

IDO-EVB3506-V1 Linux开发手册



IDO-EVB3506-V1 Linux开发手册

深圳触觉智能科技有限公司

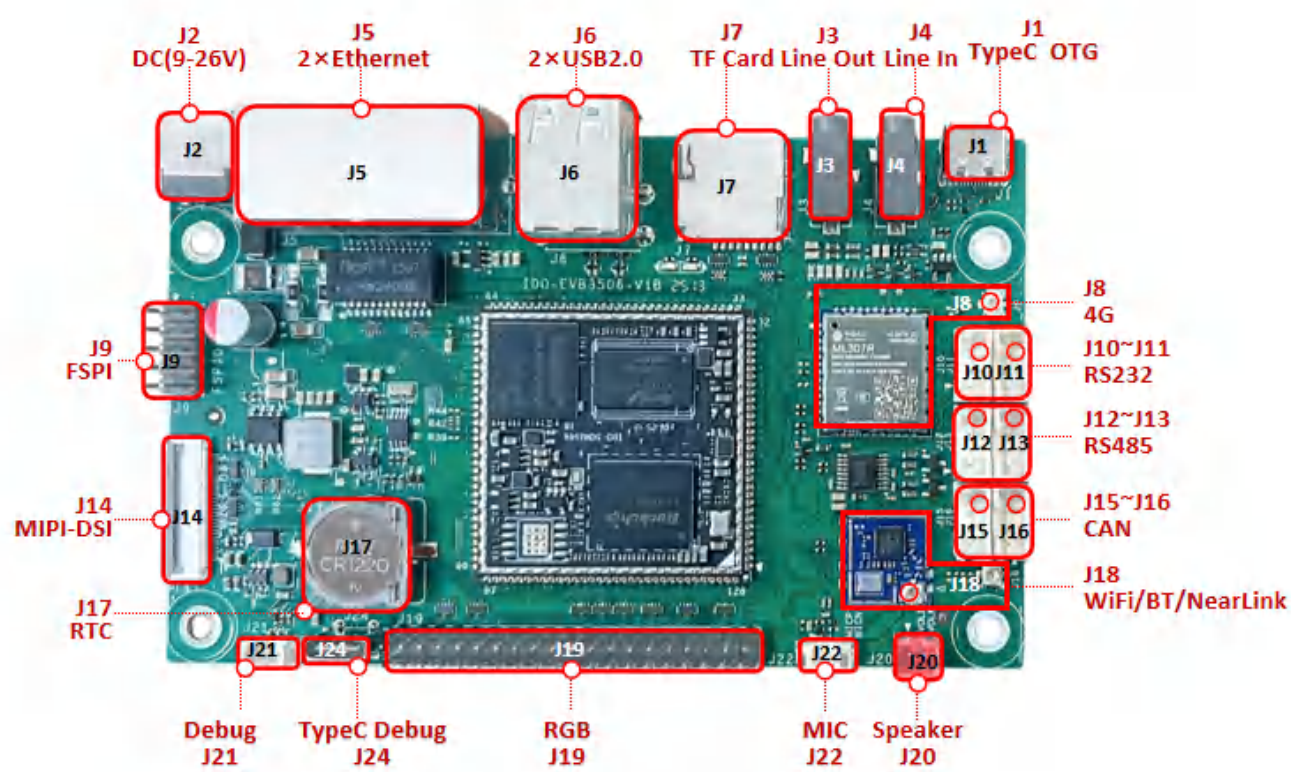
www.industio.cn

版本	PCBA版本	修订内容	修订	审核	日期
V1.0	V1D	创建文档	MHK	IDO	2025/12/22

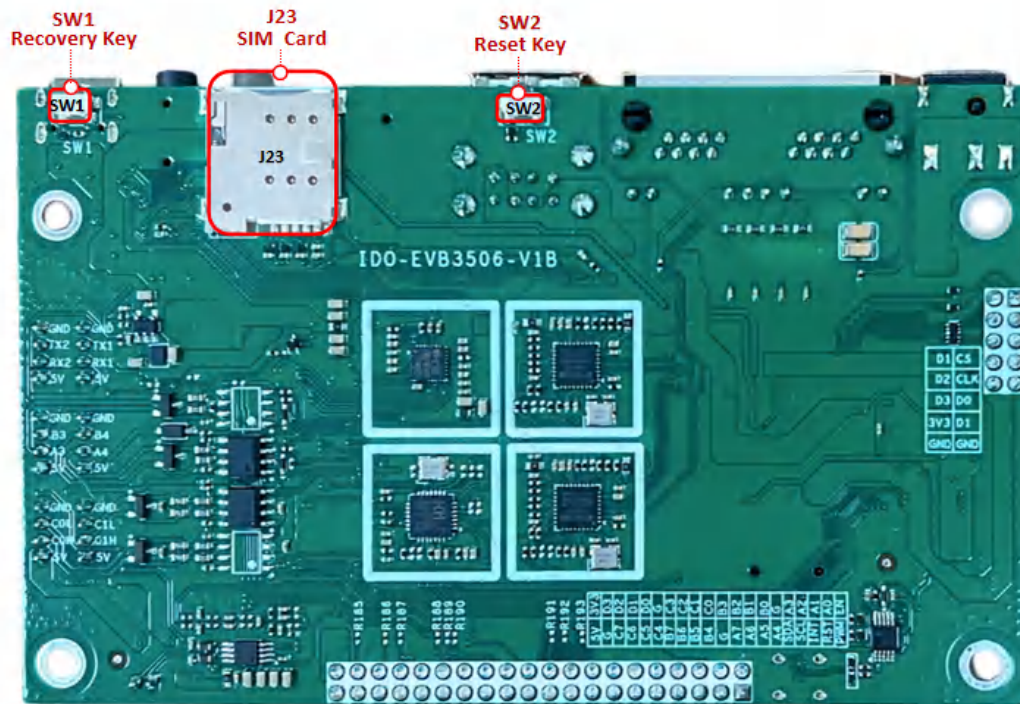
1 上手指南

1.介绍

IDO-EVB3506是搭载 Rockchip 全新芯片 RK3506B，22nm 先进制程工艺，集成了三核 ARM Cortex-A7和 Cortex-M0，主频高达 1.5GHz。支持 AMP 多核异构架构，一颗芯片可支持 Linux、RTOS、Bare-metal 灵活组合搭配，采用标准 RPMsg 核间通信机制。SDK 原生支持 LVGL 轻量级 UI 框架，内部2D硬件加速让 LVGL 运行更加流畅。



IDO-EVB3506-V1正面



IDO-EVB3506-V1背面

2 Linux开发

2.1.获取 SDK

下载路径：EVB3506/EVB3506-Linux/4. 软件资料/SDK-V2.0/Linux/SDK/

2.2.校验md5值

```
Bash |  
1 $ md5sum rk3506_linux6.1_v1.2.0.tar.gz
```

2.3.解压

首先准备一个空文件夹用于存放 SDK，建议在 home 目录下，本文以~/evb3506为例

```
1  #解压
2  $ mkdir ~/evb3506
3  $ cd ~/evb3506
4  $ tar -xvf rk3506_linux6.1_v1.2.0.tar.gz -C ./
5
6  #查看分支(默认分支为ido-evb3506)
7  $ git branch -a
8  * ido-evb3506
9      ido-evb3506-preempt-rt
10     ido-evb3506-xenomai-rt
11     remotes/origin/HEAD -> origin/master
12     remotes/origin/ido-evb3506
13     remotes/origin/ido-evb3506-preempt-rt
14     remotes/origin/ido-evb3506-xenomai-rt
15     remotes/origin/master
16
17 #获取代码
18 $ git reset --hard
```

2.4.SDK目录介绍

```
1  .
2  |— app
3  |— buildroot
4  |— build.sh -> device/rockchip/common/scripts/build.sh
5  |— device
6  |— docs
7  |— envsetup.sh -> buildroot/scripts/envsetup.sh
8  |— external
9  |— hal
10 |— kernel -> kernel-6.1
11 |— kernel-6.1
12 |— Makefile -> device/rockchip/common/Makefile
13 |— oebuild.log
14 |— prebuilts
15 |— rkbin
16 |— rkflash.sh -> device/rockchip/common/scripts/rkflash.sh
17 |— rootfs
18 |— rtos
19 |— tools
20 |— u-boot
21 |— yocto
```

```
1  app: 存放上层应用 APP, 主要是一些应用Demo。
2  buildroot: 基于 Buildroot (2024) 开发的根文件系统。
3  device/rockchip: 存放芯片板级配置以及SDK编译和打包固件的脚本和文件等。
4  docs: 存放通用开发指导文档、芯片平台相关文档、Linux 系统开发相关文档、其他参考文档等。
5  external: 存放第三方相关仓库, 包括显示、音视频、摄像头、网络、安全等。
6  kernel: 存放 Kernel 开发的代码。
7  output: 存放每次生成的固件信息、编译信息、XML、主机环境等。
8  prebuilts: 存放交叉编译工具链。
9  rkbin: 存放 Rockchip 相关二进制和工具。
10 rockdev: 存放编译输出固件, 实际软链接到 output/firmware 。
11 tools: 存放 Linux 和 Window 操作系统下常用工具。
12 u-boot: 存放基于 v2017.09 版本进行开发的 U-Boot 代码。
13 rtos: 存放rtos相关源码。
14 yocto: 存放yocto相关编译脚本和代码。
```

注意:

1. SDK 采用交叉编译, 所以要在 X86_64 电脑上使用 SDK, 不要将 SDK 下载到板子上。
2. 编译环境请使用 Ubuntu22.04 (真机或 虚拟机), 如果使用其他版本可能导致编译出错。

3. 不要在虚拟机共享文件夹以及非英文目录存放、解压SDK。
4. 获取、编译 SDK 请全程使用普通用户，不允许也不需要使用 root 权限（除非需要 apt 安装软件）

2.5.配置文件介绍

在 `device/rockchip/rk3506` 目录下，有不同板型的配置文件(xxxx_defconfig)，用于管理 SDK 每个环节的编译配置。以rockchip_ido-evb3506_v1_emmc_mipi_720x720_defconfig 做介绍：

```
1  RK_ROOTFS_EXT4=y                                #支持EXT4系统
2  RK_CHIP_FAMILY="rk3506_emmc"
3  RK_BUILDROOT_BASE_CFG="rk3506_emmc"
4  RK_ROOTFS_INSTALL_MODULES=y
5  RK_WIFIBT_CHIP="AP6256"
6  # RK_ROOTFS_LOG_GUARDIAN is not set
7  RK_UBOOT_CFG_FRAGMENTS="rk3506b rk3506_tb"      # uboot 编译配置
8  RK_UBOOT_SPL=y
9  RK_KERNEL_CFG="rk3506_emmc_defconfig"           # kernel 编译配置
10 RK_KERNEL_CFG_FRAGMENTS="rk3506-display.config" # kernel 编译追加
    配置
11 RK_KERNEL_DTS_NAME="ido-evb3506-v1a-emmc-mipi-720x720" # 板型 dts 名称
12 RK_BOOT_COMPRESSED=y
13 RK_BOOT_FIT_ITS_NAME="thunderboot.its"           #快启配置
14 RK_RECOVERY_FIT_ITS_NAME="thunderboot4recovery.its" # recovery 固件编
    译打包使用的 its 配置
15 RK_FLASH_SIZE=2048
16 RK_EXTRA_PARTITION_1_SRC="rk3506_oem"
17 RK_PARAMETER="parameter-emmc.txt"               # 打包使用的分区表
18 RK_USE_FIT_IMG=y
19 RK_EXTRA_PARTITION_1_FSTYPE="ext4"
20 #RK_EXTRA_PARTITION_1_SRC="rk3506_oem"
21 RK_EXTRA_PARTITION_2_FSTYPE="ext4"
```

2.6 分区说明

parameter-flash.txt 文件中包含了固件的分区信息，以 parameter-flash.txt 为例：

```
1  FIRMWARE_VER:8.1
2  MACHINE_MODEL:RK3506
3  MACHINE_ID:007
4  MANUFACTURER: RK3506
5  MAGIC: 0x5041524B
6  ATAG: 0x00200800
7  MACHINE: 3506
8  CHECK_MASK: 0x80
9  PWR_HLD: 0,0,A,0,1
10 TYPE: GPT
11 GROW_ALIGN: 0
12 CMDLINE:mtddparts=:0x00003000@0x00002000(uboot),0x00000800@0x00005000(mis
    c),0x00000200@0x00005800(vnvm),0x00009000@0x00005A00(recovery),0x00005000@
    0x0000EA00(boot),0x00050000@0x00013A00(rootfs),0x00008000@0x00063A00(oem),
    -@0x0006BA00(userdata:grow)
13 uuid:rootfs=614e0000-0000-4b53-8000-1d28000054a9
```

