

RK3399 USB DTS 配置说明

发布版本：1.2

作者邮箱：wulf@rock-chips.com

日期：2019-06-25

文档密级：公开资料

概述

本文档提供RK3399 USB DTS的配置方法。RK3399支持两个Type-C USB 3.0(Type-C PHY is a combination of USB 3.0 SuperSpeed PHY and DisplayPort Transmit PHY)和两个USB 2.0 Host。其中，两个Type-C USB 3.0控制器硬件都可以支持OTG(USB Peripheral和USB Host)，并且向下兼容USB2.0/1.1/1.0。此外，Type-C USB 3.0可以根据实际的应用需求，将物理接口简化设计为Type-A USB 3.0/2.0，Micro USB 3.0/2.0等多种接口类型，内核USB驱动已经兼容这几种不同类型的USB接口，只需要根据实际的硬件设计修改对应的板级DTS配置，就可以使能相应的USB接口。

产品版本

芯片名称	内核版本
RK3399	Linux4.4

读者对象 本文档（本指南）主要适用于以下工程师： 软件工程师 技术支持工程师

修订记录

日期	版本	作者	修改说明
2018-03-01	V1.0	吴良峰	初始版本
2019-01-09	V1.1	吴良峰	使用markdownlint修订格式
2019-06-25	V1.2	吴良峰	1. 增加Type-C to Type-A USB 2.0说明 2. 增加VBUS供电说明 3. 更新文档目录名称 4. 参考示例由EVB改为Sapphire Excavator Board 5. 修订一些错误

RK3399 USB DTS 配置说明

- 1 Type-C0/1 USB 3.0 DTS
 - 1.1 Type-C0 /1 USB Controller DTS
 - 1.2 Type-C0 /1 USB PHY DTS
 - 1.2.1 Type-C0 /1 USB 3.0 PHY DTS
 - 1.2.2 Type-C0 /1 USB 2.0 PHY DTS
 - 1.3 Type-C1 USB OTG Mode DTS

- 2 Type-C to Type-A USB 3.0 Host DTS
- 3 Type-C to Micro USB 3.0 OTG Mode DTS
- 4 Type-C to Micro USB 2.0 OTG Mode DTS
- 5 Type-C to Type-A USB 2.0
 - 5.1 Type-C to Type-A USB 2.0 OTG mode DTS
 - 5.2 Type-C to Type-A USB 2.0 Host only mode DTS
 - 5.3 Type-C to Type-A USB 2.0 OTG mode DTS and Support DP 4 Lane
- 6 USB 2.0 Host DTS
 - 6.1 USB 2.0 Host Controller DTS
 - 6.2 USB 2.0 Host PHY DTS
- 7 USB 3.0 force to USB 2.0
- 8 关于USB VBUS供电的说明
- 9 参考文档

1 Type-C0/1 USB 3.0 DTS

Type-C 的接口类型如下图1-1所示。

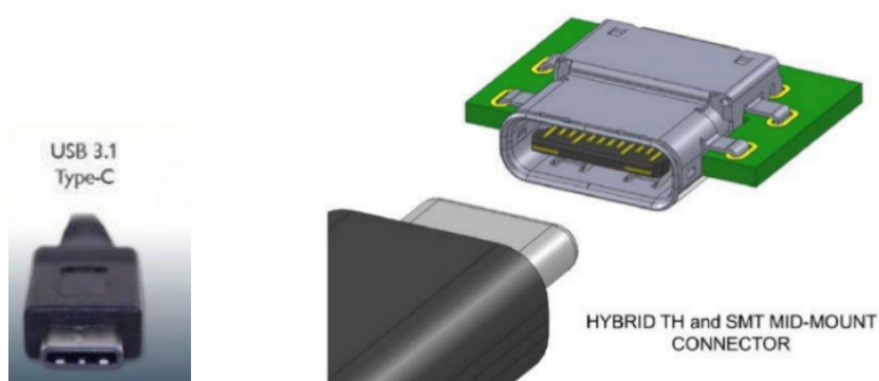


图1-1 Type-C 接口类型示意图

RK3399 SoC内部4个USB控制器与USB PHY的连接如下图1-2所示。

其中，DP是指Display Port控制器，DP与USB 3.0共用Type-C PHY。如图1-2所示，一个完整的Type-C功能，是由Type-C USB 3.0 PHY & DP PHY和USB 2.0 OTG PHY两部分组成的，这两部分PHY在芯片内部的硬件模块是独立的，供电也是独立的。

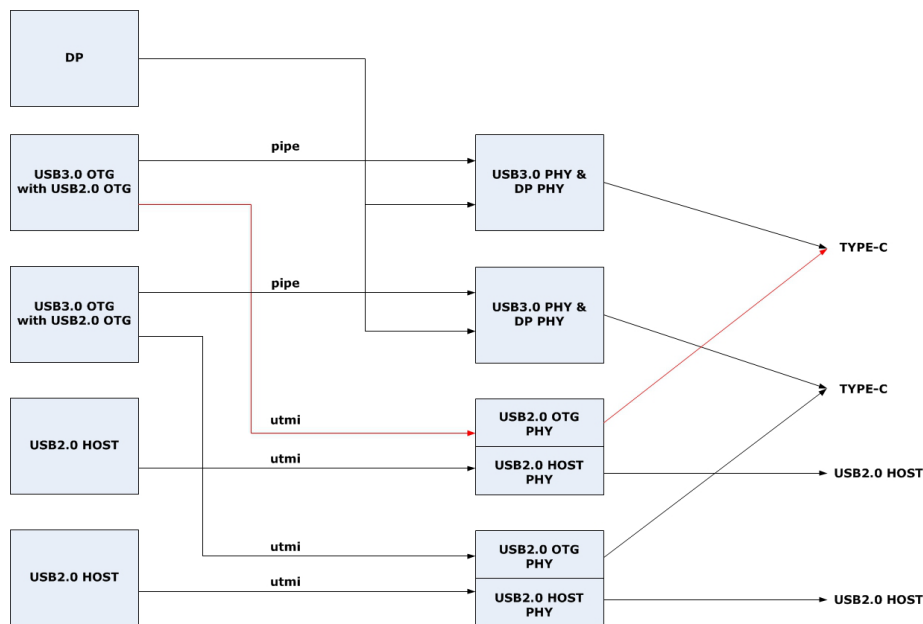


图1-2 RK3399 USB控制器&PHY连接示意图

RK3399 SDK DTS的默认配置，支持Type-C0 USB 3.0 OTG功能，Type-C1 USB 3.0 Host功能。DTS的配置主要包括DWC3控制器、Type-C USB 3.0 PHY以及USB 2.0 PHY。

