

# IDO-EVB3020 - Linux 接口配置

---

## 一、资源列表

## 二、接口配置

### 1、LCD

(1) MIPI

(2) LVDS

### 2、TP

### 3、USB

(1) USB OTG

(2) USB HOST

### 4、TF Card

### 5、以太网

### 6、WIFI/BT

### 7、4G

### 8、扬声器

### 9、耳机

### 10、麦克风

### 11、CAN

### 12、RTC

### 13、UART

### 14、ADC按键

### 15、ADC IN

# IDO-EVB3020

## Linux 接口配置

深圳触觉智能科技有限公司

[www.industio.cn](http://www.industio.cn)

### 一、资源列表

| 序号 | 名称   | 描述                 | 设备节点 |
|----|------|--------------------|------|
| 1  | 内核版本 | 4.4.189            |      |
| 2  | 文件系统 | Debian / Buildroot |      |
| 3  | 内存   | DDR4 (1GB/2GB选配)   |      |

|    |          |                                        |                                                      |
|----|----------|----------------------------------------|------------------------------------------------------|
| 4  | 存储       | eMMC 5.1<br>(8GB/16GB/32GB/128GB 可选配)  | /dev/mmcblk1                                         |
| 5  | 供电       | 12V DC @2A                             |                                                      |
| 6  | 显示       | MIPI 、 LVDS                            |                                                      |
| 7  | USB OTG  | USB 2.0 OTG X 1                        |                                                      |
| 8  | USB HOST | USB 2.0 Type-A X 1<br>USB 2.0 PH-4 X 1 |                                                      |
| 9  | TF Card  | Micro SD 卡                             | /dev/mmcblk0                                         |
| 10 | 以太网      | 自适应10/100/100Mbps以太网                   | eth0                                                 |
| 11 | WiFi/BT  | RT8723DU                               | wlan0、 hci0                                          |
| 12 | 4G       | 支持MiniPCIE座子接口的4G模组, EC20              |                                                      |
| 13 | 扬声器      | 双声道 4Ω3W 扬声器                           |                                                      |
| 14 | 耳机       | 3.5mm 耳机                               |                                                      |
| 15 | CAN      | spi转CAN X 1                            | can0                                                 |
| 16 | 摄像头      | OV5648 mipi摄像头                         |                                                      |
| 17 | RTC      | HYM8563                                | /dev/rtc0                                            |
| 18 | RS232    | RS232 X 4                              | /dev/ttyS0<br>/dev/ttyS1<br>/dev/ttyS4<br>/dev/ttyS5 |
| 19 | RS485    | RS485 X 2                              | /dev/ttyS2 (调试串口和RS485 2选1)<br>/dev/ttyS3            |
| 20 | 调试串口     | TTL X 1                                | /dev/ttyFIQ0                                         |
| 21 | ADC按键    | 1路                                     | /dev/input/event1                                    |



KERNEL 开启背光驱动:

```
.config = Linux/armv4 4.4.189 Kernel Configuration
> Device Drivers > Graphics support > Backlight & LCD device support
Backlight & LCD device support
* Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters
* are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?>
* for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable
*
* l Backlight & LCD device support
* x   < > Lowlevel LCD controls
* x   < > Lowlevel Backlight controls
* x   <*> Generic (aka Sharp Corgi) Backlight Driver
* x   <*> Generic PWM based Backlight Driver
* x   < > Qualcomm PM8941 WLED Driver
* x   < > Backlight Driver for ADP8860/ADP8861/ADP8863 using WLED
* x   < > Backlight Driver for ADP8870 using WLED
* x   < > Backlight Driver for LM3630A
* x   < > Backlight Driver for LM3639
* x   < > Backlight driver for TI LP855X
* x   < > Generic GPIO based Backlight Driver
* x   < > Sanyo LV5207LP Backlight
* x   < > Rohm BD6107 Backlight
```

配置DTS:

使能背光：

```
backlight: backlight {
    status = "okay";
    compatible = "pwm-backlight";
    pwms = <&pwm0 0 25000 1>;
    brightness-levels = <
        0  1  2  3  4  5  6  7
        8  9 10 11 12 13 14 15
        16 17 18 19 20 21 22 23
        24 25 26 27 28 29 30 31
        32 33 34 35 36 37 38 39
        40 41 42 43 44 45 46 47
        48 49 50 51 52 53 54 55
        56 57 58 59 60 61 62 63
        64 65 66 67 68 69 70 71
        72 73 74 75 76 77 78 79
        80 81 82 83 84 85 86 87
        88 89 90 91 92 93 94 95
        96 97 98 99 100 101 102 103
        104 105 106 107 108 109 110 111
        112 113 114 115 116 117 118 119
        120 121 122 123 124 125 126 127
        128 129 130 131 132 133 134 135
        136 137 138 139 140 141 142 143
        144 145 146 147 148 149 150 151
        152 153 154 155 156 157 158 159
        160 161 162 163 164 165 166 167
        168 169 170 171 172 173 174 175
        176 177 178 179 180 181 182 183
        184 185 186 187 188 189 190 191
        192 193 194 195 196 197 198 199
        200 201 202 203 204 205 206 207
        208 209 210 211 212 213 214 215
        216 217 218 219 220 221 222 223
        224 225 226 227 228 229 230 231
        232 233 234 235 236 237 238 239
        240 241 242 243 244 245 246 247
        248 249 250 251 252 253 254 255>;
    default-brightness-level = <200>;
};
```

使能PWM0：

```
&pwm0 {
    status = "okay";
};
```

## (1) MIPI

点MIPI屏主要注意：MIPI时钟、初始化数据、背光、复位、供电。

背光我们前面配好了，接下来配置MIPI。

[illegible]

### DTS配置:

```
&dsi_in_vopl {  
    status = "disabled";  
};  
  
&dsi_in_vopb {  
    status = "okay";  
};
```

```
&route_dsi {  
    status = "okay";  
};  
  
&vopb {  
    status = "okay";  
};  
  
&vopb_mmu {  
    status = "okay";  
};  
  
&vopl {  
    status = "okay";  
};  
  
&vopl_mmu {  
    status = "okay";  
};  
  
&video_phy {  
    status = "okay";  
};
```



```

&display_subsystem {
    status = "okay";
    ports = <&vopb_out>, <&vopl_out>;
    logo-memory-region = <&drm_logo>;

    route {
        route_lvds: route-lvds {
            status = "disabled";
            logo,uboot = "logo.bmp";
            logo,kernel = "logo_kernel.bmp";
            logo,mode = "center";
            charge_logo,mode = "center";
            connect = <&vopb_out_lvds>;
        };

        route_dsi: route-dsi {
            status = "disabled";
            logo,uboot = "logo.bmp";
            logo,kernel = "logo_kernel.bmp";
            logo,mode = "center";
            charge_logo,mode = "center";
            connect = <&vopb_out_dsi>;
        };

        route_rgb: route-rgb {
            status = "disabled";
            logo,uboot = "logo.bmp";
            logo,kernel = "logo_kernel.bmp";
            logo,mode = "center";
            charge_logo,mode = "center";
            connect = <&vopb_out_rgb>;
        };
    };
};

```

```

&dsi {
    status = "okay";
    rockchip,cmd_debug = <0>;
    panel@0 {
        compatible = "simple-panel-dsi";
        reg = <0>;
        power-supply = <&vcc3v3_lcd>;
        backlight = <&backlight>;
        prepare-delay-ms = <2>;
        reset-delay-ms = <1>;
        init-delay-ms = <20>;
        enable-delay-ms = <120>;
        //disable-delay-ms = <50>;
        //unprepare-delay-ms = <20>;

        //pinctrl-0 = <&lcd_en>;

        width-mm = <293>;
        height-mm = <165>;

        //enable-gpios = <&gpio0 RK_PB5 GPIO_ACTIVE_LOW>;
        //reset-gpios = <&gpio1 13 GPIO_ACTIVE_HIGH>;

        dsi,flags = <(MIPI_DSI_MODE_VIDEO | MIPI_DSI_MODE_VIDEO_BURST |
                    MIPI_DSI_MODE_LPM | MIPI_DSI_MODE_EOT_PACKET)>;
        dsi,format = <MIPI_DSI_FMT_RGB888>;
        dsi,lanes = <4>;

        panel-init-sequence = [
                                05 78 01 11
                                05 14 01 29
                                ];

        panel-exit-sequence = [
                                05 00 01 28
                                05 78 01 10
                                ];
    };
};

```

```

display-timings {
    native-mode = <&timing0>;
    compatible = "rockchip,display-timings";
    timing0: timing0 {
        clock-frequency = <148000000>;
        hactive = <1200>;
        vactive = <1920>;
        hback-porch = <130>;
        hfront-porch = <160>;
        vback-porch = <16>;
        vfront-porch = <16>;
        hsync-len = <4>;
        vsync-len = <2>;
        hsync-active = <0>;
        vsync-active = <0>;
        de-active = <0>;
        pixelclk-active = <0>;
    };
};
};
};

```

如果有打开logo显示，route\_dsi的connect的属性需要指定同一个开启的VOP。

```

&route_dsi {
    connect = <&vopb_out_dsi>;
    status = "okay";
};

```

这里我们主要关注panle 和 display-timing。

PANLE:

| 参数名                | 描述                 | 备注                  |
|--------------------|--------------------|---------------------|
| power-supply       | 屏幕电源               | 电源不能弄错              |
| compatible         | 对应驱动               | 默认 simple-panel-dsi |
| backlight          | 屏幕背光               |                     |
| prepare-delay-ms   | 配置panel接收图像数据前的延时值 |                     |
| reset-delay-ms     | 配置reset的脉冲延时值      |                     |
| init-delay-ms      | 配置panel在初始化前的延时值   |                     |
| enable-delay-ms    | 配置panel显示前的延时值     |                     |
| disable-delay-ms   | 配置panel关闭前的延时值     |                     |
| unprepare-delay-ms | 配置panel黑屏前的延时值     |                     |

|                     |                    |                                                                                |
|---------------------|--------------------|--------------------------------------------------------------------------------|
| width-mm            | 配置panel的物理宽度尺寸     |                                                                                |
| height-mm           | 配置panel的物理高度尺寸[mm] |                                                                                |
| dsi,flags           | 配置panel的使用的模式      |                                                                                |
| dsi,format          | 配置panel像素格式        |                                                                                |
| dsi,lanes           | 配置panel使用的lane数    |                                                                                |
| panel-init-sequence | 屏幕初始化数据            | byte 0: dcs 数据类型<br>byte 1: 指令发送之后的延时<br>byte 2: 指令长度<br>byte 3: 数据（寄存器+初始化数据） |
| panel-exit-sequence | 屏幕退出数据             | byte 0: dcs 数据类型<br>byte 1: 指令发送之后的延时<br>byte 2: 指令长度<br>byte 3: 数据（寄存器+初始化数据） |