CMDLINE 属性是我们关注的地方，以 uboot 为例，0x00003000@0x00002000(uboot) 中 0x00003000 为uboot 分区的起始位置，0x00002000 为分区的大小，以此类推。

3.编译

3.1.选择项目

```
▼ Bash |
1 ido@ido-virtual-machine:~/evb3506/rk3506_linux6.1_v1.2.0$ ./build.sh lunch
2 Log colors: message notice warning error fatal
3
4 Log saved at /home/ido/evb3506/rk3506_linux6.1_v1.2.0/output/sessions/2025
  -12-22_16-04-05
5 Pick a defconfig:
6
7 1. rockchip_ido_evb3506_v1_emmc_mipi_1024x600_defconfig
8 2. rockchip_ido_evb3506_v1_emmc_mipi_1280x800_defconfig
9 3. rockchip_ido_evb3506_v1_emmc_mipi_720x720_amp_defconfig
10 4. rockchip_ido_evb3506_v1_emmc_mipi_720x720_defconfig
11 5. rockchip_ido_evb3506_v1_emmc_rgb_1024x600_defconfig
12 6. rockchip_ido_evb3506_v1_flash_mipi_1024x600_defconfig
13 7. rockchip_ido_evb3506_v1_flash_mipi_1280x800_defconfig
14 8. rockchip_ido_evb3506_v1_flash_mipi_720x720_amp_defconfig
15 9. rockchip_ido_evb3506_v1_flash_mipi_720x720_defconfig
16 10. rockchip_ido_evb3506_v1_flash_rgb_1024x600_defconfig
17 ▼ Which would you like? [1]: 3
18
```

选择对应的数字即可。

注意：

1.项目命名规则介绍：

```
▼ Bash |
1 平台： rockchip
2 主板丝印： ido_evb3506_v1。
3 显示接口及屏幕分辨率： mipi-720x720,mipi_1024x600,mipi_1280x800,rgb-1024x600
4 核心板存储支持： nand-flash, emmc
```

3.2.自动编译

```
▼ Bash |
1 $ ./build.sh
```

固件编译后生成路径： rockdev/update.img

3.3.分步编译

3.3.1.编译uboot

▼

Bash |

```
1  ./build.sh uboot
```

固件编译后生成路径：rockdev/uboot.img

3.3.2.编译kernel

▼

Bash |

```
1  ./build.sh kernel
```

固件编译后生成路径：rockdev/boot.img

3.3.3.编译buildroot系统

▼

Bash |

```
1  ./build.sh buildroot
```

固件编译后生成路径：rockdev/rootfs.img

3.3.4.编译ubuntu镜像

▼

Bash |

```
1  ./build.sh ubuntu
```

注意： nand配置不支持ubuntu系统。

3.3.5.编译recovery

```
1 ./build.sh recovery
```

固件编译后生成路径：rockdev/recovery.img

3.3.6. 固件打包

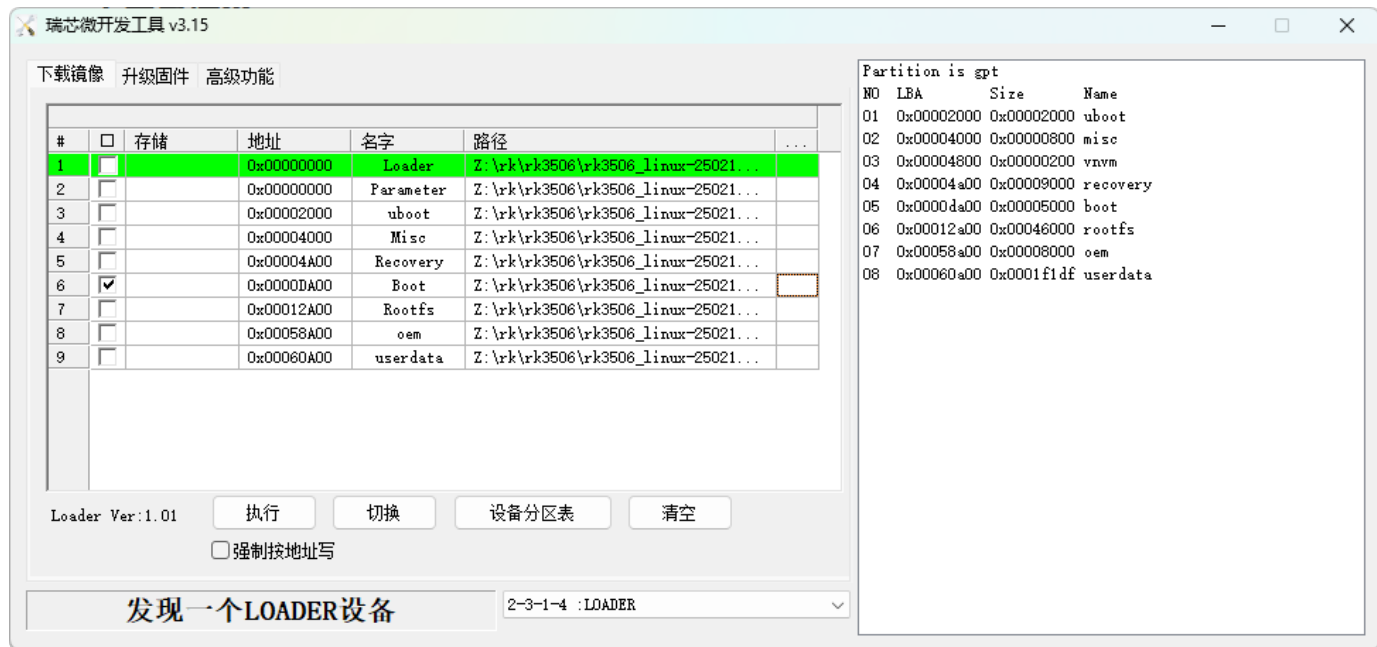
```
1 ./build.sh updateimg
```

4 烧录说明

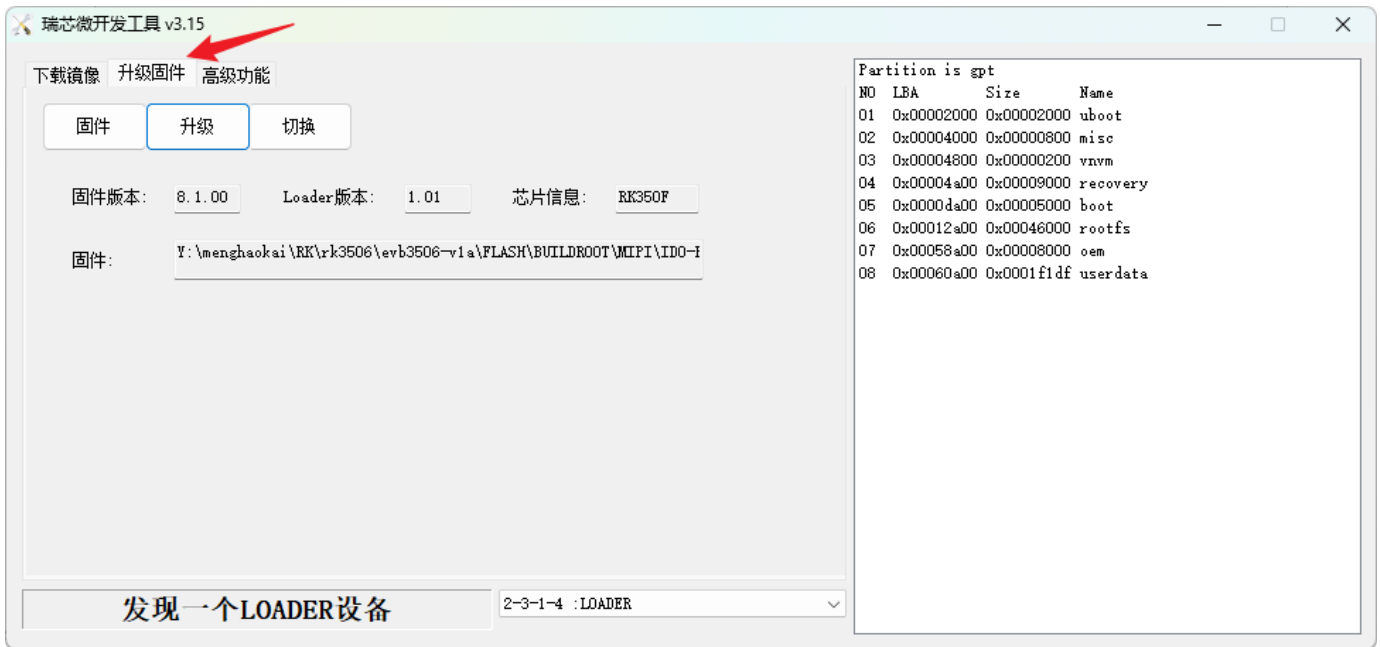
注意：板子需要进入到烧录模式，具体方法可以参考：《IDO-EVB3506-V1 开发板上手指南》

