.....	14
3.2.2. Tina 配置.....	14
3.2.3. sys_config.fex.....	15
3.2.4. RTL8188EU 模组移植相关文件.....	16
3.3. XR819.....	16
3.3.1. 内核配置.....	16
3.3.2. Tina 配置.....	17
3.3.3. sys_config.fex.....	18
3.3.4. XR819 模组移植相关文件.....	21
4. 上层定制开发及 demo.....	22
5. Declaration.....	23

1. 引言

1.1. 编写目的

本文档编写目的是让相关人员按照文档说明使用所支持的 WIFI/Bluetooth 模组,同时可以作为移植其它 WIFI/Bluetooth 模组参考。本文档以特定的方案说明 WIFI/Bluetooth 移植,在其他方案上进行该款 WIFI/Bluetooth 模组只需根据具体情况,做相应修改即可。

1.2. 适用范围

本 WIFI 和 Bluetooth 模组移植说明文档适用于 Tina 2.5 及以上 Linux 4.4 内核的 WIFI/Bluetooth 移植。相关的 Allwinner SoC 有：R18 等。如果你使用配备其他内核系统的 SoC，请参考其他系列的 WIFI/Bluetooth 模组移植说明书。

1.3. 相关人员

本文档适合方案开发人员、系统集成人员及 WIFI/Bluetooth 移植和维护人员。



2. 模块概述

本文档将以具体方案 xxxxx (xxxxx 为平台方案, 下同) 为例说明如何配置每款 WIFI 模组。RTL8723AU 暂不支持。

WIFI 模组可分 USB 接口和 SDIO 接口两种类型, 部分模组带蓝牙和 FM 功能 (FM 功能尚未支持)。WIFI 的全功能包括 station、softap 和 wifi direct。其中 station 是 WIFI 的最基本功能, 每款 WIFI 模组均支持, 在此模式下, 无线网卡发送连接与认证消息给 AP, AP 接收到后完成认证, 返回成功认证消息, 此网卡接入无线网络; softap 是让当前网卡工作在 AP 模式作为热点使用; wifi direct 模式下, 当前网卡和其他无线设备实现点对点直接连接, 而无需 AP。

本文档会不断更新, 支持的模组也会不断更新, 文档和代码对应可能会稍有差别。

模组	功能(协议)	接口	WIFI 功能			BT
			Station	Softap	wifi direct	
AP6212/AP6212A/AP6255/AP6335/AP6356S 等	802.11b/g/n Bluetooth 4.0	SDIO/UART/PCM	√	√	√	√
RTL8188EU	802.11b/g/n	USB	√	√	√	
XR819	802.11b/g/n	SDIO	√	√	√	

表 1 Tina 平台 WIFI 模组列表

3. 模组移植

进行 WIFI/Bluetooth 模组的移植，主要分为内核配置、Tina 配置、修改 sys_config.fex 文件三个步骤。

3.1. AP6212/AP6212A 等

功能：wifi station、softap、wifi direct、bluetooth

接口类型：SDIO/UART/PCM

3.1.1. 内核配置

在 Tina 目录下执行：

```
$ make kernel_menuconfig
```

选择编译 Broadcom 无线网卡驱动为模块

wifi:(编译成模块)

```
Device Drivers --->
```

```
    Network device support --->
```

```
        Wireless LAN --->
```

```
            <M> Broadcom FullMAC wireless cards support
```

```
                (/lib/firmware/fw_bcmdhd.bin) Firmware path
```

```
                (/lib/firmware/nvram.txt) NVRAM path
```

bluetooth:

```
[*]Networking support --->
```

```
    <*>Bluetooth subsystem support--->
```

```
        Bluetooth device drivers --->
```

```
            [*] UART (H4) protocol support
```

3.1.2. Tina 配置

内核选定之后，tina 进行相关配置

在 Tina 目录下执行：

```
$ make menuconfig
```

选中使用 AP6212，系统就会将 AP6212 的驱动模块拷贝到制定位置，使得系统固件烧写后在 Tina 系统中保存，并且在系统启动时能够自动加载。

以下配置实现 WIFI 驱动拷贝以及开机自动加载：

```
Kernel modules--->
```

```
    Wireless Drivers--->
```

```
        <*> kmod-net-broadcom
```

以下配置编译拷贝 wifi 的 firmware:

```
Firmware--->
```

```
    <*> ap6212-firmware.
```

3.1.3. sys_config.fex

sys_config.fex 文件位于

/tina/target/allwinner/xxxxx/configs/sys_config.fex

为 Allwinner 对于特定硬件平台进行参数配置的文件，sys_config.fex 文件决定使用哪款 WIFI 模组和 GPIO 的分配。修改 sys_config.fex 文件时需查看原理图确定相关硬件连接信息。

3.1.3.1. SDIO/MMC 配置

如图 1 所示，我们可以看到该方案使用网络 VCC-WIFI 为 AP6212 WIFI 模组供电，使用 VCC-WIFI-IO 作为上拉电源使相关 IO 保持确定的电平。图 2 说明了系统使用 SDIO1 与主控进行通讯。

