

# IDO-EVB1126B-V1 Linux开发手册

---



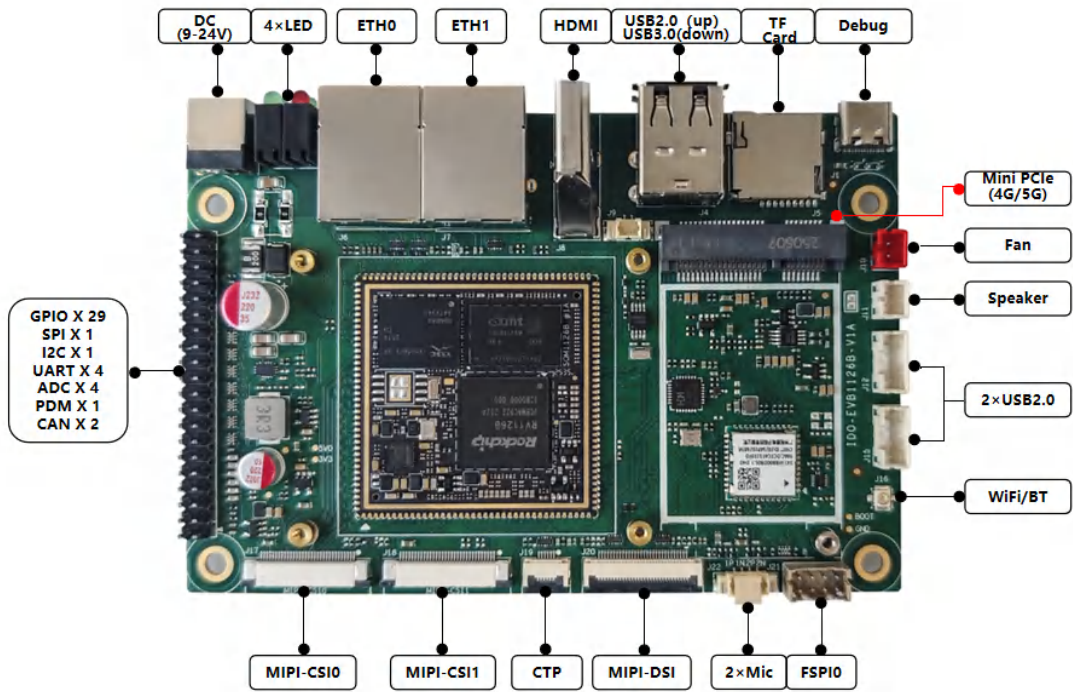
## IDO-EVB1126B-V1 Linux开发手册

深圳触觉智能科技有限公司

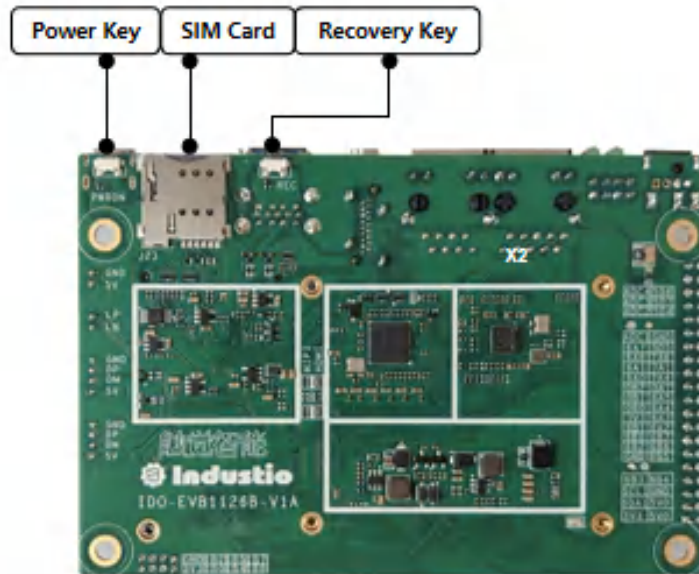
[www.industio.cn](http://www.industio.cn)

| 版本   | PCBA版本 | 修订内容 | 修订  | 审核  | 日期         |
|------|--------|------|-----|-----|------------|
| V1.0 | V1A    | 创建文档 | LJS | IDO | 2025/10/18 |
|      |        |      |     |     |            |