4.1. 整包烧录

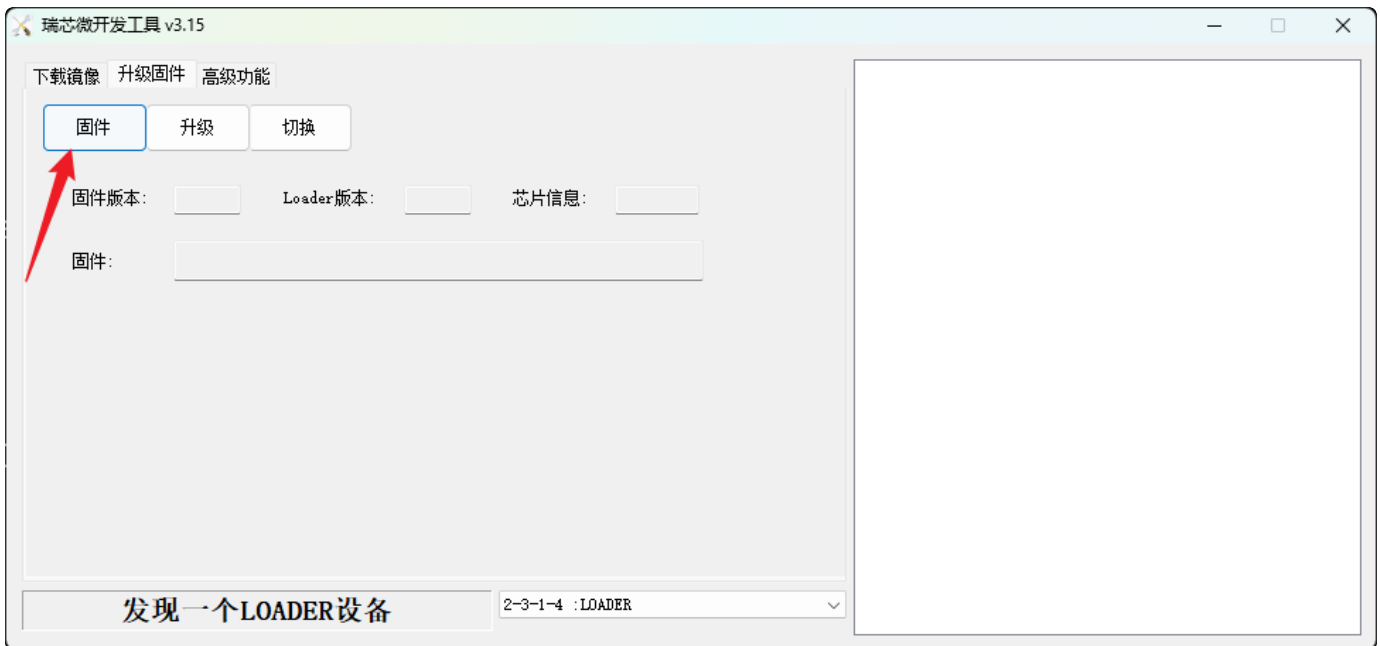
4.1.1. 运行RK开发工具



4.1.2. 点击升级固件

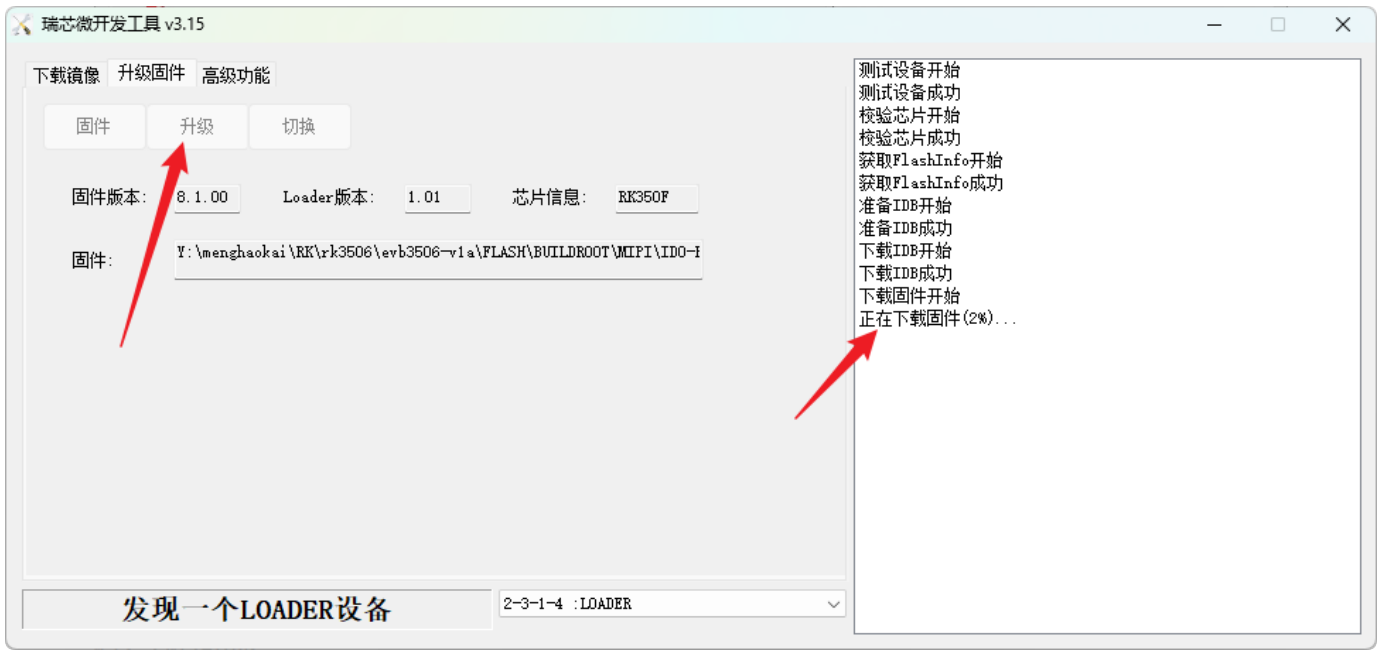


4.1.3.点击固件



选择: rockdev/update.img

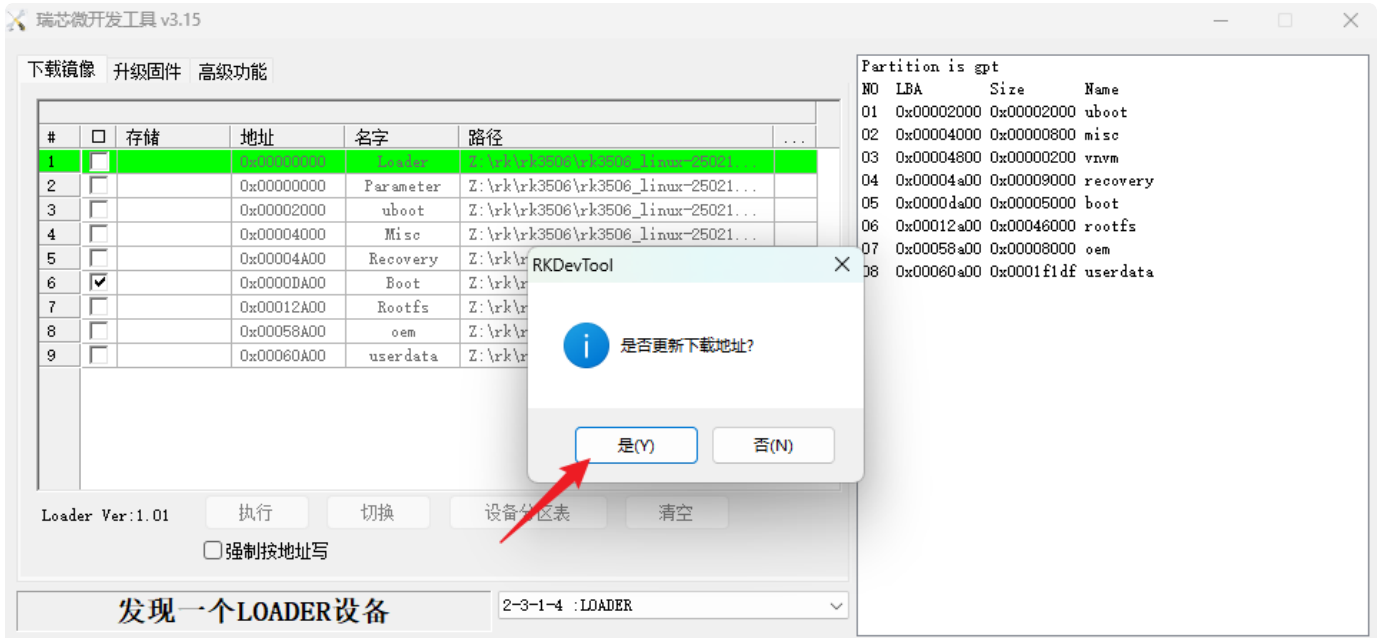
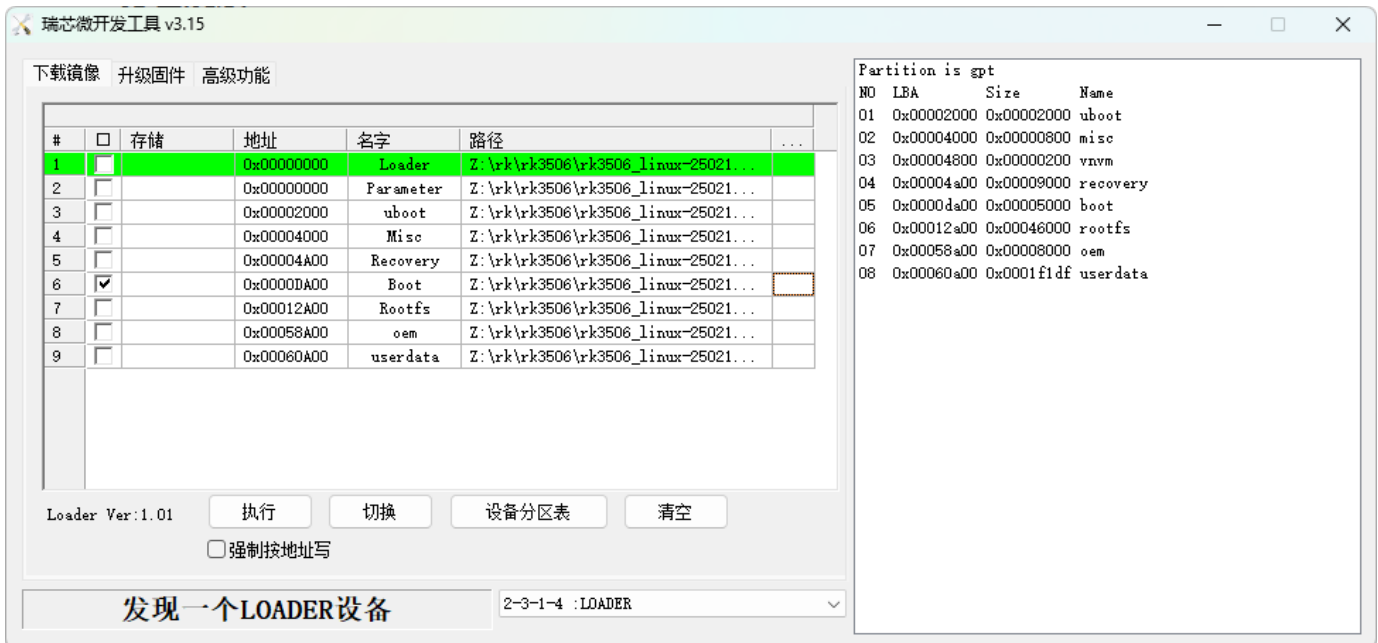
4.1.4.点击升级

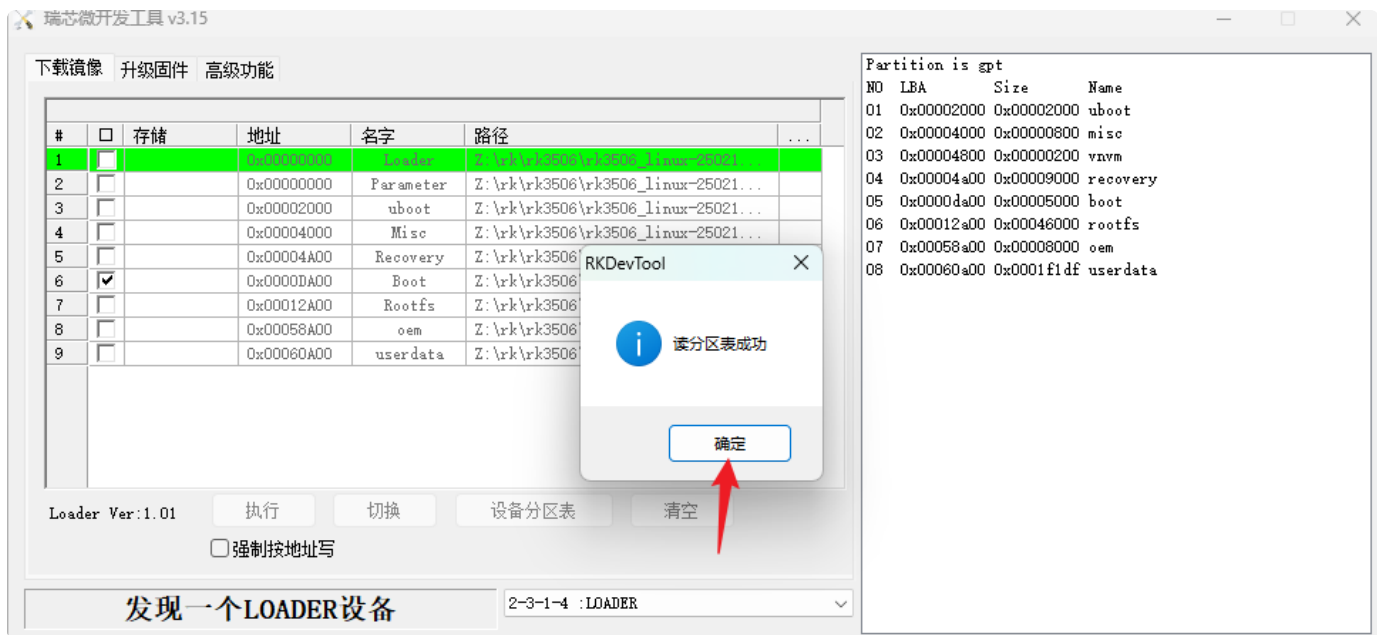


点击升级后，右侧会开始下载固件；下载完成后，板子会自动启动。

4.2.分包烧录

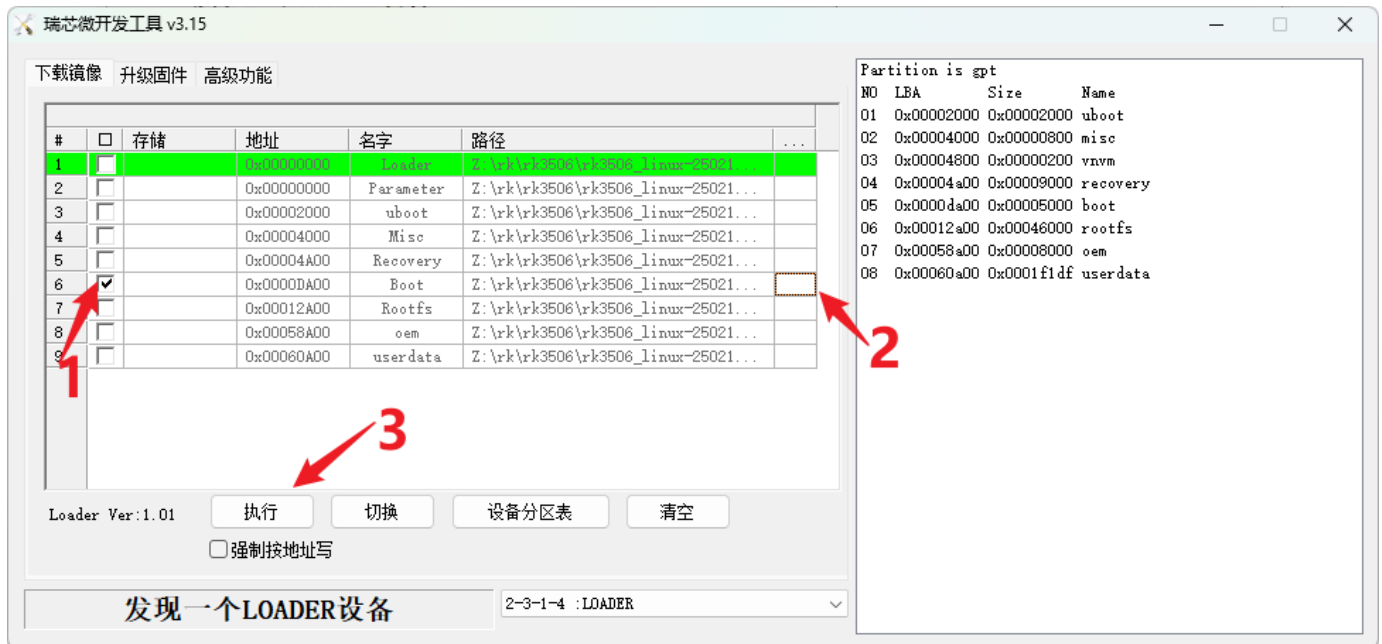
4.2.1.更新下载地址





4.2.2.烧录

1. 在对应的分区前选择对号
2. 然后点击右侧去选择对应的镜像
3. 点击执行即可烧录



5.驱动开发

5.1.CAN

5.1.1.CAN简介

CAN(Controller Area Network)总线，即控制器局域网总线，是一种有效支持分布式控制或实时控制的串行通信网络。CAN总线是一种在汽车上广泛采用的总线协议，被设计作为汽车环境中的微控制器通讯。