表中数据可在屏幕规格书或向屏幕供应商获取。

如果屏幕某些引脚需要IO控制，也可添加在PANLE中，例如添加屏幕复位IO。

```
reset-gpios = <&gpio1 13 GPIO_ACTIVE_HIGH>;
```

因为该MIPI屏不需要初始化数据，我们可以参考一下其他mipi屏的panel-init-sequence配置

```

panel-init-sequence = [
    39 00 03 F0 5A 5A
    39 00 03 F1 5A 5A
    39 00 03 FC 5A 5A
    39 00 05 CA 00 54 05 28
    39 00 02 B2 41
    39 00 02 E3 26
    39 00 05 F2 10 10 0E 0D
    39 00 13 F5 A6 B5 10 66 27 1D 4C 4C 03 03 04 22 11 31 50 2A 16 75
    39 00 02 B0 04
    39 00 02 F6 40
    39 00 02 B0 1C
    39 00 02 EF 01
    39 00 02 B0 03
    39 00 02 EF 44
    39 00 02 B0 01
    39 00 02 EF 09
    39 00 05 EE 89 67 89 67
    39 00 06 ED C0 2A 10 2A 10
    39 00 17 F7 24 25 2E 2F 18 1A 14 16 04 01 06 01 01 01 01 01 01 01 01 01 01
    39 00 17 F8 24 25 2E 2F 19 1B 15 17 05 01 07 01 01 01 01 01 01 01 01 01 01
    39 00 12 FA 10 30 13 16 0A 0D 12 10 10 18 1D 1E 20 21 22 25 2C
    39 00 12 FB 10 30 13 16 0A 0D 12 10 10 18 1D 1E 20 21 22 25 2C
    05 78 01 11
    05 14 01 29
];

```

其中的数据类型的使用方式，可以参考此文档：

[Rockchip\\_DRM\\_Panel\\_Porting\\_Guide\\_V1.6\\_20190228.pdf](#)

## display-timing:

这部分的内容很重要。所用到的参数可以在规格书中或向屏幕供应商获取。

clock-frequency算法：  $(hsync-len + hback-porch + hback-porch + hactive) * (vsync-len + vback-porch + vfront-porch + vactive) * FPS / 1048576$

有些时候，厂商给的时钟参数不一定能到我们想要的显示效果，这时候我们就需要根据显示情况调整以下参数。

```

hback-porch = <130>;
hfront-porch = <160>;
vback-porch = <16>;
vfront-porch = <16>;
hsync-len = <4>;
vsync-len = <2>;
hsync-active = <0>;

```

## 点屏注意事项:

- (1) 供电是否正常。

- (2) 座子是否虚焊。
- (3) 背光是否开启。
- (4) 屏参是否正确。（有些屏在不同的平台参数会有差异，这个可以跟屏厂确认一下。）
- (5) 时钟是否正确。（在不同的平台时钟参数可能需要进行调整的。）

## (2) LVDS

LVDS与MIPI配置类似只需要调整DTS即可。

**DTS:**

```

panel {
    status = "okay";
    compatible = "simple-panel";
    backlight = <&backlight>;
    bus-format = <MEDIA_BUS_FMT_RGB888_1X32_PADHI>; /* vesa-24 */
    //enable-gpios = <&gpio0 RK_PB5 GPIO_ACTIVE_LOW>;
    //pinctrl-0 = <&lcd_en>;
    power-supply = <&vcc3v3_lcd>;
    enable-delay-ms = <100>;
    prepare-delay-ms = <100>;
    //unprepare-delay-ms = <120>;
    //disable-delay-ms = <120>;

    width-mm = <217>;
    height-mm = <136>;

    rockchip,data-mapping = "vesa";
    //rockchip,data-mapping = "jeida";
    rockchip,data-width = <24>;
    rockchip,output = "lvds";

    display-timings {
        native-mode = <&timing1>;
        timing1: timing1 {
            clock-frequency = <51287040>;
            hactive = <1024>;
            vactive = <600>;
            hback-porch = <160>;
            hfront-porch = <136>;
            vback-porch = <23>;
            vfront-porch = <12>;
            hsync-len = <24>;
            vsync-len = <1>;
            hsync-active = <0>;
            vsync-active = <0>;
            de-active = <0>;
            pixelclk-active = <0>;
        };
    };

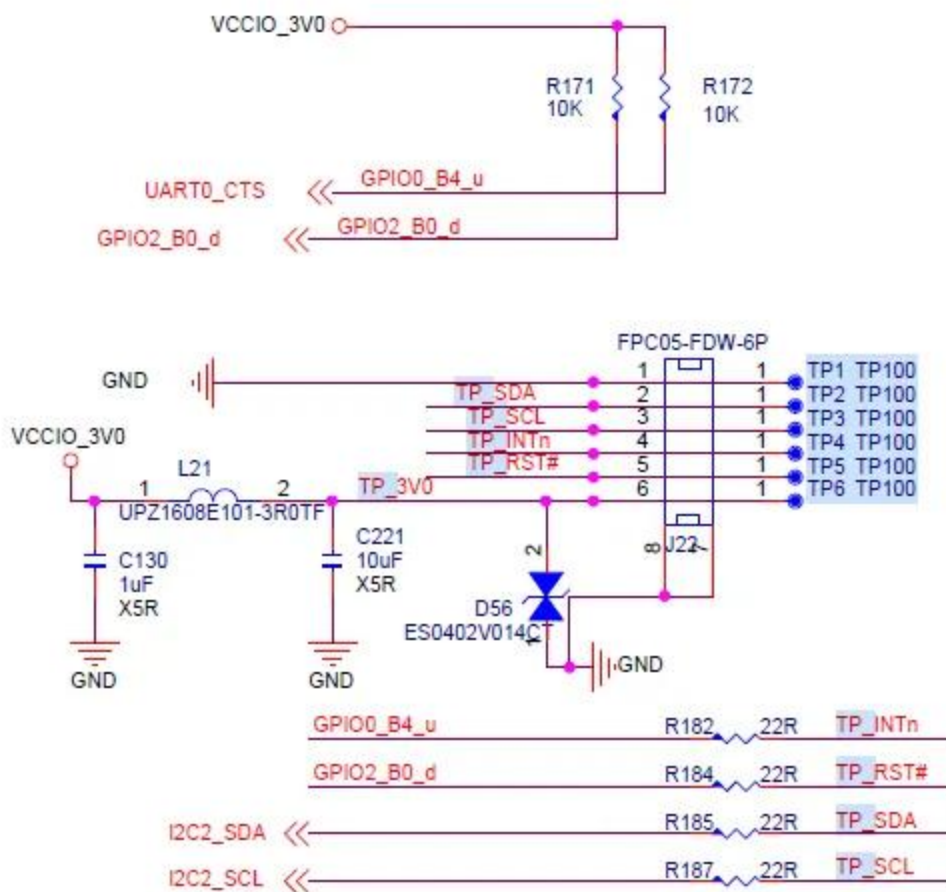
    port {
        panel_in_lvds: endpoint {
            remote-endpoint = <&lvds_out_panel>;
        };
    };
};
};

```

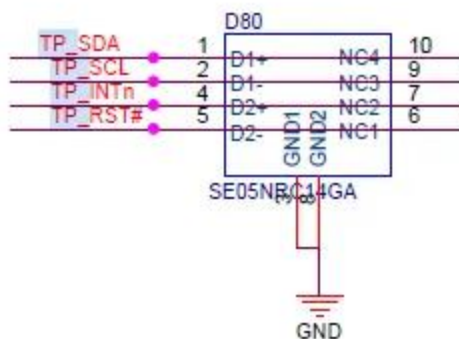
```
&lvds {  
    status = "okay";  
  
    ports {  
        port@1 {  
            reg = <1>;  
  
            lvds_out_panel: endpoint {  
                remote-endpoint = <&panel_in_lvds>;  
            };  
        };  
    };  
};
```

## 2、TP





## TP-LCD



TP原理图

由原理图可知：

| 引脚名称               | GPIO       | 功能   |
|--------------------|------------|------|
| TP_INIT_GPIO0_B4_u | GPIO0_B4_u | tp中断 |
| TP_RST_GPIO2_B0_d  | GPIO2_B0_d | tp复位 |

|          |  |  |
|----------|--|--|
| I2C2_SDA |  |  |
| I2C2_SCL |  |  |

KERNEL配置：

打开GT911的驱动配置。

```
> Device Drivers > Input device support > Touchscreens
--- Touchscreens
< > ADS7846/TSC2046/AD7873 and AD(S)7843 based touchscreens
< > AD7877 based touchscreens
< > Analog Devices AD7879-1/AD7889-1 touchscreen interface
< > Microchip AR1021 i2c touchscreen
< > Atmel mXT I2C Touchscreen
< > AUO in-cell touchscreen using Pixcir ICs
< > BU21013 based touch panel controllers
< > chipone icn8318 touchscreen controller
< > cy8c40xx touchpad based on IIC
< > cy8ctmgl10 touchscreen
< > Cypress TTSP touchscreen
< > Cypress TrueTouch Gen4 Touchscreen Driver
< > Dynapro serial touchscreen
< > Hampshire serial touchscreen
< > EETI touchscreen panel support
< > EETI eGalax multi-touch panel support
< > FT6236 I2C touchscreen
< > Fujitsu serial touchscreen
< > Goodix I2C touchscreen
< > gslx 680a
< > gslX680 d708 touchpad driver for rk3126 pad
< > gslX680 touchpad driver for pad
< > gslX680 touchpad driver for VR
< > gslX680 touchpad driver for firefly-rk3288
< > gsl3673 touchpad driver for evb
< > gsl3673 touchpad driver for rk3326 board
< > Goodix gt9xx support for rockchip platform
< > Hitec ILI210X based touchscreen
```