IDO-EVB1126B 是搭载 瑞芯微的RV1126B SoC 是一款高性能视觉处理器SoC ， 适用于机器视觉应用， 特别是与AI相关的应用。 是一款高性能、低功耗的四核应用处理器， 搭载4核Cortex-A53与自研 3Tops算力NPU。



IDO-EVB1126B-V1A 正面



IDO-EVB1126B-V1A 背面

## 1. Linux开发

### 1.1. 获取 SDK

百度网盘:EVB1126B-Linux /4.软件资料/SDK/

### 1.2. 校验md5值

```
Bash |  
1  $ md5sum rv1126b_linux6.1-rkr6-251203.tgz
```

### 1.3. 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~/evb1126b为例

```
1  $ mkdir ~/evb1126b
2  $ cd ~/evb1126b
3  $ tar -xvf rv1126b_linux6.1-rkr6-251203.tgz -C ./
4
5  $ cd evb1126b
6
7  #查看分支(默认分支为ido-evb1126b-v1a)
8  $ git branch
9  * ido-evb1126b-v1a
10     ido-som1126b-v1a
11     master
12
13 #获取代码
14 $ git reset --hard
15 正在更新文件:  20% (42627/212711)
```

注意：解压SDK 不要使用root用户

## 1.4. SDK目录介绍

```
1 $ tree -L 1
2 .
3 |— app
4 |— buildroot
5 |— build.sh -> device/rockchip/common/scripts/build.sh
6 |— common -> device/rockchip/common
7 |— Copyright_Statement.md -> docs/licenses/LICENSE
8 |— debian
9 |— device
10 |— docker.sh
11 |— docs
12 |— external
13 |— hal
14 |— kernel -> kernel-5.10
15 |— kernel-5.10
16 |— Makefile -> device/rockchip/common/Makefile
17 |— output
18 |— prebuilts
19 |— README.md -> device/rockchip/common/README.md
20 |— rkbin
21 |— rkflash.sh -> device/rockchip/common/scripts/rkflash.sh
22 |— rockdev -> output/firmware
23 |— rootfs
24 |— rtos
25 |— tools
26 |— u-boot
27 |— yocto
```

```
1 app: 存放上层应用 APP, 主要是一些应用Demo。
2 buildroot: 基于 Buildroot (2021) 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）

## 1.5. 配置文件介绍

在 `device/rockchip/rv1126b` 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。rockchip\_rv1126b\_evb1126\_v1\_\*defconfig 作为基础的配置。

以rockchip\_rv1126b\_evb1126\_v1\_hdmi\_debian11\_defconfig 做介绍：

```

1  RK_ROOTFS_PREBUILT_TOOLS=y
2  RK_WIFIBT_CHIP="AIC8800_SDIO"
3  RK_UBOOT_SPL=y
4  RK_BUILDROOT_BASE_CFG="evb1126b-v1a"           # buildroot 的defco
nfig 配置文件
5  RK_KERNEL_CFG="rockchip_linux_evb1126b_v1_defconfig" # 内核的defconfig 配
置文件
6  RK_KERNEL_DTS_NAME="ido-evb1126b-v1a-hdmi"      # 内核dts 文件
7  RK_USE_FIT_IMG=y
8  RK_PARAMETER="parameter-debian.txt"
9  RK_ROOTFS_SYSTEM_DEBIAN=y
```

## 1.6. 分区说明

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

```
1  FIRMWARE_VER: 1.0
2  MACHINE_MODEL: RV1126B
3  MACHINE_ID: 007
4  MANUFACTURER: RV1126B
5  MAGIC: 0x5041524B
6  ATAG: 0x00200800
7  MACHINE: 0xffffffff
8  CHECK_MASK: 0x80
9  PWR_HLD: 0,0,A,0,1
10 TYPE: GPT
11 GROW_ALIGN: 0
12 CMDLINE: mtdparts=:0x00002000@0x00004000(uboot),0x00002000@0x00006000(misc),0x00020000@0x00008000(boot),0x00040000@0x00028000(recovery),0x00010000@0x00068000(backup),0x00c00000@0x00078000(userdata),0x00040000@0x00c78000(overlay),-@0x00cb8000(rootfs:grow)
13 uuid:rootfs=614e0000-0000-4b53-8000-1d28000054a9
14 uuid:boot=7A3F0000-0000-446A-8000-702F00006273
```

CMDLINE 属性是我们关注的地方，以 uboot 为例，0x00002000@0x00004000(uboot) 中 0x00002000 为uboot 分区的起始位置，0x00004000 为分区的大小，以此类推。

## 2. 编译

### 2.1. 选择项目