1.1 Type-C0 /1 USB Controller DTS

Type-C0/1 USB控制器硬件都支持USB 3.0 OTG（USB Peripheral和USB Host）功能，并且向下兼容USB 2.0/1.1/1.0。但由于当前内核的USB 框架只支持一个USB 口作为Peripheral功能，所以SDK默认配置Type-C0支持OTG mode，而Type-C1仅支持Host mode。如果要配置Type-C1支持OTG mode，请参考：

1.3 Type-C1 USB OTG Mode DTS

以RK3399 Sapphire Excavator Board 的 Type-C0/C1 USB 3.0 控制器DTS配置为例：

`arch/arm64/boot/dts/rockchip/rk3399.dtsi`

```

1  usbd3_0: usb0 { /* Type-C0 USB3.0 控制器DTS配置*/
2      compatible = "rockchip,rk3399-dwc3";
3      clocks = <&cru SCLK_USB30TG0_REF>,
4      <&cru SCLK_USB30TG0_SUSPEND>,
5      <&cru ACLK_USB30TG0>, <&cru
6      ACLK_USB3_GRF>;
7      clock-names = "ref_clk",
8      "suspend_clk",
9      "bus_clk", "grf_clk";
10     power-domains = <&power
11     RK3399_PD_USB3>;
12     resets = <&cru SRST_A_USB3_OTG0>;
13     reset-names = "usb3-otg"; /* USB0 控制
14     器的 reset */
15     #address-cells = <2>;
16     #size-cells = <2>;

```

```

12         ranges;
13         status = "disabled";
14         usbdrd_dwc3_0: dwc3@fe800000 {
15             compatible = "snps,dwc3";
16             reg = <0x0 0xfe800000 0x0
17                 0x100000>;
18             interrupts = <GIC_SPI 105
19                 IRQ_TYPE_LEVEL_HIGH 0>;
20             dr_mode = "otg"; /* 支持OTG
21                 mode */
22             phys = <&u2phy0_otg>,
23                 <&tcphy0_usb3>; /* usb3/2 phy属性 */
24             phy-names = "usb2-phy",
25                 "usb3-phy";
26             phy_type = "utmi_wide";
27             snps,dis_enblslpm_quirk;
28             snps,dis-u2-freeclk-exists-
29                 quirk;
30             snps,dis_u2_susphy_quirk;
31             snps,dis-del-phy-power-chg-
32                 quirk;
33             snps,tx-ipgap-linecheck-dis-
34                 quirk;
35             snps,xhci-slow-suspend-quirk;
36             snps,xhci-trb-ent-quirk;
37             snps,usb3-warm-reset-on-
38                 resume-quirk;
39             status = "disabled";
40         };
41     };
42
43     usbdrd3_1: usb1 { /* Type-C1 USB3.0 控制器DTS配置*/
44         compatible = "rockchip,rk3399-dwc3";
45         clocks = <&cru SCLK_USB30TG1_REF>,
46                 <&cru SCLK_USB30TG1_SUSPEND>,
47                 <&cru ACLK_USB30TG1>, <&cru
48                 ACLK_USB3_GRF>;
49         clock-names = "ref_clk",
50                 "suspend_clk",
51                 "bus_clk", "grf_clk";
52         power-domains = <&power
53                 RK3399_PD_USB3>;
54         resets = <&cru SRST_A_USB3_OTG1>;
55         reset-names = "usb3-otg"; /* USB1 控制
56             器的 reset */
57         #address-cells = <2>;
58         #size-cells = <2>;
59         ranges;
60         status = "disabled";
61         usbdrd_dwc3_1: dwc3@fe900000 {
62             compatible = "snps,dwc3";

```

```

49             reg = <0x0 0xfe900000 0x0
           0x100000>;
50             interrupts = <GIC_SPI 110
           IRQ_TYPE_LEVEL_HIGH 0>;
51             dr_mode = "host"; /* 只支持Host
           mode */
52             phys = <&u2phy1_otg>,
           <&tcphy1_usb3>; /* usb3/2 phy属性 */
53             phy-names = "usb2-phy",
           "usb3-phy";
54             phy_type = "utmi_wide";
55             snps,dis_enblslpm_quirk;
56             snps,dis-u2-freeclk-exists-
           quirk;
57             snps,dis_u2_susphy_quirk;
58             snps,dis-del-phy-power-chg-
           quirk;
59             snps,tx-ipgap-linecheck-dis-
           quirk;
60             snps,xhci-slow-suspend-quirk;
61             snps,xhci-trb-ent-quirk;
62             snps,usb3-warm-reset-on-
           resume-quirk;
63             status = "disabled";
64         };
65     };

```

arch/arm64/boot/dts/rockchip/rk3399-sapphire.dtsi

```

1  &usbdrd3_0 {
2      extcon = <&fusb0>; /* 配置extcon属性，用于接收
           fusb302驱动的 UFP/DFP notifier*/
3      status = "okay";
4  };
5
6  &usbdrd3_1 {
7      /* USB1 为Type-A接口，只支持USB Host，不用配置
           extcon属性 */
8      status = "okay";
9  };
10
11 &usbdrd_dwc3_0 {
12     status = "okay";
13 };
14
15 &usbdrd_dwc3_1 {
16     dr_mode = "host"; /* 配置USB1 为Host only mode
           */
17     status = "okay";
18 };

```

1.2 Type-C0 /1 USB PHY DTS

Type-C0/1 USB PHY的硬件由USB 3.0 PHY（只支持Super-speed）和USB 2.0 PHY（支持High-speed/Full-speed/Low-speed）两部分组成。所以，对应的USB PHY DTS也包括USB 3.0 PHY和USB 2.0 PHY两部分。