DTS 配置：

```
&i2c2 {
    status = "okay";
    clock-frequency = <100000>;

    /* These are relatively safe rise/fall times; TODO: measure */
    i2c-scl-falling-time-ns = <50>;
    i2c-scl-rising-time-ns = <300>;

    gt911@14 {
        compatible = "goodix,gt9xx";
        reg = <0x14>;
        pinctrl-names = "default";
        pinctrl-0 = <&gt;gtxx_gpio>;
        goodix_irq_gpio = <&gpio0 RK_PB4 IRQ_TYPE_LEVEL_LOW>;
        goodix_rst_gpio = <&gpio2 RK_PB0 GPIO_ACTIVE_HIGH>;
        touchscreen-inverted-y;
        touchscreen-swapped-x-y;
        status = "okay";
    };
};
```

```
goodix_gpio {
    >gtxx_gpio:gtxx-gpio {
        rockchip,pins =
            <0 RK_PB4 RK_FUNC_GPIO &pcfg_pull_none>,
            <2 RK_PB0 RK_FUNC_GPIO &pcfg_pull_none>;
    };
};
```

TP调试：

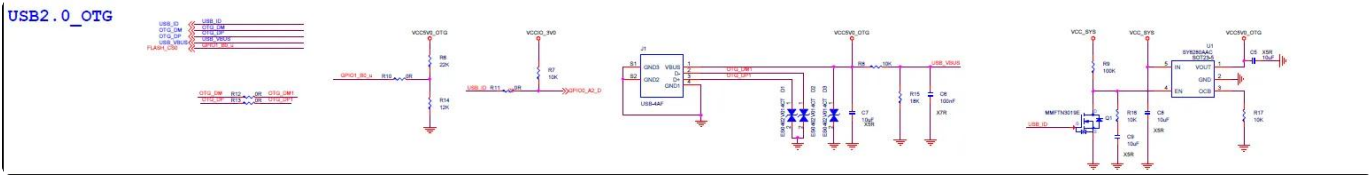
- (1) 配置好KERNEL和DTS，重新编译烧录系统，打印log（dmesg | grep good\*）检查是否配置正确。
- (2) 输入 i2cdetect -y -r 2 查看是否出现TP的设备号。
- (3) 触摸有节点，但是无反应。可以cat TP的对应节点，并按下触摸屏，检查中断是否正常相应：cat /dev/eventX 。（ps:这里也可以用示波器或电压表排查）

### 3、USB

详细配置参考：

 [Rockchip\\_Developer\\_Guide\\_USB\\_CN.pdf](#)

# (1) USB OTG



有原理图可知：

| 引脚名称       | 功能    |
|------------|-------|
| GPIO0_A2_u | OTG状态 |
| GPIO1_B0_d | USB电压 |

kernel 配置：



```

Device Drivers > USB support
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable

--- USB support
  [*] Support for Host-side USB
  [*] USB announce new devices
      *** Miscellaneous USB options ***
  [ ] Enable USB persist by default
  [ ] Dynamic USB minor allocation
  [*] OTG support
  [ ] Rely on OTG and EH Targeted Peripherals List
  [ ] Disable external hubs
  < > USB 2.0 OTG FSM implementation
  < > USB ULPI PHY interface support
  < * > USB Monitor
  < > Support WUSB Cable Based Association (CBA)
      *** USB Host Controller Drivers ***
  < > Cypress C67x00 HCD support
  < * > xHCI HCD (USB 3.0) support
      Generic xHCI driver for a platform device
  < * > EHCI HCD (USB 2.0) support
      [*] Root Hub Transaction Translators
      [*] Improved Transaction Translator scheduling
  < * > Generic EHCI driver for a platform device
  < > OXU210HP HCD support
  < > ISP116X HCD support
  < > ISP1362 HCD support
  < > FOTG210 HCD support
  < > MAX3421 HCD (USB-over-SPI) support
  < * > OHCI HCD (USB 1.1) support
      Generic OHCI driver for a platform device
  < > SL811HS HCD support
  < > R8A66597 HCD support
  [ ] HCD test mode support
      *** USB Device Class drivers ***
  < * > USB Modem (CDC ACM) support
  < * > USB Printer support
  - * - USB Wireless Device Management support
  < > USB Test and Measurement Class support
      *** NOTE: USB_STORAGE depends on SCSI but BLK_DEV_SD may ***
      *** also be needed; see USB_STORAGE Help for more info ***
  < * > USB Mass Storage support

```

```

Device Drivers > USB support
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable

Support for Rio Karma music player
SAT emulation on Cypress USB/ATA Bridge with ATACB
USB ENE card reader support
  < * > USB Attached SCSI
      *** USB Imaging devices ***
  < > USB Mustek MDC800 Digital Camera support
  < > Microtek X6USB scanner support
  < > USB/IP support
  < > Inventra Highspeed Dual Role Controller (TI, ADI, AW, ...)
  < > DesignWare USB3 DRD Core Support
  < * > DesignWare USB2 DRD Core Support
      DWC2 Mode Selection (Dual Role mode) ---->
  [ ] Enable Debugging Messages
  [ ] Enable Missed SOF Tracking
  < > Chipidea Highspeed Dual Role Controller
  < > NXP ISP 1760/1761 support

```



```

&u2phy {
    status = "okay";

    u2phy_host: host-port {
        status = "okay";
    };

    u2phy_otg: otg-port {
        status = "okay";
    };
};

&usb20_otg {
    status = "okay";
};

&usb_host0_ehci {
    vcc-supply = <&reg_gpio_usb>;
    status = "okay";
};

```

```

gpio-otg-auto {
    otg_auto_gpio: otg-gpio {
        rockchip,pins = <0 RK_PA2 RK_FUNC_GPIO &pcfg_pull_none>,
                        <1 RK_PB0 RK_FUNC_GPIO &pcfg_pull_none>;
    };
};

```

```

otg_auto {
    compatible = "ro,otg-auto";
    pinctrl-names = "default";
    pinctrl-0 = <&otg_auto_gpio>;
    gpio-otg-id = <&gpio0 RK_PA2 GPIO_ACTIVE_LOW>;
    gpio-pwr-check = <&gpio1 RK_PB0 GPIO_ACTIVE_LOW>;
    status = "okay";
};

```

上图 otg-auto驱动作用是提供一个节点用于USB HOST ----USB OTG切换。

## (2) USB HOST

由原理图可知，USB HOST 由USB HUB转出5个USB接口，相应的有5个GPIO进行控电。

USB-CTRL-GPIO列表：

| GPIO号      | USB 号 | 位号 |
|------------|-------|----|
| GPIO3_D2_d | USB2  | J5 |
| GPIO3_D3_d | USB3  | J3 |
| GPIO3_A3_d | USB4  | J4 |
| GPIO3_A0_d | USB5  | J6 |







```

reg_gpio_usb: regulator-gpio-usb {
    compatible = "regulator-gpio";
    pinctrl-names = "default";
    pinctrl-0 = <&usb_power_gpio>;
    regulator-min-microvolt = <5000000>;
    regulator-max-microvolt = <5000000>;
    regulator-name = "gpio_usb";
    regulator-type = "voltage";
    enable-gpio = <&gpio3 RK_PA4 GPIO_ACTIVE_HIGH>;
    enable-active-high;
    regulator-always-on;
    regulator-boot-on;
    states = <5000000 0x1 5000000 0x0>;
    status = "okay";
};

```

```

usb_power {
    usb_power_gpio: usb-power-gpio {
        rockchip,pins = <3 RK_PA4 RK_FUNC_GPIO &pcfg_pull_none>;
    };
};

```

```

leds: gpio-leds {
    compatible = "gpio-leds";
    pinctrl-names = "default";
    pinctrl-0 = <&leds_gpio>;
    status = "okay";

    heartbeat {
        gpios = <&gpio3 RK_PD1 GPIO_ACTIVE_LOW>;
        linux,default-trigger = "heartbeat";
        label = "Heartbeat";
    };

    // usb1_pwr {
    //     gpios = <&gpio2 RK_PC3 GPIO_ACTIVE_HIGH>;
    //     default-state="on";
    // };

    usb2_pwr {
        gpios = <&gpio3 RK_PD2 GPIO_ACTIVE_HIGH>;
        default-state="on";
    };

    usb3_pwr {
        gpios = <&gpio3 RK_PD3 GPIO_ACTIVE_HIGH>;
        default-state="on";
    };

    usb4_pwr {
        gpios = <&gpio3 RK_PA3 GPIO_ACTIVE_HIGH>;
        default-state="on";
    };

    usb5_pwr {
        gpios = <&gpio3 RK_PA0 GPIO_ACTIVE_HIGH>;
        default-state="on";
    };
};

```

```

gpio-leds {
    leds_gpio: leds-gpio {
        rockchip,pins = <3 RK_PD1 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PD2 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PD3 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PA3 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PA0 RK_FUNC_GPIO &pcfg_pull_none>,
        <0 RK_PA0 RK_FUNC_GPIO &pcfg_pull_none>,
        <0 RK_PC0 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PA5 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PA6 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PA7 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PB0 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PB5 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PC0 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PC1 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PC2 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PC3 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PC4 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PC5 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PC1 RK_FUNC_GPIO &pcfg_pull_none>,
        <3 RK_PB2 RK_FUNC_GPIO &pcfg_pull_none>,
        <0 RK_PA1 RK_FUNC_GPIO &pcfg_pull_none>;
    };
};