```
1  lijs@dell-PowerEdge-R430:~/work/ljswork/rockchip/rv1126b_linux6.1-9/linux
6.1$ ./build.sh lunch
2  Log colors: message notice warning error fatal
3
4  Log saved at /mnt/work2/lijs/ljswork/rockchip/rv1126b_linux6.1-9/linux6.1/
output/sessions/2025-10-18_15-44-56
5  Pick a defconfig:
6
7  1. rockchip_defconfig
8  2. rockchip_rv1126b_dv_64_evb1_v10_defconfig
9  3. rockchip_rv1126b_evb1126_v1_defconfig
10 4. rockchip_rv1126b_evb1126_v1_hdmi_debian11_defconfig
11 5. rockchip_rv1126b_evb1126_v1_hdmi_defconfig
12 6. rockchip_rv1126b_evb1126_v1_mipi_debian11_defconfig
13 7. rockchip_rv1126b_evb1126_v1_mipi_defconfig
14 8. rockchip_rv1126b_evb1_v10_defconfig
15 9. rockchip_rv1126b_evb4_v10_defconfig
16 10. rockchip_rv1126b_fastboot_defconfig
17 11. rockchip_rv1126b_ipc_32_evb1_v10_defconfig
18 12. rockchip_rv1126b_ipc_64_evb1_v10_defconfig
19 13. rockchip_rv1126b_robot_defconfig
20 14. rockchip_rv1126bp_evb1_v10_defconfig
21 15. rockchip_rv1126bp_fastboot_defconfig
22 16. rockchip_rv1126bp_ipc_32_evb1_v10_defconfig
23 17. rockchip_rv1126bp_ipc_64_evb1_v10_defconfig
24 18. rockchip_rv1126bp_robot_defconfig
25 Which would you like? [1]: 4
26
```

选择对应的数字即可。

**注意：**编译时选择以下这四个，以对应EVB1126B开发板配置

```
1  4. rockchip_rv1126b_evb1126_v1_hdmi_debian11_defconfig
2  5. rockchip_rv1126b_evb1126_v1_hdmi_defconfig
3  6. rockchip_rv1126b_evb1126_v1_mipi_debian11_defconfig
4  7. rockchip_rv1126b_evb1126_v1_mipi_defconfig
```

**注意：**

1.项目命名规则介绍：

▼

Bash

```
1  主板丝印: som1126b。
2  显示接口及屏幕分辨率: mipi-1024x600, hdmi-1920x1080。
3  文件系统支持列表: buildroot, debian。
```

## 2.2. 自动编译

▼

Bash

```
1  $ ./build.sh
```

固件编译后生成路径: rockdev/update.img

## 2.3. 分步编译

### 2.3.1. 编译uboot

▼

Bash

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

固件编译后生成路径: rockdev/uboot.img

### 2.3.2. 编译kernel

▼

Bash

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

固件编译后生成路径: rockdev/boot.img

### 2.3.3. 编译buildroot系统

提供dl.zip 压缩包, 放在<sdk>buildroot/ 目录下, 执行unzip dl.zip 解压

▼ Bash |

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

固件编译后生成路径：rockdev/rootfs.img

### 2.3.4. 编译debian镜像

▼ Bash |

