



Linux Type-C 开发指南

版本号: 1.2

发布日期: 2024.11.27

版本历史

版本号	日期	制/修订人	内容描述
1.0	2024.04.14	AWA1881	初始版本。
1.1	2024.06.11	AWA1881	根据评审意见进行修改。
1.2	2024.11.27	AWA1881	更新 menuconfig、dts 配置及源码说明。

目 录

1 前言	1
1.1 文档简介	1
1.2 目标读者	1
1.3 适用范围	1
2 模块介绍	2
2.1 模块功能介绍	2
2.2 相关术语介绍	2
2.3 模块应用介绍	3
2.4 模块硬件介绍	5
2.4.1 硬件通路	5
2.4.1.1 USB TYPE-C 插座端视图	5
2.4.1.2 USB TYPE-C 插头引出端视图	5
2.4.1.3 USB TYPE-C 典型应用设计	6
2.5 模块配置介绍	6
2.5.1 kernel menuconfig 配置说明	6
2.5.1.1 注意事项	8
2.5.2 Device Tree 配置说明	8
2.5.3 board.dts 配置说明	8
2.5.3.1 路径定义	8
2.5.3.2 配置示例	9
2.5.3.3 注意事项	13
2.5.3.4 相关配置	13
2.5.3.5 注意事项	14
2.5.3.6 扩展说明	14
2.6 源码说明	15
2.6.1 源码路径	15
2.6.2 源码结构	15
2.6.3 注意事项	16
2.6.4 软件设计	16
3 日志分析	18
3.1 驱动加载日志分析	18
3.2 Device 日志分析	19
3.3 Host 日志分析	20
3.4 快充日志分析	22
3.5 Type-C DP OUT 日志分析	23
3.6 Audio 日志分析	25

4	Type-C 调试方法	27
4.1	FSA4480 调试	27
4.2	PS8743 调试	27
4.3	CC Logic 调试	28
5	FAQ	29
5.1	如何在 Linux-4.9 内核上支持快充和 OTG	29
5.2	MTP 从 PC 拷贝文件至样机进度条不动	29
5.3	如何支持外设插拔唤醒	29
5.4	总线被拉低导致驱动加载失败	29
6	附录	31
6.1	相关补丁支持情况	31
6.1.1	音频模式支持	31
6.1.2	多 Switch 支持	32
6.2	参考资料	32
6.3	DRP 状态	33



插图

图 2-1	Type-C 架构设计	3
图 2-2	Type-C Environment 示意图	4
图 2-3	USB TYPE-C 插座端视图	5
图 2-4	USB TYPE-C 插头引出端视图	5
图 2-5	USB TYPE-C 典型应用设计	6
图 2-6	USB Type-C Support menuconfig 界面	7
图 2-7	USB Type-C Switch Support menuconfig 界面	7
图 2-8	DisplayPort Alternate Mode Support menuconfig 界面	8
图 2-9	Type-C 框架图	17
图 6-1	标准 Type-C 线缆	32
图 6-2	Type-C_ 线缆梳理	33
图 6-3	DRP_with_Accessory_and_TrySNK	34
图 6-4	DRP_with_Accessory_and_TrySRC	35

1 前言

1.1 文档简介

介绍 Type-C 接口相关模块配置和调试方法，为软件开发者提供一站式与 Type-C 接口相关的模块（USB2.0/3.0DRD + 快充 + DP-OUT + Audio）解决方案。

1.2 目标读者

Type-C 接口软硬件开发、维护人员。

1.3 适用范围

表 1-1: 适用内核列表

内核版本	驱动文件
Linux-5.15	drivers/usb/typec/*、bsp/drivers/usb/typec
Linux-6.6	drivers/usb/typec/*、bsp/drivers/usb/typec

说明

注：本文档描述仅适用于 Linux-5.10 及以上的内核版本，一般不区分产品和平台。

表 1-2: 物料支持列表

类别	内核原生支持	AW 私有支持
CC Logic	FUSB302、RT1711 等	HUSB311
快充支持	unknown	ETA6973、ETA6974、AXP519
USB&DP 支持	PI3USB30532	PS8743B、GMA340、PhySwitcher[Virtual]
USB&Audio 支持	FSA4480	BCT4480、ASW5480

2 模块介绍

2.1 模块功能介绍

Type-C 接口又叫 USB Type-C 接口，是一种外形标准，拥有比 Type-A 及 Type-B 更小的体积，既可以应用于 PC（主设备）又可以应用于外部设备（从设备，如手机）的接口类型。此外，Type-C 接口的配件模式可用于支持 DP，原生支持的 PD 协议可用于支持快充。

2.2 相关术语介绍

表 2-1: Type-C 接口相关术语介绍：

术语	解释说明
TCPM	USB Type-C Port Manager，即 USB Type-C 端口管理器，文档内指 Type-C 端口管理框架。
TCPC	USB Type-C Port Controller，即 USB Type-C 端口控制器。
TCPCI	USB Type-C Port Controller Interface，即 USB Type-C 端口控制器接口。
Sunxi	指 Allwinner 的一系列 SoC 硬件平台。
USB	Universal Serial Bus, 通用串行总线。
xHCI	eXtensible Host Controller Interface，可扩展的主机控制器接口。
DWC3	DesignWare USB3.0/3.1/3.2 控制器及驱动。
AXP	全志 PMU 的名称，如 AXP2202，AXP515 等。
TWI	同等于 I2C，AXP 的使用需要用到 I2C 的通讯进行读写寄存器。
PMU	电源管理单元。

说明

Type-C 控制器接口相关的介绍参考文档《[USB Type-C Port Controller Interface Specification](#)》，这里仅展示以下架构图用以理解 Type-C 的 TCPC 接口。

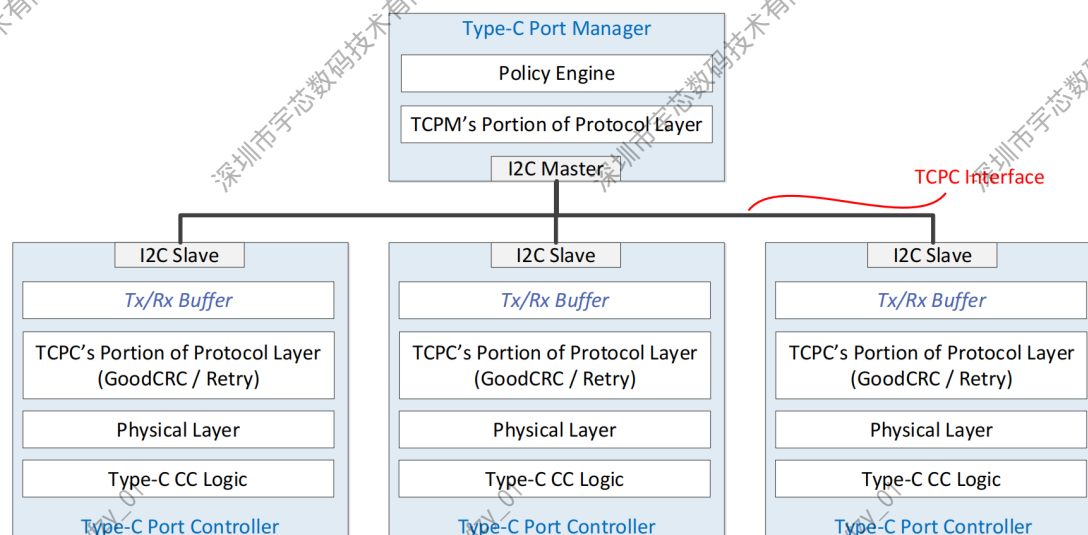


Figure 1-1. USB Type-C Port Manager to USB Type-C Port Controller Interface

图 2-1: Type-C 架构设计

2.3 模块应用介绍

此模块的主要使用场景如下：（1）支持 USB3 的 DRD；（2）支持 PD 协议；（3）支持 Type-C DP。

Type-C 相对于其它接口的优势：

- 无需区分正反
- 最高可以支持 5A 大电流，
- 在 USB4 标准支持下，最高可传输速率可达 40Gbps。
- 正确插入后发出声音。
- 通过 USB Power Delivery 技术，可用于 3C 产品（如笔记本电脑、智能手机）的充电。
- 增进的电磁干扰（EMI）与射频干扰（RFI mitigation）特性。
- 可选集成 DisplayPort、HDMI、MHL。
- 可选集成 Thunderbolt。
- 可选集成 USB4。