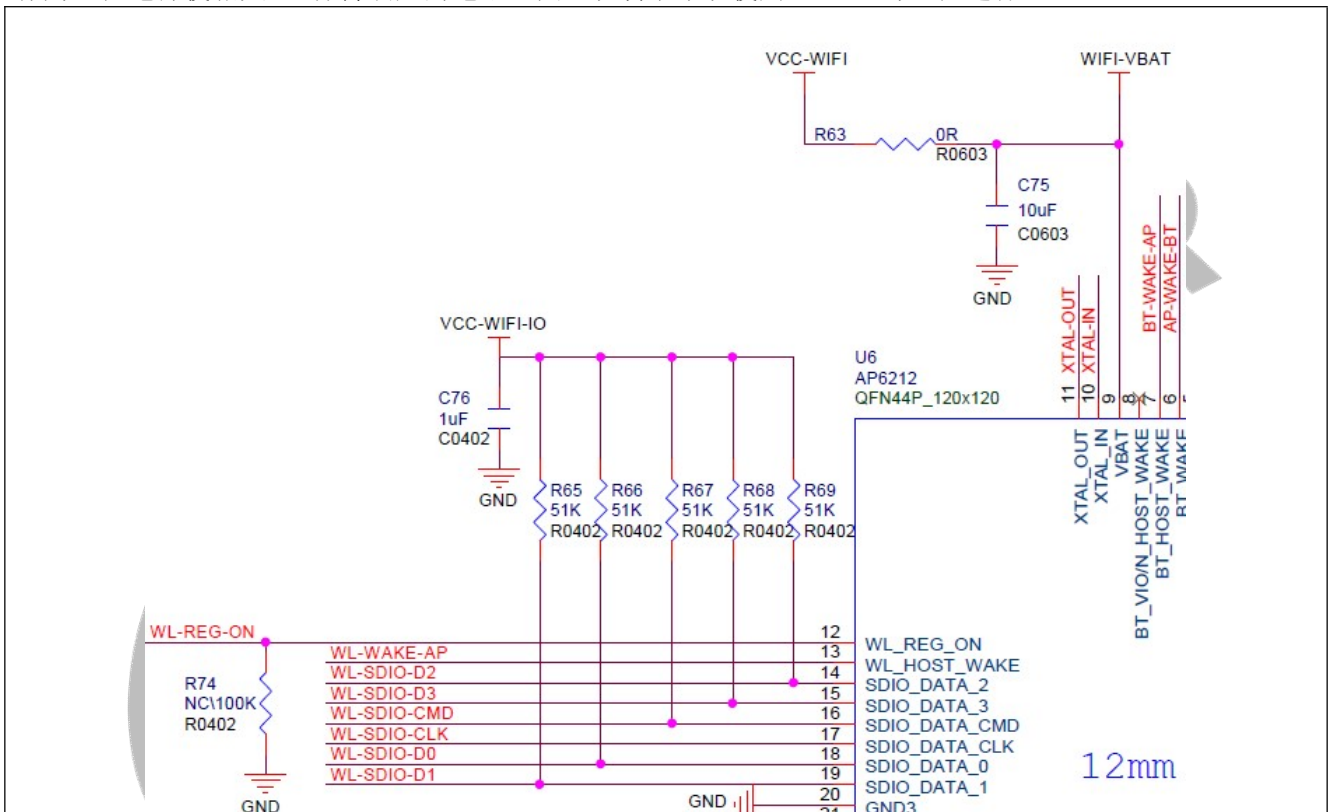


图 1 AP6212 WIFI 原理图

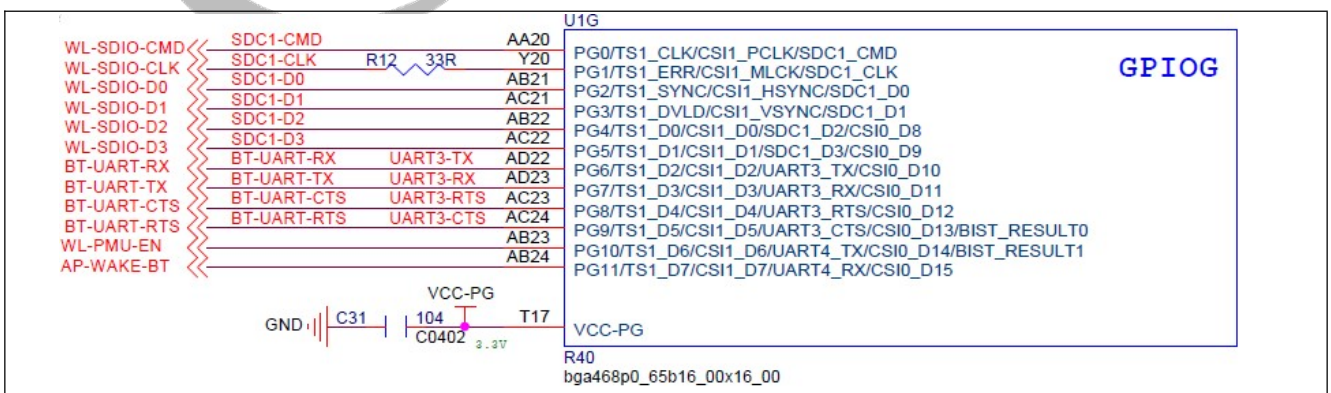


图 2 AP6212 GPIO 使用情况

根据图 1 和图 2, 使用 AP6212 模组需要把 sys_config.fex 文件的 SDIO 相关部分改成如下(部分代码):

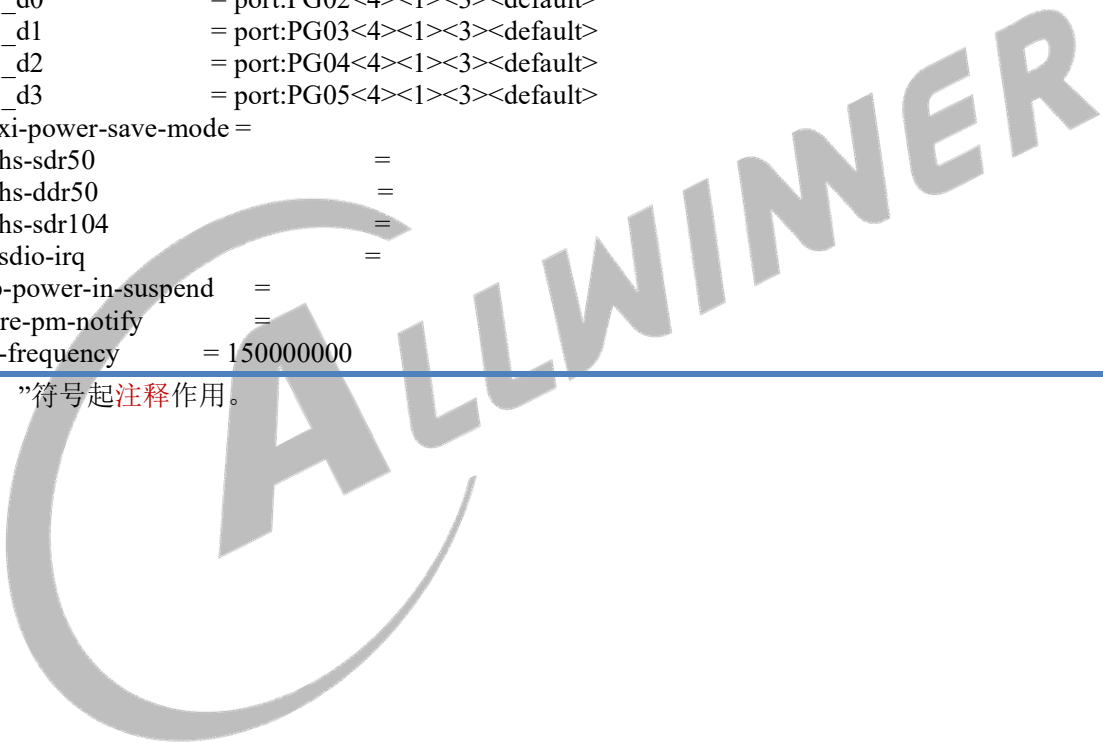
```

;-----
; SDMMC PINS MAPPING          ; -----
;   Config Guide
;   sdc_used: 1-enable card, 0-disable card
;   non-removable:if you use as main memory,you should set it,for example eMMC
;   bus-width: card bus width, 1-1bit, 4-4bit, 8-8bit
;   sunxi-power-save-mode: if use sdio card,should not set it
;   vmmc:regulator for card/emmc power
;   vqmmc:regulator for card/emmc io power
;   vdmcc:regulator for card detect pin pull up power
;   other: GPIO Mapping configuration
;-----

[sdc1]
sdc1_used          = 1
bus-width          = 4
sdc1_clk           = port:PG00<4><1><3><default>
sdc1_cmd           = port:PG01<4><1><3><default>
sdc1_d0            = port:PG02<4><1><3><default>
sdc1_d1            = port:PG03<4><1><3><default>
sdc1_d2            = port:PG04<4><1><3><default>
sdc1_d3            = port:PG05<4><1><3><default>
;sunxi-power-save-mode =
sd-uhs-sdr50       =
sd-uhs-ddr50       =
sd-uhs-sdr104      =
cap-sdio-irq       =
keep-power-in-suspend =
ignore-pm-notify   =
max-frequency       = 150000000