```
1 ./build.sh debian
```

固件编译后生成路径：rockdev/rootfs.img

### 2.3.5. 编译recovery

▼ Bash |

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

固件编译后生成路径：rockdev/recovery.img

### 2.3.6. 固件打包

▼ Bash |

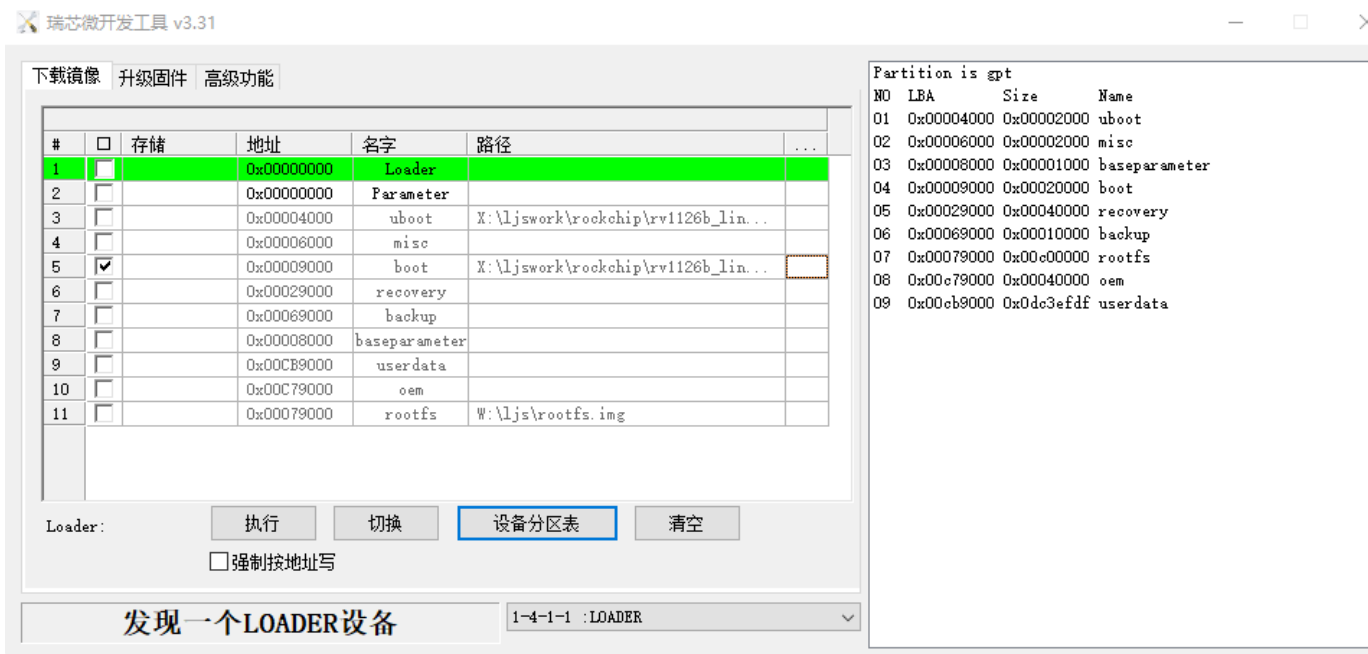
```
1 ./build.sh updateimg
```

## 3. 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-EVB1126B-V1 上手指南》

### 3.1. 整包烧录

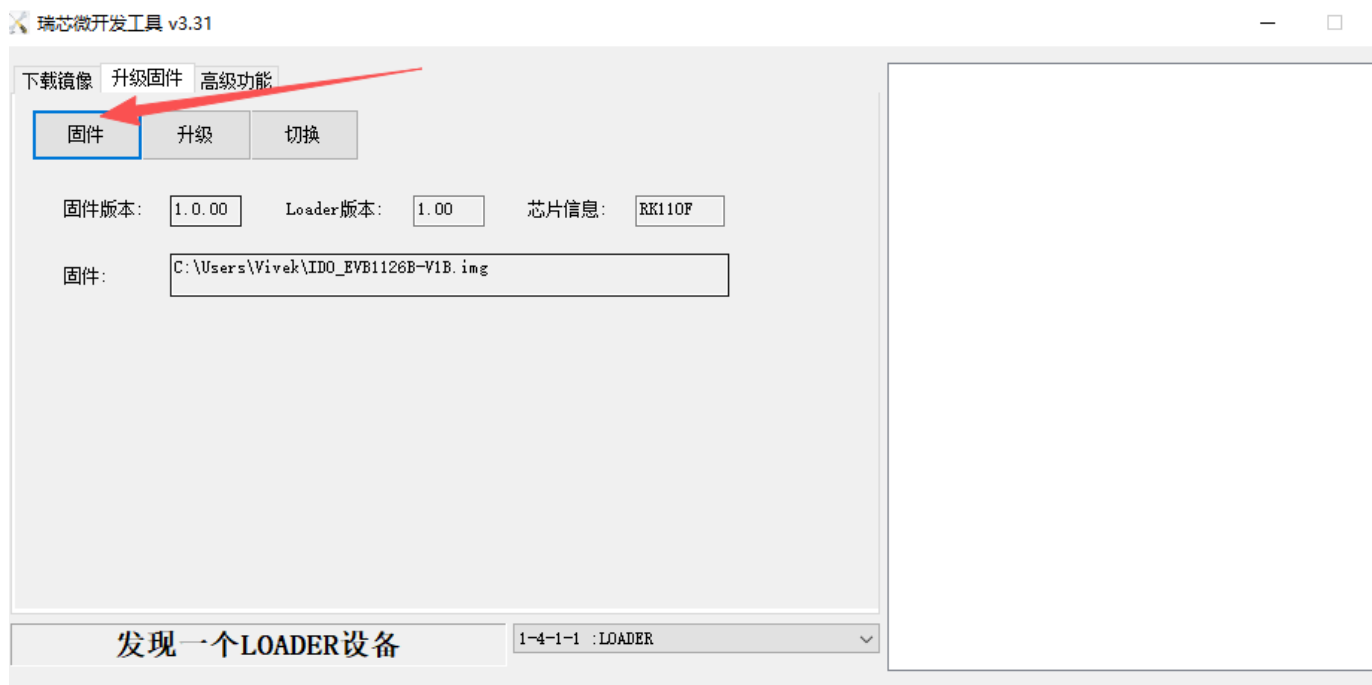
#### 3.1.1. 运行RK开发工具



### 3.1.2. 点击升级固件



### 3.1.3. 点击固件



选择: rockdev/update.img

### 3.1.4. 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。

## 3.2. 分包烧录

### 3.2.1. 更新下载地址

下载镜像 升级固件 高级功能

| #  | <input type="checkbox"/>            | 存储 | 地址         | 名字            | 路径                                 | ... |
|----|-------------------------------------|----|------------|---------------|------------------------------------|-----|
| 1  | <input checked="" type="checkbox"/> |    | 0x00000000 | Loader        |                                    |     |
| 2  | <input type="checkbox"/>            |    | 0x00000000 | Parameter     |                                    |     |
| 3  | <input type="checkbox"/>            |    | 0x00004000 | uboot         | X:\ljswork\rockchip\rv1126b_lin... |     |
| 4  | <input type="checkbox"/>            |    | 0x00006000 | misc          |                                    |     |