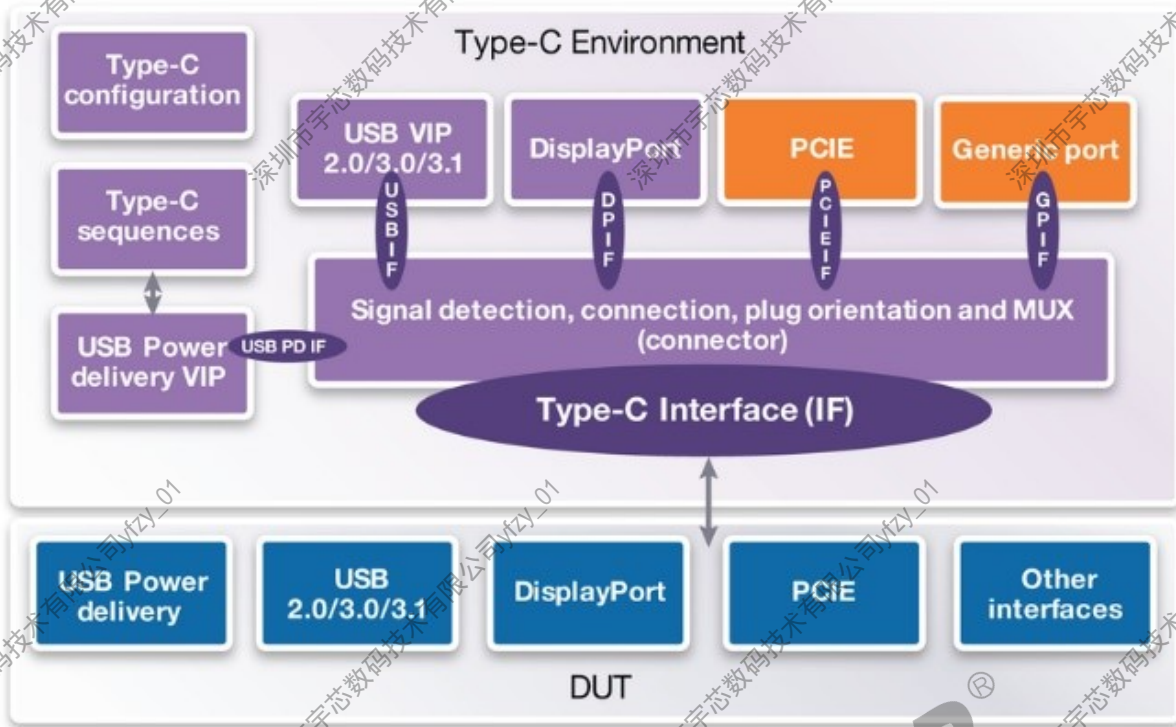


图 2-2: Type-C Environment 示意图



2.4 模块硬件介绍

2.4.1 硬件通路

2.4.1.1 USB TYPE-C 插座端视图

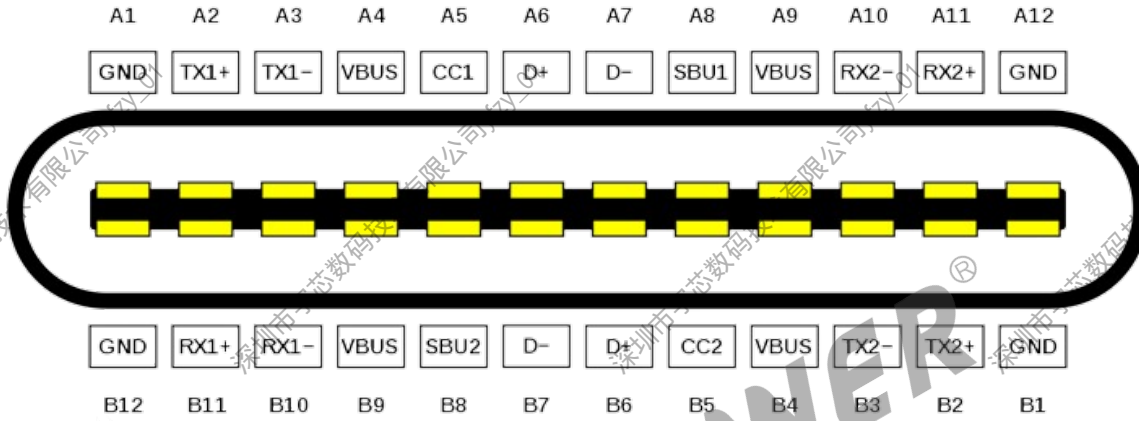


图 2-3: USB TYPE-C 插座端视图

2.4.1.2 USB TYPE-C 插头引出端视图

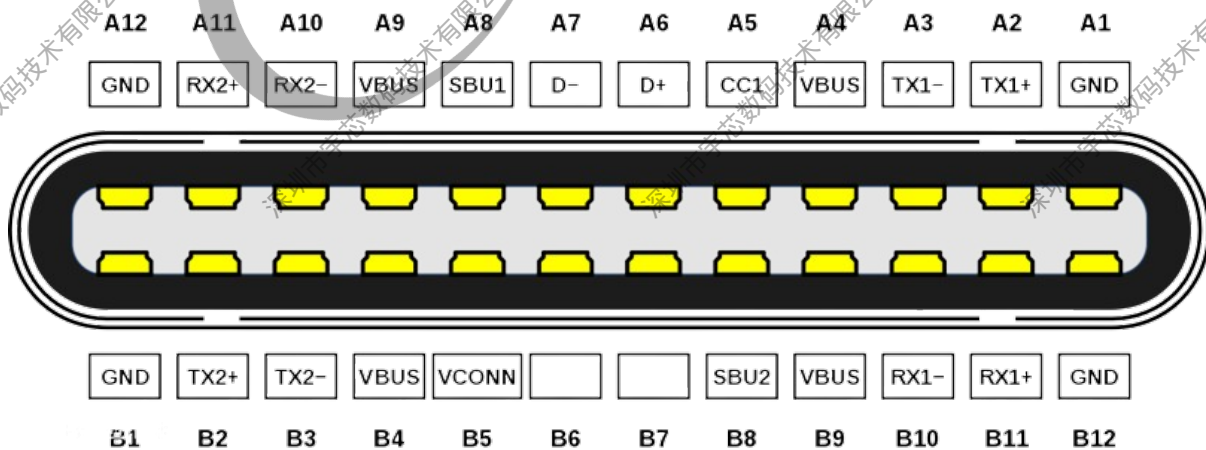


图 2-4: USB TYPE-C 插头引出端视图

2.4.1.3 USB TYPE-C 典型应用设计

TYPICAL APPLICATION CIRCUIT

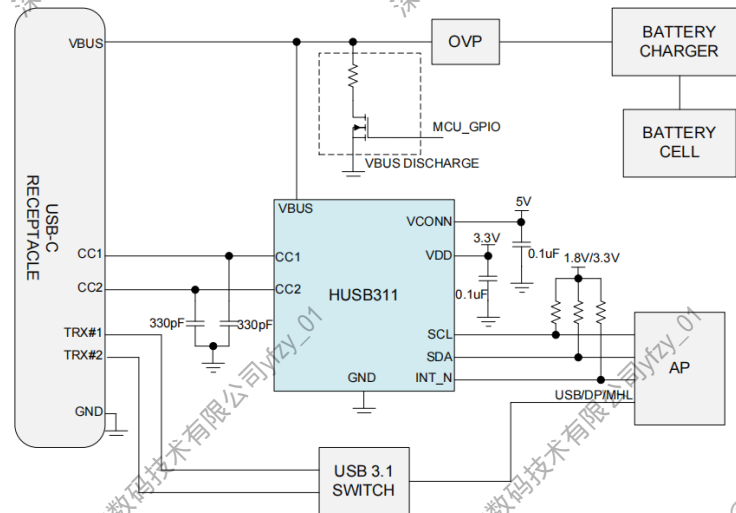


Figure 1. HUSB311 Typical Application Circuit

图 2-5: USB TYPE-C 典型应用设计

以 {IC} {BOARD} 板为例，完整的 Type-C 硬件功能设计如下：

- 快充部分通过 ETA6973 支持；
- USB3/DP 超速 TXRX 信号通过 PS8743B 切换；
- USB2.0/Audio 高速 DP/DM 信号通过 BCT4480/ASW5480 切换；
- PD 协商和 CC Logci 通过 HUSB311 支持。

特别强调，硬件设计上，HUSB311 的 VDD 需要确保使用非掉电域的 3.3V 进行供电，否则可能会导致休眠唤醒后接入 OTG 外设无响应。

2.5 模块配置介绍

2.5.1 kernel menuconfig 配置说明

在 SDK 根目录下，执行 ./build.sh menuconfig，进入配置主界面，并按以下步骤操作：

如果支持 Type-C Port Controller，需要按照以下步骤打开对应配置项（注：必须外挂 HUSB311 以及 PS8743 芯片才需要打开，若外挂其他 CC Logic 以及 Switch 芯片可参考对应驱动进行开发）。

回到主界面，依次选择 Allwinner BSP > Device Drivers > USB Drivers > USB Type-C Support，如下图所示：

<1> 开启 **Allwinner USB Type-C support**，把 **CONFIG_USB_SUNXI_TYPEC** 配置为 Y；

<2> 开启 **Hynetek HUSB311 Type-C chip driver**，把 **CONFIG_TYPEC_HUSB311** 配置为 Y；

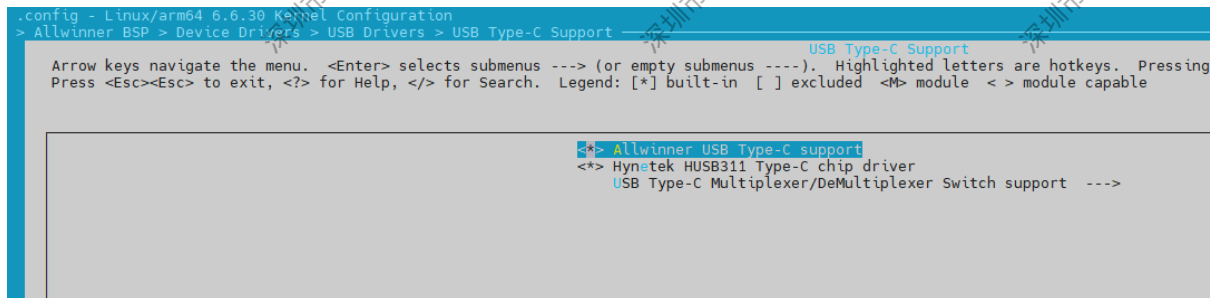


图 2-6: USB Type-C Support menuconfig 界面

<3> 进入 **USB Type-C Multiplexer/DeMultiplexer Switch support** 页，开启 **Parade PS8743 Type-C cross switch driver**，把 **CONFIG_TYPEC_MUX_PS8743** 配置为 Y；

<4> 开启 **ON Semi FSA4480 Analog Audio Switch driver**，把 **CONFIG_TYPEC_MUX_FSA4480** 配置为 Y。

需要注意的是，目前全志平台支持 **Allwinner Mux/Switcher based on external PHY**，通过抽象 Mux 的相关操作，将 Type-C 接口的正反插及模式切换参数传递给指定模块，实现类似外挂 Switch 的功能。

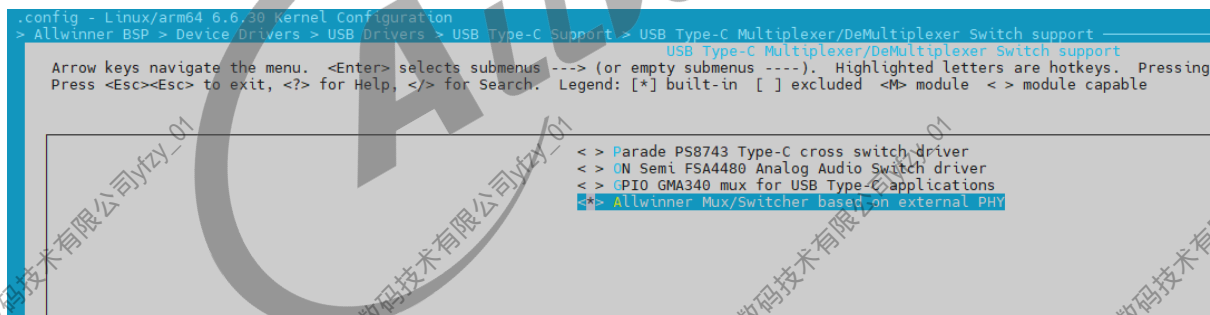


图 2-7: USB Type-C Switch Support menuconfig 界面

<5> 如需使用 Type-C DP OUT 功能，需开启 **DisplayPort Alternate Mode driver**，把 **CONFIG_TYPEC_DP_ALTMODE** 配置为 Y。

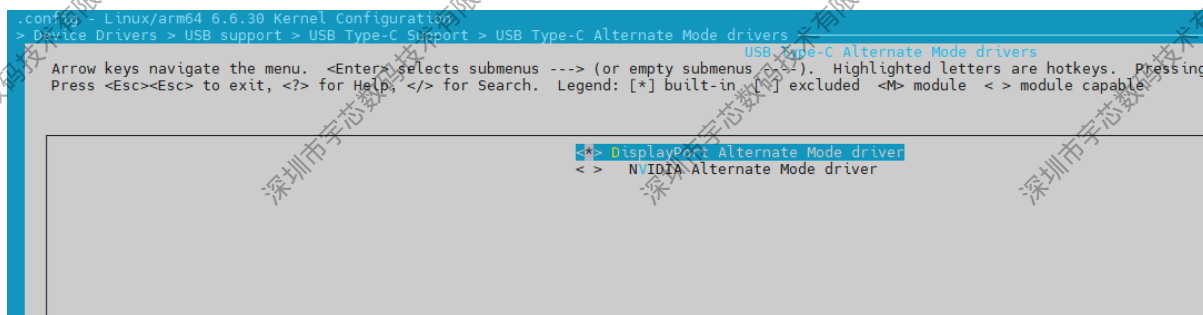


图 2-8: DisplayPort Alternate Mode Support menuconfig 界面

2.5.1.1 注意事项

说明

- 模拟音频设备需要打开 `CONFIG_SND_SOC_SUNXI_AUDIO` 配置确保声卡正常加载，数字音频设备需要开启 `CONFIG_SND_USB_AUDIO` 配置；
- USB 相关配置参考《Linux_USB2_开发指南.pdf》和《Linux_USB3_开发指南.pdf》，快充相关配置参考《Linux_快充_开发指南.pdf》，DP-OUT 相关配置参考《Linux_EDP_开发指南.pdf》。

2.5.2 Device Tree 配置说明

设备树中定义的是该类芯片对应于 IC 规格的所有配置，设备树文件路径如下：

```
32/64 位平台：bsp/configs/linux-x.x/{CHIP}.dtsi
```

硬件接口形态以及外挂 IC 一般跟随板级，因此，无需进行 dtsi 配置，仅添加 pd.h 的引用即可。

2.5.3 board.dts 配置说明

2.5.3.1 路径定义

board.dts 用于保存每一个板级平台的设备信息（如 demo 板，perf1 板等），里面的配置信息会覆盖上面的 Device Tree 中 dtsi 默认配置信息。不同 IC、版型及内核版本对应的 board.dts 具体路径如下。

```
device/config/chips/IC/configs/{BOARD}/${内核版本}/board.dts
```