1.2.1 Type-C0 /1 USB 3.0 PHY DTS

以RK3399 Sapphire Excavator Board Type-C0 /1 USB 3.0 PHY DTS配置为例：

`arch/arm64/boot/dts/rockchip/rk3399.dtsi`

```
1  tcphy0: phy@ff7c0000 {
2      compatible = "rockchip,rk3399-typec-
   phy";
3      reg = <0x0 0xff7c0000 0x0 0x40000>;
4      #phy-cells = <1>;
5      clocks = <&cru SCLK_UPHY0_TCPDCORE>,
6              <&cru
   SCLK_UPHY0_TCPDPHY_REF>;
7      clock-names = "tcpdcore", "tcpdphy-
   ref";
8      assigned-clocks = <&cru
   SCLK_UPHY0_TCPDCORE>;
9      assigned-clock-rates = <50000000>;
10     power-domains = <&power
   RK3399_PD_TCPD0>;
11     resets = <&cru SRST_UPHY0>,
12             <&cru SRST_UPHY0_PIPE_L00>,
13             <&cru SRST_P_UPHY0_TCPHY>;
14     reset-names = "uphy", "uphy-pipe",
   "uphy-tcphy";
15     rockchip,grf = <&grf>;
16     rockchip,typec-conn-dir = <0xe580 0
   16>;
17     rockchip,usb3tousb2-en = <0xe580 3
   19>;
18     rockchip,usb3-host-disable = <0x2434
   0 16>;
19     rockchip,usb3-host-port = <0x2434 12
   28>;
20     rockchip,external-psm = <0xe588 14
   30>;
21     rockchip,pipe-status = <0xe5c0 0 0>;
22     rockchip,uphy-dp-sel = <0x6268 19
   19>;
23     status = "disabled";
24
25     tcphy0_dp: dp-port {
26         #phy-cells = <0>;
27     };
```

```

28
29         tcphy0_usb3: usb3-port { /* Type-C0
USB3.0 port */
30             #phy-cells = <0>;
31         };
32     };
33
34     tcphy1: phy@ff800000 {
35         compatible = "rockchip,rk3399-typec-
phy";
36         reg = <0x0 0xff800000 0x0 0x40000>;
37         #phy-cells = <1>;
38         clocks = <&cru SCLK_UPHY1_TCPDCORE>,
39             <&cru
SCLK_UPHY1_TCPDPHY_REF>;
40         clock-names = "tcpdcore", "tcpdphy-
ref";
41         assigned-clocks = <&cru
SCLK_UPHY1_TCPDCORE>;
42         assigned-clock-rates = <500000000>;
43         power-domains = <&power
RK3399_PD_TCPD1>;
44         resets = <&cru SRST_UPHY1>,
45             <&cru SRST_UPHY1_PIPE_L00>,
46             <&cru SRST_P_UPHY1_TCPHY>;
47         reset-names = "uphy", "uphy-pipe",
"uphy-tcphy";
48         rockchip,grf = <&grf>;
49         rockchip,typec-conn-dir = <0xe58c 0
16>;
50         rockchip,usb3tousb2-en = <0xe58c 3
19>;
51         rockchip,usb3-host-disable = <0x2444
0 16>;
52         rockchip,usb3-host-port = <0x2444 12
28>;
53         rockchip,external-psm = <0xe594 14
30>;
54         rockchip,pipe-status = <0xe5c0 16
16>;
55         rockchip,uphy-dp-se1 = <0x6268 3 19>;
56         status = "disabled";
57
58         tcphy1_dp: dp-port {
59             #phy-cells = <0>;
60         };
61
62         tcphy1_usb3: usb3-port { /* Type-C1
USB3.0 port */
63             #phy-cells = <0>;
64         };
65     };

```

```

1  &tcphy0 {
2      extcon = <&fusb0>;
3      status = "okay";
4  };
5
6  &tcphy1 {
7      /* Type-C1 使用的是Type-A USB接口，不用配置extcon
   属性 */
8      status = "okay";
9  };
10
11 &pinctrl {
12     .....
13     fusb30x {
14         fusb0_int: fusb0-int { /* 配置TypeC0
   fusb302 中断 */
15             rockchip,pins = <1 2
   RK_FUNC_GPIO &pcfg_pull_up>;
16             };
17     };
18 };
19
20 &i2c4 { /* 配置fusb302芯片的i2c */
21     status = "okay";
22     i2c-scl-rising-time-ns = <475>;
23     i2c-scl-falling-time-ns = <26>;
24
25     fusb0: fusb30x@22 {
26         compatible = "fairchild,fusb302";
27         reg = <0x22>;
28         pinctrl-names = "default";
29         pinctrl-0 = <&fusb0_int>;
30         int-n-gpios = <&gpio1 2
   GPIO_ACTIVE_HIGH>;
31         vbus-5v-gpios = <&gpio2 0
   GPIO_ACTIVE_HIGH>;
32         status = "okay";
33     };
34 };

```

1.2.2 Type-C0 /1 USB 2.0 PHY DTS

RK3399 有两个 USB 2.0 combphy（一个PHY支持两个port，一个port连接OTG，连接port连接Host），本文档称之为USB 2.0 PHY0和PHY1（参考图1-2）。其中，PHY0的port0作为Type-C0 USB的USB 2.0 PHY，PHY1的port0作为Type-C1 USB的USB 2.0 PHY。

以RK3399 Sapphire Excavator Board Type-C0 /1 USB2.0 PHY DTS配置为例：


```

1  grf: syscon@ff770000 {
2      compatible = "rockchip,rk3399-grf",
   "syscon", "simple-mfd";
3      .....
4      u2phy0: usb2-phy@e450 {
5          compatible =
   "rockchip,rk3399-usb2phy";
6          reg = <0xe450 0x10>;
7          clocks = <&cru
   SCLK_USB2PHY0_REF>;
8          clock-names = "phyclk";
9          #clock-cells = <0>;
10         clock-output-names =
   "clk_usbphy0_480m";
11         status = "disabled";
12
13         u2phy0_otg: otg-port { /* 配置
   Type-C0 USB2.0 PHY0 port0 */
14             #phy-cells = <0>;
15             interrupts = <GIC_SPI
   103 IRQ_TYPE_LEVEL_HIGH 0>,
16                             <GIC_SPI
   104 IRQ_TYPE_LEVEL_HIGH 0>,
17                             <GIC_SPI
   106 IRQ_TYPE_LEVEL_HIGH 0>;
18             interrupt-names =
   "otg-bvalid", "otg-id",
19             "linestate";
20             status = "disabled";
21         };
22
23         .....
24     };
25
26     u2phy1: usb2-phy@e460 { /* 配置Type-C1
   USB2.0 PHY1 port0 */
27         compatible =
   "rockchip,rk3399-usb2phy";
28         reg = <0xe460 0x10>;
29         clocks = <&cru
   SCLK_USB2PHY1_REF>;
30         clock-names = "phyclk";
31         #clock-cells = <0>;
32         clock-output-names =
   "clk_usbphy1_480m";
33         status = "disabled";
34
35         u2phy1_otg: otg-port { /*
   Type-C1 USB2.0 PHY1 port0*/