```

其中：“;”符号起注释作用。



3.1.3.2. PMU 电源端口配置

图 3 说明了 VCC-WIFI-IO 连接至电源管理芯片的 DLDO1 端，VCC-WIFI 连接至电源管理芯片的 DLDO2/3 端。

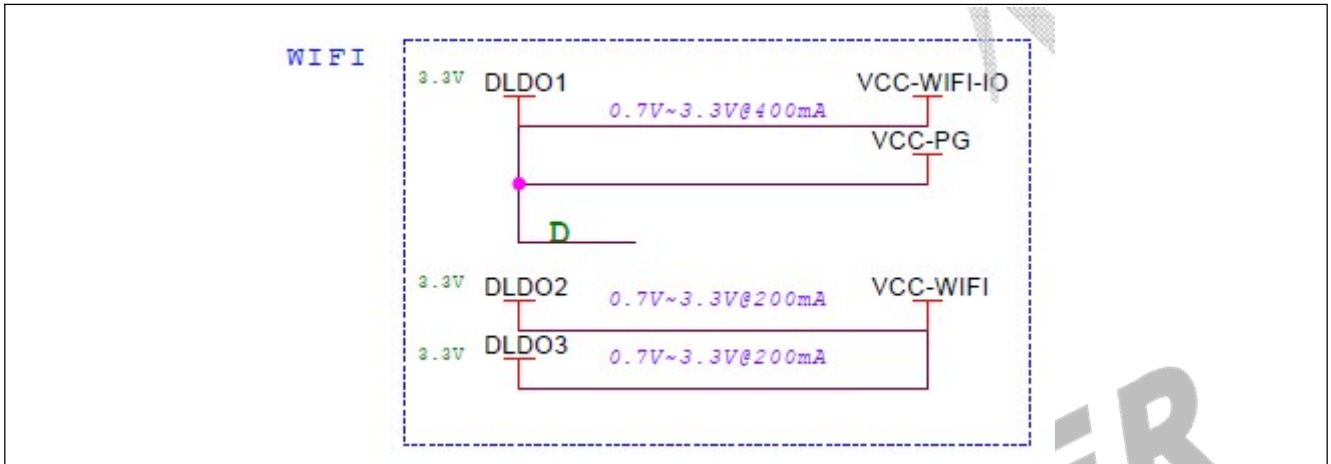


图 3 WIFI 电源网络

该方案的电源管理端口配置如下：

```

;-----
;compatible                ---regulator0 name, support:axp221s-regulator
;regulator_count          ---pmu ldo count, changed with different pmu
;regulator(1~regulator_count)---ldo details
;-----
[regulator0]
compatible                = "axp221s-regulator"
regulator_count = 20
regulator1                = "axp221s_dcdc1 none vcc-hdmi vcc-io vcc-dsi vcc-usb vdd-efuse vcc-hp vcc-audio
vcc-emmc vcc-card vcc-pe vcc-pd vcc-3v vcc-tvout vcc-tvin vcc-emmcv vcc-sdcv vcc-sdcvq33 vcc-sdcvd
vcc-nand vcc-sdcv-p3 vcc-sdcvq33-p3 vcc-sdcvd-p3"
regulator2                = "axp221s_dcdc2 none vdd-cpu"
regulator3                = "axp221s_dcdc3 none vdd-sys vdd-gpu"
regulator4                = "axp221s_dcdc4 none"
regulator5                = "axp221s_dcdc5 none vcc-dram"
regulator6                = "axp221s_rtc none vcc-rtc"
regulator7                = "axp221s_aldo1 none vcc-25 csi-avdd"
regulator8                = "axp221s_aldo2 none vcc-pa ehpy-vdd25"
regulator9                = "axp221s_aldo3 none avcc vcc-pll"
regulator10               = "axp221s_dldo1 none vcc-io-wifi vcc-pg "
regulator11               = "axp221s_dldo2 none vcc-wifi1"
regulator12               = "axp221s_dldo3 none vcc-wifi2"
regulator13               = "axp221s_dldo4 none vdd-sata-25 vcc-pf"
regulator14               = "axp221s_eldo1 none vcc-pe csi-iovcc csi-afvcc"
regulator15               = "axp221s_eldo2 none csi-dvdd"
regulator16               = "axp221s_eldo3 none vdd-sata-12"
regulator17               = "axp221s_ldoio0 none vcc-ctp"
regulator18               = "axp221s_ldoio1 none vcc-i2s-18"
regulator19               = "axp221s_dc1sw none ephy-dvdd33"
regulator20               = "axp221s_dc5ldo none"

```

由图 3 可以看到该 WIFI 模组是由两路电源(dldo2、dldo3)供电,故上述配置中将其分别命名为 vcc-wifi1、vcc-wifi2,以示区别 (regulator11 和 regulator12 项)。如果读者使用的方案只有一路电源供电,请命名为 vcc-wifi1(数字 1 必不可少);两路电源则分别命名为 vcc-wifi1、vcc-wifi2;其他多路供电时以此类推。vcc-wifi 为设置的电源网络标号,命名可任意,但需与使用时的标号一致(见 wifi_para/IO 口配置部分)。