2.5.3.2 配置示例

搭配 SWITCH 芯片（可选）以及 CC Logic 芯片参考以下配置（此处以 FSA4480、PS8743 和 HUSB311 为例）：

```
1 phy_switcher: phy_switcher@10 {
2     compatible = "allwinner,sunxi-phy-switcher";
3     reg = <0x0 0x10 0x0 0x0>;
4     hotplug = <&pio PH 4 GPIO_ACTIVE_HIGH>; /* Switch通知DP热插拔事件的io配置，仅使用USB3.0可以不配置。 */
5     aux_p = <&r_pio PL 10 GPIO_ACTIVE_HIGH>;
6     aux_n = <&r_pio PL 11 GPIO_ACTIVE_HIGH>;
7     phys = <&combo0_dp>, <&combo0_usb>;
8     phy-names = "dp-phy", "usb-phy";
9
10    status = "okay";
11
12    mode-switch; /* 用于Type-C内核框架获取模式的切换。 */
13    orientation-switch; /* 用于Type-C内核框架获取正反插方向的切换。 */
14    svid = <0xff01>; /* 用于识别配件模式altmode为DP。 */
15    port {
16        #address-cells = <1>;
17        #size-cells = <0>;
18        usbdp_orientation_switch: endpoint@0 {
19            reg = <0>;
20            remote-endpoint = <&usb2_orien_sw>; /* 使用port描述，phy_switcher与远程hub311设备关联，用于切换方向。 */
21        };
22        usbdp_dp_altmode_mux: endpoint@1 {
23            reg = <1>;
24            remote-endpoint = <&dp_altmode_mux>; /* 使用port描述，phy_switcher与远程hub311设备关联，用于切换MUX。 */
25        };
26    };
27 };
28
29 &twi5 {
30     fsa4480: fsa4480@43 {
31         compatible = "fcs,fsa4480"; /* 设备匹配名，对应Type-C模拟音频芯片的型号，目前BCT4480/ASW5480可复用此配置。 */
32         reg = <0x43>;
33         interrupts-extended;
34         vcc-supply; /* power supply (2.7V-5.5V) */
35         status = "okay";
36
37         mode-switch;
38         orientation-switch;
39         accessory;
40         // svid = <0xff01>; /* BCT4480要添加 accessory; 属性而不要添加 svid = <0xff01>; 属性。 */
41         port {
42             fsa4480_sbu_mux: endpoint@0 {
43                 remote-endpoint = <&typec_sbu_out>; /* 使用port描述，fsa4480与远程hub311设备关联，用于音频模式切换。 */
44             };
45         };
46     };
47 };
48 ps8743: ps8743@11 {
49     compatible = "parade,ps8743"; /* 设备匹配名，对应Type-C DP Switch芯片的型号。 */
```

```
50     reg = <0x11>;
51     hotplug = <&r_pio PL 5 GPIO_ACTIVE_HIGH>; /* Switch通知DP热插拔事件的io配置, 仅使用USB3.0可以不配置 */
52     aw,rx_eq_emphasis_quirk; /* USB3.0 RX端EQ参数是否需要调整。 */
53     aw,rx_eq_emphasis = <0x10>; /* USB3.0 RX端EQ参数调整值, 默认为2.4dB, 结合Spec配置为0x10, 表示需要调整
    为5dB。 */
54     status = "okay";
55
56     orientation-switch; /* 用于Type-C内核框架获取正反插方向的切换。 */
57     svid = <0xff01>; /* 用于识别配件模式altmode为DP。 */
58 #if 1
59     port {
60         #address-cells = <1>;
61         #size-cells = <0>;
62         usbdp_orientation_switch: endpoint@0 {
63             reg = <0>;
64             remote-endpoint = <&usb2_orien_sw>; /* 使用port描述, ps8743与远程hub311设备关联, 用于切换方
    向。 */
65         };
66         usbdp_dp_altmode_mux: endpoint@1 {
67             reg = <1>;
68             remote-endpoint = <&dp_altmode_mux>; /* 使用port描述, ps8743与远程hub311设备关联, 用于切换
    MUX。 */
69         };
70     };
71 #endif
72 };
73
74     hub311: hub311@4e {
75         compatible = "hynetek,hub311"; /* 设备匹配名, 对应Type-C PD控制器的型号。 */
76         reg = <0x4e>;
77         interrupt-parent = <&r_pio>;
78         hub311,intr_gpio = <&r_pio PL 13 GPIO_ACTIVE_LOW>; /* CC Logic的中断pin, 根据中断触发TCPM状态机。 */
79         /* interrupts = <PM 5 GPIO_ACTIVE_LOW>;可按照上述配置, 增加“wakeup-source;”支持HUSB311作为唤醒源 */
80         /* interrupt-names = "wakeup"; */
81         vbus-supply = <&reg_qc_drivevbus>; /* VBUS输出的开关, 根据CC状态通过regulator框架设置Vbus的开启和关闭。
    */
82         det_usb_supply = <&charger_power_supply>; /* 通过TCPC的ops在PD握手过程中进行电压和电流的设置, 实现快
    充等功能。 */
83         aw,vbus-debounce-quirk; 用于规避UGREEN的CD275适配器未接入电源时连接到HUSB311导致反复对外放电异
    常。 */
84         /* aw,vbus-tryon-debounce = <1200>; Vbus开启与上一次关闭的时间差。 */
85         /* aw,vbus-check-debounce = <600>; Vbus关闭与上一次开启的时间差。 */
86         /* aw,port-reset-quirk; 用于配置HUSB311唤醒SoC后是否进行复位。 */
87         wakeup-source;
88         status = "okay";
89
90     ports {
91         #address-cells = <1>;
92         #size-cells = <0>;
93
94         port@0 {
95             reg = <0>;
96             usbc2_role_sw: endpoint@0 {
97                 remote-endpoint = <&dwc3_0_role_switch>; /* 使用port描述, hub311与远程dwc3控制器关联, 用于
    实现OTG角色切换。 */
98             };
99         };
100     };
101
102     usb_con: connector {
```

```

103 compatible = "usb-c-connector"; /* 设备匹配名，USB连接器的型号，一般有A、B、C。 */
104 label = "USB-C"; /* 标签，标注当前USB口为C口。 */
105 data-role = "dual"; /* 数据传输方向：dual：双向，host/device：主机/从机。 */
106 power-role = "dual"; /* 充电/供电方向：dual：双向，sink/source：充电/放电。 */
107 try-power-role = "source"; /* 优先匹配充电/供电方向，sink/source：充电/放电；对于DP设备误识别，可配置
为“source”。 */
108 op-sink-microwatt = <1000000>; /* 最小充电功率。 */
109 // typec-power-opmode = "default";
110 // pd-disable;
111 slow-charger-loop; /* 匹配档位时，是否从最低默认档位开始。 */
112 sink-pdos = < PDO_FIXED(5000, 500, (0x1<<29)|(0x1<<25))
113         PDO_VAR(5000, 5000, 2000)
114         PDO_VAR(9000, 9000, 2000)
115         /* PDO_VAR(12000, 12000, 3000)
116         PDO_VAR(15000, 15000, 3000)
117         PDO_VAR(20000, 20000, 2250) */
118         >; /* 配置充电档位，例子中配置的档位如下：默认档：5V 0.5A，档1：5V 2A，档2：9V 2A。 */
119 source-pdos = < PDO_FIXED(5000, 500, (0x1<<29)|(0x1<<25))
120         PDO_VAR(5000, 5000, 1200) >; /* 配置放电档位，例子中配置的档位如下：默认档：5V 0.5A，档
：5V 1.2A。 */
121 /* fixme */
122 sink-vdos = <0xd1002e99 0x00 0x03110000>; /* 定义VDO消息，使TCPM框架可以获取caps。 */
123 sink-vdos-v1 = <0xd1002e99 0x00 0x03110000>; /* 定义VDO v1消息。 */
124
125 altmodes {
126     #address-cells = <1>;
127     #size-cells = <0>;
128
129     altmode@0 {
130         reg = <0>;
131         svid = <0xff01>; /* 用于识别配件模式altmode为DP。 */
132         vdo = <0xffffffff >; /* 定义VDO消息，用于displayport.c加载驱动。 */
133     };
134 };
135
136 ports {
137     #address-cells = <1>;
138     #size-cells = <0>;
139
140 #if 1
141     port@0 {
142         reg = <0>;
143         usb2_orien_sw: endpoint {
144             remote-endpoint = <&usbdp_orientation_switch>; /* 使用port描述，hub311与远程ps8743关
联，用于切换方向。 */
145         };
146     };
147     port@1 {
148         reg = <1>;
149         dp_altmode_mux: endpoint {
150             remote-endpoint = <&usbdp_dp_altmode_mux>; /* 使用port描述，hub311与远程ps8743设备关
联，用于切换MUX。 */
151         };
152     };
153     port@2 {
154         reg = <2>;
155         typec_sbu_out: endpoint {
156             remote-endpoint = <&fsa4480_sbu_mux>;
157         };
158     };

```

```
159 #else
160     port {
161         typec_sbu_out: endpoint {
162             remote-endpoint = <&fsa4480_sbu_mux>;
163         };
164     };
165 #endif
166 };
167 };
168 };
169 };
170
171 &twi6 {
172     qc0: qc@6b {
173         compatible = "eta,eta6973";
174         reg = <0x6b>;
175         status = "okay";
176         quick-charge,drive-vbus-en;
177         wakeup-source;
178         // watchdog_enable = <1>;
179
180         charger_power_supply: charger_power_supply {
181             compatible = "eta,eta6973-charger-power-supply";
182             status = "okay";
183             wakeup-source;
184             det_battery_supply = <&bat_power_supply>;
185             pmu_chargerpc_vol = <5000>;
186             pmu_chargerpc_cur = <500>;
187             pmu_chargerad_vol = <5000>;
188             pmu_chargerad_cur = <2400>;
189             battery_exist_sets = <0>;
190             qc_det_gpio = <&r_pio PL 4 IRQ_TYPE_LEVEL_LOW>;
191             qc_nce_gpio = <&r_pio PL 5 GPIO_PULL_DOWN>;
192             bat_state_gpio = <&r_pio PL 7 GPIO_ACTIVE_LOW>;
193         };
194         regulator4: regulators@4 {
195             reg_qc_drivevbus: drivevbus {
196                 regulator-name = "eta6973-drivevbus";
197                 regulator-enable-ramp-delay = <1000>;
198             };
199         };
200     };
201 };
```

2.5.3.3 注意事项

说明

- 若编译 PDO_FIXED 报错，在 dtsi 中添加 #include <dt-bindings/usb/pd.h>;
- PS8743 不要添加 accessory; 属性而要添加 svid = <0xff01>; 属性; BCT4480 要添加 accessory; 属性而不要添加 svid = <0xff01>; 属性。
- mode-switch: 用于 Type-C 框架注册模式切换端口;
- orientation-switch: 用于 Type-C 框架注册正反插端口;
- svid: 用于识别配件模式 altmode 为 DP;
- usb-role-switch: 用于 DWC3 框架识别是否通过 Type-C 进行 DRD 切换;
- vbus-supply: 此节点用于 TCPM 框架根据 CC 状态设置 Vbus 的开启和关闭;
- det_usb_supply: 此节点用于 TCPM 框架设置电压、电流，通过 PowerSupply 框架接口回调快充驱动的函数;
- sink-pdos/source-pdos: 此节点用于设置 charger 芯片支持的档位，如 9V2A、5V3A;
- qc0: 此节点的相关配置参考《Linux_快充_开发指南.pdf》，目前还支持 axp519。
- 若仅支持音频 Switch，将 #if 1 修改为 #if 0，并把 ps8743 节点配置为 “disabled” 即可，若仅支持 DP Switch，将 fsa4480 配置为 “disabled” 即可。
- 目前 hub311 的中断支持两种配置方式，若使用默认配置方式，则需要添加 wakeup-source 属性注册唤醒源，若使用 interrupt-names = “wakeup”；会自动通过 i2c 框架注册唤醒源。

2.5.3.4 相关配置

```

1  &xhci2 {
2      dr_mode = "otg";
3      usb-role-switch; /* 用于DWC3框架识别是否通过Type-C框架进行DRD切换。*/
4      /* usb-psy-name = "eta6973-charger";用于DWC3的核心层代码获取usb_psy，并给Gadget层vbus_draw接口设置限
5       流，还存在问题。*/
6      port {
7          #address-cells = <1>;
8          #size-cells = <0>;
9          dwc3_0_role_switch: endpoint@0 {
10             reg = <0>;
11             remote-endpoint = <&usb2_role_sw>; /* 使用port描述，dwc3控制器与远程hub311关联，用于实现OTG角色
12              切换。*/
13         };
14     };
15 };
16
17 &usb2 {
18     device_type = "usb2";
19     drvbus-supply = <&reg_drivevbus>;
20     aw,vbus-shared-quirk; /* 若配置此项，则drvbus-supply不生效。*/
21     status = "okay";
22 };
23
24 &xhci2 {
25     dr_mode = "host"; /* 控制器支持的主从模式，可选择的配置为：host, peripheral, otg。*/
26     status = "okay";
27 };
28
29 &u2phy {
30     status = "okay";
31 };
32
33 &combophy {