```

```

36                                     #phy-cells = <0>;
37                                     interrupts = <GIC_SPI
108 IRQ_TYPE_LEVEL_HIGH 0>,
38                                     <GIC_SPI
109 IRQ_TYPE_LEVEL_HIGH 0>,
39                                     <GIC_SPI
111 IRQ_TYPE_LEVEL_HIGH 0>;
40                                     interrupt-names =
"otg-bvalid", "otg-id",
41                                     "linestate";
42                                     status = "disabled";
43                                     };
44
45                                     .....
46                                     };

```

arch/arm64/boot/dts/rockchip/rk3399-sapphire.dtsi

```

1  &u2phy0 {
2      status = "okay";
3      extcon = <&fusb0>; /* extcon 属性*/
4
5      u2phy0_otg: otg-port {
6          status = "okay";
7      };
8      .....
9  };
10
11 &u2phy1 {
12     status = "okay";
13     /* u2phy1 只支持USB Host, 不用配置 extcon 属性 */
14
15     u2phy1_otg: otg-port {
16         status = "okay";
17     };
18     .....
19 };
20

```

1.3 Type-C1 USB OTG Mode DTS

在[1.1 Type-C0 /1 USB Controller DTS](#)中已经提到，由于当前的内核USB框架只能支持一个USB 口作为Peripheral功能，所以RK3399 SDK默认配置Type-C0作为OTG mode 支持USB Peripheral功能，而Type-C1只支持Host mode。实际产品中，可以根据应用需求，配置Type-C1为OTG mode，支持USB Peripheral功能，需要修改的地方有两个：

- DTS的“dr_mode”属性

```

1 &usbdrd_dwc3_1 {
2     status = "okay";
3     dr_mode = "otg"; /* 配置Type-C1 USB控制器为OTG mode */
4     extcon = <&fusb1>; /* 注意: extcon 属性要根据实际的硬件电路设计来配置 */
5 };

```

- init.rk30board.usb.rc 的USB控制器地址（适用于Android平台）
设置USB控制器的地址为Type-C1 USB控制器的基地址：

```
setprop sys.usb.controller "fe900000.dwc3"
```

2 Type-C to Type-A USB 3.0 Host DTS

Type-A USB 3.0的接口类型如下图2-1所示。

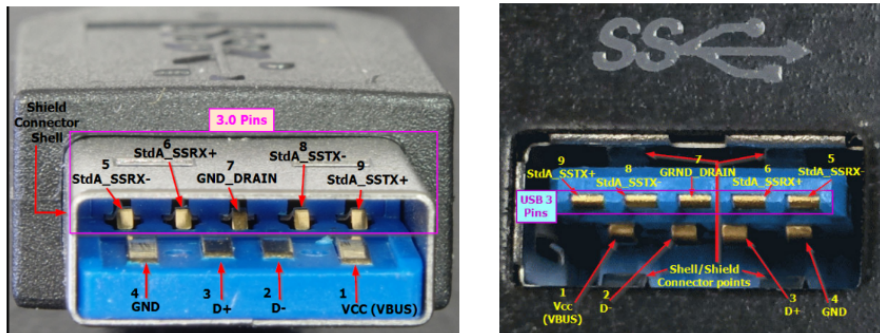


图2-1 Type-A USB3.0接口类型示意图

Type-C USB可以配置为Type-A USB使用。如RK3399 Sapphire Excavator Board平台的Type-C1 USB默认设计为Type-A USB 3.0 Host。这种设计，USB Vbus 5V一般为常供电，不需要单独的GPIO控制，也不需要fusb302芯片，但Type-C的三路供电需要正常开启，如下图2-2所示，才能支持USB 3.0 Super-speed。



图2-2 Type-C 供电电路

Type-A USB3.0 Host DTS配置的注意点如下：

- 对应的fusb节点不要配置，因为Type-A USB3.0不需要fusb302芯片
- 对应的USB控制器父节点（usbdrd3）和PHY的节点（tcphy和u2phy）都要删除extcon属性
- 对应的USB控制器子节点（usbdrd_dwc3）的dr_mode属性要配置为"host"

以RK3399 Sapphire Excavator Board 平台为例（Type-C0 配置为Type-C接口，Type-C1配置为Type-A USB 3.0 接口），Type-A USB 3.0 Host DTS对应的配置方法如下：

```
arch/arm64/boot/dts/rockchip/rk3399-sapphire.dtsi
```

```

1  /* Enable Type-C1 USB 3.0 PHY */
2  &tcphy1 {
3      /* Type-C1 使用的是Type-A USB接口，不用配置extcon
   属性 */
4      status = "okay";
5  };
6
7  /* Enable Type-C1 USB 2.0 PHY */
8  &u2phy1 {
9      status = "okay";
10     /* u2phy1 只支持USB Host，不用配置 extcon 属性 */
11
12     u2phy1_otg: otg-port {
13         status = "okay";
14     };
15     .....
16 };
17
18 /* Configure and Enable Type-C1 USB 3.0 Controller
   */
19 &usbdrd3_1 {
20     status = "okay";
21 };
22
23 &usbdrd_dwc3_1 {
24     /* 配置dr_mode为host，表示只支持Host only mode，
   并且不用配置 extcon 属性 */
25     dr_mode = "host";
26     status = "okay";
27 };

```

3 Type-C to Micro USB 3.0 OTG Mode DTS

Micro USB 3.0 OTG的接口类型如下图3-1所示。

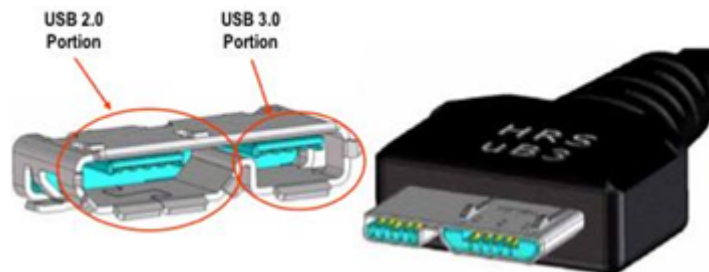


图3-1 Micro USB3.0 OTG接口类型示意图

为了节省硬件成本，Type-C USB可以配置为Micro USB 3.0 OTG使用。这种设计，硬件上不需要fusb302芯片，USB Vbus 5V一般由GPIO控制，Type-C的三路供电与2 [Type-C to Type-A USB 3.0 Host DTS](#)的硬件电路一样，需要正常开启。