3.1.3.3.wifi_para/IO 口配置

根据以上 IO 连接信息和 PMU 端口分配信息，可以设置 WIFI 相关的配置信息如下：

```

;-----
;wlan configuration
;clocks:                                     32k clk
;wlan_power_num:                             the number of inputs for wifi power
;wlan_power(n):                               wifi power(n)
;wlan_io_regulator:                           the power of wifi io
;wlan_busnum:                                 no. of bus(usb or bus)
;wlan_regon:                                  wifi function enable io
;wlan_hostwake:                               wifi device wake-up host
;status:                                       okay
;-----
[wlan]
wlan_used                                     = 1
compatible                                    = "allwinner,sunxi-wlan"
clocks                                         = "outa"
;pinctrl-names                                = "default"
;bt_clk_gpio                                  = port:PI12<4><0><0><0>
wlan_power_num                                = 2 (详见①)
wlan_power1                                   = "vcc-wifi1" (详见②)
wlan_power2                                   = "vcc-wifi2"
wlan_io_regulator                             = "vcc-io-wifi" (详见②)
wlan_busnum                                    = 1 (详见③)
wlan_regon                                    = port:PG10<1><1><1><0> (详见④)
wlan_hostwake                                 = port:ower0<0><default><default><default> (详见⑤)

```

说明：

① wlan_power_num 代表供电电源数，wlan_power(n)数字 n(1≤n≤5)必不可少

② “wlan_power”和“wlan_io_regulator”的配置值取决于 PMU 的 sys_config.fex 配置信息，如本示例方案中 PMU 配置信息：

```

regulator10 = "axp221s_dldo1 none vcc-io-wifi vcc-pg "
regulator11 = "axp221s_dldo2 none vcc-wifi1"
regulator12 = "axp221s_dldo3 none vcc-wifi2"

```

③ wlan_busnum 代表使用的 SDIO，如本方案使用 SDIO1。

④ “wlan_regon”为配置控制 wifi enable 的 gpio，对应模块的 Pin 12 脚；

⑤ “wlan_hostwake”为配置 wifi 休眠唤醒主控的 gpio，对应模块的 Pin 13 脚；

3.1.3.4. UART

本方案的AP6212 Bluetooth模组连接情况如图4。

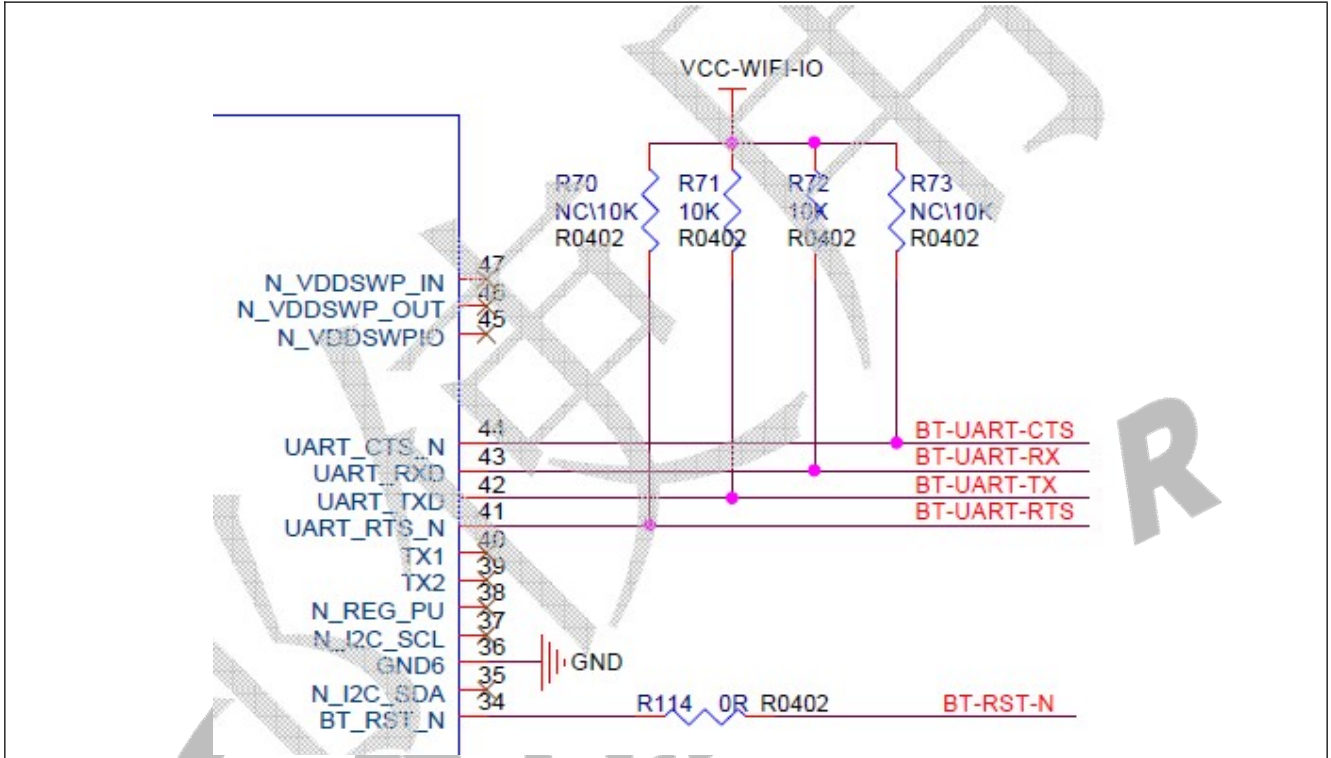


图 4 AP6212 Bluetooth 原理图

结合图 2 可知该方案使用 UART3，UART3 的配置如下：

```
[uart3]
uart3_used      = 1
uart3_port      = 3
uart3_type      = 4
uart3_tx        = port:PG06<4><1><default><default>
uart3_rx        = port:PG07<4><1><default><default>
uart3_rts       = port:PG08<4><1><default><default>
uart3_cts       = port:PG09<4><1><default><default>
```

3.1.3.5. bt_para/Bluetooth IO

蓝牙使用与 WIFI 相同的主电源、相同的上拉电源。主电源设置与 WIFI 模组类似，bt_power_num 代表供电电源数，wlan_power(n) 数字 n(1≤n≤5)必不可少，为 PMU 配置相应的电源网络。

```

;-----
;bt configuration
;clocks:                                     32k clk
;bt_power_num                               the number of inputs for wifi power
;bt_power(n):                               bt power (n)
;wlan_io_regulator:                         the power of bt io
;bt_rst_n:                                  bt function enable io
;status:                                     okay
;-----
[bt]
bt_used                                     = 1
compatible                                 = "allwinner,sunxi-bt"
clocks                                     = "outa"
pinctrl-names                             = "default"
clock_io                                   = port:PI12<4><0><0><0>
bt_power_num                              = 2
bt_power1                                  = "vcc-wifi1"
bt_power2                                  = "vcc-wifi2"
bt_io_regulator                           = "vcc-io-wifi"
bt_rst_n                                   = port:PH12<1><1><1><0> (详见①)
;-----
;btlpm configuration
;uart_index:                               uart0, uart1 or uart3
;bt_wake:                                  host wake-up bluetooth device
;bt_wak_host:                              bt device wake-up host
;status:                                     okay
;-----
[btlpm]
btlpm_used                                 = 0
compatible                                 = "allwinner,sunxi-btlpm"
uart_index                                = 3
bt_wake                                   = port:PG11<1><1><1><0> (详见①)
bt_host_wake                              = port:power1<0><default><default><default> (详见①)
    
```