```

```
32 phy_use_sel = <1>; /* combophy的使用者配置, 0:PCIE; 1:USB3; 2:PCIE_USB3_U2 */
33 status = "okay";
34 };
35
36 &codec {
37     ...
38     extcon = <&fsa4480>; /* 使用内核extcon框架接收模拟音频的断开和连接, 以及耳机类型 (三节 or 四节) */
39     ...
40     status = "okay";
41 };
42
43 &codec_mach {
44     soundcard-mach,jack-support = <2>;
45     status = "okay";
46     ...
47 };
48
49 &edp0 {
50     compatible = "allwinner,sunxi-dp0";
51     edp_colordepth = <8>;
52     edp_color_fmt = <0>;
53     edp_lane_rate = <1>;
54     edp_lane_cnt = <4>;
55     status = "okay";
56     ...
57 };
```

2.5.3.5 注意事项

📖 说明

其他相关节点 `usbc2`、`xhci2`、`u2phy`、`combophy`、`codec`、`codec_mach`、`edp0` 参考相关配置章节进行配置即可。

- `drv_vbus-supply`: 可由 `vbus` 节点提供, 也可由 PMU 模块提供, 作为 host 时给外设供电, OTG 模式下推荐由 HUSB311 操作 `vbus`;
- `aw,vbus-shared-quirk`: 可选配置, 若配置此选项, 表示与 USB2 控制器驱动共用 `vbus`, USB3 控制器驱动不会操作 `vbus`;
- `dr_mode`: 表示将控制器配置的模式, 这里推荐配置为 `otg` 模式, `dr_mode option: host, peripheral, otg`;
- `phy_use_sel`: `PCIE_USB3_U2` (即配置为 2) 为新增模式, 用于 PCIE 和 USB3.0 同时工作的场景, 其中, PCIE 使用 `combophy`, XHCI 控制器使用 `U2-PHY`, 此时 USB3.0 只支持高速传输, 使用此模式时需要确认 `combophy` 驱动中已支持该项功能;
- `extcon`: 使用原生 `extcon` 框架, `fsa4480` 驱动在切换到音频模式后报告耳机类型和连接/断开信息给音频驱动, 确保 `CONFIG_EXTCON` 配置项已开启;
- `soundcard-mach,jack-support`: 这里需要配置为 2, 详见《Linux_Audio_开发指南.pdf》相关说明。
- `edp0` 节点的 `compatible`: 这里需要配置为 “`allwinner,sunxi-dp0`”, 用于配置 `edp` 控制器作为 `dp` 模式使用。详见《Linux_EDP_开发指南.pdf》或《AW_EDP_开发指南.pdf》相关说明。

2.5.3.6 扩展说明

很多设备的数据接口是通过子节点 `port` “来描述, 一个 `port` 的配置取决于参与到数据传输的其他设备通过 ‘`endpoint` 字节点’ 描述。

```
1 device {
2     ...
3     ports {
4         #address-cells = <1>;
5         #size-cells = <0>;
6
7         port@0 {
8             ...
9             endpoint@0 { ... };
10            endpoint@1 { ... };
11        };
12        port@1 { ... };
13    };
14};
```

- 如果一个 port 配置到和同一总线的多个远程设备工作，则必须为它们每一个提供一个“endpoint”子节点，如果在一个设备节点上存在多个 port 或者在一个 port 上有多个 endpoint 或者 port 节点需要与一个选定的硬件接口联系，通常的方案是使用 #address-cells #size-cells 和 reg 属性。
- 所有的 port 可以分组到 ports 节点下，允许 #address-cells、#size-cells 属性标识 port、endpoint 节点和一个设备可能含有的所有子节点。
- 两个 endpoint 节点通过“remote-endpoint”句柄彼此联系。设备的 endpoint 子节点包含对于这个设备和其他设备交换数据所有需要的属性。
- 允许一个 port 下的多个 endpoint 处于激活状态。

2.6 源码说明

2.6.1 源码路径

Type-C 源码的驱动位于 bsp/drivers/usb/typec 目录下。

2.6.2 源码结构

Type-C 整体框架基于内核 TCPM 进行实现，因此，代码结构与内核保持一致。

```
1 drivers/usb/typec/
2 |-- Kconfig
3 |-- Makefile
4 |-- mux
5 |   |-- Kconfig
6 |   |-- Makefile
7 |   |-- fsa4480.c
8 |   |-- gpio-gma340-mux.c
9 |   |-- ps8743.c
10 |   |-- sunxi-phy-switcher.c
11 |-- tcpm
```

```
12 | Kconfig
13 | Makefile
14 | -- tcpci_husb311.c
```

2.6.3 注意事项

说明

- **mux**: Type-C 接口 Switch 芯片源码，可以向内核 mux 框架注册 switch 和 mux 操作，用于进行 Type-C 的正反插和配件模式 Accessory Mode、Altmode 切换。
USB2.0 和模拟音频 Switch 已支持 fsa4480 驱动，用于 USB 和 Audio 的 Accessory Mode 配件模式切换，同型号的 BCT4480、ASW5480 已验证兼容性。
USB3.0 和 EDP1.3 Switch 已支持 ps8743 驱动（注意，某些平台可以使用 sunxi-phy-switcher 驱动代替外挂 Switch），用于 USB 和 DP 的 Altmode 模式切换。
- **tcpm**: Type-C 接口 CC Logic 芯片源码，初始化 CC 芯片，用于向 tcpci 框架注册回调实现 Vendor 操作。
CC Logic 以及 PD 3.0 协议已支持 husb311 驱动，用于 CC 状态检测和 PD 握手，实现 USB2.0/3.0 DRD、快充、DP-OUT 以及 Audio 模式的切换。

2.6.4 软件设计

Type-C 驱动按照 TCPCI 接口协议进行软件架构分层设计，HUSB311 作为 Type-C 端口控制器，是实现 Type-C 完整功能的核心和中枢，在 Type-C 设备插拔以及模式切换、PD 握手过程中发挥重要作用。

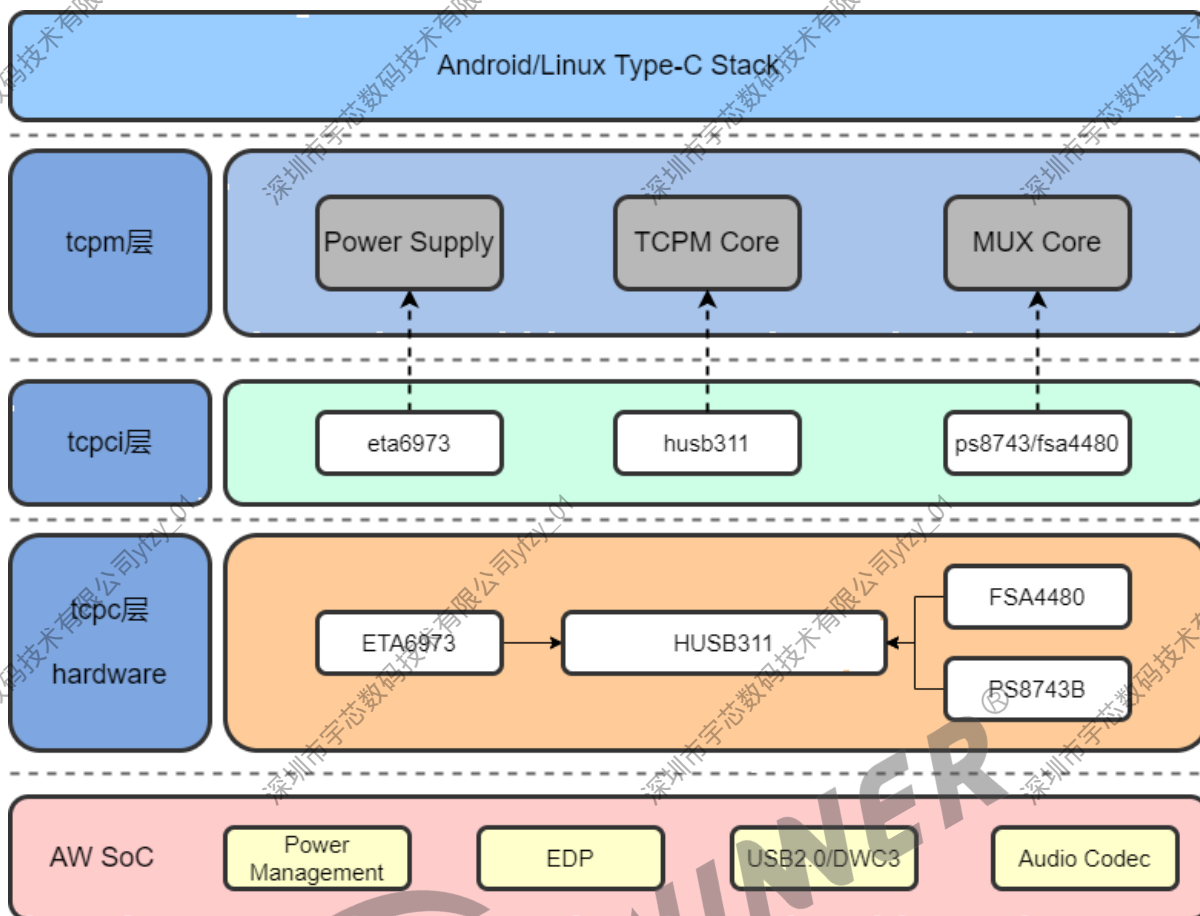


图 2-9: Type-C 框架图

3 日志分析

本章节主要提供 kernel 阶段初始化时相关的日志，以供分析和参考。不同的平台和不同版本的 kernel 之间可能会有轻微差异，但是大致流程是不变的。

以下按照初始化的时间顺序来展示 USB3.0、快充、DP-OUT、Audio 相关的 log。

3.1 驱动加载日志分析

- Combo Phy 驱动加载后，初始化 PCIE 和 USB3 共用的 Sub System，打印如下。

```
[ 2.230944] inno-combphy 4f00000.phy: Sub System Version: 0x10000
```

```
[ 2.015468] sunxi-cadence-combophy 6c00000.serdes: Sub System Version v2.0.0
```

- USB2.0 Phy 驱动加载后，用于初始化 phy，打印如下。

```
[ 2.727800] sunxi-plat-phy 4e00000.phy: phy provider register success
```