5.1.2.DTS 节点配置

公共配置：kernel/arch/arm/boot/dts/rk3506.dtsi

```
1  can0: can@ff320000 {
2      compatible = "rockchip,rk3506-canfd", "rockchip,rk3576-canfd";
3      reg = <0xff320000 0x1000>;
4      interrupts = <GIC_SPI 45 IRQ_TYPE_LEVEL_HIGH>;
5      clocks = <&cru CLK_CAN0>, <&cru HCLK_CAN0>;
6      clock-names = "baudclk", "apb_pclk";
7      resets = <&cru SRST_CAN0>, <&cru SRST_H_CAN0>;
8      reset-names = "can", "can-apb";
9      assigned-clocks = <&cru CLK_CAN0>;
10     assigned-clock-rates = <300000000>;
11     status = "disabled";
12 };
13
14  can1: can@ff330000 {
15     compatible = "rockchip,rk3506-canfd", "rockchip,rk3576-canfd";
16     reg = <0xff330000 0x1000>;
17     interrupts = <GIC_SPI 46 IRQ_TYPE_LEVEL_HIGH>;
18     clocks = <&cru CLK_CAN1>, <&cru HCLK_CAN1>;
19     clock-names = "baudclk", "apb_pclk";
20     resets = <&cru SRST_CAN1>, <&cru SRST_H_CAN1>;
21     reset-names = "can", "can-apb";
22     assigned-clocks = <&cru CLK_CAN1>;
23     assigned-clock-rates = <300000000>;
24     status = "disabled";
25 };
```

配置CAN1

板级配置：ido_evb3506_v1a-emmc.dtsi

```

1 //和rgb冲突
2 &can0 {
3     assigned-clocks = <&cru CLK_CAN0>;
4     assigned-clock-rates = <200000000>;
5     pinctrl-0 = <&rm_io25_can0_tx &rm_io26_can0_rx>;
6     pinctrl-names = "default";
7     status = "disabled";
8 };
9
10 &can1 {
11     assigned-clocks = <&cru CLK_CAN1>;
12     assigned-clock-rates = <200000000>;
13     pinctrl-0 = <&rm_io17_can1_tx &rm_io18_can1_rx>;
14     pinctrl-names = "default";
15     status = "okay";
16 };

```

双CAN配置

板级配置: ido_evb3506_v1a-emmc.dtsi

```

1 //和rgb冲突
2 &can0 {
3     assigned-clocks = <&cru CLK_CAN0>;
4     assigned-clock-rates = <200000000>;
5     pinctrl-0 = <&rm_io25_can0_tx &rm_io26_can0_rx>;
6     pinctrl-names = "default";
7     status = "okay";
8 };
9
10 &can1 {
11     assigned-clocks = <&cru CLK_CAN1>;
12     assigned-clock-rates = <200000000>;
13     pinctrl-0 = <&rm_io17_can1_tx &rm_io18_can1_rx>;
14     pinctrl-names = "default";
15     status = "okay";
16 };

```

板级配置: ido_evb3506_v1a-emmc-rgb-1024x600.dts

```
1 ▾ &rgb {
2     status = "disabled";
3     pinctrl-names = "default";
4     pinctrl-0 = <&rgb888_pins>;
5
6 ▾     ports {
7 ▾         rgb_out: port@1 {
8             reg = <1>;
9             #address-cells = <1>;
10            #size-cells = <0>;
11
12 ▾         rgb_out_panel: endpoint@0 {
13             reg = <0>;
14             remote-endpoint = <&panel_in_rgb>;
15         };
16     };
17 };
18 };
19
```

另外可设置时钟频率 `assigned-clock-rates`，如果 CAN 的比特率低于等于 3M 建议修改 CAN 时钟到 100M，信号更稳定。高于 3M 比特率的，时钟设置 200M 就可以。

通信测试

```
1  #检查can设备
2  $ ip link show
3
4  #在收发端关闭can0设备
5  $ ip link set can0 down
6
7  #设置仲裁段1M波特率, 数据段1M波特率
8  $ ip link set can0 type can bitrate 1000000 dbitrade 1000000 fd on
9  [ 63.834376] rk3576_canfd ff330000.can can0: bitrate error 0.3%
10 [ 63.834512] rk3576_canfd ff330000.can can0: bitrate error 0.3%
11
12 #在收发端打开can0设备
13 $ ip link set can0 up
14
15 #在接收端执行candump, 阻塞等待报文
16 $ candump can0
17
18 #在发送端执行cansend, 发送报文
19 $ cansend can0 123#1122334455667788
20
21 #检查是否启用成功
22 $ ip link show can0
```

总结调试过程中遇到的几个问题及解决方法：

1.无法正常收发

检查总线 CAN_H 和 CAN_L， 杜邦线是否松动或者接反。

2.CAN时钟频率配置

如果CAN的比特率低于等于3M建议修改CAN时钟到100M,信号更稳定。高于3M比特率的, 时钟设置200M就可以。

CAN时钟频率修改方法参考如下：

```

1 @@ -48,7 +48,7 @@ can0: can@ff320000 {
2     resets = <&cru SRST_CAN0>, <&cru SRST_H_CAN0>;
3     reset-names = "can", "can-apb";
4     assigned-clocks = <&cru CLK_CAN0>;
5     - assigned-clock-rates = <300000000>;
6     + assigned-clock-rates = <200000000>;
7     status = "disabled";
8 };
9
10 @@ -61,7 +61,7 @@ can1: can@ff330000 {
11     resets = <&cru SRST_CAN1>, <&cru SRST_H_CAN1>;
12     reset-names = "can", "can-apb";
13     assigned-clocks = <&cru CLK_CAN1>;
14     - assigned-clock-rates = <300000000>;
15     + assigned-clock-rates = <200000000>;
16     status = "disabled";
17 };

```

在某些时钟频率下，CAN的bitrate无法获得准确的速率，可以自行调整assigned-clock-rates去解决。

查看是否得到所需的bitrate：

```

1 ip -d link show can0

```

注意： evb3506的can0和RGB屏幕有冲突，默认dts中只打开can1节点，此节点在系统下注册成can0。

5.2.Ethernet

公共配置

```
kernel/arch/arm/boot/dts/rk3506.dtsi
```

```

1  ▾ gmac0: ethernet@ff4c8000 {
2      compatible = "rockchip,rk3506-gmac", "snps,dwmac-4.20a";
3      reg = <0xff4c8000 0x2000>;
4      interrupts = <GIC_SPI 66 IRQ_TYPE_LEVEL_HIGH>,
5                  <GIC_SPI 69 IRQ_TYPE_LEVEL_HIGH>;
6      interrupt-names = "macirq", "eth_wake_irq";
7      rockchip,grf = <&grf>;
8      clocks = <&cru CLK_MAC0>, <&cru CLK_MAC0_PTP>,
9              <&cru PCLK_MAC0>, <&cru ACLK_MAC0>;
10     clock-names = "stmmaceth", "ptp_ref",
11                  "pclk_mac", "aclk_mac";
12     resets = <&cru SRST_A_MAC0>;
13     reset-names = "stmmaceth";
14
15     snps,mixed-burst;
16     snps,tso;
17
18     snps,axi-config = <&gmac0_stmmac_axi_setup>;
19     snps,mtl-rx-config = <&gmac0_mtl_rx_setup>;
20     snps,mtl-tx-config = <&gmac0_mtl_tx_setup>;
21
22     phy-mode = "rmii";
23     status = "disabled";
24
25  ▾ mdio0: mdio {
26      compatible = "snps,dwmac-mdio";
27      #address-cells = <0x1>;
28      #size-cells = <0x0>;
29  };
30
31  ▾ gmac0_stmmac_axi_setup: stmmac-axi-config {
32      snps,wr_osr_lmt = <4>;
33      snps,rd_osr_lmt = <8>;
34      snps,blen = <0 0 0 0 16 8 4>;
35  };
36
37  ▾ gmac0_mtl_rx_setup: rx-queues-config {
38      snps,rx-queues-to-use = <1>;
39  ▾ queue0 {
40      status = "okay";
41  };
42  };
43
44  ▾ gmac0_mtl_tx_setup: tx-queues-config {

```

```

45     snps,tx-queues-to-use = <1>;
46     queue0 {
47         status = "okay";
48     };
49 };
50 };
51
52 gmac1: ethernet@ff4d0000 {
53     compatible = "rockchip,rk3506-gmac", "snps,dwmac-4.20a";
54     reg = <0xff4d0000 0x2000>;
55     interrupts = <GIC_SPI 70 IRQ_TYPE_LEVEL_HIGH>,
56                 <GIC_SPI 73 IRQ_TYPE_LEVEL_HIGH>;
57     interrupt-names = "macirq", "eth_wake_irq";
58     rockchip,grf = <&grf>;
59     clocks = <&cru CLK_MAC1>, <&cru CLK_MAC1_PTP>,
60             <&cru PCLK_MAC1>, <&cru ACLK_MAC1>;
61     clock-names = "stmmaceth", "ptp_ref",
62                 "pclk_mac", "aclk_mac";
63     resets = <&cru SRST_A_MAC1>;
64     reset-names = "stmmaceth";
65
66     snps,mixed-burst;
67     snps,tso;
68
69     snps,axi-config = <&gmac1_stmmac_axi_setup>;
70     snps,mtl-rx-config = <&gmac1_mtl_rx_setup>;
71     snps,mtl-tx-config = <&gmac1_mtl_tx_setup>;
72
73     phy-mode = "rmii";
74     status = "disabled";
75
76     mdio1: mdio {
77         compatible = "snps,dwmac-mdio";
78         #address-cells = <0x1>;
79         #size-cells = <0x0>;
80     };
81
82     gmac1_stmmac_axi_setup: stmmac-axi-config {
83         snps,wr_osr_lmt = <4>;
84         snps,rd_osr_lmt = <8>;
85         snps,blen = <0 0 0 0 16 8 4>;
86     };
87
88     gmac1_mtl_rx_setup: rx-queues-config {
89         snps,rx-queues-to-use = <1>;
90         queue0 {
91             status = "okay";

```

```
92     };
93 };
94
95 gmac1_mtl_tx_setup: tx-queues-config {
96     snps,tx-queues-to-use = <1>;
97     queue0 {
98         status = "okay";
99     };
100 };
101
```