Micro USB3.0 OTG DTS配置的注意点如下：

- 对应的fusb节点不要配置，因为Micro USB3.0不需要fusb302芯片
- 对应的USB PHY节点（tcphy和u2phy）都要删除extcon属性
- 对应的USB控制器父节点（usbdrd3）中，extcon属性引用为u2phy的节点
- 对应的USB控制器子节点（usbdrd_dwc3）的dr_mode属性要配置为"otg"
- 对应的USB2 PHY节点（u2phy）中，配置Vbus regulator
- Micro USB 3.0 OTG 是根据ID脚的电平变化（与Micro USB 2.0 OTG相同）来切换Peripheral mode和Host mode

以Type-C0 USB配置为Micro USB 3.0 OTG为例：

```

1  /* Enable Type-C0 USB 3.0 PHY */
2  &tcphy0 {
3      /* Type-C0 使用的是Micro USB 3.0接口，不用配置
4      extcon属性 */
5      status = "okay";
6  };
7
8  /* Enable Type-C0 USB 2.0 PHY */
9  &u2phy0 {
10     /* USB 2.0 PHY 不用配置extcon属性 */
11     status = "okay";
12     otg-vbus-gpios = <&gpio3 RK_PC6
13     GPIO_ACTIVE_HIGH>; /* Vbus GPIO配置，见Note1 */
14
15     u2phy1_otg: otg-port {
16         status = "okay";
17     };
18     .....
19 };
20
21 /* Configure and Enable Type-C0 USB 3.0 Controller
22 */
23 &usbdrd3_0 {
24     /* USB控制器的extcon属性必须引用u2phy0，才能支持
25     Peripheral mode和Host mode切换 */
26     extcon = <&u2phy0>;
27     status = "okay";
28 };
29
30 &usbdrd_dwc3_0 {
31     /* USB控制器的dr_mode必须配置为otg */
32     dr_mode = "otg";
33     status = "okay";
34 };

```

Note1.

Kernel 4.4最新的代码，已经将OTG USB Vbus的控制改为regulator的方式，对应的提交信息如下：

commit a1ca1be8f6ed “phy: rockchip-inno-usb2: use fixed-regulator for vbus power”

参考文档:

Documentation/devicetree/bindings/phy/phy-rockchip-inno-usb2.txt

DTS中对USB Vbus的控制，应该改为:

```
1  vcc_otg_vbus: otg-vbus-regulator {
2      compatible = "regulator-fixed";
3      gpio = <&gpio3 RK_PC6
4          GPIO_ACTIVE_HIGH>;
5      pinctrl-names = "default";
6      pinctrl-0 = <&otg_vbus_drv>;
7      regulator-name = "vcc_otg_vbus";
8      regulator-min-microvolt = <5000000>;
9      regulator-max-microvolt = <5000000>;
10     enable-active-high;
11
12     };
13
14     &pinctrl {
15         .....
16         usb {
17             otg_vbus_drv: otg-vbus-drv {
18                 rockchip,pins = <3 RK_PC6
19                     RK_FUNC_GPIO &pcfg_pull_none>;
20             };
21         };
22     };
23
24     &u2phy0 {
25         status = "okay";
26
27         u2phy0_otg: otg-port {
28             vbus-supply = <&vcc_otg_vbus>; /*配置
29             vbus regulator属性 */
30             status = "okay";
31         };
32     };
33     .....
34 }
```

4 Type-C to Micro USB 2.0 OTG Mode DTS

Micro USB 2.0 OTG的接口类型如下图4-1所示。



图4-1 Micro USB2.0 OTG接口类型示意图

为了节省硬件成本，Type-C USB可以配置为Micro USB 2.0 OTG使用。这种设计，硬件上不需要fusb302芯片，USB Vbus 5V一般由GPIO控制，因为不需要支持USB3.0，所以对应的Type-C三路供电（USB_AVDD_0V9，USB_AVDD_1V8，USB_AVDD_3V3）可以关闭。

Micro USB2.0 OTG DTS配置的注意点如下：

- 对应的fusb节点不要配置，因为Micro USB2.0不需要fusb302芯片
- Disable对应的USB3 PHY节点（tcphy）
- 对应的USB2 PHY节点（u2phy）要删除extcon属性，并且配置Vbus regulator
- 对应的USB控制器父节点（usbdrd3）中，extcon属性引用为u2phy
- 对应的USB控制器子节点（usbdrd_dwc3）的dr_mode属性要配置为"otg"，maximum-speed 属性配置为high-speed，phys 属性只引用USB2 PHY节点

以Type-C0 USB配置为Micro USB2.0 OTG为例：

```

1  /* Disable Type-C0 USB 3.0 PHY */
2  &tcphy0 {
3      status = "disabled";
4  };
5
6  /* Enable Type-C0 USB 2.0 PHY */
7  &u2phy0 {
8      /* USB 2.0 PHY 不用配置extcon属性 */
9      status = "okay";
10     otg-vbus-gpios = <&gpio3 RK_PC6
        GPIO_ACTIVE_HIGH>; /* Vbus GPIO配置，见Note1 */
11
12     u2phy1_otg: otg-port {
13         status = "okay";
14     };
15     .....
16 };
17
18 /* Configure and Enable Type-C0 USB 3.0 Controller
   */
19 &usbdrd3_0 {
20     /* USB控制器的extcon属性必须引用u2phy0，才能支持
        Peripheral mode和Host mode切换 */
21     extcon = <&u2phy0>;

```

```

22         status = "okay";
23     };
24
25     &usbdrd_dwc3_0 {
26         dr_mode = "otg"; /* USB控制器的dr_mode配置为otg
27         */
28         maximum-speed = "high-speed"; /* maximum-
29         speed 属性配置为high-speed */
30         phys = <&u2phy0_otg>; /* phys 属性只引用USB2
31         PHY节点 */
32         phy-names = "usb2-phy";
33         status = "okay";
34     };

```

Note1.