- DWC3 平台驱动注册，当内核 core 层驱动还未注册并初始化控制器时将返回 EPROBE_DEFER，之后将通过框架重新 probe；

hub311 在通过 tcpci_register_port 注册 port 时，core 层还未注册 usb_role_switch，将返回错误码。

当依赖的资源加载后，内核框架会重新尝试 probe 对应的驱动。

axp2202 会识别为快充，fsa4480 根据 DeviceID 识别到当前贴片为 BCT4480，ps8743 初始化 eq 参数并加载成功，hub311 识别到电池存在并加载成功。

```
[ 3.609238] i2c 5-004e: Fixing up cyclic dependency with 4d00000.xhci2-controller
[ 3.617800] I2C functionality : OK...
[ 3.621916] hub311 5-004e: no power_gpios info, ret = -2
[ 3.628635] sunxi:twi-2503400.twi5:[INFO]: v2.7.2 probe success
[ 3.663060] sunxi:dwc3_sunxi_plat:[INFO]: get usb_detect_type is fail, 22
[ 3.670691] sunxi:dwc3_sunxi_plat:[INFO]: get usb_detect_mode is fail, 22
[ 3.678309] sunxi:dwc3_sunxi_plat:[INFO]: get gma340-oe is fail, 22
[ 3.685335] sunxi:dwc3_sunxi_plat:[INFO]: get gma340-sel is fail, 22
[ 3.693044] sunxi:dwc3_sunxi_plat-12.usbc2:[INFO]: DesignWare USB3 Allwinner Glue Layer Driver(v1.0.15 2024-03-14
18:30)
[ 3.705567] sunxi:axp2202_battery:[INFO]: axp2202:used quick charge
[ 3.714262] sunxi:axp2202_battery-axp2202-bat-power-supply.0:[ERR]: axp2202 model need update!
[ 4.803535] sunxi:axp2202_battery:[INFO]: radio:100
```

```
[ 4.803863] sunxi:axp2202_battery:[WARN]: bat_vol:4699
[ 4.803863]
[ 4.817129] thermal thermal_zones5: power_allocator: sustainable_power will be estimated
[ 4.829343] axp2202-bat-power-supply axp2202-bat-power-supply.0: IRQ charge_start not found
[ 4.838749] axp2202-bat-power-supply axp2202-bat-power-supply.0: IRQ charge_done not found
[ 4.848020] axp2202-bat-power-supply axp2202-bat-power-supply.0: IRQ SOC_low_warning1 not found
[ 4.857775] axp2202-bat-power-supply axp2202-bat-power-supply.0: IRQ SOC_low_warning2 not found
[ 4.877308] I2C functionality : OK...
[ 4.881427] hub311 5-004e: no power_gpios info, ret = -2
[ 4.888767] hub311 5-004e: fail to register tcpci port, -517
[ 4.898859] fsa4480 5-0043: Found BCT4480 v1.1 (Vendor ID = 0)
[ 4.907676] ps8743 5-0011: setting rx_eq_emphasis: 0x10
[ 4.913548] ps8743 5-0011: Revision ID:0x0b00, Chip ID:0x8741, probe success
[ 4.921496] I2C functionality : OK...
[ 4.925602] hub311 5-004e: no power_gpios info, ret = -2
[ 4.934966] hub311 5-004e: vbus is already Off
[ 4.941363] ps8743 5-0011: orientation:0, set sw conf 0x00 -> 0xa8
[ 4.948435] ps8743 5-0011: orientation:0, set sw conf 0xb8 -> 0xa8
[ 4.955366] fsa4480 5-0043: set sw conf 0 -> 0 [NORMAL]
[ 4.961257] fsa4480 5-0043: set mux conf 0 -> 0 [USB]
[ 4.967883] hub311 5-004e: battery exist: yes
[ 4.968092] hub311 5-004e: vbus is already Off
[ 4.973091] hub311 5-004e: Vendor ID:0x2e99, Product ID:0x0311, probe success
[ 4.978581] hub311 5-004e: Setting voltage/current limit 0 mV 0 mA
[ 4.986474] psci_checker: PSCI checker started using 8 CPUs
[ 4.993888] ps8743 5-0011: orientation:0, set sw conf 0xb8 -> 0xa8
[ 4.999333] psci_checker: Starting hotplug tests
[ 5.006231] ps8743 5-0011: orientation:0, set sw conf 0xb8 -> 0xa8
[ 5.011404] psci_checker: Trying to turn off and on again all CPUs
[ 5.018328] fsa4480 5-0043: set sw conf 0 -> 0 [NORMAL]
[ 5.031127] fsa4480 5-0043: set mux conf 0 -> 0 [USB]
.....
```

注意：hub311 和 ps8743 仅对于支持 OTG 模式的 USB3.0 的 DRD 需要加载，fsa4480 仅对于需要支持模拟音频设备需要加载。

3.2 Device 日志分析

- 当通过 Type-A 转 Type-C 或者 Type-C 转 Type-C 的线缆连接至 PC 时，会根据实际情况识别为 SuperSpeed 设备或者 HighSpeed 设备。

```
[ 5.271990] ps8743 5-0011: orientation:1, set sw conf 0xb8 -> 0xb8
[ 5.279443] hub311 5-004e: get current limit 2400 mA
[ 5.285150] hub311 5-004e: Setting voltage/current limit 5000 mV 500 mA
[ 5.292933] hub311 5-004e: vbus is already Off
[ 5.616387] ps8743 5-0011: orientation:1, set sw conf 0xb8 -> 0xb8
[ 6.150485] android_work: sent uevent USB_STATE=CONNECTED
[ 6.183556] android_work: sent uevent USB_STATE=CONFIGURED
[ 6.274151] hub311 5-004e: get current limit 2400 mA
[ 6.279886] hub311 5-004e: Setting voltage/current limit 5000 mV 500 mA
[ 6.601105] ps8743 5-0011: orientation:1, set sw conf 0xb8 -> 0xb8
[ 7.258892] hub311 5-004e: get current limit 2400 mA
[ 7.264620] hub311 5-004e: Setting voltage/current limit 5000 mV 500 mA
[ 7.583205] hub311 5-004e: get current limit 2400 mA
[ 7.588924] hub311 5-004e: Setting voltage/current limit 5000 mV 2400 mA
```

3.3 Host 日志分析

- 当使用 OTG 线（也可以是 Type-C 转 Type-C 的线缆连接外设）接入各种外设后，将会通过 Type-C 的 TCPM 框架加载 XHCI 主机控制器驱动。

连接不同的设备将会根据设备特性（SuperSpeed、HighSpeed、FullSpeed、LowSpeed）匹配总线并进行枚举，从以下 Log 中不难看出，xhci 控制器在初始化过程中同时注册总线 3 和总线 4，其中总线 3 是高速总线，用于向下兼容 USB2.0 外设，总线 4 是超高速总线，用于识别 USB3.0 外设。

```
[ 40.675111] hub311 5-004e: set vbus On
[ 40.679482] dwc3 4d00000.xhci2-controller: request (<__ptrval__>) was not queued to ep0out
[ 40.689028] android_work: sent uevent USB_STATE=DISCONNECTED
[ 40.696361] xhci-hcd xhci-hcd.24.auto: xHCI Host Controller
[ 40.702673] xhci-hcd xhci-hcd.24.auto: new USB bus registered, assigned bus number 3
[ 40.711542] xhci-hcd xhci-hcd.24.auto: hcc params 0x0110ffc4 hci version 0x110 quirks 0x0000000000010010
[ 40.722204] xhci-hcd xhci-hcd.24.auto: irq 511, io mem 0x04d00000
[ 40.729274] xhci-hcd xhci-hcd.24.auto: xHCI Host Controller
[ 40.735562] xhci-hcd xhci-hcd.24.auto: new USB bus registered, assigned bus number 4
[ 40.744294] xhci-hcd xhci-hcd.24.auto: Host supports USB 3.1 Enhanced SuperSpeed
[ 40.752705] usb usb3: New USB device found, idVendor=1d6b, idProduct=0002, bcdDevice= 5.15
[ 40.762015] usb usb3: New USB device strings: Mfr=3, Product=2, SerialNumber=1
[ 40.770149] usb usb3: Product: xHCI Host Controller
[ 40.775632] usb usb3: Manufacturer: Linux 5.15.119 xhci-hcd
[ 40.781896] usb usb3: SerialNumber: xhci-hcd.24.auto
[ 40.787819] hub 3-0:1.0: USB hub found
[ 40.792042] hub 3-0:1.0: 1 port detected
[ 40.796683] usb usb4: We don't know the algorithms for LPM for this host, disabling LPM.
[ 40.805820] usb usb4: New USB device found, idVendor=1d6b, idProduct=0003, bcdDevice= 5.15
[ 40.815095] usb usb4: New USB device strings: Mfr=3, Product=2, SerialNumber=1
[ 40.823202] usb usb4: Product: xHCI Host Controller
[ 40.828665] usb usb4: Manufacturer: Linux 5.15.119 xhci-hcd
[ 40.834911] usb usb4: SerialNumber: xhci-hcd.24.auto
[ 40.840765] hub 4-0:1.0: USB hub found
[ 40.844988] hub 4-0:1.0: 1 port detected
```

注意：需要说明的是，当配置为 Host 模式，在驱动加载过程中将会直接初始化 Host 控制器；配置为 Combo 模式，将会通过 AXP2202 的 CC 状态进行 Host 的动态加载和卸载。

- 当接入 3.1 的 U 盘时，会被识别为 SuperSpeedPlus 设备，并挂到总线 4 上。

```
[ 85.953376] usb 4-1: new SuperSpeed Plus Gen 2x1 USB device number 2 using xhci-hcd
[ 85.982285] usb 4-1: New USB device found, idVendor=090c, idProduct=2320, bcdDevice= 1.00
[ 85.992800] usb 4-1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[ 86.003661] usb 4-1: Product: 512GB thinkplus
[ 86.009711] usb 4-1: Manufacturer: Lenovo
[ 86.015268] usb 4-1: SerialNumber: AA00000000000000221
[ 86.032901] scsi host1: uas
[ 86.039349] scsi 1:0:0:0: Direct-Access  Lenovo 512GB thinkplus 1000 PQ: 0 ANSI: 6
[ 86.059094] sd 1:0:0:0: [sdb] 1000215216 512-byte logical blocks: (512 GB/477 GiB)
[ 86.072200] sd 1:0:0:0: [sdb] Write Protect is off
[ 86.079192] sd 1:0:0:0: [sdb] Mode Sense: 43 00 00 00
[ 86.091058] sd 1:0:0:0: [sdb] Write cache: enabled, read cache: enabled, doesn't support DPO or FUA
[ 86.107629] sdb: sdb1
```

```
[ 86.112338] sd 1:0:0:0: [sdb] Attached SCSI disk
```

- 当接入 3.0 的 U 盘时，会被识别为 SuperSpeed 设备，并挂到总线 4 上。

```
[ 41.183450] usb 4-1: new SuperSpeed USB device number 2 using xhci-hcd
[ 41.209077] usb 4-1: New USB device found, idVendor=152d, idProduct=0578, bcdDevice= 5.08
[ 41.218285] usb 4-1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[ 41.226292] usb 4-1: Product: USB to ATA/ATAPI Bridge
[ 41.231958] usb 4-1: Manufacturer: JMicron
[ 41.236545] usb 4-1: SerialNumber: 0123456789ABCDEF
[ 41.245801] scsi host0: uas
[ 41.249801] scsi 0:0:0:0: Direct-Access JMicron Generic 0508 PQ: 0 ANSI: 6
[ 42.855529] sd 0:0:0:0: [sda] 250069680 512-byte logical blocks: (128 GB/119 GiB)
[ 42.863956] sd 0:0:0:0: [sda] 4096-byte physical blocks
[ 42.870146] sd 0:0:0:0: [sda] Write Protect is off
[ 42.875563] sd 0:0:0:0: [sda] Mode Sense: 53 00 00 08
[ 42.881591] sd 0:0:0:0: [sda] Disabling FUA
[ 42.886280] sd 0:0:0:0: [sda] Write cache: enabled, read cache: enabled, doesn't support DPO or FUA
[ 42.896756] sd 0:0:0:0: [sda] Optimal transfer size 33553920 bytes not a multiple of physical block size (4096 bytes)
[ 43.254334] sda: sda1
[ 43.259027] sd 0:0:0:0: [sda] Attached SCSI disk
```

- 当接入 2.0 的 U 盘时，会被识别为 HighSpeed 设备，并挂到总线 3 上。

```
[ 52.411037] usb 3-1: new high-speed USB device number 2 using xhci-hcd
[ 52.563927] usb 3-1: New USB device found, idVendor=23a9, idProduct=ef18, bcdDevice= 1.00
[ 52.573140] usb 3-1: New USB device strings: Mfr=1, Product=2, SerialNumber=0
[ 52.581162] usb 3-1: Product: Teclast CoolFlash
[ 52.586243] usb 3-1: Manufacturer: Generic
[ 52.591466] usb-storage 3-1:1.0: USB Mass Storage device detected
[ 52.598908] scsi host0: usb-storage 3-1:1.0
[ 53.631814] scsi 0:0:0:0: Direct-Access Teclast CoolFlash 1.00 PQ: 0 ANSI: 4
[ 53.641578] sd 0:0:0:0: [sda] 61472960 512-byte logical blocks: (31.5 GB/29.3 GiB)
[ 53.650391] sd 0:0:0:0: [sda] Write Protect is off
[ 53.655777] sd 0:0:0:0: [sda] Mode Sense: 03 00 00 00
[ 53.661578] sd 0:0:0:0: [sda] No Caching mode page found
[ 53.667551] sd 0:0:0:0: [sda] Assuming drive cache: write through
[ 53.697096] sda: sda1
[ 53.701188] sd 0:0:0:0: [sda] Attached SCSI removable disk
```