板级的配置

```

1 ▾ &gmac0 {
2     phy-mode = "rmii";
3     clock_in_out = "input";
4
5     snps,reset-gpio = <&gpio0 RK_PC3 GPIO_ACTIVE_LOW>;
6     snps,reset-active-low;
7     snps,reset-delays-us = <0 20000 100000>;
8
9     pinctrl-names = "default";
10    pinctrl-0 = <&eth_rmii0_miim_pins
11                &eth_rmii0_tx_bus2_pins
12                &eth_rmii0_rx_bus2_pins
13                &eth_rmii0_clk_pins>;
14
15    phy-handle = <&rmii_phy0>;
16    status = "okay";
17 };
18
19 ▾ &gmac1 {
20     phy-mode = "rmii";
21     clock_in_out = "input";
22
23     snps,reset-gpio = <&gpio0 RK_PC4 GPIO_ACTIVE_LOW>;
24     snps,reset-active-low;
25     snps,reset-delays-us = <0 20000 100000>;
26
27     pinctrl-names = "default";
28     pinctrl-0 = <&eth_rmii1_miim_pins
29                &eth_rmii1_tx_bus2_pins
30                &eth_rmii1_rx_bus2_pins
31                &eth_rmii1_clk_pins>;
32
33     phy-handle = <&rmii_phy1>;
34     status = "okay";
35 };
36

```

5.3.LCD

IDO-EVB3506-V1 有一路 MIPI DSI 显示输出接口，支持 2 Lane 的数据输出，最高可输出 720x720@60Hz。

mts修改:

mti-720x720:

```

1 // SPDX-License-Identifier: (GPL-2.0+ OR MIT)
2 /*
3  * Copyright (c) 2024 Rockchip Electronics Co., Ltd.
4  */
5
6
7 #include <dt-bindings/display/drm_mipi_dsi.h>
8 #include <dt-bindings/input/rk-input.h>
9 #include <dt-bindings/suspend/rockchip-rk3506.h>
10 #include "ido_evb3506_v1a-emmc.dtsi"
11
12 / {
13     backlight: backlight {
14         compatible = "pwm-backlight";
15         pwms = <&pwm0_4ch_2 0 25000 0>;
16         brightness-levels = <
17             0 20 20 21 21 22 22 23
18             23 24 24 25 25 26 26 27
19             27 28 28 29 29 30 30 31
20             31 32 32 33 33 34 34 35
21             35 36 36 37 37 38 38 39
22             40 41 42 43 44 45 46 47
23             48 49 50 51 52 53 54 55
24             56 57 58 59 60 61 62 63
25             64 65 66 67 68 69 70 71
26             72 73 74 75 76 77 78 79
27             80 81 82 83 84 85 86 87
28             88 89 90 91 92 93 94 95
29             96 97 98 99 100 101 102 103
30             104 105 106 107 108 109 110 111
31             112 113 114 115 116 117 118 119
32             120 121 122 123 124 125 126 127
33             128 129 130 131 132 133 134 135
34             136 137 138 139 140 141 142 143
35             144 145 146 147 148 149 150 151
36             152 153 154 155 156 157 158 159
37             160 161 162 163 164 165 166 167
38             168 169 170 171 172 173 174 175
39             176 177 178 179 180 181 182 183
40             184 185 186 187 188 189 190 191
41             192 193 194 195 196 197 198 199
42             200 201 202 203 204 205 206 207
43             208 209 210 211 212 213 214 215
44             216 217 218 219 220 221 222 223

```

```

45         224 225 226 227 228 229 230 231
46         232 233 234 235 236 237 238 239
47         240 241 242 243 244 245 246 247
48         248 249 250 251 252 253 254 255
49     >;
50     default-brightness-level = <200>;
51     status = "okay";
52 };
53 };
54
55
56 &display_subsystem {
57     logo-memory-region = <&drm_logo>;
58     status = "okay";
59 };
60
61 &pwm0_4ch_2 {
62     pinctrl-names = "active";
63     pinctrl-0 = <&rm_io4_pwm0_ch2>;
64     status = "okay";
65 };
66
67 &dsi {
68     status = "okay";
69     //rockchip, lane-rate = <850>;
70     dsi_panel: panel@0 {
71         status = "okay";
72         compatible = "simple-panel-dsi";
73         reg = <0>;
74         backlight = <&backlight>;
75         reset-gpios = <&gpio0 RK_PB5 GPIO_ACTIVE_LOW>;
76         prepare-delay-ms = <5>;
77         reset-delay-ms = <1>;
78         init-delay-ms = <80>;
79         disable-delay-ms = <10>;
80         unprepare-delay-ms = <5>;
81
82         width-mm = <68>;
83         height-mm = <121>;
84
85         dsi, flags = <(MIPI_DSI_MODE_VIDEO | MIPI_DSI_MODE_VIDEO_BURST |
86             MIPI_DSI_MODE_LPM | MIPI_DSI_MODE_NO_EOT_PACKET)>;
87         dsi, format = <MIPI_DSI_FMT_RGB888>;
88         dsi, lanes = <2>;
89         panel-init-sequence = [
90             39 00 04 B9 F1 12 87
91             39 00 04 B2 B4 03 70

```

```

92      39 00 0B B3 10 10 28 28 03 FF 00 00 00 00
93      15 00 02 B4 80
94      39 00 03 B5 0A 0A
95      39 00 03 B6 8D 8D
96      39 00 05 B8 26 22 F0 13
97      39 00 1C BA 31 81 05 F9 0E 0E 20 00 00 00 00 00 00 00 44 25 00 91 0
A 00 00 01 4F 01 00 00 37
98      15 00 02 BC 47
99      39 00 06 BF 02 10 00 80 04
100     39 00 0A C0 73 73 50 50 00 00 12 73 00
101     39 00 12 C1 36 00 32 32 77 E1 77 77 CC CC FF FF 11 11 00 00 32
102     39 00 0D C7 10 00 0A 00 00 00 00 00 ED C5 00 A5
103     39 00 05 C8 10 40 1E 03
104     15 00 02 CC 0B
105     39 00 23 E0 00 0A 0F 2A 33 3F 44 39 06 0C 0E 14 15 13 15 10 18 00 0
A 0F 2A 33 3F 44 39 06 0C 0E 14 15 13 15 10 18
106     39 00 08 E1 11 11 91 00 00 00 00
107     39 00 0F E3 07 07 0B 0B 0B 0B 00 00 00 00 FF 04 C0 10
108     39 00 40 E9 C8 10 0A 00 00 80 81 12 31 23 4F 86 A0 00 47 08 00 00 0
C 00 00 00 00 00 0C 00 00 00 98 02 8B AF 46 02 88 88 88 88 88 98 13 8B A
F 57 13 88 88 88 88 88 00 00 00 00 00 00 00 00 00 00 00 00 00 00
109     39 00 3E EA 97 0C 09 09 09 78 00 00 00 00 00 00 9F 31 8B A8 31 75 8
8 88 88 88 88 9F 20 8B A8 20 64 88 88 88 88 88 23 00 00 02 71 00 00 00 0
0 00 00 00 00 00 00 00 00 00 00 00 40 80 81 00 00 00 00
110     39 00 04 EF FF FF 01
111     05 96 01 11
112     05 14 01 29
113 ];
114
115 disp_timings0: display-timings {
116     native-mode = <&dsi_timing0>;
117     dsi_timing0: timing0 {
118         clock-frequency = <48000000>;
119         hactive = <720>;
120         vactive = <720>;
121         hfront-porch = <106>;
122         hsync-len = <8>;
123         hback-porch = <120>;
124         vfront-porch = <20>;
125         vsync-len = <4>;
126         vback-porch = <20>;
127         hsync-active = <0>;
128         vsync-active = <0>;
129         de-active = <0>;
130         pixelclk-active = <0>;
131     };
132 };

```

```

133
134     ports {
135         #address-cells = <1>;
136         #size-cells = <0>;
137
138         port@0 {
139             reg = <0>;
140             panel_in_dsi: endpoint {
141                 remote-endpoint = <&dsi_out_panel>;
142             };
143         };
144     };
145 };
146
147     ports {
148         #address-cells = <1>;
149         #size-cells = <0>;
150
151         port@1 {
152             reg = <1>;
153             dsi_out_panel: endpoint {
154                 remote-endpoint = <&panel_in_dsi>;
155             };
156         };
157     };
158
159 };
160
161     &dsi_dphy {
162         status = "okay";
163     };
164
165     &dsi_in_vop {
166         status = "okay";
167     };
168
169     &dsi_panel {
170         power-supply = <&vcc3v3_lcd_n>;
171     };
172
173     &route_dsi{
174         status = "okay";
175         connect = <&vop_out_dsi>;
176     };
177
178     &i2c0 {
179         status = "okay";

```

```

180     pinctrl-names = "default";
181     pinctrl-0 = <&rm_io0_i2c0_scl
182         &rm_io1_i2c0_sda>;
183
184     gt911@14 {
185         compatible = "goodix,gt9xx";
186         reg = <0x14>;
187         pinctrl-0 = <&touch_gpio>;
188         touch-gpio = <&gpio0 RK_PA2 IRQ_TYPE_LEVEL_LOW>;
189         reset-gpio = <&gpio0 RK_PA3 GPIO_ACTIVE_HIGH>;
190         max-x = <720>;
191         max-y = <720>;
192         tp-size = <9110>;
193         status = "okay";
194     };
195 };
196
197     &vop {
198         status = "okay";
199     };
200
201     &pinctrl {
202         touch {
203             touch_gpio: touch-gpio {
204                 rockchip,pins = <0 RK_PA2 RK_FUNC_GPIO &pcfg_pull_none>;
205             };
206
207             reset_gpio: reset-gpio {
208                 rockchip,pins = <0 RK_PA3 RK_FUNC_GPIO &pcfg_pull_none>;
209             };
210         };

```

mipi-1024x600:

```

1 // SPDX-License-Identifier: (GPL-2.0+ OR MIT)
2 /*
3  * Copyright (c) 2024 Rockchip Electronics Co., Ltd.
4  */
5
6
7 #include <dt-bindings/display/drm_mipi_dsi.h>
8 #include <dt-bindings/input/rk-input.h>
9 #include <dt-bindings/suspend/rockchip-rk3506.h>
10 #include "ido_evb3506_v1a-emmc.dtsi"
11
12 / {
13     backlight: backlight {
14         compatible = "pwm-backlight";
15         pwms = <&pwm0_4ch_2 0 25000 0>;
16         brightness-levels = <
17             0 20 20 21 21 22 22 23
18             23 24 24 25 25 26 26 27
19             27 28 28 29 29 30 30 31
20             31 32 32 33 33 34 34 35
21             35 36 36 37 37 38 38 39
22             40 41 42 43 44 45 46 47
23             48 49 50 51 52 53 54 55
24             56 57 58 59 60 61 62 63
25             64 65 66 67 68 69 70 71
26             72 73 74 75 76 77 78 79
27             80 81 82 83 84 85 86 87
28             88 89 90 91 92 93 94 95
29             96 97 98 99 100 101 102 103
30             104 105 106 107 108 109 110 111
31             112 113 114 115 116 117 118 119
32             120 121 122 123 124 125 126 127
33             128 129 130 131 132 133 134 135
34             136 137 138 139 140 141 142 143
35             144 145 146 147 148 149 150 151
36             152 153 154 155 156 157 158 159
37             160 161 162 163 164 165 166 167
38             168 169 170 171 172 173 174 175
39             176 177 178 179 180 181 182 183
40             184 185 186 187 188 189 190 191
41             192 193 194 195 196 197 198 199
42             200 201 202 203 204 205 206 207
43             208 209 210 211 212 213 214 215
44             216 217 218 219 220 221 222 223

```

```

45         224 225 226 227 228 229 230 231
46         232 233 234 235 236 237 238 239
47         240 241 242 243 244 245 246 247
48         248 249 250 251 252 253 254 255
49     >;
50     default-brightness-level = <200>;
51     status = "okay";
52 };
53 };
54
55
56 &display_subsystem {
57     logo-memory-region = <&drm_logo>;
58     status = "okay";
59 };
60
61 &pwm0_4ch_2 {
62     pinctrl-names = "active";
63     pinctrl-0 = <&rm_io4_pwm0_ch2>;
64     status = "okay";
65 };
66
67 &dsi {
68     status = "okay";
69     //rockchip, lane-rate = <850>;
70     dsi_panel: panel@0 {
71         status = "okay";
72         compatible = "simple-panel-dsi";
73         reg = <0>;
74         backlight = <&backlight>;
75         reset-gpios = <&gpio0 RK_PB5 GPIO_ACTIVE_LOW>;
76         prepare-delay-ms = <5>;
77         reset-delay-ms = <1>;
78         init-delay-ms = <80>;
79         disable-delay-ms = <10>;
80         unprepare-delay-ms = <5>;
81
82         width-mm = <68>;
83         height-mm = <121>;
84
85         dsi,flags = <(MIPI_DSI_MODE_VIDEO | MIPI_DSI_MODE_VIDEO_BURST |
86             MIPI_DSI_MODE_LPM | MIPI_DSI_MODE_NO_EOT_PACKET)>;
87         dsi,format = <MIPI_DSI_FMT_RGB888>;
88         dsi,lanes = <2>;
89         panel-init-sequence = [
90     #if 0
91         39 00 04 B9 F1 12 87