Kernel 4.4最新的代码，已经将OTG USB Vbus的控制改为regulator的方式（commit a1ca1be8f6ed “phy: rockchip-inno-usb2: use fixed-regulator for vbus power”），参考文档：

Documentation/devicetree/bindings/phy/phy-rockchip-inno-usb2.txt

所以，DTS中对OTG USB Vbus的控制，请参考[3 Type-C to Micro USB 3.0 OTG Mode DTS](#)中Vbus regulator的配置方法。

5 Type-C to Type-A USB 2.0

Type-C to Type-A USB 2.0的硬件设计方案（ID脚悬空），可以细化为三种不同的实现形式，分别是：

1. Type-C to Type-A USB 2.0 OTG mode DTS
2. Type-C to Type-A USB 2.0 Host only mode DTS
3. Type-C to Type-A USB 2.0 OTG mode DTS and Support DP 4 Lane

以下章节详细说明上述三种Type-C to Type-A USB 2.0方案的DTS配置。

5.1 Type-C to Type-A USB 2.0 OTG mode DTS

该方案的特点是，支持USB 2.0 OTG功能，Vbus为常供电或者通过GPIO/PMIC控制。系统启动后，需要应用层通过内核的接口设置USB控制器工作于Host mode。Type-C的三路供电（见图2-2）可以关闭。

以Type-C0 USB配置为Type-C to Type-A USB 2.0 OTG mode为例，其中，Vbus通过GPIO3_PC6控制

```

1  /* Disable Type-C0 USB 3.0 PHY */
2  &tcphy0 {
3      status = "disabled";
4  };
5
6  /* 配置vbus regulator属性 */

```



```

7 vcc_otg_vbus: otg-vbus-regulator {
8     compatible = "regulator-fixed";
9     gpio = <&gpio3 RK_PC6
    GPIO_ACTIVE_HIGH>;
10     pinctrl-names = "default";
11     pinctrl-0 = <&otg_vbus_drv>;
12     regulator-name = "vcc_otg_vbus";
13     regulator-min-microvolt = <5000000>;
14     regulator-max-microvolt = <5000000>;
15     enable-active-high;
16 };
17
18 &pinctrl {
19     .....
20     usb {
21         otg_vbus_drv: otg-vbus-drv {
22             rockchip,pins = <3 RK_PC6
    RK_FUNC_GPIO &pcfg_pull_none>;
23         };
24     };
25 };
26
27 /* Enable Type-C0 USB 2.0 PHY */
28 &u2phy0 {
29     status = "okay";
30
31     u2phy0_otg: otg-port {
32         vbus-supply = <&vcc_otg_vbus>; /* 配置
    vbus regulator属性，见Note1 */
33         status = "okay";
34     };
35     .....
36 };
37
38 /* Configure and Enable Type-C0 USB 3.0 Controller
    */
39 &usbdrd3_0 {
40     /* USB控制器不用配置extcon属性，通过内核节点来切换
    Peripheral mode和Host mode，见Note2 */
41     status = "okay";
42 };
43
44 &usbdrd_dwc3_0 {
45     dr_mode = "otg"; /* USB控制器的dr_mode配置为otg
    */
46     maximum-speed = "high-speed"; /* maximum-
    speed 属性配置为high-speed */
47     phys = <&u2phy0_otg>; /* phys 属性只引用USB2
    PHY节点 */
48     phy-names = "usb2-phy";
49     status = "okay";
50 };

```

Note1

假如Vbus为常供电（也即系统开机后，Vbus一直为高），则不需要配置“vbus-supply”属性，但需要增加如下的DTS属性，否则，会出现USB ADB无法正常连接的情况。

```
1 &u2phy0_otg {
2     rockchip,vbus-always-on;
3 };
```

Note2

内核中切换USB控制器工作在Peripheral mode或Host mode的接口：

旧的接口使用方法：

```
1 1.Force host mode
2     echo host >
  sys/kernel/debug/usb@fe800000/rk_usb_force_mode
3 2.Force peripheral mode
4     echo peripheral >
  sys/kernel/debug/usb@fe800000/rk_usb_force_mode
```

新的接口使用方法：

```
1 1.Force host mode
2     echo host > sys/kernel/debug/usb0/dwc3_mode
3 2.Force peripheral mode
4     echo peripheral > sys/kernel/debug/usb0/dwc3_mode
```

5.2 Type-C to Type-A USB 2.0 Host only mode DTS

该方案的特点是，只支持Host 功能，Vbus为常供电，进系统后不需要Device功能，但可以支持固件烧录。Type-C的三路供电（见图2-2）可以关闭。

以Type-C0 USB配置为Type-C to Type-A USB 2.0 Host mode为例，其中，Vbus为常供电，不需要软件控制

```
1 /* Disable Type-C0 USB 3.0 PHY */
2 &tcphy0 {
3     status = "disabled";
4 };
5
6 /* Enable Type-C0 USB 2.0 PHY */
7 &u2phy0 {
8     status = "okay";
9
10    u2phy0_otg: otg-port {
11        /* 不需要配置vbus regulator属性 */
12        status = "okay";
13    };
```

```

14         .....
15     };
16
17     /* Configure and Enable Type-C0 USB 3.0 Controller
18     */
19     &usbdrd3_0 {
20         /* USB控制器不用配置extcon属性，通过内核节点来切换
21         peripheral mode和Host mode */
22         status = "okay";
23     };
24
25     &usbdrd_dwc3_0 {
26         dr_mode = "host"; /* USB控制器的dr_mode配置为
27         host only mode */
28         maximum-speed = "high-speed"; /* maximum-
29         speed 属性配置为high-speed */
30         phys = <&u2phy0_otg>; /* phys 属性只引用USB2
31         PHY节点 */
32         phy-names = "usb2-phy";
33         status = "okay";
34     };

```

5.3 Type-C to Type-A USB 2.0 OTG mode DTS and Support DP 4 Lane

该方案的特点是，支持USB 2.0 OTG功能，同时USB 3.0的Tx和Rx配置给DP使用，以支持DP 4 lanes的功能。Type-C的三路供电（见图2-2）需要正常开启。Rockchip SDK Kernel默认没有支持该方案，如果要支持该方案，需要正确配置DTS，同时，还要增加新的驱动drivers/extcon/extcon-pd-virtual.c，该驱动的作用是替代fusb302驱动，发通知给Type-C PHY驱动和DP驱动，以配置DP 4 lanes。如果有该功能需求，请提交Issue到Rockchip的Redmine平台，或者发邮件给本文档的作者wulf@rock-chips.com

DTS配置参考如下：

```

1  /* 配置VPD驱动，用于发送通知给Type-C PHY驱动和DP驱动，以配置
2  DP 4 lanes */
3  vpd0:virtual-pd0{
4      compatible = "linux,extcon-pd-virtual";
5      pinctrl-names = "default";
6      pinctrl-0 = <&vpd0_int>;
7      dp-det-gpios = <&gpio4 25 GPIO_ACTIVE_LOW>;
8      hdmi-5v-gpios = <&gpio4 29 GPIO_ACTIVE_LOW>;
9
10     /* 0: positive, 1: negative*/
11     vpd,init-flip = <0>;
12     /* 0: u2, 1: u3*/
13     vpd,init-ss = <0>;
14     /* 0: dfp, 1: ufp, 2: dp 3: dp/ufp */
15     vpd,init-mode = <2>;
16     status = "okay";
17 };