- 当接入全速的 U 盘时，会被识别为 FullSpeed 设备，并挂到总线 3 上。

```
[ 171.162478] usb 3-1: new full-speed USB device number 5 using xhci-hcd
[ 171.314932] usb 3-1: New USB device found, idVendor=0c76, idProduct=0003, bcdDevice= 1.00
[ 171.324122] usb 3-1: New USB device strings: Mfr=0, Product=0, SerialNumber=0
[ 171.338763] usb-storage 3-1:1.0: USB Mass Storage device detected
[ 171.346040] scsi host0: usb-storage 3-1:1.0
[ 172.475360] scsi host0: scsi scan: INQUIRY result too short (5), using 36
[ 172.483005] scsi 0:0:0:0: Direct-Access USB FLASH DISK 1.00 PQ: 0 ANSI: 2
[ 172.494891] sd 0:0:0:0: [sda] 512000 512-byte logical blocks: (262 MB/250 MiB)
[ 172.503734] sd 0:0:0:0: [sda] Write Protect is off
[ 172.509117] sd 0:0:0:0: [sda] Mode Sense: 0b 00 00 08
[ 172.515687] sd 0:0:0:0: [sda] No Caching mode page found
[ 172.521673] sd 0:0:0:0: [sda] Assuming drive cache: write through
```

```
[ 172.699465] sda: sda1
[172.705241] sd 0:0:0:0: [sda] Attached SCSI removable disk
```

- 当接入低速鼠标或者键盘时，会被识别为 LowSpeed 设备，并挂到总线 3 上。

```
[ 254.907157] usb 3-1: new low-speed USB device number 4 using xhci-hcd
[ 255.066800] usb 3-1: New USB device found, idVendor=18f8, idProduct=1286, bcdDevice= 1.00
[ 255.076019] usb 3-1: New USB device strings: Mfr=0, Product=1, SerialNumber=0
[ 255.084027] usb 3-1: Product: USB GAMING MOUSE
[ 255.100965] input: USB GAMING MOUSE as /devices/platform/soc@3000000/12.usbc2/4d00000.xhci2-controller/xhci-hcd.24.auto/usb3/3-1/3-1:1.0/0003:18F8:1286.0003/input/input5
[ 255.175173] hid-generic 0003:18F8:1286.0003: input: USB HID v1.10 Mouse [USB GAMING MOUSE] on usb-xhci-hcd.24.auto-1/input0
[ 255.191380] input: USB GAMING MOUSE Keyboard as /devices/platform/soc@3000000/12.usbc2/4d00000.xhci2-controller/xhci-hcd.24.auto/usb3/3-1/3-1:1.1/0003:18F8:1286.0004/input/input6
[ 255.267415] hid-generic 0003:18F8:1286.0004: input: USB HID v1.10 Keyboard [USB GAMING MOUSE] on usb-xhci-hcd.24.auto-1/input1
```

- 当断开 OTG 线后，将会通过 Type-C 的 TCPM 框架卸载 XHCI 主机控制器驱动，并释放已分配的总线号。

```
[ 444.969601] usb 3-1: USB disconnect, device number 4
[ 444.976461] hub311 5-004e: set vbus Off
[ 444.981545] hub311 5-004e: Setting voltage/current limit 0 mV 0 mA
[ 444.989186] ps8743 5-0011: orientation:0, set sw conf 0xb8 -> 0xa8
[ 445.010038] xhci-hcd xhci-hcd.24.auto: remove, state 4
[ 445.015830] usb usb4: USB disconnect, device number 1
[ 445.022216] xhci-hcd xhci-hcd.24.auto: USB bus 4 deregistered
[ 445.037688] xhci-hcd xhci-hcd.24.auto: remove, state 1
[ 445.043493] usb usb3: USB disconnect, device number 1
[ 445.115314] xhci-hcd xhci-hcd.24.auto: USB bus 3 deregistered
```

3.4 快充日志分析

- 当连接快充适配器时，从最低档位开始匹配电压和电流；在与适配器进行 PD 协商成功后，将会设置为匹配的电压和限流。

```
[ 704.105459] ps8743 5-0011: orientation:1, set sw conf 0xb8 -> 0xb8
[ 704.112383] ps8743 5-0011: orientation:1, set sw conf 0xb8 -> 0xb8
[ 704.119281] fsa4480 5-0043: set sw conf 0 -> 1 [NORMAL]
[ 704.125438] fsa4480 5-0043: set mux conf 0 -> 1 [USB]
[ 704.131862] hub311 5-004e: Setting voltage/current limit 5000 mV 500 mA
[ 704.139839] hub311 5-004e: vbus is already Off
[ 704.275386] hub311 5-004e: Setting voltage/current limit 9000 mV 2000 mA
```

- 当断开适配器时，关闭 Vbus 并设置限流为 0。

```
[ 858.247891] husb311 5-004e: vbus is already Off
[ 858.263204] husb311 5-004e: Setting voltage/current limit 0 mV 0 mA
[ 858.280516] ps8743 5-0011: orientation:0, set sw conf 0xb8 -> 0xa8
[ 858.287439] ps8743 5-0011: orientation:0, set sw conf 0xb8 -> 0xa8
[ 858.294347] fsa4480 5-0043: set sw conf 1 -> 0 [NORMAL]
[ 858.300415] fsa4480 5-0043: set mux conf 1 -> 0 [USB]
```

3.5 Type-C DP OUT 日志分析

当 DisplayPort 显示器接入时，对应流程为：

- CC logic 芯片能顺利完成 PD 协商，切换为 DisplayPort Alternate Mode。
- 通过 switch IC 完成数据 lane 的使能和 remap。DisplayPort Alternate Mode 对应的 switch 有多种模式可以切换，视实际显示器的支持情况而定，分别为：0xe8（4lane DP 正插）、0xec（4lane DP 反插）、0xf8（2lane DP + USB 正插）、0xfc（2lane DP + USB 反插）。
- 拉起 hotplug gpio，触发 edp 驱动的热插拔信号。
- edp 显示驱动完成热插拔上报，显示中间件 HWC 打开及绑定显示设备，成功绑定显示设备后（attached ok）安卓显示框架开始送显。

```
[ 394.721817][ T433] ps8743 5-0011: orientation:2, set sw conf 0xbc -> 0xbc
[ 394.735015][ T225] dwc3 4d00000.xhci2-controller: request 0000000000000000 was not queued to ep0out
[ 394.746338][ T2238] android_work: did not send uevent (0 0 0000000000000000)
[ 394.762548][ T225] xhci-hcd xhci-hcd.24.auto: xHCI Host Controller
[ 394.790974][ T225] xhci-hcd xhci-hcd.24.auto: new USB bus registered, assigned bus number 3
[ 394.801497][ T225] xhci-hcd xhci-hcd.24.auto: hcc params 0x0110ffc4 hci version 0x110 quirks 0x000000000010010
[ 394.819829][ T225] xhci-hcd xhci-hcd.24.auto: irq 492, io mem 0x04d00000
[ 394.827815][ T225] xhci-hcd xhci-hcd.24.auto: xHCI Host Controller
[ 394.834839][ T225] xhci-hcd xhci-hcd.24.auto: new USB bus registered, assigned bus number 4
[ 394.844285][ T225] xhci-hcd xhci-hcd.24.auto: Host supports USB 3.1 Enhanced SuperSpeed
[ 394.853521][ T225] usb usb3: New USB device found, idVendor=1d6b, idProduct=0002, bcdDevice= 5.15
[ 394.863546][ T225] usb usb3: New USB device strings: Mfr=3, Product=2, SerialNumber=1
[ 394.872401][ T225] usb usb3: Product: xHCI Host Controller
[ 394.878631][ T225] usb usb3: Manufacturer: Linux 5.15.119-tablet-a523-android14-v1.0rc4 xhci-hcd
[ 394.888545][ T225] usb usb3: SerialNumber: xhci-hcd.24.auto
[ 394.896137][ T225] hub 3-0:1.0: USB hub found
[ 394.901210][ T225] hub 3-0:1.0: 1 port detected
[ 394.907607][ T225] usb usb4: We don't know the algorithms for LPM for this host, disabling LPM.
[ 394.917640][ T225] usb usb4: New USB device found, idVendor=1d6b, idProduct=0003, bcdDevice= 5.15
[ 394.927695][ T225] usb usb4: New USB device strings: Mfr=3, Product=2, SerialNumber=1
[ 394.936544][ T225] usb usb4: Product: xHCI Host Controller
[ 394.942773][ T225] usb usb4: Manufacturer: Linux 5.15.119-tablet-a523-android14-v1.0rc4 xhci-hcd
[ 394.952685][ T225] usb usb4: SerialNumber: xhci-hcd.24.auto
[ 394.960200][ T2238] ps8743 5-0011: mode:0, set mux conf 0xbc -> 0xac
[ 394.967775][ T225] hub 4-0:1.0: USB hub found
[ 394.972920][ T225] hub 4-0:1.0: 1 port detected
[ 394.986879][ T225] ps8743 5-0011: mode:0, set mux conf 0xac -> 0xac
[ 394.998430][ T433] ps8743 5-0011: mode:4, set mux conf 0xac -> 0xec
[ 395.429947][ T227] type=1400 audit(1705646316.528:282): avc: denied { search } for comm="MQ-Thread" name="
  exp2202-usb-power-supply.0" dev="sysfs" ino=38249 scontext=u:r:hal_graphics_composer_default:s0 tcontext=u:
  object_r:sysfs_batteryinfo:s0 tclass=dir permissive=0
[ 395.430376][ T410] sunxi:edp2:[ERR]: [EDP-ERR]: no standard mode found!
[ 395.466103][ T410] sunxi:disp:[WARN]: [DE]: edp_panel_fun[0].cfg_close_flow is NULL
[ 395.475071][ T410] sunxi:disp:[WARN]: [DE]: de_smb_l_enable_ahb_read=0
```

```
[ 395.485543][ T410] sunxi:disp:[WARN]:[DE]: de_dci_enable_ahb_read=0
[ 395.502973][ T410] sunxi:disp:[WARN]:[DE]: pwm device hdl is NULL
[ 395.510143][ T410] sunxi:disp:[WARN]:[DE]: edp_panel_fun[0].cfg_open_flow is NULL
[ 395.519892][ T410] sunxi:disp:[WARN]:[DE]: irq for 1 already registered
[ 395.527507][ T410] sunxi:disp:[WARN]:[DE]: not expected val=f500fd0, disp=1
[ 395.536465][ T410] sunxi:disp:[WARN]:[DE]: tcon 3
[ 395.544895][ T410] sunxi:disp:[WARN]:[DE]: de_smb_l_enable_ahb_read=1
[ 395.552188][ T410] sunxi:disp:[WARN]:[DE]: de_dci_enable_ahb_read=1
[ 395.608205][ T410] sunxi:disp:[WARN]:[DE]: disp_sys_pwm_Config, handle is NULL
[ 395.616511][ T410] sunxi:disp:[WARN]:[DE]: disp_sys_pwm_Set_Polarity, handle is NULL
[ 395.625408][ T410] sunxi:disp:[WARN]:[DE]: pwm device hdl is NULL
[ 395.643289][ T410] sunxi:disp:[WARN]:[DE]: attached ok, mgr1<->dev1
[ 395.651111][ T410] sunxi:disp:[WARN]:[DE]: type:32,mode:10,fmt:rgb,bits:8bits,eotf:4,cs:0 dvi_hdmi:2, range:2 scan:0
ratio:8
[ 395.999438][ T433] ps8743 5-0011: mode:4, set mux conf 0xec -> 0xec
[ 397.008739][ T433] ps8743 5-0011: mode:4, set mux conf 0xec -> 0xec
[ 398.017425][ T433] ps8743 5-0011: mode:4, set mux conf 0xec -> 0xec
[ 399.028396][ T433] ps8743 5-0011: mode:4, set mux conf 0xec -> 0xec
```

- 在支持 phy_switcher 的平台，接入显示器或者带有 DP 功能的 HUB 时，将会打印其 Type-C 接口的类型，目前支持的接口类型有 **STATE_DP_C**、**STATE_DP_D** 和 **STATE_DP_E**。

```
[ 161.177102][ T672] switcher 10.phy_switcher: conf:0x0 status:0x0 mode:1, set mux state STATE_SAFE -> STATE_USB
[ 161.188748][ T672] switcher 10.phy_switcher: conf:0x0 status:0x0 mode:1, set mux state STATE_USB -> STATE_USB
[ 161.202963][ T672] hub311 13-004e: set vbus On
[ 161.493638][ T509] switcher 10.phy_switcher: conf:0x0 status:0x0 mode:0, set mux state STATE_USB -> STATE_SAFE
[ 161.505500][ T509] switcher 10.phy_switcher: conf:0x0 status:0x0 mode:0, set mux state STATE_SAFE -> STATE_SAFE
[ 161.600997][ T509] switcher 10.phy_switcher: conf:0x806 status:0x1a mode:0, set mux state STATE_SAFE ->
STATE_SAFE
[ 161.643146][ T509] switcher 10.phy_switcher: conf:0x806 status:0x1a mode:0, set mux state STATE_SAFE ->
STATE_SAFE
[ 161.677499][ T672] switcher 10.phy_switcher: conf:0x806 status:0x1a mode:5, set mux state STATE_SAFE ->
STATE_DP_D
[ 161.763470][ T672] switcher 10.phy_switcher: conf:0x806 status:0x1a mode:5, set mux state STATE_DP_D ->
STATE_DP_D
```