```

```

92      39 00 04 B2 B4 03 70
93      39 00 0B B3 10 10 28 28 03 FF 00 00 00 00
94      15 00 02 B4 80
95      39 00 03 B5 0A 0A
96      39 00 03 B6 8D 8D
97      39 00 05 B8 26 22 F0 13
98      39 00 1C BA 31 81 05 F9 0E 0E 20 00 00 00 00 00 00 00 44 25 00 91 0
A 00 00 01 4F 01 00 00 37
99      15 00 02 BC 47
100     39 00 06 BF 02 10 00 80 04
101     39 00 0A C0 73 73 50 50 00 00 12 73 00
102     39 00 12 C1 36 00 32 32 77 E1 77 77 CC CC FF FF 11 11 00 00 32
103     39 00 0D C7 10 00 0A 00 00 00 00 00 ED C5 00 A5
104     39 00 05 C8 10 40 1E 03
105     15 00 02 CC 0B
106     39 00 23 E0 00 0A 0F 2A 33 3F 44 39 06 0C 0E 14 15 13 15 10 18 00 0
A 0F 2A 33 3F 44 39 06 0C 0E 14 15 13 15 10 18
107     39 00 08 E1 11 11 91 00 00 00 00
108     39 00 0F E3 07 07 0B 0B 0B 0B 00 00 00 00 FF 04 C0 10
109     39 00 40 E9 C8 10 0A 00 00 80 81 12 31 23 4F 86 A0 00 47 08 00 00 0
C 00 00 00 00 00 0C 00 00 00 98 02 8B AF 46 02 88 88 88 88 98 13 8B A
F 57 13 88 88 88 88 88 00 00 00 00 00 00 00 00 00 00 00 00 00 00
110     39 00 3E EA 97 0C 09 09 09 78 00 00 00 00 00 00 9F 31 8B A8 31 75 8
8 88 88 88 88 9F 20 8B A8 20 64 88 88 88 88 88 23 00 00 02 71 00 00 00 0
0 00 00 00 00 00 00 00 00 00 00 00 40 80 81 00 00 00 00
111     39 00 04 EF FF FF 01
112     #endif
113     /* mipi init : F700035 */
114     15 00 02 B2 10
115     15 00 02 80 AC
116     15 00 02 81 B8
117     15 00 02 82 09
118     15 00 02 83 78
119     15 00 02 84 7F
120     15 00 02 85 BB
121     15 00 02 86 70
122
123     05 96 01 11
124     05 14 01 29
125 ];
126
127     disp_timings0: display-timings {
128         native-mode = <&dsi_timing1>;
129         dsi_timing0: timing0 {
130             clock-frequency = <48000000>;
131             hactive = <720>;
132             vactive = <720>;

```

```

133         hfront-porch = <106>;
134         hsync-len = <8>;
135         hback-porch = <120>;
136         vfront-porch = <20>;
137         vsync-len = <4>;
138         vback-porch = <20>;
139         hsync-active = <0>;
140         vsync-active = <0>;
141         de-active = <0>;
142         pixelclk-active = <0>;
143     };
144     dsi_timing1: timing1 {
145         clock-frequency = <52000000>;
146         hactive = <1024>;
147         vactive = <600>;
148         hfront-porch = <5>;
149         hsync-len = <10>;
150         hback-porch = <60>;
151         vfront-porch = <12>;
152         vsync-len = <2>;
153         vback-porch = <23>;
154         hsync-active = <0>;
155         vsync-active = <0>;
156         de-active = <0>;
157         pixelclk-active = <0>;
158     };
159 };
160
161 ports {
162     #address-cells = <1>;
163     #size-cells = <0>;
164
165     port@0 {
166         reg = <0>;
167         panel_in_dsi: endpoint {
168             remote-endpoint = <&dsi_out_panel>;
169         };
170     };
171 };
172 };
173
174 ports {
175     #address-cells = <1>;
176     #size-cells = <0>;
177
178     port@1 {
179         reg = <1>;

```

```

180         dsi_out_panel: endpoint {
181             remote-endpoint = <&panel_in_dsi>;
182         };
183     };
184 };
185
186 };
187
188 &dsi_dphy {
189     status = "okay";
190 };
191
192 &dsi_in_vop {
193     status = "okay";
194 };
195
196 &dsi_panel {
197     power-supply = <&vcc3v3_lcd_n>;
198 };
199
200 &route_dsi{
201     status = "okay";
202     connect = <&vop_out_dsi>;
203 };
204
205 &i2c0 {
206     status = "okay";
207     pinctrl-names = "default";
208     pinctrl-0 = <&rm_io0_i2c0_scl
209         &rm_io1_i2c0_sda>;
210
211     gt911@14 {
212         compatible = "goodix,gt9xx";
213         reg = <0x14>;
214         pinctrl-0 = <&touch_gpio>;
215         touch-gpio = <&gpio0 RK_PA2 IRQ_TYPE_LEVEL_LOW>;
216         reset-gpio = <&gpio0 RK_PA3 GPIO_ACTIVE_HIGH>;
217         max-x = <1024>;
218         max-y = <600>;
219         tp-size = <9110>;
220         status = "okay";
221     };
222 };
223
224 &vop {
225     status = "okay";
226 };

```

```

227 ▾
228 ▾ &pinctrl {
229     touch {
230         touch_gpio: touch-gpio {
231             rockchip,pins = <0 RK_PA2 RK_FUNC_GPIO &pcfg_pull_none>;
232 ▾     };
233
234         reset_gpio: reset-gpio {
235             rockchip,pins = <0 RK_PA3 RK_FUNC_GPIO &pcfg_pull_none>;
236         };
237     };
238 1.

```

mipi-1280x800:

```

1 // SPDX-License-Identifier: (GPL-2.0+ OR MIT)
2 /*
3  * Copyright (c) 2024 Rockchip Electronics Co., Ltd.
4  */
5
6
7 #include <dt-bindings/display/drm_mipi_dsi.h>
8 #include <dt-bindings/input/rk-input.h>
9 #include <dt-bindings/suspend/rockchip-rk3506.h>
10 #include "ido_evb3506_v1a-emmc.dtsi"
11
12 / {
13     backlight: backlight {
14         compatible = "pwm-backlight";
15         pwms = <&pwm0_4ch_2 0 25000 0>;
16         brightness-levels = <
17             0 20 20 21 21 22 22 23
18             23 24 24 25 25 26 26 27
19             27 28 28 29 29 30 30 31
20             31 32 32 33 33 34 34 35
21             35 36 36 37 37 38 38 39
22             40 41 42 43 44 45 46 47
23             48 49 50 51 52 53 54 55
24             56 57 58 59 60 61 62 63
25             64 65 66 67 68 69 70 71
26             72 73 74 75 76 77 78 79
27             80 81 82 83 84 85 86 87
28             88 89 90 91 92 93 94 95
29             96 97 98 99 100 101 102 103
30             104 105 106 107 108 109 110 111
31             112 113 114 115 116 117 118 119
32             120 121 122 123 124 125 126 127
33             128 129 130 131 132 133 134 135
34             136 137 138 139 140 141 142 143
35             144 145 146 147 148 149 150 151
36             152 153 154 155 156 157 158 159
37             160 161 162 163 164 165 166 167
38             168 169 170 171 172 173 174 175
39             176 177 178 179 180 181 182 183
40             184 185 186 187 188 189 190 191
41             192 193 194 195 196 197 198 199
42             200 201 202 203 204 205 206 207
43             208 209 210 211 212 213 214 215
44             216 217 218 219 220 221 222 223

```

```

45         224 225 226 227 228 229 230 231
46         232 233 234 235 236 237 238 239
47         240 241 242 243 244 245 246 247
48         248 249 250 251 252 253 254 255
49     >;
50     default-brightness-level = <200>;
51     status = "okay";
52 };
53 };
54
55
56 &display_subsystem {
57     logo-memory-region = <&drm_logo>;
58     status = "okay";
59 };
60
61 &pwm0_4ch_2 {
62     pinctrl-names = "active";
63     pinctrl-0 = <&rm_io4_pwm0_ch2>;
64     status = "okay";
65 };
66
67 &dsi {
68     status = "okay";
69     //rockchip, lane-rate = <850>;
70     dsi_panel: panel@0 {
71         status = "okay";
72         compatible = "simple-panel-dsi";
73         reg = <0>;
74         backlight = <&backlight>;
75         reset-gpios = <&gpio0 RK_PB5 GPIO_ACTIVE_LOW>;
76         prepare-delay-ms = <5>;
77         reset-delay-ms = <1>;
78         init-delay-ms = <80>;
79         disable-delay-ms = <10>;
80         unprepare-delay-ms = <5>;
81
82         width-mm = <68>;
83         height-mm = <121>;
84
85         dsi, flags = <(MIPI_DSI_MODE_VIDEO | MIPI_DSI_MODE_VIDEO_BURST |
86             MIPI_DSI_MODE_LPM | MIPI_DSI_MODE_NO_EOT_PACKET)>;
87         dsi, format = <MIPI_DSI_FMT_RGB888>;
88         dsi, lanes = <2>;
89         panel-init-sequence = [
90             39 00 04 FF 98 81 03    /* Generic_Long_Write_3P(0xFF,0x98,0x81,0x0
91 3) */

```

91	15 00 02 01 00	/* Generic_Short_Write_1P(0x01,0x00) */
92	15 00 02 02 00	/* Generic_Short_Write_1P(0x02,0x00) */
93	15 00 02 03 73	/* Generic_Short_Write_1P(0x03,0x73) */
94	15 00 02 04 13	/* Generic_Short_Write_1P(0x04,0x13) */
95	15 00 02 05 00	/* Generic_Short_Write_1P(0x05,0x00) */
96	15 00 02 06 0A	/* Generic_Short_Write_1P(0x06,0x0A) */
97	15 00 02 07 05	/* Generic_Short_Write_1P(0x07,0x05) */
98	15 00 02 08 00	/* Generic_Short_Write_1P(0x08,0x00) */
99	15 00 02 09 28	/* Generic_Short_Write_1P(0x09,0x28) */
100	15 00 02 0A 00	/* Generic_Short_Write_1P(0x0A,0x00) */
101	15 00 02 0B 00	/* Generic_Short_Write_1P(0x0B,0x00) */
102	15 00 02 0C 00	/* Generic_Short_Write_1P(0x0C,0x00) */
103	15 00 02 0D 28	/* Generic_Short_Write_1P(0x0D,0x28) */
104	15 00 02 0E 00	/* Generic_Short_Write_1P(0x0E,0x00) */
105	15 00 02 0F 28	/* Generic_Short_Write_1P(0x0F,0x28) */
106	15 00 02 10 28	/* Generic_Short_Write_1P(0x10,0x28) */
107	15 00 02 11 00	/* Generic_Short_Write_1P(0x11,0x00) */
108	15 00 02 12 00	/* Generic_Short_Write_1P(0x12,0x00) */
109	15 00 02 13 00	/* Generic_Short_Write_1P(0x13,0x00) */
110	15 00 02 14 00	/* Generic_Short_Write_1P(0x14,0x00) */
111	15 00 02 15 00	/* Generic_Short_Write_1P(0x15,0x00) */
112	15 00 02 16 00	/* Generic_Short_Write_1P(0x16,0x00) */
113	15 00 02 17 00	/* Generic_Short_Write_1P(0x17,0x00) */
114	15 00 02 18 00	/* Generic_Short_Write_1P(0x18,0x00) */
115	15 00 02 19 00	/* Generic_Short_Write_1P(0x19,0x00) */
116	15 00 02 1A 00	/* Generic_Short_Write_1P(0x1A,0x00) */
117	15 00 02 1B 00	/* Generic_Short_Write_1P(0x1B,0x00) */
118	15 00 02 1C 00	/* Generic_Short_Write_1P(0x1C,0x00) */
119	15 00 02 1D 00	/* Generic_Short_Write_1P(0x1D,0x00) */
120	15 00 02 1E 40	/* Generic_Short_Write_1P(0x1E,0x40) */
121	15 00 02 1F 80	/* Generic_Short_Write_1P(0x1F,0x80) */
122	15 00 02 20 06	/* Generic_Short_Write_1P(0x20,0x06) */
123	15 00 02 21 01	/* Generic_Short_Write_1P(0x21,0x01) */
124	15 00 02 22 00	/* Generic_Short_Write_1P(0x22,0x00) */
125	15 00 02 23 00	/* Generic_Short_Write_1P(0x23,0x00) */
126	15 00 02 24 00	/* Generic_Short_Write_1P(0x24,0x00) */
127	15 00 02 25 00	/* Generic_Short_Write_1P(0x25,0x00) */
128	15 00 02 26 00	/* Generic_Short_Write_1P(0x26,0x00) */
129	15 00 02 27 00	/* Generic_Short_Write_1P(0x27,0x00) */
130	15 00 02 28 33	/* Generic_Short_Write_1P(0x28,0x33) */
131	15 00 02 29 33	/* Generic_Short_Write_1P(0x29,0x33) */
132	15 00 02 2A 00	/* Generic_Short_Write_1P(0x2A,0x00) */
133	15 00 02 2B 00	/* Generic_Short_Write_1P(0x2B,0x00) */
134	15 00 02 2C 04	/* Generic_Short_Write_1P(0x2C,0x04) */
135	15 00 02 2D 04	/* Generic_Short_Write_1P(0x2D,0x04) */
136	15 00 02 2E 05	/* Generic_Short_Write_1P(0x2E,0x05) */
137	15 00 02 2F 05	/* Generic_Short_Write_1P(0x2F,0x05) */