```

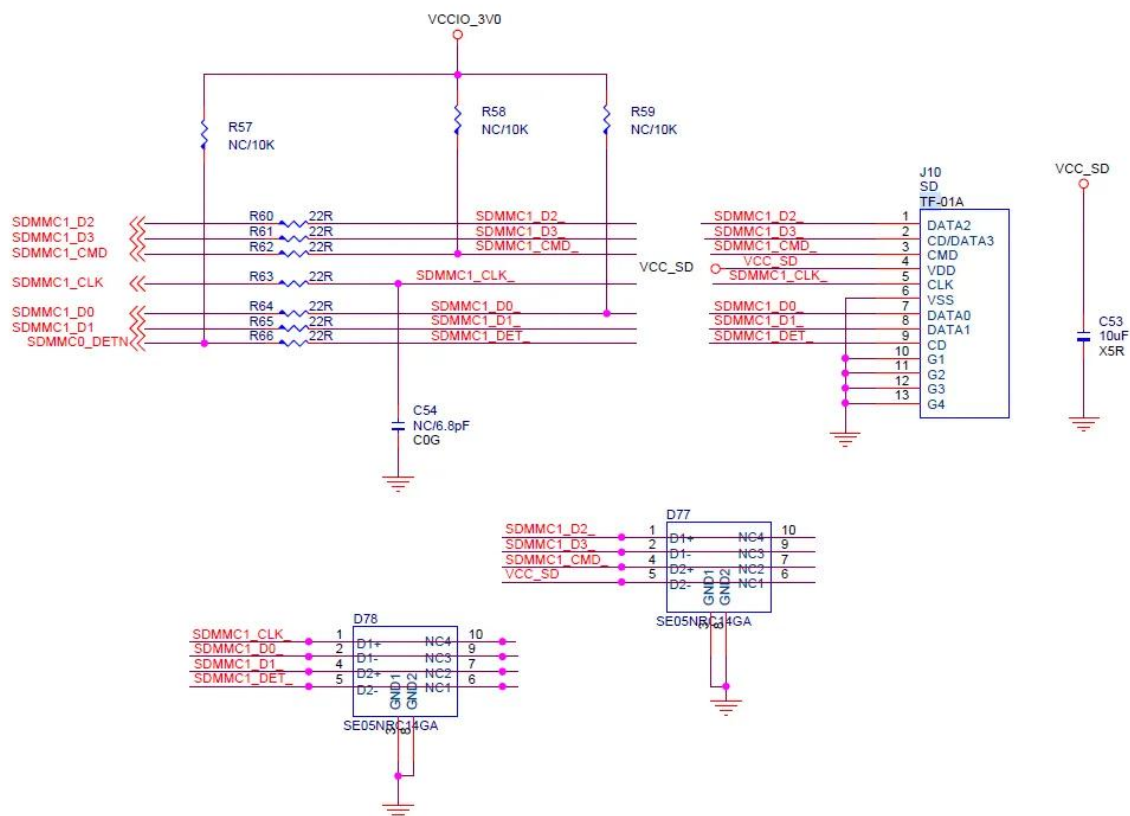
到这里USB 就配置完成了。

注意事项：

- 1、USB不识别，可以检查是否有5V供电。
- 2、5V没供电，可以dmesg | grep gpio ， 查看是否有GPIO冲突。
- 3、USB OTG切换无法生效，检查相应的控制IO配置跟原理图是否匹配。
- 4、供电电压是否匹配。

## 4、TF Card

## TF Card



由原理图可知，我们需要单独对TF卡的检测脚进行配置：

| 引脚名称         | GPIO       | 功能     |
|--------------|------------|--------|
| SDMMC0_DET_N | GPIO0_A3_u | TF卡检测脚 |
|              |            |        |

供电分别是VCC\_SD/VCCIO\_SD。

kernel 配置：

```

> Device Drivers > MMC/SD/SDIO card support
x Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted
x are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to
x for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable
x
x l----- MMC/SD/SDIO card support
x x      [ ] MMC debugging
x x      [ ] MMC embedded SDIO device support (EXPERIMENTAL)
x x      [ ] Enable paranoid SD card initialization (EXPERIMENTAL)
x x      *** MMC/SD/SDIO Card Drivers ***
x x      <*> MMC block device driver
x x      (32)   Number of minors per block device
x x      [*]   Use bounce buffer for simple hosts
x x      < > SDIO UART/GPS class support
x x      <*> MMC host test driver
x x      [ ]   Turn on maximum speed control per block device
x x      *** MMC/SD/SDIO Host Controller Drivers ***
x x      < > ARM AMBA Multimedia Card Interface support
x x      < > Secure Digital Host Controller Interface support
x x      < > MMC/SD/SDIO over SPI
x x      <*> Synopsys DesignWare Memory Card Interface
x x      -*-* Synopsys Designware MCI Support as platform device
x x      < > Exynos specific extensions for Synopsys DW Memory Card Interface
x x      < > K3 specific extensions for Synopsys DW Memory Card Interface
x x      <*> Rockchip specific extensions for Synopsys DW Memory Card Interface
x x      < > VUB300 USB to SDIO/SD/MMC Host Controller support
x x      < > USB SD Host Controller (USHC) support
x x      < > Renesas USDHI6ROL0 SD/SDIO Host Controller support
x x      < > MediaTek SD/MMC Card Interface support
x x

```

kernel 需要开启MMC/SD/SDIO card support 。

DTS 配置：

```

1
2  &sdmmc {
3      bus-width = <4>;
4      cap-mmc-highspeed;
5      cap-sd-highspeed;
6      supports-sd;
7      card-detect-delay = <800>;
8      ignore-pm-notify;
9      cd-gpios = <&gpio0 RK_PA3 GPIO_ACTIVE_LOW>;
10 //      cd-inverted;
11      sd-uhs-sdr12;
12      sd-uhs-sdr25;
13      sd-uhs-sdr50;
14      sd-uhs-sdr104;
15      vqmmc-supply = <&vccio_sd>;
16      vmmc-supply = <&vcc_sd>;
17      status = "okay";
18 };
19 &i2c0 {
20     status = "okay";
21     regulators {
22         vccio_sd: LD0_REG5 {
23             regulator-always-on;
24             regulator-boot-on;
25             regulator-min-microvolt = <3000000>;
26             regulator-max-microvolt = <3000000>;
27
28             regulator-name = "vccio_sd";
29             regulator-state-mem {
30                 regulator-on-in-suspend;
31                 regulator-suspend-microvolt = <3000000>;
32             };
33         };
34
35         vcc_sd: LD0_REG6 {
36             regulator-min-microvolt = <3000000>;
37             regulator-max-microvolt = <3000000>;
38
39             regulator-name = "vcc_sd";
40             regulator-state-mem {
41                 regulator-on-in-suspend;
42                 regulator-suspend-microvolt = <3000000>;
43
44     };

```

```

45                                     };
46                                     };
47                                     };
48 };
49

```

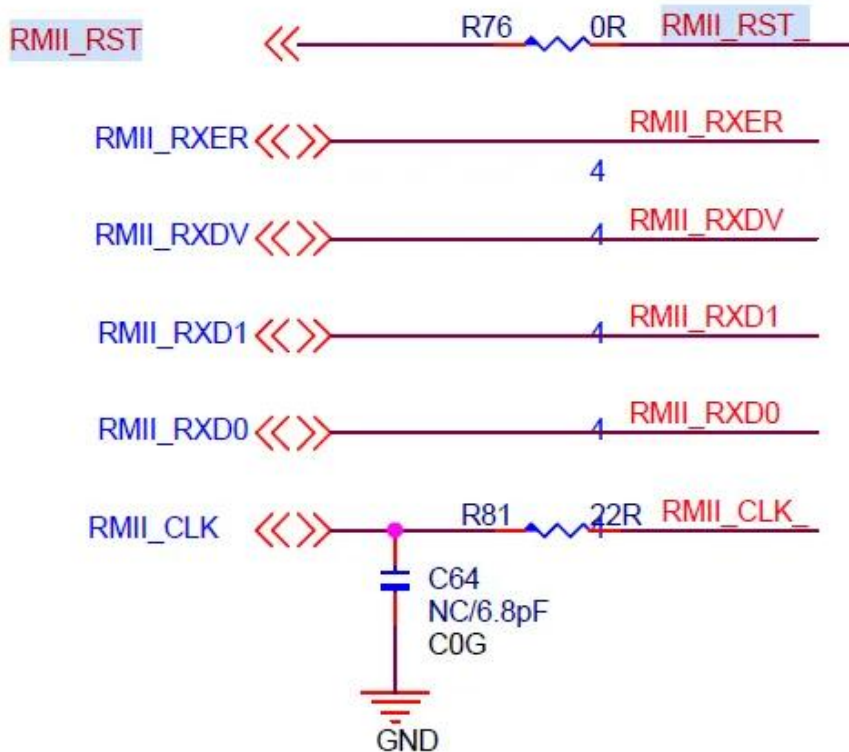
详细配置参考：

[Rockchip\\_Developer\\_Guide\\_SDMMC\\_SDIO\\_eMMC\\_CN.pdf](#)

注意事项：

- 1、注意SDMMC\_DET脚，由dts配置，无需硬件单独上拉，否则会导致检测卡报错。
- 2、供电电压是否匹配。

## 5、以太网



**CLOSE TO PHY**



由原理图可知，我们需要单独对以太网的复位脚进行配置：

| 引脚名称     | GPIO       | 功能 |
|----------|------------|----|
| RMII_RST | GPIO2_B5_d | 复位 |
|          |            |    |

kernel 配置：

开启以太网phy：

```

> Device Drivers > Network device support > PHY Device support and infrastructure
|----- PHY Device support and infrastructure -----|
x Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters x
x are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> x
x for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable x
x
x l-----|
x x          --- PHY Device support and infrastructure x x
x x          *** MII PHY device drivers *** x x
x x          < > Drivers for the Aquantia PHYs x x
x x          < > Drivers for Atheros AT803X PHYs x x
x x          < > Drivers for the AMD PHYs x x
x x          < > Drivers for Marvell PHYs x x
x x          < > Drivers for Davicom PHYs x x
x x          < > Drivers for Quality Semiconductor PHYs x x
x x          < > Drivers for the Intel LXT PHYs x x
x x          < > Drivers for the Cicada PHYs x x
x x          < > Drivers for the Vitesse PHYs x x
x x          < > Drivers for the Teranetics PHYs x x
x x          < > Driver for Rockchip Ethernet PHYs x x
x x          < > Drivers for SMSC PHYs x x