- 当 DisplayPort 显示器断开连接时，关闭 Vbus 和限流，并将模式恢复为 USB，显示接收热插拔出事件。

```
[ 677.572578][ T433] ps8743 5-0011: mode:0, set mux conf 0xec -> 0xac
[ 677.581201][ T336] healthd: battery none chg=
[ 677.583530][ T193] selinux: SELinux: Could not stat /sys/devices/platform/soc@3000000/2503400.twi5/i2c-5/5-004e/
typec/port1/port1-partner/port1-partner.0: No such file or directory.
[ 677.583987][ T433] hub311 5-004e: vbus is already Off
[ 677.585404][ T433] ps8743 5-0011: orientation:0, set sw conf 0xac -> 0xa8
[ 677.586547][ T225] xhci-hcd xhci-hcd.24.auto: remove, state 4
[ 677.604686][ T193] selinux:
[ 677.612184][ T225] usb usb4: USB disconnect, device number 1
[ 677.632345][ T225] xhci-hcd xhci-hcd.24.auto: USB bus 4 deregistered
[ 677.640830][ T225] xhci-hcd xhci-hcd.24.auto: remove, state 4
[ 677.693374][ T225] usb usb3: USB disconnect, device number 1
[ 677.705056][ T225] xhci-hcd xhci-hcd.24.auto: USB bus 3 deregistered
[ 678.345418][ T227] type=1400 audit(1705646599.444:283): avc: denied { search } for comm="MQ-Thread" name="
axp2202-usb-power-supply.0" dev="sysfs" ino=38249 scontext=u:r:hal_graphics_composer_default:s0 tcontext=u:
object_r:sysfs_batteryinfo:s0 tclass=dir permissive=0
```

3.6 Audio 日志分析

- 当接入 Type-C 模拟耳机时，FSA4480 被 TCPM 配置为 Audio 模式，FSA4480 将连接状态和音频设备类型通知 Audio 驱动，同时加载 USB Host 控制器驱动。

```
[ 13.129853] ps8743 5-0011: orientation:1, set sw conf 0xb8 -> 0xb8
[ 13.136813] ps8743 5-0011: orientation:1, set sw conf 0xb8 -> 0xb8
[ 13.143751] fsa4480 5-0043: set sw conf 0 -> 1 [NORMAL]
[ 13.149913] fsa4480 5-0043: set mux conf 0 -> 5 [Audio]
[ 13.157852] dwc3 4d00000.xhci2-controller: request (____ptrval____) was not queued to ep0out
[ 13.167355] android_work: did not send uevent (0 0 0000000000000000)
[ 13.174593] sunxi:sound-jack:[INFO]: 52 sunxi_jack_plugin_notifier(): event -> 1
[ 13.182866] sunxi:sound-codec:[WARN]: 1633 sunxi_jack_extcon_irq_clean(): unsupport jack irq type
[ 13.193649] xhci-hcd xhci-hcd.24.auto: xHCI Host Controller
[ 13.199899] xhci-hcd xhci-hcd.24.auto: new USB bus registered, assigned bus number 3
[ 13.208670] xhci-hcd xhci-hcd.24.auto: hcc params 0x0110ffc4 hci version 0x110 quirks 0x0000000000010010
[ 13.219285] xhci-hcd xhci-hcd.24.auto: irq 510, io mem 0x04d00000
[ 13.226161] xhci-hcd xhci-hcd.24.auto: xHCI Host Controller
[ 13.232386] xhci-hcd xhci-hcd.24.auto: new USB bus registered, assigned bus number 4
[ 13.241036] xhci-hcd xhci-hcd.24.auto: Host supports USB 3.1 Enhanced SuperSpeed
[ 13.249367] usb usb3: New USB device found, idVendor=1d6b, idProduct=0002, bcdDevice=5.15
[ 13.258597] usb usb3: New USB device strings: Mfr=3, Product=2, SerialNumber=1
[ 13.266711] usb usb3: Product: xHCI Host Controller
[ 13.272161] usb usb3: Manufacturer: Linux 5.15.137 xhci-hcd
[ 13.278374] usb usb3: SerialNumber: xhci-hcd.24.auto
[ 13.284020] sunxi:sound-jack:[INFO]: 192 sunxi_jack_det_scan_work(): 4 pole SBU2 to MIC, SBU1 to Audio Ground [8]
[ 13.295478] sunxi:sound-jack:[ERR]: 330 sunxi_jack_typec_mode_set(): modes map or jack pins is NULL
[ 13.305713] hub 3-0:1.0: USB hub found
[ 13.309917] hub 3-0:1.0: 1 port detected
[ 13.314505] usb usb4: We don't know the algorithms for LPM for this host, disabling LPM.
[ 13.314597] sunxi:sound-mach:[WARN]: 367 asoc_simple_parse_ucfmt(): set data late to default
[ 13.323559] sunxi:sound-jack:[ERR]: 330 sunxi_jack_typec_mode_set(): modes map or jack pins is NULL
[ 13.332962] sunxi:sound-mach:[ERR]: 486 simple_parse_of(): simple_dai_link_of failed
[ 13.343101] usb usb4: New USB device found, idVendor=1d6b, idProduct=0003, bcdDevice=5.15
[ 13.351816] sunxi:sound-mach:[WARN]: 367 asoc_simple_parse_ucfmt(): set data late to default
[ 13.360941] usb usb4: New USB device strings: Mfr=3, Product=2, SerialNumber=1
[ 13.370365] sunxi:sound-mach:[ERR]: 486 simple_parse_of(): simple_dai_link_of failed
[ 13.378428] usb usb4: Product: xHCI Host Controller
[ 13.392510] usb usb4: Manufacturer: Linux 5.15.137 xhci-hcd
[ 13.398733] usb usb4: SerialNumber: xhci-hcd.24.auto
[ 13.404592] hub 4-0:1.0: USB hub found
[ 13.408796] hub 4-0:1.0: 1 port detected
[ 13.413385] sunxi:sound-mach:[WARN]: 367 asoc_simple_parse_ucfmt(): set data late to default
[ 13.422821] sunxi:sound-mach:[ERR]: 486 simple_parse_of(): simple_dai_link_of failed
[ 13.431568] sunxi:sound-mach:[WARN]: 367 asoc_simple_parse_ucfmt(): set data late to default
[ 13.441001] sunxi:sound-mach:[ERR]: 486 simple_parse_of(): simple_dai_link_of failed
[ 13.449739] sunxi:sound-mach:[WARN]: 367 asoc_simple_parse_ucfmt(): set data late to default
[ 13.459170] sunxi:sound-mach:[ERR]: 486 simple_parse_of(): simple_dai_link_of failed
[ 13.466466] [sound] jack report -> HEADSET
```

- 当模拟耳机断开连接时，关闭 Vbus 和限流，并将模式恢复为 USB，FSA4480 通知音频驱动设备已断开。

```
[ 528.380859] husb311 5-004e: vbus is already Off
[ 528.386871] husb311 5-004e: Setting voltage/current limit 0 mV 0 mA
[ 528.395213] ps8743 5-0011: orientation:0, set sw conf 0xb8 -> 0xa8
```

```
[ 528.402137] ps8743 5-0011: orientation:0, set sw conf 0xb8 -> 0xa8
[ 528.409042] fsa4480 5-0043: set sw conf 1 -> 0 [NORMAL]
[ 528.415388] fsa4480 5-0043: set mux conf 5 -> 0 [USB]
[ 528.421151] xhci-hcd xhci-hcd.24.auto: remove, state 4
[ 528.426919] sunxi:sound-jack:[INFO]: 52 sunxi_jack_plugin_notifier(): event -> 0
[ 528.435181] sunxi:sound-codec:[WARN]: 1633 sunxi_jack_extcon_irq_clean(): unsupport jack irq type
[ 528.445116] usb usb4: USB disconnect, device number 1
[ 528.451370] xhci-hcd xhci-hcd.24.auto: USB bus 4 deregistered
[ 528.457820] xhci-hcd xhci-hcd.24.auto: remove, state 4
[ 528.463616] usb usb3: USB disconnect, device number 1
[ 528.469578] sunxi:sound-jack:[INFO]: 195 sunxi_jack_det_scan_work(): audio disconnect [0]
[ 528.478814] sunxi:sound-jack:[ERR]: 330 sunxi_jack_typec_mode_set(): modes map or jack pins is NULL
[ 528.489211] xhci-hcd xhci-hcd.24.auto: USB bus 3 deregistered
[ 528.495692] [sound] jack report -> OUT
```



4 Type-C 调试方法

4.1 FSA4480 调试

BCT4480 支持通过外部 IO 设置 TWI 设备地址，ADDR 高时地址为 **0x43**，ADDR 低时地址为 **0x42**。

模拟耳机以及数字耳机测试方法如下：

- 将 audiocodec 模块的 menuconfig 配置打开，目的是加载 audiocodec 声卡；
- 开机后输入 `cat /proc/asound/cards` 确认声卡是否顺利加载，成功的话会显示 audiocodec 声卡；
- 推工具到小机端（tinymix、tinyplay、tinycap），以及 wav 音频文件，相关测试文件位于以下路径：

Linux_Type-C_ 开发指南/tools/audio

- 执行命令：tinymix 用于切换控件，tinyplay 用于播发音频文件，tinycap 用于 mic 录音，测试时使用 tinymix 把 mic3 switch 打开；

```
./tinymix 26 1
./tinyplay ***.wav
./tinycap ***wav -D 0 -c 1
```

- 确认耳机能播放音乐。

4.2 PS8743 调试

外挂 Type-C 的 Switch 芯片较为简单，仅需按照如下配置即可完成正反插和配件模式 AltMode 的切换。

1	4-Lane DP(正插)	0x00h <bit7:2>	0xe8	11101000
2	4-Lane DP(反插)	0x00h <bit7:2>	0xec	11101100
3	2-Lane DP+USB3.1(正插)	0x00h <bit7:2>	0xf8	11111000

4	2-Lane DP+USB3.1(反插)	0x00h <bit7:2>	0xfc	11111100
5	USB3.1(正插)	0x00h <bit7:2>	0xb8	10111000
6	USB3.1(反插)	0x00h <bit7:2>	0xbc	10111100

注：主要是 offset 0x00 的值，是设置 8743 工作模式的；先把模式设置好，如果有信号兼容性问题，再根据 Spec 调整 EQ。

- 调试时可以通过 i2c 调试工具直接访问和读写寄存器的值进行芯片功能验证。

```
i2cget -y -f 5 0x11
i2cset -y 5 0x11 0x00 0xb8
i2cset -y 5 0x11 0x00 0xbc
i2cdump -y -f 5 0x11
```

- 也可通过 8743 驱动内置 debug 节点完成 I2C 的读写。

```
cat /sys/class/typec_mux/5-0011-mux/device/ps8743_attr/dump_all_reg
echo 0xXX 0xYY > /sys/class/typec_mux/5-0011-mux/device/ps8743_attr/i2c_write (XX 为寄存器地址，YY 为想写的值)
```

4.3 CC Logic 调试

目前内核原生支持 FUSB302、RT1711 等几款 CC 芯片，HUSB311 驱动也基于此进行开发，整个 TCPM 框架的状态机较为复杂，推荐使用以下方式进行功能开发和调试。

```
mount -t debugfs none /sys/kernel/debug
cat /sys/kernel/debug/usb/tcpm-5-004e/log
```

若尚未开启 HUSB311 软件驱动，则可以通过 i2c 调试工具直接访问寄存器的值。

```
i2cdump -y -f 5 0x4e
```

若已开启软件驱动，则可以通过 regmap 节点对寄存器进行访问，也可以直接使用 i2c-tools 进行调试。

```
cat /sys/kernel/debug/regmap/${dev-name}/registers
```