说明：

- ① “bt_rst_n”，“bt_wake”，“bt_host_wake” 参照具体硬件原理图配置，这里不再赘举；
- ② 其他选项请用默认配置即可。

3.1.4. AP6212 模组移植相关文件

以下文件是与 AP6212 模组移植相关的，无需再对这些文件作修改，只需了解即可。

- ① AP6212 驱动代码：

```
tina/lichee/linux-4.4/drivers/net/wireless/bcmdhd
```

- ② 电源及 GPIO 控制

```
tina/lichee/linux-4.4/drivers/misc/sunxi-rf/
```

注：

- a. broadcom 的 ap6xxx 系列模组均是使用相同的 bcmdhd 驱动；
- b. broadcom 的 ap6xxx 系列通过 fw 来表现不同的 feature；

3.2. RTL8188EU

功能：wifi station、softap、wifi direct
接口类型：USB

3.2.1. 内核配置

在 Tina 目录下执行：

```
make kernel_menuconfig
```

选择编译 RTL8188EU 为模块

```
Device Drivers --->
```

```
Network device support --->
```

```
Wireless LAN --->
```

```
<M> Realtek 8188E USB WIFI
```

3.2.2. Tina 配置

内核选定之后，Tina 进行相关配置。

在 Tina 目录下执行：

```
$ make menuconfig
```

选中使用 RTL8188EU，系统就会将 RTL8188EU 的驱动模块拷贝到制定位置，使得系统固件烧写后在 Tina 系统中保存，并且在系统启动时能够自动加载。

以下配置实现 wifi 驱动的拷贝以及开机自动加载：

```
Kernel modules--->
```

```
Wireless Drivers--->
```

```
<*> kmod-net-rtl8188eu
```

以下配置编译拷贝 wifi 的 firmware

```
Firmware--->
```

```
<*> r8188eu-firmware.
```

3.2.3. sys_config.fex

sys_config.fex 文件位于

tina/target/allwinner/xxxxx/configs/sys_config.fex

为 Allwinner 对于特定硬件平台进行参数配置的文件。修改 sys_config.fex 文件时需查看原理图确定相关硬件连接信息。

如图 5 所示，我们可以看到该方案使用网络 VCC-IO-WIFI 为 RTL8188EU WIFI 模组供电，图 6 说明了 VCC-IO-WIFI 连接至电源管理芯片的 DLDO1 端；系统使用 USB2 与主控进行通讯。

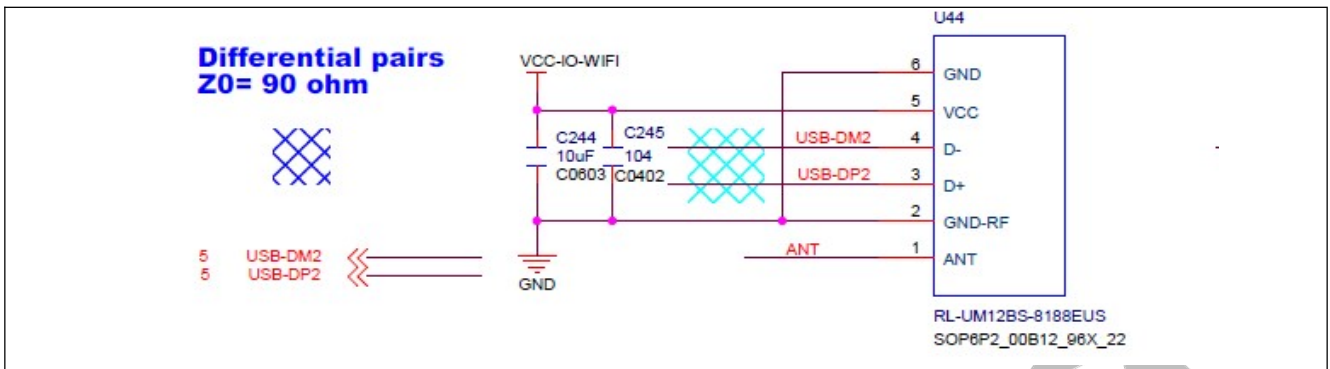


图 5 RTL8188EU 连接图

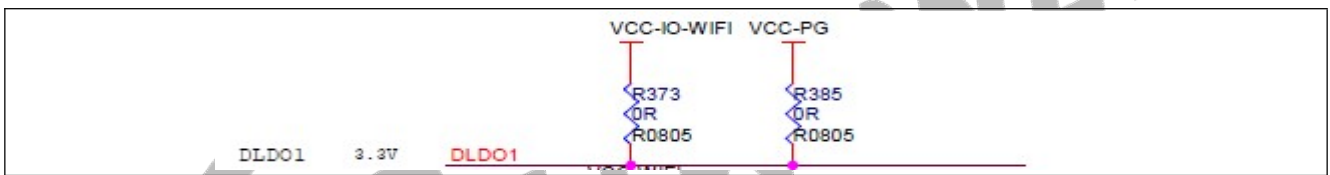


图 6 电源网络连接

根据电路原理图分别修改 sys_config.fex 文件如下：

3.2.3.1. 修改 USB 配置

```

;-----
;---      USB2 CONFIG
;-----
;
[usbc2]
usbc2_used      = 1
usb_drv_vbus_gpio =
usb_host_init_state = 1
usb_regulator_io   = "nocare"
usb_wakeup_suspend = 0
    
```

其中：设置 usbc2_used=1 表示使用 USB2 接口；“；”符号起注释作用。

3.2.3.2. 修改 wlan 设置

```

;-----
;wlan configuration
;clocks:                                     32k clk
;wlan_power_num:                             the number of inputs for wifi power
;wlan_power(n):                               wifi power(n)
;wlan_io_regulator:                           the power of wifi io
;wlan_busnum:                                 no. of bus(usb or bus)
;wlan_regon:                                  wifi function enable io
;wlan_hostwake:                               wifi device wake-up host
;status:                                       okay
;-----
[wlan]
wlan_used                                     = 1
compatible                                   = "allwinner,sunxi-wlan"
;clocks                                       = "outa"
;pinctrl-names                               = "default"
;bt_clk_gpio                                  = port:PI12<4><0><0><0>
wlan_power_num                               = 1
wlan_power1                                  = "vcc-io-wifi"
wlan_io_regulator                            = "vcc-io-wifi"
wlan_busnum                                  = 2
wlan_regon                                    = port:PG10<1><1><1><0>
wlan_hostwake                                = port:ower0<0><default><default><default>
    
```