```

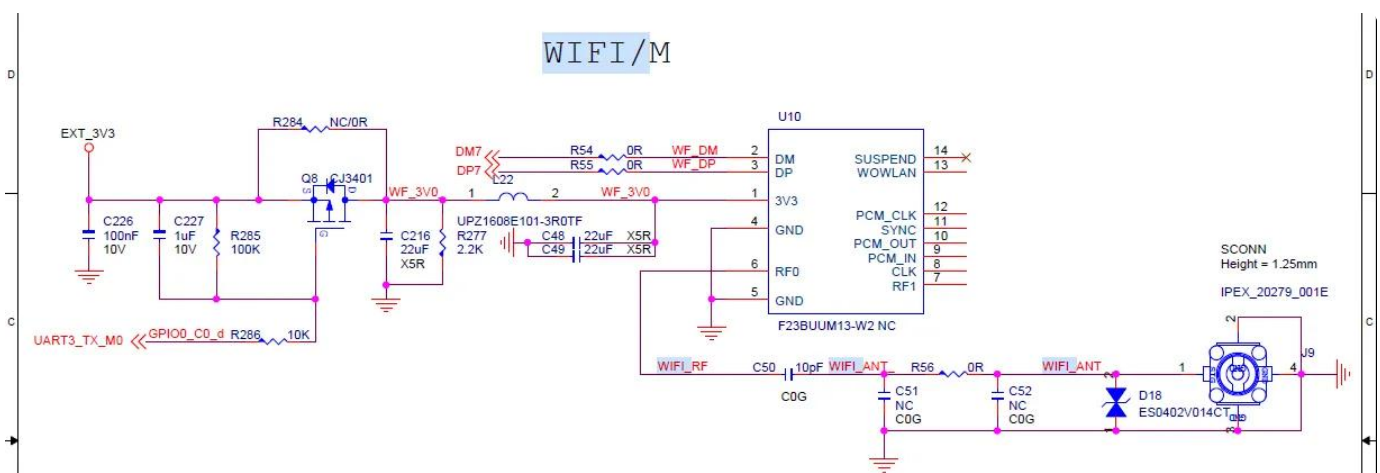
DTS 配置：

```

1
2 &gmac {
3     phy-supply = <&vcc_phy>;
4     assigned-clocks = <&cru SCLK_GMAC>;
5     assigned-clock-parents = <&gmac_clkin>;
6     clock_in_out = "input";
7     //clock_in_out = "output";
8     snps,reset-gpio = <&gpio2 RK_PB5 GPIO_ACTIVE_HIGH>;
9     snps,reset-active-low;
10    snps,reset-delays-us = <0 50000 50000>;
11    tx_delay = <0x30>;
12    rx_delay = <0x10>;
13    status = "okay";
14 };
15
16 /{
17     vcc_phy: vcc-phy-regulator {
18         compatible = "regulator-fixed";
19         regulator-name = "vcc_phy";
20         regulator-always-on;
21         regulator-boot-on;
22     };
23
24
25 };

```

## 6、WIFI/BT



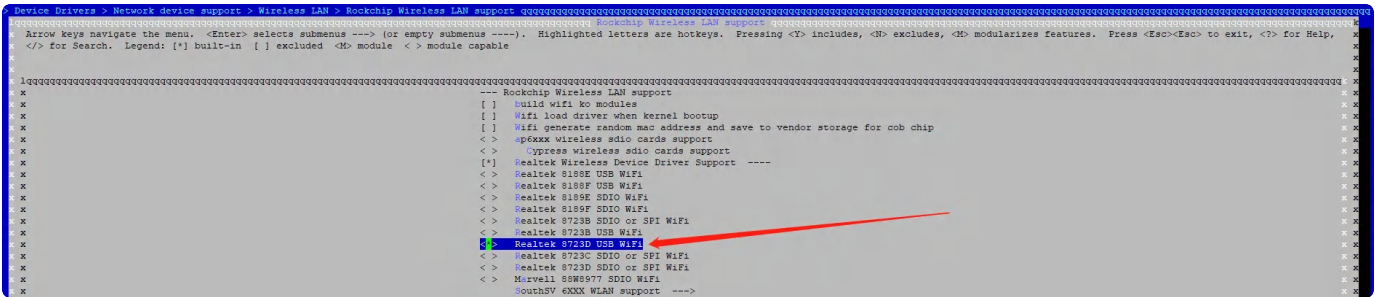
由原理图可知，有使用GPIO控制WIFI/BT模块的供电，并且WIFI蓝牙芯片使用的是USB接口，由于前面已配好，就不需要单独配置了。

| 引脚名称        | GPIO       | 功能   |
|-------------|------------|------|
| UART3_TX_M0 | GPIO0_C0_d | 模块复位 |
|             |            |      |

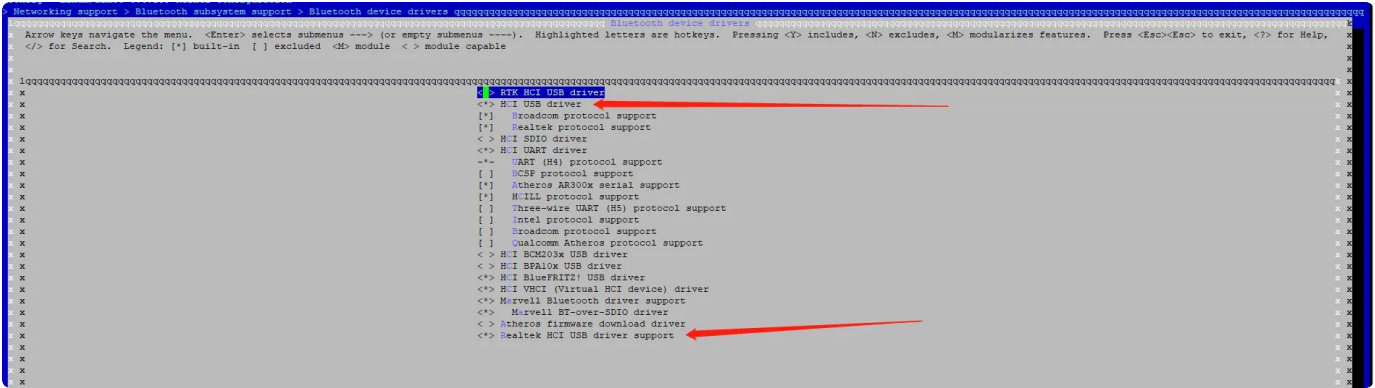
kernel 配置：

开发板用的RTL8723DU模块，相应的内核使能该驱动。

使能wifi驱动：



使能蓝牙驱动：



DTS 配置：

用LED驱动控制该IO：

```

1  &pinctrl {
2      gpio-leds {
3          leds_gpio: leds-gpio {
4              rockchip,pins = <0 RK_PA0 RK_FUNC_GPIO &pcfg_pull_
              none>;
5          };
6
7      };
8  };
9
10 leds: gpio-leds {
11     compatible = "gpio-leds";
12     pinctrl-names = "default";
13     pinctrl-0 =<&leds_gpio>;
14     status = "okay";
15     wlan_pwr {
16         gpios = <&gpio0 RK_PC0 GPIO_ACTIVE_LOW>;
17         default-state="on";
18     };
19 };
20
21 //注意该IO用的UART3_M0, 配置串口IO注意不要复用了。
22 &uart3 {
23     pinctrl-names = "default";
24     pinctrl-0 = <&uart3m1_xfer>;
25     status = "okay";
26 };

```

由于蓝牙模块需要相应的固件才能正常使用，需要将蓝牙固件拷贝到文件系统的对应文件夹中。

默认放在/lib/firmware目录下：

```

[root@px30_64:/]# ls lib/firmware/
rtl8723du_config rtl8723du_fw rtl_bt

```

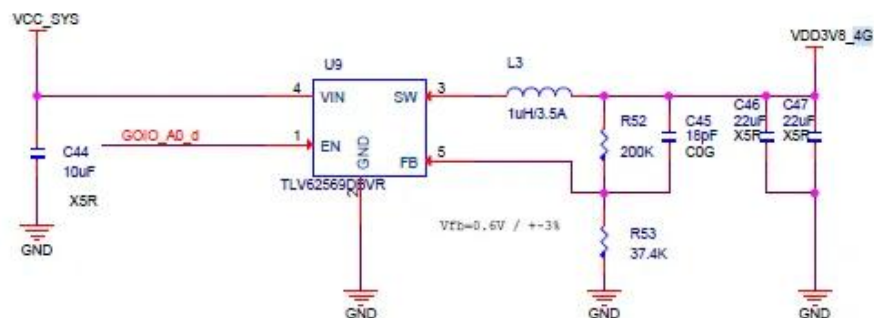
详细配置参考：

 [Rockchip\\_Developer\\_Guide\\_Linux\\_WIFI\\_BT\\_CN.pdf](#)

## 7、4G



## POWER VCC3V8



由原理图可知，GPIO0\_A0\_d作为USB5电源控制。

| 引脚名称     | GPIO       | 功能        |
|----------|------------|-----------|
| REF_CLKO | GPIO0_A0_d | power ctl |
|          |            |           |