138	15 00 02 30 00	/* Generic_Short_Write_1P(0x30,0x00) */
139	15 00 02 31 00	/* Generic_Short_Write_1P(0x31,0x00) */
140	15 00 02 32 31	/* Generic_Short_Write_1P(0x32,0x31) */
141	15 00 02 33 00	/* Generic_Short_Write_1P(0x33,0x00) */
142	15 00 02 34 00	/* Generic_Short_Write_1P(0x34,0x00) */
143	15 00 02 35 0A	/* Generic_Short_Write_1P(0x35,0x0A) */
144	15 00 02 36 00	/* Generic_Short_Write_1P(0x36,0x00) */
145	15 00 02 37 08	/* Generic_Short_Write_1P(0x37,0x08) */
146	15 00 02 38 00	/* Generic_Short_Write_1P(0x38,0x00) */
147	15 00 02 39 00	/* Generic_Short_Write_1P(0x39,0x00) */
148	15 00 02 3A 00	/* Generic_Short_Write_1P(0x3A,0x00) */
149	15 00 02 3B 00	/* Generic_Short_Write_1P(0x3B,0x00) */
150	15 00 02 3C 00	/* Generic_Short_Write_1P(0x3C,0x00) */
151	15 00 02 3D 00	/* Generic_Short_Write_1P(0x3D,0x00) */
152	15 00 02 3E 00	/* Generic_Short_Write_1P(0x3E,0x00) */
153	15 00 02 3F 00	/* Generic_Short_Write_1P(0x3F,0x00) */
154	15 00 02 40 00	/* Generic_Short_Write_1P(0x40,0x00) */
155	15 00 02 41 00	/* Generic_Short_Write_1P(0x41,0x00) */
156	15 00 02 42 00	/* Generic_Short_Write_1P(0x42,0x00) */
157	15 00 02 43 08	/* Generic_Short_Write_1P(0x43,0x08) */
158	15 00 02 44 00	/* Generic_Short_Write_1P(0x44,0x00) */
159		
160	15 00 02 50 01	/* Generic_Short_Write_1P(0x50,0x01) */
161	15 00 02 51 23	/* Generic_Short_Write_1P(0x51,0x23) */
162	15 00 02 52 44	/* Generic_Short_Write_1P(0x52,0x44) */
163	15 00 02 53 67	/* Generic_Short_Write_1P(0x53,0x67) */
164	15 00 02 54 89	/* Generic_Short_Write_1P(0x54,0x89) */
165	15 00 02 55 AB	/* Generic_Short_Write_1P(0x55,0xAB) */
166	15 00 02 56 01	/* Generic_Short_Write_1P(0x56,0x01) */
167	15 00 02 57 23	/* Generic_Short_Write_1P(0x57,0x23) */
168	15 00 02 58 45	/* Generic_Short_Write_1P(0x58,0x45) */
169	15 00 02 59 67	/* Generic_Short_Write_1P(0x59,0x67) */
170	15 00 02 5A 89	/* Generic_Short_Write_1P(0x5A,0x89) */
171	15 00 02 5B AB	/* Generic_Short_Write_1P(0x5B,0xAB) */
172	15 00 02 5C CD	/* Generic_Short_Write_1P(0x5C,0xCD) */
173	15 00 02 5D EF	/* Generic_Short_Write_1P(0x5D,0xEF) */
174		
175	15 00 02 5E 11	/* Generic_Short_Write_1P(0x5E,0x11) */
176	15 00 02 5F 02	/* Generic_Short_Write_1P(0x5F,0x02) */
177	15 00 02 60 08	/* Generic_Short_Write_1P(0x60,0x08) */
178	15 00 02 61 0E	/* Generic_Short_Write_1P(0x61,0x0E) */
179	15 00 02 62 0F	/* Generic_Short_Write_1P(0x62,0x0F) */
180	15 00 02 63 0C	/* Generic_Short_Write_1P(0x63,0x0C) */
181	15 00 02 64 0D	/* Generic_Short_Write_1P(0x64,0x0D) */
182	15 00 02 65 17	/* Generic_Short_Write_1P(0x65,0x17) */
183	15 00 02 66 01	/* Generic_Short_Write_1P(0x66,0x01) */
184	15 00 02 67 01	/* Generic_Short_Write_1P(0x67,0x01) */

```

185      15 00 02 68 02      /* Generic_Short_Write_1P(0x68,0x02) */
186      15 00 02 69 02      /* Generic_Short_Write_1P(0x69,0x02) */
187      15 00 02 6A 00      /* Generic_Short_Write_1P(0x6A,0x00) */
188      15 00 02 6B 00      /* Generic_Short_Write_1P(0x6B,0x00) */
189      15 00 02 6C 02      /* Generic_Short_Write_1P(0x6C,0x02) */
190      15 00 02 6D 02      /* Generic_Short_Write_1P(0x6D,0x02) */
191      15 00 02 6E 16      /* Generic_Short_Write_1P(0x6E,0x16) */
192      15 00 02 6F 16      /* Generic_Short_Write_1P(0x6F,0x16) */
193      15 00 02 70 06      /* Generic_Short_Write_1P(0x70,0x06) */
194      15 00 02 71 06      /* Generic_Short_Write_1P(0x71,0x06) */
195      15 00 02 72 07      /* Generic_Short_Write_1P(0x72,0x07) */
196      15 00 02 73 07      /* Generic_Short_Write_1P(0x73,0x07) */
197      15 00 02 74 02      /* Generic_Short_Write_1P(0x74,0x02) */
198      15 00 02 75 02      /* Generic_Short_Write_1P(0x75,0x02) */
199      15 00 02 76 08      /* Generic_Short_Write_1P(0x76,0x08) */
200      15 00 02 77 0E      /* Generic_Short_Write_1P(0x77,0x0E) */
201      15 00 02 78 0F      /* Generic_Short_Write_1P(0x78,0x0F) */
202      15 00 02 79 0C      /* Generic_Short_Write_1P(0x79,0x0C) */
203      15 00 02 7A 0D      /* Generic_Short_Write_1P(0x7A,0x0D) */
204      15 00 02 7B 17      /* Generic_Short_Write_1P(0x7B,0x17) */
205
206      15 00 02 7C 01      /* Generic_Short_Write_1P(0x7C,0x01) */
207      15 00 02 7D 01      /* Generic_Short_Write_1P(0x7D,0x01) */
208      15 00 02 7E 02      /* Generic_Short_Write_1P(0x7E,0x02) */
209      15 00 02 7F 02      /* Generic_Short_Write_1P(0x7F,0x02) */
210      15 00 02 80 00      /* Generic_Short_Write_1P(0x80,0x00) */
211      15 00 02 81 00      /* Generic_Short_Write_1P(0x81,0x00) */
212      15 00 02 82 02      /* Generic_Short_Write_1P(0x82,0x02) */
213      15 00 02 83 02      /* Generic_Short_Write_1P(0x83,0x02) */
214      15 00 02 84 16      /* Generic_Short_Write_1P(0x84,0x16) */
215      15 00 02 85 16      /* Generic_Short_Write_1P(0x85,0x16) */
216      15 00 02 86 06      /* Generic_Short_Write_1P(0x86,0x06) */
217      15 00 02 87 06      /* Generic_Short_Write_1P(0x87,0x06) */
218      15 00 02 88 07      /* Generic_Short_Write_1P(0x88,0x07) */
219      15 00 02 89 07      /* Generic_Short_Write_1P(0x89,0x07) */
220      15 00 02 8A 02      /* Generic_Short_Write_1P(0x8A,0x02) */
221
222      39 00 04 FF 98 81 04      /* Generic_Long_Write_3P(0xFF,0x98,0x81,0x0
4) */
223      15 00 02 6E 1A      /* Generic_Short_Write_1P(0x6E,0x1A) */
224      15 00 02 6F 37      /* Generic_Short_Write_1P(0x6F,0x37) */
225      15 00 02 3A A4      /* Generic_Short_Write_1P(0x3A,0xA4) */
226      15 00 02 8D 1F      /* Generic_Short_Write_1P(0x8D,0x1F) */
227      15 00 02 87 BA      /* Generic_Short_Write_1P(0x87,0xBA) */
228      15 00 02 B2 D1      /* Generic_Short_Write_1P(0xB2,0xD1) */
229      15 00 02 88 0B      /* Generic_Short_Write_1P(0x88,0x0B) */
230      15 00 02 38 01      /* Generic_Short_Write_1P(0x38,0x01) */

```

```

231      15 00 02 39 00      /* Generic_Short_Write_1P(0x39,0x00) */
232      15 00 02 B5 02      /* Generic_Short_Write_1P(0xB5,0x02) */
233      15 00 02 31 25      /* Generic_Short_Write_1P(0x31,0x25) */
234      15 00 02 3B C0      /* Generic_Short_Write_1P(0x3B,0xC0) */
235
236      39 00 04 FF 98 81 01 /* Generic_Long_Write_3P(0xFF,0x98,0x81,0x0
1) */
237      15 00 02 22 09      /* Generic_Short_Write_1P(0x22,0x09) */
238      15 00 02 31 00      /* Generic_Short_Write_1P(0x31,0x00) */
239      15 00 02 40 53      /* Generic_Short_Write_1P(0x40,0x53) */
240      15 00 02 53 5A      /* Generic_Short_Write_1P(0x53,0x5A) */
241      15 00 02 55 69      /* Generic_Short_Write_1P(0x55,0x69) */
242      15 00 02 50 9E      /* Generic_Short_Write_1P(0x50,0x9E) */
243      15 00 02 51 99      /* Generic_Short_Write_1P(0x51,0x99) */
244      15 00 02 60 06      /* Generic_Short_Write_1P(0x60,0x06) */
245      15 00 02 62 20      /* Generic_Short_Write_1P(0x62,0x20) */
246      15 00 02 B6 90      /* Generic_Short_Write_1P(0xB6,0x90) */
247      15 00 02 B7 03      /* Generic_Short_Write_1P(0xB7,0x03) */
248
249      /* Gamma参数设置 */
250      15 00 02 A0 00      /* Generic_Short_Write_1P(0xA0,0x00) */
251      15 00 02 A1 17      /* Generic_Short_Write_1P(0xA1,0x17) */
252      15 00 02 A2 26      /* Generic_Short_Write_1P(0xA2,0x26) */
253      15 00 02 A3 13      /* Generic_Short_Write_1P(0xA3,0x13) */
254      15 00 02 A4 16      /* Generic_Short_Write_1P(0xA4,0x16) */
255      15 00 02 A5 29      /* Generic_Short_Write_1P(0xA5,0x29) */
256      15 00 02 A6 1E      /* Generic_Short_Write_1P(0xA6,0x1E) */
257      15 00 02 A7 1F      /* Generic_Short_Write_1P(0xA7,0x1F) */
258      15 00 02 A8 8B      /* Generic_Short_Write_1P(0xA8,0x8B) */
259      15 00 02 A9 1D      /* Generic_Short_Write_1P(0xA9,0x1D) */
260      15 00 02 AA 2A      /* Generic_Short_Write_1P(0xAA,0x2A) */
261      15 00 02 AB 7B      /* Generic_Short_Write_1P(0xAB,0x7B) */
262      15 00 02 AC 1A      /* Generic_Short_Write_1P(0xAC,0x1A) */
263      15 00 02 AD 19      /* Generic_Short_Write_1P(0xAD,0x19) */
264      15 00 02 AE 4E      /* Generic_Short_Write_1P(0xAE,0x4E) */
265      15 00 02 AF 24      /* Generic_Short_Write_1P(0xAF,0x24) */
266      15 00 02 B0 29      /* Generic_Short_Write_1P(0xB0,0x29) */
267      15 00 02 B1 4F      /* Generic_Short_Write_1P(0xB1,0x4F) */
268      15 00 02 B2 5C      /* Generic_Short_Write_1P(0xB2,0x5C) */
269      15 00 02 B3 23      /* Generic_Short_Write_1P(0xB3,0x23) */
270
271      15 00 02 C0 00      /* Generic_Short_Write_1P(0xC0,0x00) */
272      15 00 02 C1 17      /* Generic_Short_Write_1P(0xC1,0x17) */
273      15 00 02 C2 26      /* Generic_Short_Write_1P(0xC2,0x26) */
274      15 00 02 C3 13      /* Generic_Short_Write_1P(0xC3,0x13) */
275      15 00 02 C4 16      /* Generic_Short_Write_1P(0xC4,0x16) */
276      15 00 02 C5 29      /* Generic_Short_Write_1P(0xC5,0x29) */