```

```

17
18 &pinctrl {
19     vpd {
20         vpd0_int: vpd0-int {
21             rockchip,pins =
22                 <4 25 RK_FUNC_GPIO
23         &pcfg_pull_up>;
24     };
25 };
26
27 /* 配置DP */
28 &cdn_dp {
29     status = "okay";
30     extcon = <&vpd0>;
31     dp_vop_sel = <1>;
32 };
33
34 /* 配置Type-C0 PHY */
35 &tcphy0 {
36     extcon = <&vpd0>;
37     status = "okay";
38 };
39
40 /* 配置USB2 PHY */
41 &u2phy0 {
42     status = "okay";
43     /* 这里不需要配置extcon属性 */
44
45     u2phy0_otg: otg-port {
46         status = "okay";
47     };
48
49     u2phy0_host: host-port {
50         phy-supply = <&vcc5v0_host>;
51         status = "okay";
52     };
53 };
54
55 /* 配置USB控制器 */
56 &usbdrd3_0 {
57     /* extcon不是必要的。如果用micro OTG接口，需要配
58     置。如果用Type-A接口，不用配置，见Note1 */
59     extcon = <&u2phy0>;
60     status = "okay";
61 };
62 &usbdrd_dwc3_0 {
63     dr_mode = "otg"; /* 配置为otg mode，支持
64     peripheral和host切换 */
65     maximum-speed = "high-speed"; /* 配置最高支持
66     high-speed */

```

```

65         phys = <&u2phy0_otg>, <&tcphy0_usb3>; /* 必须要
           配置U2和U3 PHY */
66         phy-names = "usb2-phy", "usb3-phy";
67         status = "okay";
68     };

```

Note1

如果用Type-A接口，系统启动后，需要应用层通过内核提供的OTG mode切换节点，配置USB控制器工作Peripheral mode或者Host mode。配置方法参考[5.1 Type-C to Type-A USB 2.0 OTG mode DTS](#)

6 USB 2.0 Host DTS

RK3399 支持两个USB2.0 Host接口，对应的USB控制器为EHCI&OHCI，相比Type-C接口的多种硬件设计方案，USB2.0 Host的接口一般只有一种设计方案，即Type-A USB2.0 Host接口，对应的DTS配置，包括控制器DTS配置和PHY DTS配置。

实际方案中，用户一般不用重新配置Host Controller DTS，只需要根据实际硬件电路的USB VBUS设计，修改Host PHY DTS的“phy-supply”属性。

6.1 USB 2.0 Host Controller DTS

以RK3399 Sapphire Excavator Board USB2.0 Host 控制器 DTS配置为例：

`arch/arm64/boot/dts/rockchip/rk3399.dtsi`

```

1  usb_host0_ehci: usb@fe380000 {
2      compatible = "generic-ehci";
3      reg = <0x0 0xfe380000 0x0 0x20000>;
4      interrupts = <GIC_SPI 26
           IRQ_TYPE_LEVEL_HIGH 0>;
5      clocks = <&cru HCLK_HOST0>, <&cru
           HCLK_HOST0_ARB>,
6              <&cru
           SCLK_USBPHY0_480M_SRC>;
7      clock-names = "hclk_host0",
           "hclk_host0_arb", "usbphy0_480m";
8      phys = <&u2phy0_host>;
9      phy-names = "usb";
10     power-domains = <&power
           RK3399_PD_PERIHP>;
11     status = "disabled";
12 };
13
14 usb_host0_ohci: usb@fe3a0000 {
15     compatible = "generic-ohci";
16     reg = <0x0 0xfe3a0000 0x0 0x20000>;
17     interrupts = <GIC_SPI 28
           IRQ_TYPE_LEVEL_HIGH 0>;

```

```

18         clocks = <&cru HCLK_HOST0>, <&cru
HCLK_HOST0_ARB>,
19         <&cru
SCLK_USBPHY0_480M_SRC>;
20         clock-names = "hclk_host0",
"hclk_host0_arb", "usbphy0_480m";
21         phys = <&u2phy0_host>;
22         phy-names = "usb";
23         power-domains = <&power
RK3399_PD_PERIHP>;
24         status = "disabled";
25     };
26
27     usb_host1_ehci: usb@fe3c0000 {
28         compatible = "generic-ehci";
29         reg = <0x0 0xfe3c0000 0x0 0x20000>;
30         interrupts = <GIC_SPI 30
IRQ_TYPE_LEVEL_HIGH 0>;
31         clocks = <&cru HCLK_HOST1>, <&cru
HCLK_HOST1_ARB>,
32         <&cru
SCLK_USBPHY1_480M_SRC>;
33         clock-names = "hclk_host1",
"hclk_host1_arb", "usbphy1_480m";
34         phys = <&u2phy1_host>;
35         phy-names = "usb";
36         power-domains = <&power
RK3399_PD_PERIHP>;
37         status = "disabled";
38     };
39
40     usb_host1_ohci: usb@fe3e0000 {
41         compatible = "generic-ohci";
42         reg = <0x0 0xfe3e0000 0x0 0x20000>;
43         interrupts = <GIC_SPI 32
IRQ_TYPE_LEVEL_HIGH 0>;
44         clocks = <&cru HCLK_HOST1>, <&cru
HCLK_HOST1_ARB>,
45         <&cru
SCLK_USBPHY1_480M_SRC>;
46         clock-names = "hclk_host1",
"hclk_host1_arb", "usbphy1_480m";
47         phys = <&u2phy1_host>;
48         phy-names = "usb";
49         power-domains = <&power
RK3399_PD_PERIHP>;
50         status = "disabled";
51     };
52

```

```

1  &usb_host0_ehci {
2      status = "okay";
3  };
4
5  &usb_host0_ohci {
6      status = "okay";
7  };
8
9  &usb_host1_ehci {
10     status = "okay";
11 };
12
13 &usb_host1_ohci {
14     status = "okay";
15 };

```

6.2 USB 2.0 Host PHY DTS

以RK3399 Sapphire Excavator Board USB2.0 Host PHY DTS配置为例:

`arch/arm64/boot/dts/rockchip/rk3399.dtsi`

```

1  grf: syscon@fff770000 {
2      compatible = "rockchip,rk3399-grf",
3      "syscon", "simple-mfd";
4      reg = <0x0 fff770000 0x0 0x10000>;
5      #address-cells = <1>;
6      #size-cells = <1>;
7      .....
8      u2phy0: usb2-phy@e450 {
9          compatible =
10         "rockchip,rk3399-usb2phy";
11         reg = <0xe450 0x10>;
12         clocks = <&cru
13         SCLK_USB2PHY0_REF>;
14         clock-names = "phyclk";
15         #clock-cells = <0>;
16         clock-output-names =
17         "clk_usbphy0_480m";
18         status = "disabled";
19         .....
20         u2phy0_host: host-port { /* 配
21         置USB2.0 Host0 PHY节点 */
22         #phy-cells = <0>;
23         interrupts = <GIC_SPI
24         27 IRQ_TYPE_LEVEL_HIGH 0>;
25         interrupt-names =
26         "linestate";
27         status = "disabled";
28     };
29 };

```

```

23
24         u2phy1: usb2-phy@e460 {
25             compatible =
26 "rockchip,rk3399-usb2phy";
27             reg = <0xe460 0x10>;
28             clocks = <&cru
29 SCLK_USB2PHY1_REF>;
30             clock-names = "phyclk";
31             #clock-cells = <0>;
32             clock-output-names =
33 "clk_usbphy1_480m";
34             status = "disabled";
35             .....
36             u2phy1_host: host-port { /* 配
37 置USB2.0 Host1 PHY节点 */
38             #phy-cells = <0>;
39             interrupts = <GIC_SPI
40 31 IRQ_TYPE_LEVEL_HIGH 0>;
41             interrupt-names =
42 "linestate";
43             status = "disabled";
44             };
45         };

```

arch/arm64/boot/dts/rockchip/rk3399-sapphire.dtsi

```

1  /* 配置USB2.0 Host Vbus regulator */
2  vcc5v0_host: vcc5v0-host-regulator {
3      compatible = "regulator-fixed";
4      enable-active-high;
5      gpio = <&gpio4 25 GPIO_ACTIVE_HIGH>;
6      pinctrl-names = "default";
7      pinctrl-0 = <&host_vbus_drv>;
8      regulator-name = "vcc5v0_host";
9      regulator-always-on;
10 };
11
12 &pinctrl {
13     .....
14     usb2 {
15         host_vbus_drv: host-vbus-drv {
16             rockchip,pins =
17                 <4 25 RK_FUNC_GPIO
18                 &pcfg_pull_none>;
19         };
20     };
21 };
22
23 &u2phy0 {
24     status = "okay";

```



```

25     ...
26     u2phy0_host: host-port {
27         /* 配置USB2.0 Host0 phy-supply属性，用于
    控制Vbus */
28         phy-supply = <&vcc5v0_host>;
29         status = "okay";
30     };
31 };
32
33 &u2phy1 {
34     status = "okay";
35     ...
36     u2phy1_host: host-port {
37         /* 配置USB2.0 Host1 phy-supply属性，用于
    控制Vbus */
38         phy-supply = <&vcc5v0_host>;
39         status = "okay";
40     };
41 };

```

7 USB 3.0 force to USB 2.0

该功能是指在USB 3.0 Tx/Rx连接的情况下，要强制让USB运行在USB 2.0的速率。这种应用场景，一般用于硬件设计问题导致USB 3.0工作异常或者某些特殊的场景需求，需要去掉USB 3.0功能，只要支持USB 2.0。由于这不是常规功能，所以SDK驱动默认没有支持该功能。Rockchip以独立的补丁形式，发布给有这类需求的客户。如果有该功能需求，请提交Issue到Rockchip的Redmine平台，或者发邮件给本文档的作者wulf@rock-chips.com

8 关于USB VBUS供电的说明

RK3399 平台的USB Vbus供电硬件电路设计，主要有三种方案：

1. 使用GPIO控制电源稳压芯片输出Vbus 5V供电电压；

方案1是Rockchip平台常用的方案，可以用于Type-C USB 接口，Type-A USB接口，Micro USB接口等，不同接口，对应的DTS配置方案不同，具体如下：

（1）Type-C USB接口的Vbus GPIO配置，参考[1.2.1 Type-C0 /1 USB 3.0 PHY DTS](#)中“vbus-5v-gpios”属性的配置；

（2）Type-A USB接口的Vbus GPIO配置，参考[6.2 USB 2.0 Host PHY DTS](#)

（3）Micro USB的Vbus GPIO配置，参考[3 Type-C to Micro USB 3.0 OTG Mode DTS](#)

2. 使用PMIC（如RK817/RK818）输出Vbus 5V供电电压；

（1）如果PMIC使用的是RK8xx（RK809除外），DTS不用配置Vbus的属性，如果配置，反而可能会导致Vbus供电异常。这种方案，驱动会通过发送 EXTCON_USB_VBUS_EN 的通知给PMIC驱动，以控制Vbus的供电。

(2) 如果PMIC使用的是其他Vendor的芯片，请参考驱动drivers/power/rk818_charger.c实现接收EXTCON_USB_VBUS_EN 通知的逻辑。

(3) 如果PMIC使用的是RK809，由于该PMIC只有Vbus输出5V的供电功能，没有充电功能，所以不适用于使用发送EXTCON_USB_VBUS_EN 的通知的方法。可参考：

`arch/arm64/boot/dts/rockchip/rk3326-evb-ai-v11.dts`

```
1 rk809: pmic@20 {
2     regulators {
3         vcc5v0_host: SWITCH_REG1 {
4             regulator-name =
5             "vcc5v0_host";
6         };
7     };
8     &u2phy {
9         status = "okay";
10        u2phy_host: host-port {
11            status = "okay";
12        };
13        u2phy_otg: otg-port {
14            vbus-supply = <&vcc5v0_host>;
15            status = "okay";
16        };
17    };
```

3. 开机后，硬件直接输出Vbus 5V供电电压，不需要软件控制，一般用于USB Host接口；

9 参考文档

1. Documentation/devicetree/bindings/usb/generic.txt
2. Documentation/devicetree/bindings/usb/dwc3.txt
3. Documentation/devicetree/bindings/usb/rockchip,dwc3.txt
4. Documentation/devicetree/bindings/usb/usb-ehci.txt
5. Documentation/devicetree/bindings/usb/usb-ohci.txt
6. Documentation/devicetree/bindings/phy/phy-rockchip-typec.txt
7. Documentation/devicetree/bindings/phy/phy-rockchip-inno-usb2.txt