(详见③)

(详见①)

(详见②)

说明:

① wlan_power1 = "vcc-io-wifi"为设置 WIFI 使用的电源网络，“wlan_power1”的配置值取决于 PMU 的 sys_config.fex 配置信息，如本示例方案中 PMU 配置信息 regulator10 = "axp221s_dldo1 none vcc-io-wifi vcc-pg"（同 3.1.3 方案电源管理配置部分），故 wlan_power1 = "vcc-io-wifi"；

② wlan_busnum = 2 表示 WIFI 使用 USB2。

③ wlan_power_num 代表供电电源数，wlan_power(n)数字 n(1≤n≤5)必不可少。

3.2.4. RTL8188EU 模组移植相关文件

以下文件是与 RTL8188EU 模组移植相关的，无需再对这些文件作修改，只需了解即可。

1. RTL8188EU驱动代码

```
tina/lichee/linux-4.4/drivers/net/wireless/rtl8188eu/
```

2. 电源控制

```
tina/lichee/linux-4.4/drivers/misc/sunxi-rf/sunxi-wlan.c
```

```
tina/lichee/linux-4.4/drivers/net/wireless/rtl8188eu/platform/platform_ARM_SUNxI_usb.c
```

其中 platform_ARM_SUNxI_usb.c 是 RTL8188EU 模组电源控制和 USB 接口选择实现文件，它调用了 sunxi-wlan.c 提供的用于 WIFI 驱动启动的相关接口。

3.3. XR819

功能: wifi station、softap、wifi direct

接口类型: SDIO

3.3.1. 内核配置

在 Tina 目录下执行:

全志科技版权所有，侵权必究

Copyright © 2018 by Allwinner. All rights reserved

Page 16 of 23


```
$ make kernel_menuconfig
```

选择编译 XRadio 无线网卡驱动为模块
wifi:(编译成模块)

```
Device Drivers --->
```

```
    Network device support --->
```

```
        Wireless LAN --->
```

```
            <M> XRadio WLAN support --->
```

3.3.2. Tina 配置

内核选定之后，tina 进行相关配置

在 Tina 目录下执行：

```
$ make menuconfig
```

选中使用 XR819，系统就会将 XR819 的驱动模块拷贝到制定位置，使得系统固件烧写后在 Tina 系统中保存，并且在系统启动时能够自动加载。

以下配置实现 WIFI 驱动拷贝以及开机自动加载：

```
Kernel modules--->
```

```
    Wireless Drivers--->
```

```
        *- komd-cfg8021
```

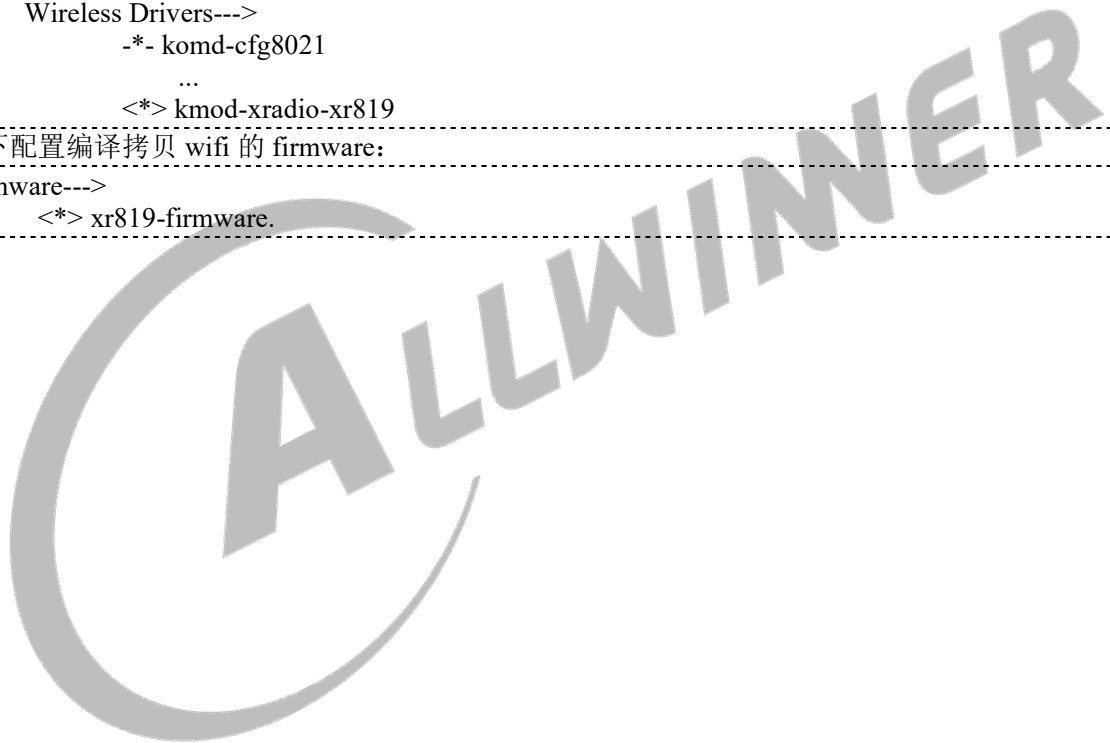
```
        ...
```

```
        <*> kmod-xradio-xr819
```

以下配置编译拷贝 wifi 的 firmware：

```
Firmware--->
```

```
    <*> xr819-firmware.
```



3.3.3. sys_config.fex

sys_config.fex 文件位于

/tina/target/allwinner/xxxxx/configs/sys_config.fex

为 Allwinner 对于特定硬件平台进行参数配置的文件，sys_config.fex 文件决定使用哪款 WIFI 模组和 GPIO 的分配。修改 sys_config.fex 文件时需查看原理图确定相关硬件连接信息。

3.3.3.1. SDIO/MMC 配置

如图 1 所示，我们可以看到该方案直接使用直流电源 VCC-3V3 为 XR819 WIFI 模组供电。图 2 说明了使用 WIFI 时系统使用 SDIO0 与主控进行通讯。