## DTS配置:

Plain Text |

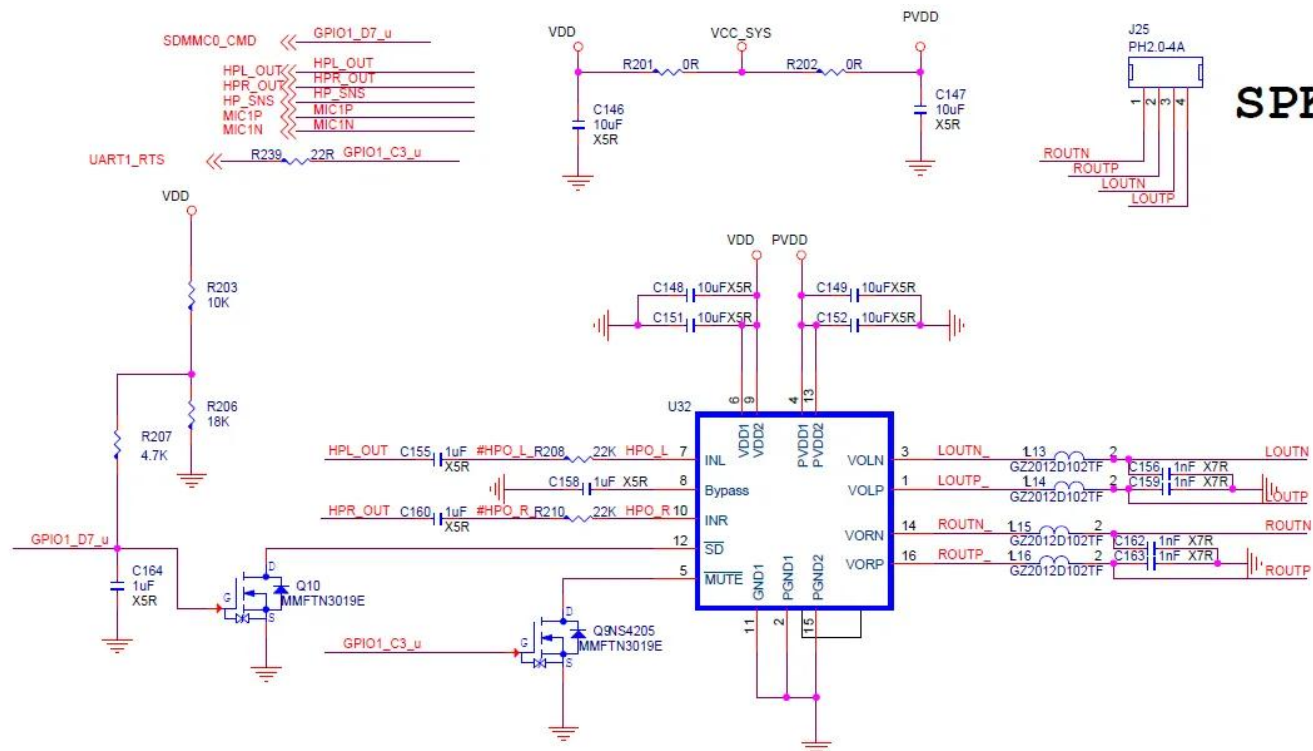
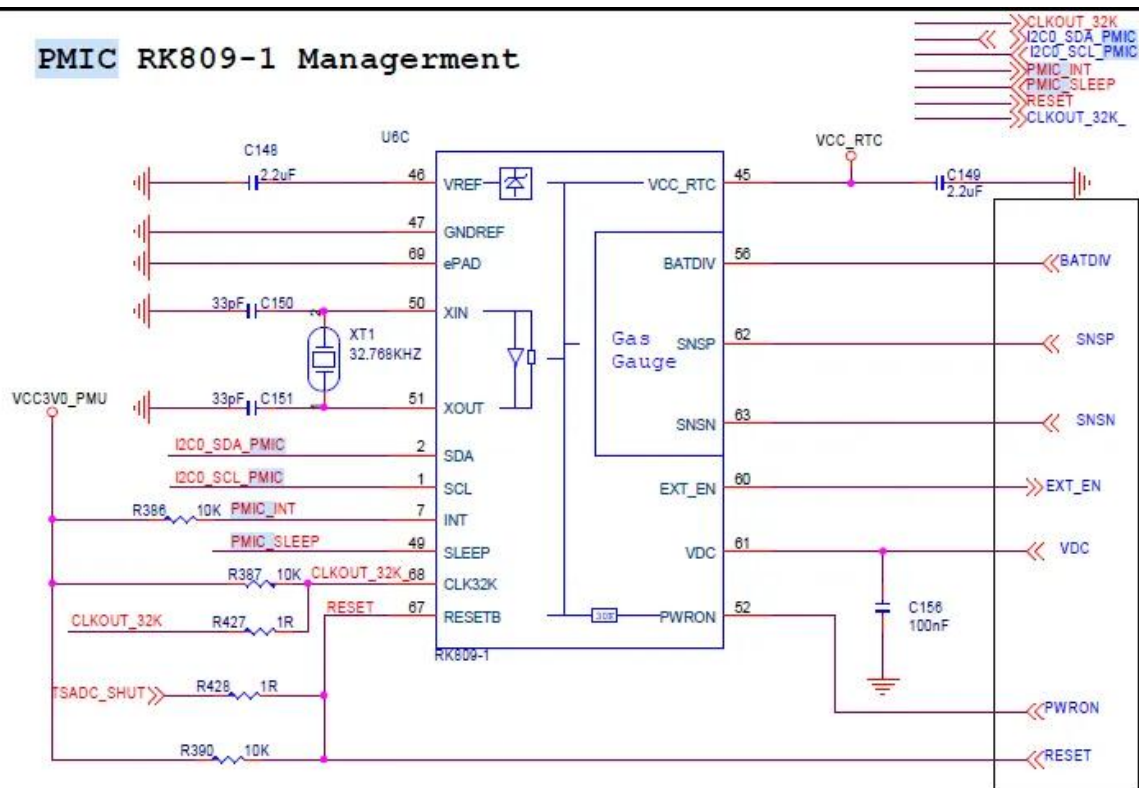
```

1  leds: gpio-leds {
2      compatible = "gpio-leds";
3      pinctrl-names = "default";
4      pinctrl-0 = <&leds_gpio>;
5      status = "okay";
6      usb5_pwr {
7          gpios = <&gpio3 RK_PA0 GPIO_ACTIVE_HIGH>;
8          default-state="on";
9      };
10
11 };
12 &pinctrl {
13     gpio-leds {
14         leds_gpio: leds-gpio {
15             rockchip,pins = <0 RK_PA0 RK_FUNC_GPIO &pcfg_pull_none>;
16         };
17     };

```

## 8、扬声器

## PMIC RK809-1 Managerment





由原理图可知，SPEAKER是用的RK809的，所以需要优先配置RK809。

| 引脚名称          | 模块         |
|---------------|------------|
| I2C0_SDA_PMIC | RK809 PMIC |
| I2C0_SCL_PMIC |            |

这里可以知道分别有两个IO控制：

| 引脚名称       | GPIO       | 功能           |
|------------|------------|--------------|
| UART1_RTS  | GPIO1_C3_u | Speaker mute |
| SDMMC0_CMD | GPIO1_D7_u | Speaker ctrl |

kernel 配置：

配置RK809:

```

Symbol: I2C [=y]
Type : tristate
Prompt: I2C support
Location:
  -> Device Drivers
(1)  -> I2C support
Defined at drivers/i2c/Kconfig:7
Selects: RT_MUTEXES [=y]
Selected by: CAN_PEAK_PCIEC [=n] && NET [=y] && CAN [=y] && CAN_DEV [=y] && CAN_SJA1000 [=n] && CAN_PEAK_P

```



## DTS 配置:

RK809接的I2C0, 所以需要使能I2C0。

```
1  &i2c0 {
2      status = "okay";
3
4      rk809: pmic@20 {
5          compatible = "rockchip,rk809";
6          reg = <0x20>;
7          interrupt-parent = <&gpio0>;
8          interrupts = <7 IRQ_TYPE_LEVEL_LOW>;
9          pinctrl-names = "default", "pmic-sleep",
10             "pmic-power-off", "pmic-reset";
11          pinctrl-0 = <&pmic_int>;
12          pinctrl-1 = <&soc_slppin_slp>, <&rk817_slppin_slp>;
13          pinctrl-2 = <&soc_slppin_gpio>, <&rk817_slppin_pwrn>;
14          pinctrl-3 = <&soc_slppin_rst>, <&rk817_slppin_rst>;
15          rockchip,system-power-controller;
16          wakeup-source;
17          #clock-cells = <1>;
18          clock-output-names = "rk808-clkout1", "rk808-clkout2";
19          //fb-inner-reg-idxs = <2>;
20          /* 1: rst regs (default in codes), 0: rst the pmic */
21          pmic-reset-func = <1>;
22
23          vcc1-supply = <&vcc5v0_sys>;
24          vcc2-supply = <&vcc5v0_sys>;
25          vcc3-supply = <&vcc3v3_sys>;
26          vcc4-supply = <&vcc5v0_sys>;
27          vcc5-supply = <&vcc3v3_sys>;
28          vcc6-supply = <&vcc3v3_sys>;
29          vcc7-supply = <&vcc3v3_sys>;
30          vcc8-supply = <&vcc3v3_sys>;
```

```

1  &pinctrl {
2      pmic {
3          pmic_int: pmic_int {
4              rockchip,pins =
5                  <0 RK_PA7 RK_FUNC_GPIO &pcfg_pull_up>;
6          };
7
8          soc_slppin_gpio: soc_slppin_gpio {
9              rockchip,pins =
10                 <0 RK_PA4 RK_FUNC_GPIO &pcfg_output_low>;
11          };
12
13          soc_slppin_slp: soc_slppin_slp {
14              rockchip,pins =
15                 <0 RK_PA4 RK_FUNC_1 &pcfg_pull_none>;
16          };
17
18          soc_slppin_rst: soc_slppin_rst {
19              rockchip,pins =
20                 <0 RK_PA4 RK_FUNC_2 &pcfg_pull_none>;
21          };
22      };
23  };
24

```

扬声器配置:

```

1  &pinctrl {
2      head_ctl {
3          hp_ctl_gpio: hp-ctl-gpio {
4              rockchip,pins =
5                  <1 RK_PD7 RK_FUNC_GPIO &pcfg_pull_none>,
6                  <1 RK_PC3 RK_FUNC_GPIO &pcfg_pull_none>;
7          };
8      };
9  };

```

```

rk809_codec: codec {
    #sound-dai-cells = <1>;
    compatible = "rockchip,rk809-codec", "rockchip,rk817-codec";
    clocks = <&cru SCLK_I2S1_OUT>;
    clock-names = "mclk";
    pinctrl-names = "default";
    pinctrl-0 = <&i2s1_2ch_mclk>;
    pinctrl-1 = <&hp_ctl_gpio>;
    hp-volume = <20>;//3-255,3:max
    spk-volume = <20>;//3-255,3:max
    capture-volume = <20>;
    #if 1 //rk817 mic
    mic-in-differential;
    /// #else //pdm
    pdmdata-out-enable;
    adc-for-loopback;
    #endif

    spk-ctl-gpios = <&gpio1 RK_PD7 GPIO_ACTIVE_LOW>; //GPIO1_D4
    spk-mute-gpios = <&gpio1 RK_PC3 GPIO_ACTIVE_LOW>;//Speaker mute GPIO0_A2
    spk-mute-delay-ms = <30>;

    status = "okay";
};
};
};