5 FAQ

5.1 如何在 Linux-4.9 内核上支持快充和 OTG

特别地，在 {IC} Linux-4.9 内核移植大量代码后也能够进行快充以及 OTG 识别，如有需要请联系 AW 技术支持。

内部链接：<http://gerrit.allwinnertech.com:8081/#/c/product/B300/linux-4.9/+255899/>

5.2 MTP 从 PC 拷贝文件至样机进度条不动

调整 “aw,rx_eq_emphasis” 参数即可，现象及分析过程详见《Linux_USB3_开发指南.pdf》，此处不作赘述。

5.3 如何支持外设插拔唤醒

系统休眠场景下，USB2.0 支持 USB Standby，USB3.0 仅支持 SuperStandby。

连接 USB3.0 外设后等待系统进入休眠，当断开 USB3.0 外设时，无法唤醒系统，此时 Vbus 仍对外输出，当连接 PC 时 Vbus 的对外输出会影响 AXP2202 对充电接入和移除的判断。

因此，可以把 HUSB311 配置为唤醒源，并保持其在休眠时的 VDD 供电，就能够在插拔时产生中断唤醒系统。

5.4 总线被拉低导致驱动加载失败

访问未初始化的 LT7911D 同一总线上的 i2c 设备（如 HUSB311），出现以下报错：

```
i2c i2c-5: SCL is stuck low, exit recovery
sunxi-twi 2503400.twi5: sunxi:twi_sunxi:twi bus barrier failed, sda is still low!
```

LT7911D 在不上电的时候，会把 i2c 总线拉低，上电后才正常。裸芯片测量 i2c 对地 150K，硬件无法控制，只能上电解决。

若需要调试同一 i2c 总线上的其他设备，可以先去掉 LT7911D 进行排查，也可以将在 husb311 节点添加如下属性，其中 PL5 根据实际板级硬件进行配置。

```
power_gpios = <&r_pio.PL 5 GPIO_ACTIVE_LOW>;
```

6 附录

6.1 相关补丁支持情况

6.1.1 音频模式支持

对于模拟音频设备还需要获取如下补丁才能实现模拟音频设备的识别以及类型的检测（三节 or 四节），补丁路径如下：

Linux_Type-C_开发指南/patches

在该路径下会有三个补丁，分别为：**0001-aw-usb-typec-tcpm-support-sink-port-for-debounce.patch**、**0002-aw-usb-typec-tcpm-support-mux-for-accessory-mode.patch**、**0003-aw-extcon-Introduce-EXTCON_PROP_JACK_POLE-property.patch**。

其中，**0001-aw-usb-typec-tcpm-support-sink-port-for-debounce.patch** 主要针对绿联以及华为等非标准 Type-C 线缆（CC 双 Rp 上拉到 Vbus）。

Table 3-13 USB Type-C to USB 2.0 Standard-A Cable Assembly Wiring

USB Type-C Plug		Wire		USB 2.0 Standard-A plug	
Pin	Signal Name	Wire Number	Signal Name	Pin	Signal Name
A1, B1, A12, B12	GND	1	GND_PWRrt1	4	GND
A4, B4, A9, B9	VBUS	2	PWR_VBUS1	1	VBUS
A5	CC	See Note 1			
B5	VCONN				
A6	Dp1	3	UTP_Dp	3	D+
A7	Dn1	4	UTP_Dn	2	D-
Shell	Shield	Outer shield	Shield	Shell	Shield

Notes:

- Pin A5 (CC) of the USB Type-C plug shall be connected to VBUS through a resistor Rp (56 kΩ ± 5%). See Section 4.5.3.2.2 and Table 4-26 for the functional description and value of Rp.
- Contacts B6 and B7 should not be present in the USB Type-C plug.
- All VBUS pins shall be connected together within the USB Type-C plug. Bypass capacitors are not required for the VBUS pins in this cable.
- All Ground return pins shall be connected together within the USB Type-C plug.
- Shield and GND grounds shall be connected within the USB Type-C and USB 2.0 Standard-A plugs on both ends of the cable assembly.
- All USB Type-C plug pins that are not listed in this table shall be open (not connected).

图 6-1: 标准 Type-C 线缆

如上图，标准的 Type-C 转 USB2.0 线缆上仅有一个 56k 上拉电阻连接到 Vbus 上。

6.1.2 多 Switch 支持

目前在 AW 的产品包中已支持单个 Switch（USB3.0&DP 或者 USB2.0&Audio 二选一）的 Mux 切换，在已发布的 Linux-5.15 内核中不支持多个 Switch，但在高版本内核中已经进行支持。因此，如果需要在支持全功能 Type-C 的功能，可以联系技术支持获取相关补丁，补丁路径如下：

Linux_Type-C_ 开发指南/patches/linux-5.15

在该路径下会有三个补丁，分别为：**0001-usb-typec-mux-Introduce-indirection.patch**、**0002-usb-typec-mux-Allow-multiple-mux_devs-per-mux.patch**、**0003-device-property-Add-helper-to-match-multiple-connections.patch**。

6.2 参考资料

- USB TYPE-C: <https://www.usbzn.com/article/forum-41.html>

- USB Type-C Port Controller Interface Specification: https://www.usb.org/sites/default/files/documents/usb-port_controller_specification_rev2.0_v1.0_0.pdf
- Type-C 线缆：详见上述 Spec 或以下表格。

类别	线缆类型	规格描述	具体说明	扩展用途
供电	CtoC	支持PD100W	必须为CtoC线缆，主要需使用其两根CC线进行PD协议握手；	支持USB2.0信号，可选支持USB3.1信号。（不一定支持USB3.1）
			可选支持E-marker（电子标记线缆），电子标记允许角色切换、交叠模式选择和增强的电源选项。	支持所有USB3.1信号
		建议选保标注支持PD20W/100W的线缆。		
DP	CtoC	支持4K/8K高清视频	必须为CtoC线缆，主要需使用其两根CC线进行Altmode协商，使用SBU线缆作为备用模式AUX通信及HPD检测；	兼容支持USB3.1。
		建议选保标注支持4K/8K视频传输的线缆。		
Audio	CtoA/3.5mm	支持模拟耳机	一般为模拟音频转接头，有三节或四节之分，支持MIC及HeadPhone功能。	
	Atoc	支持USB2.0	Type-A口公端一般为白色，有些厂家可能会做成紫色或橙色，只有4根线缆。	
USB2.0	CtoA	USB2.0 OTG转接头	一般为OTG转接头，一端为OTG Type-C口，一端为方形Type-A母口，一般为黑色或白色。	
	CtoC	支持USB2.0	一般只有CC、VBUS、GND、DP/DM线缆。	
USB3.1	Atoc	支持USB3.1 Gen1/Gen2	Type-A口公端一般为蓝色，有8根线缆，相比2.0多了8根立着的线缆。	
	CtoA	USB3.0 OTG转接头	一般为OTG转接头，一端为OTG Type-C口，一端为方形Type-A母口，一般为蓝色。	
	CtoC	支持USB3.1 Gen1/Gen2	除2.0的必要线缆外还有额外TX/RX正反共8根线缆。	如果再追加兼容DP。

图 6-2: Type-C_ 线缆梳理

6.3 DRP 状态

标准 Type-C 的连接状态按照以下图示进行变化，排查异常状态需要核对其状态机变化是否符合协议。

Figure 4-16 Connection State Diagram: DRP with Accessory and Try.SRC Support

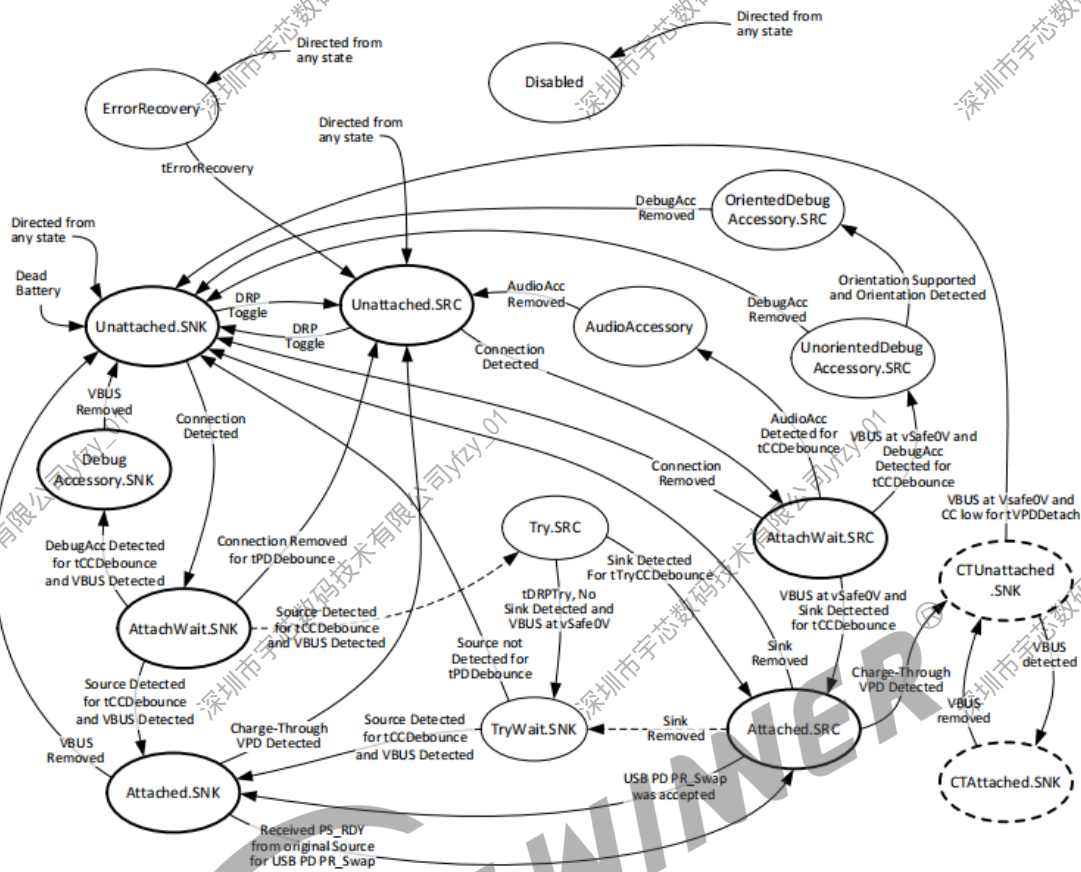


图 6-4: DRP_with_Accessory_and_TrySRC




著作权声明

版权所有 ©2024 珠海全志科技股份有限公司。保留一切权利。

本文档及内容受著作权法保护，其著作权由珠海全志科技股份有限公司（“全志”）拥有并保留一切权利。

本文档是全志的原创作品和版权财产，未经全志书面许可，任何单位和个人不得擅自摘抄、复制、修改、发表或传播本文档内容的部分或全部，且不得以任何形式传播。

商标声明

、、**全志科技**、（不完全列举）均为珠海全志科技股份有限公司的商标或者注册商标。在本文档描述的产品中出现的其它商标，产品名称，和服务名称，均由其各自所有人拥有。

免责声明

您购买的产品、服务或特性应受您与珠海全志科技股份有限公司（“全志”）之间签署的商业合同和条款的约束。本文档中描述的全部或部分产品、服务或特性可能不在您所购买或使用的范围内。使用前请认真阅读合同条款和相关说明，并严格遵循本文档的使用说明。您将自行承担任何不当使用行为（包括但不限于如超压，超频，超温使用）造成的不利后果，全志概不负责。

本文档作为使用指导仅供参考。由于产品版本升级或其他原因，本文档内容有可能修改，如有变更，恕不另行通知。全志尽全力在本文档中提供准确的信息，但并不确保内容完全没有错误，因使用本文档而发生损害（包括但不限于间接的、偶然的、特殊的损失）或发生侵犯第三方权利事件，全志概不负责。本文档中的所有陈述、信息和建议并不构成任何明示或暗示的保证或承诺。

本文档未以明示或暗示或其他方式授予全志的任何专利或知识产权。在您实施方案或使用产品的过程中，可能需要获得第三方的权利许可。请您自行向第三方权利人获取相关的许可。全志不承担也不代为支付任何关于获取第三方许可的许可费或版税（专利税）。全志不对您所使用的第三方许可技术做出任何保证、赔偿或承担其他义务。