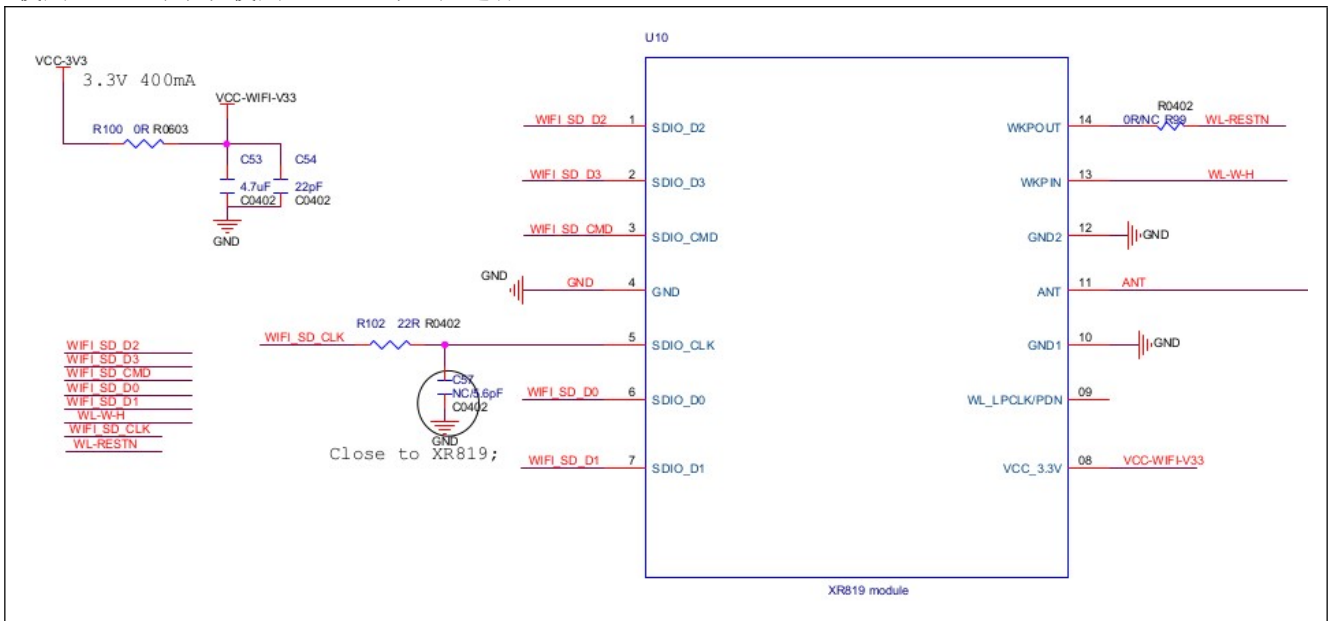


图 1 XR819 WIFI 原理图

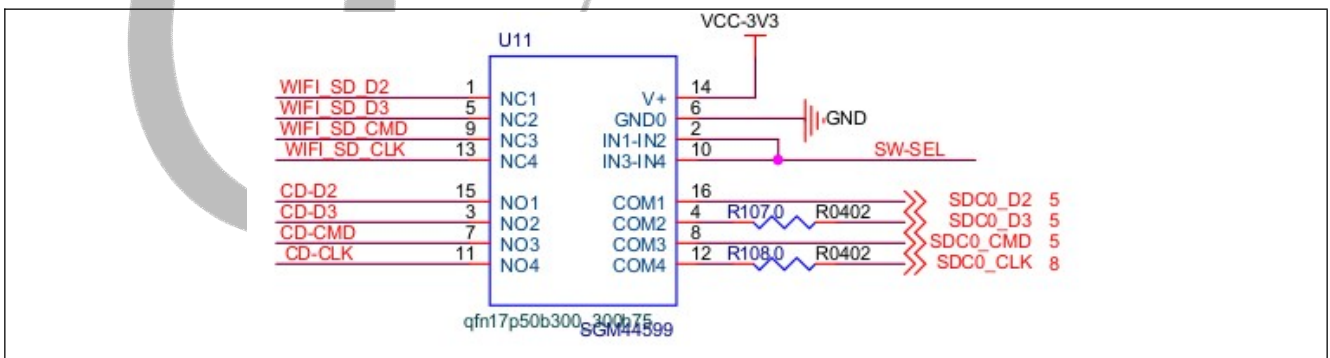


图 2 GPIO 使用情况

根据图 1 和图 2, 使用 AP6212 模组需要把 sys_config.fex 文件的 SDIO 相关部分改成如下 (部分代码):

```

-----
; SDMMC PINS MAPPING
;
; Config Guide
; sdc_used: 1-enable card, 0-disable card
; non-removable:if you use as main memory,you should set it,for example eMMC
; bus-width: card bus width, 1-1bit, 4-4bit, 8-8bit
; sunxi-power-save-mode: if use sdio card,should not set it
; vmmc:regulator for card/emmc power
; vqmmc:regulator for card/emmc io power
; vdmcc:regulator for card detect pin pull up power
; other: GPIO Mapping configuration
;
; Note:
;
;
-----
[sdc0]
sdc0_used          = 1
non-removeable    =
bus-width         = 4
sdc0_d1           = port:PF00<2><1><3><default>
sdc0_d0           = port:PF01<2><1><3><default>
sdc0_clk          = port:PF02<2><1><3><default>
sdc0_cmd          = port:PF03<2><1><3><default>
sdc0_d3           = port:PF04<2><1><3><default>
sdc0_d2           = port:PF05<2><1><3><default>
;cd-gpios         = port:PE02<0><1><3><default>
;broken-cd        =
vmmc              = "none"
vqmmc             = "none"
vdmcc             = "none"
ctl-spec-caps     = 0x8

```

其中：“;”符号起注释作用。

3.3.3.2. PMU 电源端口配置

由于示例方案使用了直流电源网络直接给 WIFI 模组供电，因此无需软件上进行电源端口的配置。如实际方案使用了电源管理芯片对 WIFI 模组电源进行控制，请参考其他模组 PMU 电源端口配置部分进行配置。