```

注意：

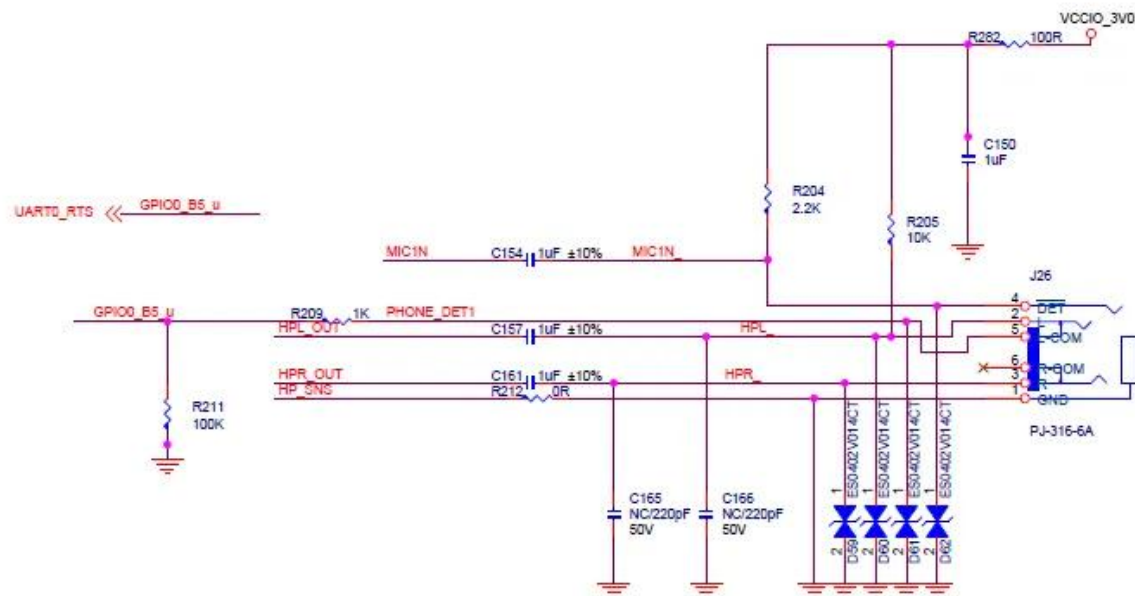
1、配置需注意UART1\_RTS不要开启，防止IO复用。

详情参考：

[📎Rockchip\\_RK809\\_Developer\\_Guide\\_CN.pdf](#)

## 9、耳机

## Headset



由原理图可知，耳机接口同样使用的RK809。因为在扬声器章节我们配置好了，所以不需要单独配置。图中可以指导GPIO0\_B5作为耳机的检测脚。

| 引脚名称      | GPIO       | 功能        |
|-----------|------------|-----------|
| UART0_RTS | GPIO0_B5_u | PHONE_DET |
|           |            |           |

kernel 配置:

[illegible]

**DTS 配置:**



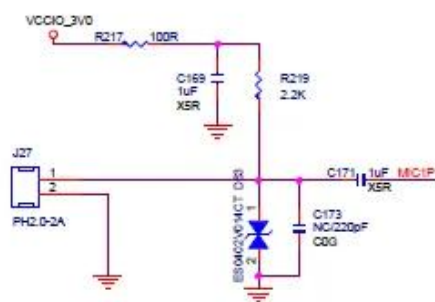
```

1  rk_headset: rk-headset {
2      status = "okay";
3      compatible = "rockchip_headset";
4      headset_gpio = <&gpio0 RK_PB5 GPIO_ACTIVE_HIGH>;
5      pinctrl-names = "default";
6      pinctrl-0 = <&hp_det>;
7  };
8
9
10 &pinctrl {
11     headphone {
12         hp_det: hp-det {
13             rockchip,pins = <0 RK_PB5 RK_FUNC_GPIO &pcfg_pull_none>;
14         };
15     };
16 };

```

## 10、麦克风

### MIC



由原理图可知，我们直接用的RK809的不需要单独配置。

DTS 配置：

```

rk809_codec: codec {
    #sound-dai-cells = <1>;
    compatible = "rockchip,rk809-codec", "rockchip,rk817-codec";
    clocks = <&cru SCLK_I2S1_OUT>;
    clock-names = "mclk";
    pinctrl-names = "default";
    pinctrl-0 = <&i2s1_2ch_mclk>;
    pinctrl-1 = <&hp_ctl_gpio>;
    hp-volume = <20>;//3-255,3:max
    spk-volume = <20>;//3-255,3:max
    capture-volume = <20>;
    #if 1 //rk817 mic
mic-in-differential;
// #else //pdm
pdmdata-out-enable;
adc-for-loopback;
#endif

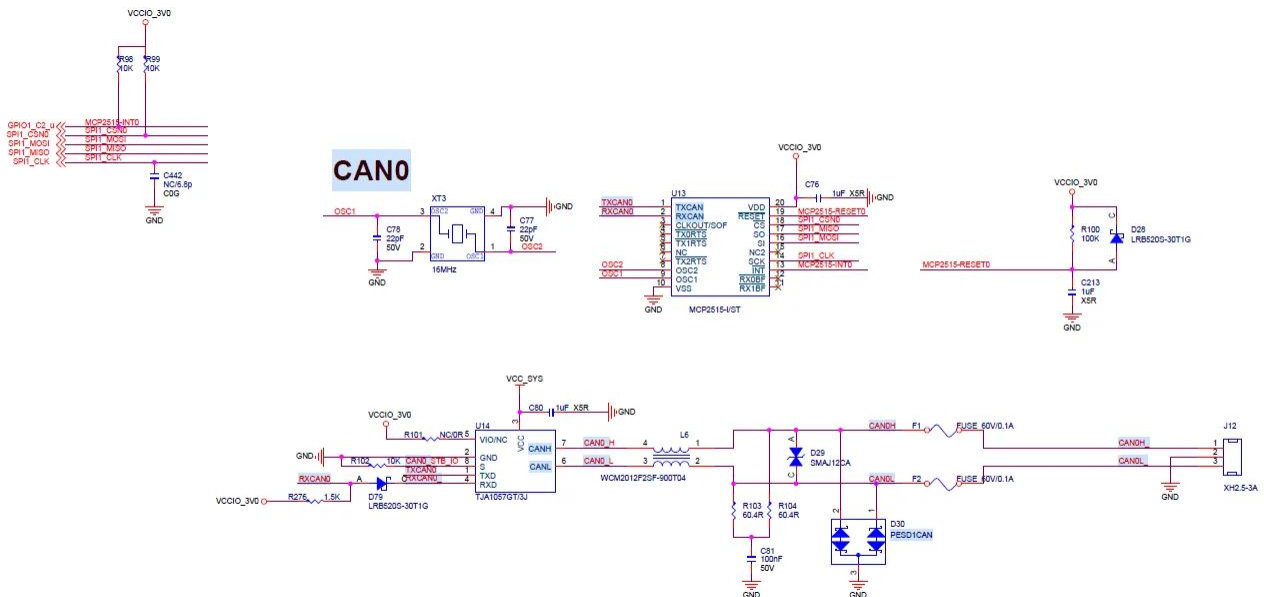
    spk-ctl-gpios = <&gpio1 RK_PD7 GPIO_ACTIVE_LOW>; //GPIO1_D4
    spk-mute-gpios = <&gpio1 RK_PC3 GPIO_ACTIVE_LOW>;//Speaker mute GPIO0_A2
    spk-mute-delay-ms = <30>;

    status = "okay";

};
};

```

## 11、CAN



由原理图可知，CAN模块（MCP2515）只用的SPI转的，我们只需要配置该模块的启动以及SPI即可。

引脚名称

功能

|            |      |
|------------|------|
| GPIO1_C2_u | 中断检测 |
| SPI1_CSN0  | SPI1 |
| SPI1_MOSI  |      |
| SPI1_MISO  |      |
| SPI1_CLK   |      |

kernel 配置:

使能SPI:

```
> Device Drivers > SPI support
--- SPI support
[ ] Debug support for SPI drivers
*** SPI Master Controller Drivers ***
< > Altera SPI Controller
< * > Utilities for Bitbanging SPI masters
< > Cadence SPI controller
< > GPIO-based bitbanging SPI Master
< > Freescale SPI controller and Aeroflex Gaisler GRLIB SPI controller
< > OpenCores tiny SPI
< > ARM AMBA PL022 SSP controller
< * > Rockchip SPI controller driver
< > NXP SC18IS602/602B/603 I2C to SPI bridge
< > Analog Devices AD-FMCOMMS1-EBZ SPI-I2C-bridge driver
< > Xilinx SPI controller common module
< > Xilinx ZynqMP GQSPI controller
< > DesignWare SPI controller core support
*** SPI Protocol Masters ***
< * > User mode SPI device driver support
< > Infineon TLE62X0 (for power switching)
```

使能MCP2515驱动:

```
> Networking support > CAN bus subsystem support > CAN Device Drivers > CAN SPI interfaces
--- CAN SPI interfaces
[ ] Microchip MCP251x and MCP25625 SPI CAN controllers
```

DTS 配置:

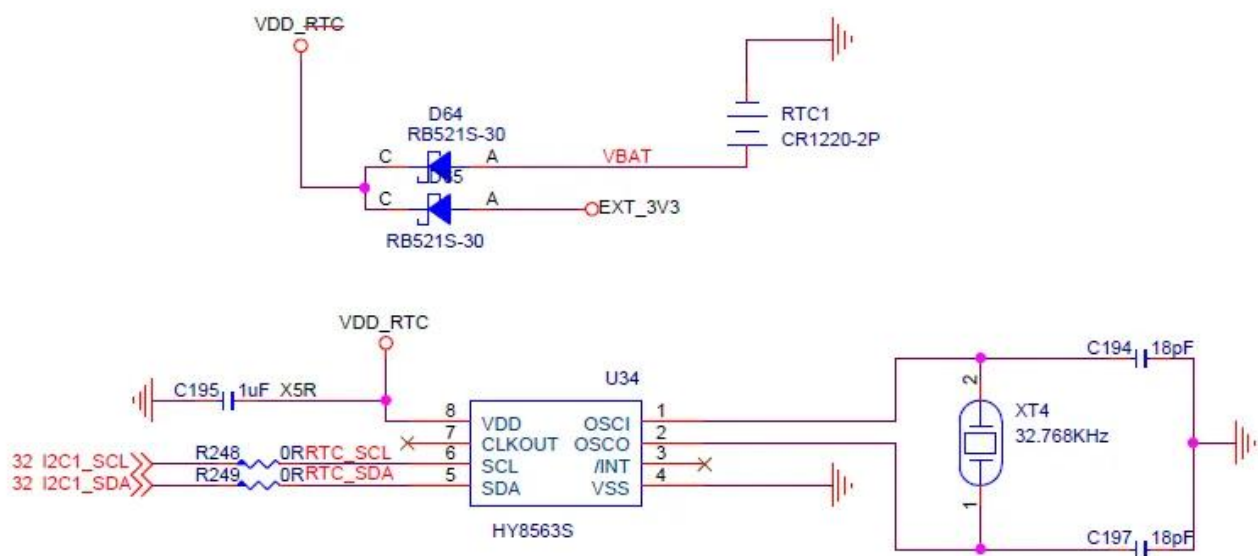
```

1  &spi1 {
2      status = "okay";
3      max-freq = <50000000>;
4      pinctrl-0 = <&spi1_clk &spi1_csn0 &spi1_miso &spi1_mosi>;
5
6      can@0 {
7          compatible = "microchip,mcp2515";
8          reg = <0>;
9          clocks = <&mcp2515_clock>;
10         interrupt-parent = <&gpio1>;
11         interrupts = <RK_PC2 IRQ_TYPE_LEVEL_LOW>;
12         spi-max-frequency = <20000000>;
13         vdd-supply = <&vcc3v0_pmu>;
14         xceiver-supply = <&vcc3v0_pmu>;
15         status = "okay";
16     };
17 };

```

## 12、RTC

### RTC 8563T



由原理图可知，使用的外部RTC 8563T，通过I2C1连接。

| 引脚名称     | 功能  |
|----------|-----|
| I2C1_SCL | I2C |
| I2C_SDA  |     |

kernel 配置：

使能RTC驱动：

```
> Device Drivers > Real Time Clock
x Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenu ----). Highlighted letters x
x are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> x
x for Help, </> for Search. Legend: [*] built-in [ ] excluded <M> module <> module capable
x
x l----- Real Time Clock
x x
x x [*] Set system time from RTC on startup and resume
x x (rtcl) RTC used to set the system time
x x [*] Set the RTC time based on NTP synchronization
x x (rtcl) RTC used to synchronize NTP adjustment
x x [ ] RTC debug support
x x *** RTC interfaces ***
x x [*] /sys/class/rtc/rtcN (sysfs)
x x [*] /proc/driver/rtc (procfs for rtcN)
x x [*] /dev/rtcN (character devices)
x x [ ] RTC UIE emulation on dev interface
x x <> Test driver/device
x x *** I2C RTC drivers ***
x x <> Abracon AB-RTCMC-32.768kHz-B5ZE-S3
x x <> Abracon ABx80x
x x <> Dallas/Maxim DS1307/37/38/39/40, ST M41T00, EPSON RX-8025
x x <> Dallas/Maxim DS1374
x x <> Dallas/Maxim DS1672
x x <> Dallas/Maxim DS3232
x x <> FAKE RTC
x x <*> Haoyu Microelectronics HYM8563
x x <> Maxim MAX6900
x x <*> Rockchip RK808/RK816 RTC
x x <> Rockchip timer RTC
```

注意这里需要指定实际的硬件时钟 HWCLOCK才会生效。

DTS 配置：

```

1  &i2c1 {
2      status = "okay";
3      clock-frequency = <100000>;
4
5      /* These are relatively safe rise/fall times; TODO: measure */
6      i2c-scl-falling-time-ns = <50>;
7      i2c-scl-rising-time-ns = <300>;
8
9      hym8563@51 {
10         compatible = "haoyu,hym8563";
11         reg = <0x51>;
12         status = "okay";
13     };
14 };

```

## 13、UART

UART配置只需要在DTS中开启即可。

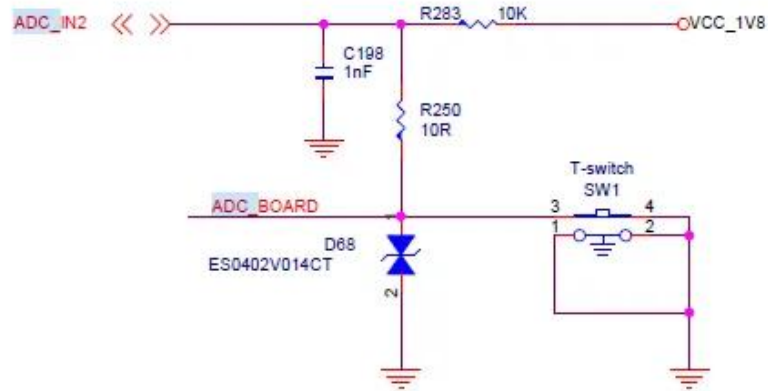
DTS配置：



```
1  &uart0 {
2      pinctrl-names = "default";
3      pinctrl-0 = <&uart0_xfer>;
4      status = "okay";
5  };
6
7  &uart1 {
8      pinctrl-names = "default";
9      pinctrl-0 = <&uart1_xfer>;
10     status = "okay";
11 };
12
13 &uart2 {
14     pinctrl-names = "default";
15     pinctrl-0 = <&uart2m1_xfer>;
16     status = "disabled";
17 };
18
19 &uart3 {
20     pinctrl-names = "default";
21     pinctrl-0 = <&uart3m1_xfer>;
22     status = "okay";
23 };
24
25 &uart4 {
26     pinctrl-names = "default";
27     pinctrl-0 = <&uart4_xfer>;
28     status = "okay";
29 };
30
31 &uart5 {
32     pinctrl-names = "default";
33     pinctrl-0 = <&uart5_xfer>;
34     status = "okay";
35 };
```

## 14、ADC按键

# REC



KERNEL配置:

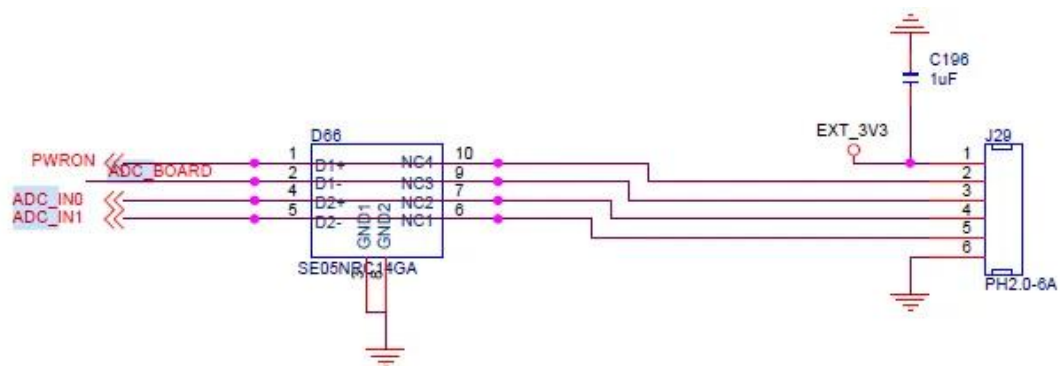
```
.config - Linux/arm64 4.4.189 Kernel Configuration
> Device Drivers > Industrial I/O support > Analog to digital converters
    Analog to digital converters
    Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----).
    Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes
    features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*]
    built-in [ ] excluded <M> module < > module capable

< > Analog Devices AD7265/AD7266 ADC driver
< > Analog Devices AD7291 ADC driver
< > Analog Devices AD7298 ADC driver
< > Analog Devices AD7476 and similar 1-channel ADCs driver
< > Analog Devices AD7791 ADC driver
< > Analog Devices AD7793 and similar ADCs driver
< > Analog Devices AD7887 ADC driver
< > Analog Devices AD7923 and similar ADCs driver
< > Analog Devices AD799x ADC driver
< > Cosmic Circuits 10001 ADC driver
< > GPIO MUXADC driver
< > Holt Integrated Circuits HI-8435 threshold detector
< > Maxim max1027 ADC driver
< > Maxim max1363 ADC driver
< > Microchip Technology MCP3x01/02/04/08
< > Microchip Technology MCP3422/3/4/6/7/8 driver
< > Nuvoton NAU7802 ADC driver
<*> Rockchip SARADC driver
< > Texas Instruments ADC081C021/027
< > Texas Instruments ADC128S052/ADC122S021
< > Freescale vf610 ADC driver
```

DTS配置:

```
1  adc-keys {
2      compatible = "adc-keys";
3      io-channels = <&saradc 2>;
4      io-channel-names = "buttons";
5      poll-interval = <100>;
6      keyup-threshold-microvolt = <1800000>;
7
8      esc-key {
9          linux,code = <KEY_ESC>;
10         label = "esc";
11         press-threshold-microvolt = <1310000>;
12     };
13
14     home-key {
15         linux,code = <KEY_HOME>;
16         label = "home";
17         press-threshold-microvolt = <624000>;
18     };
19
20     menu-key {
21         linux,code = <KEY_MENU>;
22         label = "menu";
23         press-threshold-microvolt = <987000>;
24     };
25
26     vol-down-key {
27         linux,code = <KEY_VOLUMEDOWN>;
28         label = "volume down";
29         press-threshold-microvolt = <300000>;
30     };
31
32     vol-up-key {
33         linux,code = <KEY_VOLUMEUP>;
34         label = "volume up";
35         press-threshold-microvolt = <17000>;
36     };
37 };
38
39 &saradc {
40     status = "okay";
41     vref-supply = <&vcc1v8_soc>;
42 };
43
```

## 15、ADC IN



ADC

由于我们前面配置ADC按键，所以ADC只需要开启saradc即可。

DTS配置：

```
1 &saradc {
2     status = "okay";
3     vref-supply = <&vcc1v8_soc>;
4 };
```