```

```

277      15 00 02 C6 1E      /* Generic_Short_Write_1P(0xC6,0x1E) */
278      15 00 02 C7 1F      /* Generic_Short_Write_1P(0xC7,0x1F) */
279      15 00 02 C8 8B      /* Generic_Short_Write_1P(0xC8,0x8B) */
280      15 00 02 C9 1D      /* Generic_Short_Write_1P(0xC9,0x1D) */
281      15 00 02 CA 2A      /* Generic_Short_Write_1P(0xCA,0x2A) */
282      15 00 02 CB 7B      /* Generic_Short_Write_1P(0xCB,0x7B) */
283      15 00 02 CC 1A      /* Generic_Short_Write_1P(0xCC,0x1A) */
284      15 00 02 CD 19      /* Generic_Short_Write_1P(0xCD,0x19) */
285      15 00 02 CE 4E      /* Generic_Short_Write_1P(0xCE,0x4E) */
286      15 00 02 CF 24      /* Generic_Short_Write_1P(0xCF,0x24) */
287      15 00 02 D0 29      /* Generic_Short_Write_1P(0xD0,0x29) */
288      15 00 02 D1 4D      /* Generic_Short_Write_1P(0xD1,0x4D) */
289      15 00 02 D2 5C      /* Generic_Short_Write_1P(0xD2,0x5C) */
290      15 00 02 D3 23      /* Generic_Short_Write_1P(0xD3,0x23) */
291
292      39 00 04 FF 98 81 00 /* Generic_Long_Write_3P(0xFF,0x98,0x81,0x0
0) */
293
294
295
296
297      05 96 01 11
298      05 14 01 29
299  ];
300
301  disp_timings0: display-timings {
302      native-mode = <&dsi_timing2>;
303
304      dsi_timing2: timing2 {
305          clock-frequency = <73400000>;
306          hactive = <800>;
307          vactive = <1280>;
308          hfront-porch = <52>;
309          hsync-len = <4>;
310          hback-porch = <12>;
311          vfront-porch = <60>;
312          vsync-len = <18>;
313          vback-porch = <50>;
314          hsync-active = <0>;
315          vsync-active = <0>;
316          de-active = <0>;
317          pixelclk-active = <1>;
318      };
319
320  };
321
322  ports {

```

```

323     #address-cells = <1>;
324     #size-cells = <0>;
325
326     port@0 {
327         reg = <0>;
328         panel_in_dsi: endpoint {
329             remote-endpoint = <&dsi_out_panel>;
330         };
331     };
332 };
333 };
334
335 ports {
336     #address-cells = <1>;
337     #size-cells = <0>;
338
339     port@1 {
340         reg = <1>;
341         dsi_out_panel: endpoint {
342             remote-endpoint = <&panel_in_dsi>;
343         };
344     };
345 };
346 };
347 };
348
349 &dsi_dphy {
350     status = "okay";
351 };
352
353 &dsi_in_vop {
354     status = "okay";
355 };
356
357 &dsi_panel {
358     power-supply = <&vcc3v3_lcd_n>;
359 };
360
361 &route_dsi{
362     status = "okay";
363     connect = <&vop_out_dsi>;
364 };
365
366 &i2c0 {
367     status = "okay";
368     pinctrl-names = "default";
369     pinctrl-0 = <&rm_io0_i2c0_scl

```

```

370         &rm_io1_i2c0_sda>;
371
372     gt911@14 {
373         compatible = "goodix,gt9xx";
374         reg = <0x14>;
375         pinctrl-0 = <&touch_gpio>;
376         touch-gpio = <&gpio0 RK_PA2 IRQ_TYPE_LEVEL_LOW>;
377         reset-gpio = <&gpio0 RK_PA3 GPIO_ACTIVE_HIGH>;
378         max-x = <800>;
379         max-y = <1280>;
380         tp-size = <9110>;
381         touchscreen-inverted-x;
382         touchscreen-inverted-y;
383         // touchscreen-swapped-x-y;
384         status = "okay";
385     };
386 };
387
388 &vop {
389     status = "okay";
390 };
391
392 &pinctrl {
393     touch {
394         touch_gpio: touch-gpio {
395             rockchip,pins = <0 RK_PA2 RK_FUNC_GPIO &pcfg_pull_none>;
396         };
397
398         reset_gpio: reset-gpio {
399             rockchip,pins = <0 RK_PA3 RK_FUNC_GPIO &pcfg_pull_none>;
400         };
401     };

```

注意事项

在配置 MIPI DSI的时候，如果出现异常现象，如黑屏，画面拉伸，显示有噪点等，需要注意排查：

1. 显示时序是否配置正确，尤其是 DCLK 的配置。
2. 上下电时序是否正确，在文件 `kernel/drivers/gpu/drm/panel/panel-simple.c` 中的 `panel_simple_prepare` 和 `panel_simple_unprepare` 函数内，调用了设备树中所配置的上下电时序和 gpio 口。

如果选择在 uboot 阶段开启开机 logo, 那么还需要排查 `u-boot/drivers/video/drm/rockchip_panel.c` 文件内的 `panel_simple_prepare` 和 `panel_simple_unprepare` 函数。

3. 搭示波器看一下上电时序是否正确, 主要是确认 LCD 使能引脚、复位引脚和屏幕上电指令间的时序是否正确。

调试方法

获取系统中正在使用的 Video Port (与所连接的显示控制器) 信息:

```
▼ Bash |
1 root@rk3506-buildroot:/# cat /sys/kernel/debug/dri/0/summary
2 ▾ VOP [ff600000.vop]: ACTIVE
3     Connector: DSI-1
4 ▾ bus_format[100a]: RGB888_1X24
5 ▾ overlay_mode[0] output_mode[0] color-encoding[1] color-range[1]
6     Display mode: 720x720p66
7 ▾ dclk[48000 kHz] real_dclk[47536 kHz] aclk[294912 kHz] type[48] flag[a]
8     H: 720 826 834 954
9     V: 720 740 744 764
10    win1-0: ACTIVE
11 ▾ format: BG24 little-endian (0x34324742) SDR[0] color-encoding[0] color-range[0]
12 ▾ csc: y2r[0] r2r[0] r2y[0] csc mode[0]
13     zpos: 0
14 ▾ src: pos[0x0] rect[720x720]
15 ▾ dst: pos[0x0] rect[720x720]
16 ▾ buf[0]: addr: 0x0f000000 pitch: 2304 offset: 0
17 ▾ post: sdr2hdr[0] hdr2sdr[0]
18 ▾ pre : sdr2hdr[0]
19 | post CSC: r2y[0] y2r[0] CSC mode[...]
```

一般如果遇到屏幕无法显示的问题, 都需要先执行上面的命令去看一下连接状态和分辨率是否正确。

注意: rk3506linux sdk 只有一个显示接口、可以支持 RGB/MIPI 两种接口、但是同时只能使用一个。

5.4.RTC

IDO-EVB3506-V1 采用 HYM8563 作为 RTC(*Real Time Clock*) , 需要接入 RTC 电池给 RTC 芯片供电才可以保证在短时间系统断电后 RTC 能正常运行。

DTS配置参考: `kernel/arch/arm/boot/dts/rk3502-ido-evb3506_v1a.dtsi`

驱动参考: `kernel/drivers/rtc/rtc-hym8563.c`

Linux下的rtc使用方法为:

```
▼ Bash |
1  #同步网络时间
2  $ ntpdate cn.pool.ntp.org
3
4  #查看当前 RTC 的日期和时间:
5  $ cat /sys/class/rtc/rtc0/date
6  2021-01-01
7
8  $ cat /sys/class/rtc/rtc0/time
9  17:18:14
10
11 $ 查看系统时间
12 $ date
13 Wed Mar 12 18:46:59 CST 2025
14
15 #设置系统时间
16 $ date -s "2024-10-09 14:02:30"
17
18 #将rtc时间调整为与目前的系统时间一致
19 $ hwclock -w
20
21 #获取硬件rtc当前时间 (断电重启读取时间没有太大偏差)
22 $ hwclock -r
23 2024-10-09 14:02:35.945604+00:00
24
25 #将rtc时间设置为系统时间
26 hwclock --systohc
```

5.5.UART

Debug口:

debug串口支持两种波特率: 115200 and 1500000; 默认配置为1500000。将波特率修改为115200如下所示:

修改内核打印波特率:

以emmc版本为例:

```

1  $ cd $sdk
2  $ vim kernel/arch/arm/boot/dts/ido-evb3506-v1a-emmc.dtsi
3  fiq_debugger: fiq-debugger {
4      compatible = "rockchip,fiq-debugger";
5      rockchip,serial-id = <0>;
6      rockchip,wake-irq = <0>;
7      rockchip,irq-mode-enable = <1>;
8      rockchip,baudrate = <115200>;    /* Only 115200 and 150000
    0 */
9      interrupts = <GIC_SPI 115 IRQ_TYPE_LEVEL_HIGH>;
10 };

```

修改 rockchip,baudrate = <115200>;

修改loader打印波特率:

修改rkbin/tools/ddrbin_param.txt的uart baudrate:

```

1  $ cd $sdk
2  $ git diff rkbin/tools/ddrbin_param.txt
3  diff --git a/rkbin/tools/ddrbin_param.txt b/rkbin/tools/ddrbin_param.txt
4  index 0dfdd3183..82ade7e7c 100644
5  --- a/rkbin/tools/ddrbin_param.txt
6  +++ b/rkbin/tools/ddrbin_param.txt
7  @@ -11,7 +11,7 @@ lp5_freq=
8
9  uart id=
10 uart iomux=
11 -uart baudrate=
12 +uart baudrate=115200
13
14 sr_idle=
15 pd_idle=

```

重新生成loader:

```
1 $ cd $sdk
2 $ cd rkbin/tools/
3 $ ./ddrbin_tool rk3506 ddrbin_param.txt ../bin/rk35/rk3506b_ddr_750MHz_v1.0
  4.bin
```

重新编译uboot和kernel:

```
1 ./build.sh uboot
2 ./build.sh kernel
```

编译完成, 更新loader、uboot和kernel分区。

普通串口:

DTS配置

```
1 &uart1 {
2     status = "okay";
3     pinctrl-names = "default";
4     pinctrl-0 = <&rm_io15_uart1_tx &rm_io16_uart1_rx>;
5 };
```

配置好串口后, 硬件接口对应软件上的节点为:

```
1 UART1:    /dev/ttyS1
```

收发测试

使用microcom可以进行串口收发测试, 命令如下:

```
1 root@rk3506-buildroot:/# microcom -s 115200 /dev/ttyS1
```

注意：测试完成，按Ctrl+x退出。

6.开发案例

6.1.Qt5使用

详情参考

百度网盘：EVB3506/Linux/3.开发手册/《IDO-EVB3506-V1 QT应用开发手册》.pdf

6.2.MQTT使用

详情参考

百度网盘：EVB3506/Linux/3.开发手册/《IDO-EVB3506-V1 MQTT应用案例》.pdf

6.3.GDB使用

详情参考

百度网盘：EVB3506/Linux/3.开发手册/《IDO-EVB3506-V1 GDB开发手册》.pdf

6.4.Docker使用

详情参考

百度网盘：EVB3506/Linux/3.开发手册/《IDO-EVB3506-V1 DOCKER容器安装使用手册》.pdf

6.5. LVGL使用

详情参考

百度网盘：EVB3506/Linux/3.开发手册/《IDO-EVB3506-V1 LVGL应用开发手册》.pdf

6.6.RT-Linux使用

详情参考

百度网盘：EVB3506/Linux/3.开发手册/《IDO-EVB3506-V1 RT-Linux使用手册》.pdf