3.3.3.3. wifi_para/IO 口配置

根据以上 IO 连接信息和 PMU 端口分配信息，可以设置 WIFI 相关的配置信息如下：

```

;-----
;wlan configuration
;clocks:                                     32k clk
;wlan_power_num:                             the number of inputs for wifi power
;wlan_power(n):                               wifi power(n)
;wlan_io_regulator:                           the power of wifi io
;wlan_busnum:                                 no. of bus(usb or bus)
;wlan_regon:                                  wifi function enable io
;wlan_hostwake:                               wifi device wake-up host
;status:                                       okay
;-----
[wlan]
wlan_used                                     = 1
compatible                                   = "allwinner,sunxi-wlan"
clocks                                       = "outa"
pinctrl-names                               = "default"
bt_clk_gpio                                  = port:PI12<4><0><0><0>
;wlan_power_num                             = 2 (详见①)
;wlan_power1                                = "vcc-wifi1" (详见②)
;wlan_power2                                = "vcc-wifi2" (详见②)
wlan_io_regulator                            = "vcc-io-wifi" (详见③)
wlan_busnum                                  = 1
wlan_regon                                   = port:PD16<1><1><3><0> (详见④)
wlan_hostwake                                = port:PE02<6><default><default><default> (详见⑤)
    
```

说明：

- ① wlan_power_num 代表供电电源数，wlan_power(n) 数字 n(1≤n≤5) 必不可少
- ② “wlan_power”和“wlan_io_regulator”的配置值取决于 PMU 的 sys_config.fex 配置信息，如本示例方案中 PMU 配置信息：

```

regulator10 = "axp221s_dldo1 none vcc-io-wifi vcc-pg "
regulator11 = "axp221s_dldo2 none vcc-wifi1"
regulator12 = "axp221s_dldo3 none vcc-wifi2"
    
```

- ③ wlan_busnum 代表使用的 SDIO，如本方案使用 SDIO0。
- ④ “wlan_regon”为配置控制 wifi enable 的 gpio，对应模组的 Pin 14 脚(随 XR819 模组版本的不同，引脚号可能不同，具体请咨询 WIFI 模组服务支持商或原厂)；

注意：如果是 R6-perfl 标案或者参考 R6-perfl 标案修改设计的方案，需注意 wlan_regon 使用的是通过 IIC 芯片扩展的虚拟 GPIO，则 wlan_regon 的配置是直接电源管理驱动 lichee/linux-4.4/drivers/misc/sunxi-rf/sunxi-wlan.c 里进行：

```

#if defined(VIRTUAL_EXTENDED_GPIO) //如果是 R6 标案
printf("-----SUNXI_RF: Set regon for SUN3IW1P1_R6!-----\n");
data->gpio_wlan_regon = 2042; //指定 wlan_regon 使用的虚拟扩展 GPIO
if (!gpio_is_valid(data->gpio_wlan_regon)) {
    
```

```

        dev_err(dev, "get gpio wlan_regon failed\n");
    } else {
        ret = devm_gpio_request(dev, data->gpio_wlan_regon, "wlan_regon");
        if (ret < 0) {
            dev_err(dev, "can't request wlan_regon gpio %d\n",
                    data->gpio_wlan_regon);
            goto end;
        }
    }
}
#else
...

```

如需更改，请直接修改 sunxi-wlan.c 源码。

⑤ “wlan_hostwake”为配置 wifi 休眠唤醒主控的 gpio，对应模块的 Pin 13 脚(随 XR819 模组版本的不同，引脚号可能不同，具体请咨询 WIFI 模组服务支持商或原厂)；

3.3.4. XR819 模组移植相关文件

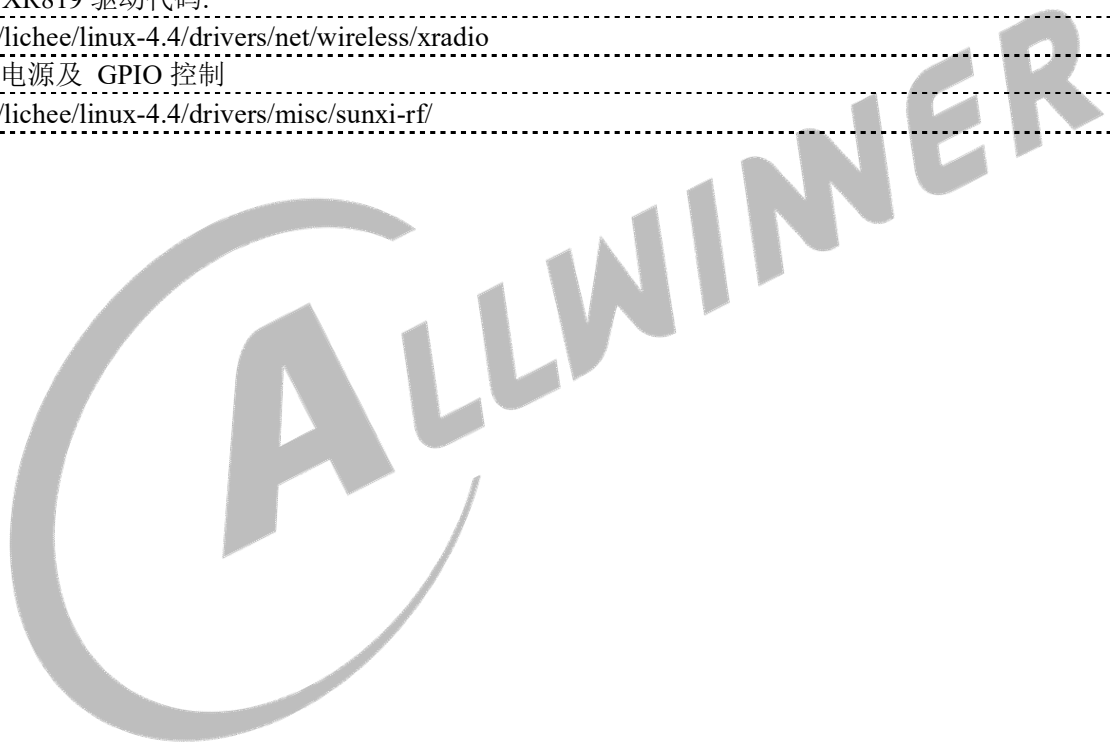
以下文件是与 XR819 模组移植相关的，无需再对这些文件作修改，只需了解即可。

③ XR819 驱动代码:

tina/lichee/linux-4.4/drivers/net/wireless/xradio

④ 电源及 GPIO 控制

tina/lichee/linux-4.4/drivers/misc/sunxi-rf/



4. 上层定制开发及 demo

请参考《Tina Linux Bluetooth API》以及《Tina Linux Wifimanager API》



5. Declaration

This document is the original work and copyrighted property of Allwinner Technology (“Allwinner”). Reproduction in whole or in part must obtain the written approval of Allwinner and give clear acknowledgment to the copyright owner.

The information furnished by Allwinner is believed to be accurate and reliable. Allwinner reserves the right to make changes in circuit design and/or specifications at any time without notice. Allwinner does not assume any responsibility and liability for its use. Nor for any infringements of patents or other rights of the third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Allwinner. This datasheet neither states nor implies warranty of any kind, including fitness for any particular application.