| 5  | <input checked="" type="checkbox"/> |    | 0x00009000 | boot          | X:\ljswork\rockchip\rv1126b_lin... |     |
| 6  | <input type="checkbox"/>            |    | 0x00029000 | recovery      |                                    |     |
| 7  | <input type="checkbox"/>            |    | 0x00069000 | backup        |                                    |     |
| 8  | <input type="checkbox"/>            |    | 0x00080000 | baseparameter |                                    |     |
| 9  | <input type="checkbox"/>            |    | 0x00079000 | userdata      |                                    |     |
| 10 | <input type="checkbox"/>            |    | 0x00C79000 | oem           |                                    |     |
| 11 | <input type="checkbox"/>            |    | 0x00CB9000 | rootfs        | W:\ljs\rootfs.img                  |     |

Loader:

执行

切换

设备分区表

清空

☐ 强制按地址写

发现一个LOADER设备

1-4-1-1 :LOADER

下载镜像 升级固件 高级功能

| #  | <input type="checkbox"/>            | 存储 | 地址         | 名字            | 路径                                 | ... |
|----|-------------------------------------|----|------------|---------------|------------------------------------|-----|
| 1  | <input checked="" type="checkbox"/> |    | 0x00000000 | Loader        |                                    |     |
| 2  | <input type="checkbox"/>            |    | 0x00000000 | Parameter     |                                    |     |
| 3  | <input type="checkbox"/>            |    | 0x00004000 | uboot         | X:\ljswork\rockchip\rv1126b_lin... |     |
| 4  | <input type="checkbox"/>            |    | 0x00006000 | misc          |                                    |     |
| 5  | <input checked="" type="checkbox"/> |    | 0x00009000 | boot          | X:\ljswork\rockchip\rv1126b_lin... |     |
| 6  | <input type="checkbox"/>            |    | 0x00029000 | recovery      |                                    |     |
| 7  | <input type="checkbox"/>            |    | 0x00069000 | backup        |                                    |     |
| 8  | <input type="checkbox"/>            |    | 0x00080000 | baseparameter |                                    |     |
| 9  | <input type="checkbox"/>            |    | 0x00079000 | userdata      |                                    |     |
| 10 | <input type="checkbox"/>            |    | 0x00C79000 | oem           |                                    |     |
| 11 | <input type="checkbox"/>            |    | 0x00CB9000 | rootfs        | W:\ljs\rootfs.img                  |     |

Loader:

执行

切换

设备分区表

清空

☐ 强制按地址写

发现一个LOADER设备

1-4-1-1 :LOADER

RKDevTool



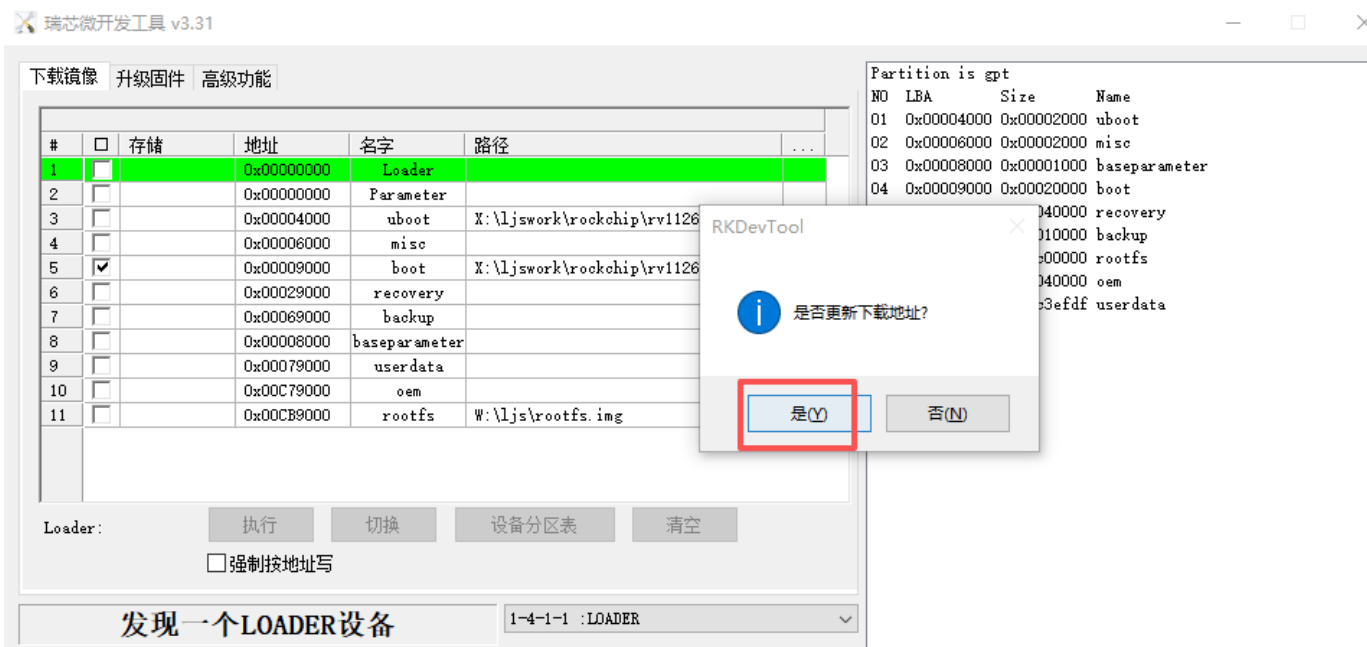
是否更新下载地址?

是(Y)

否(N)

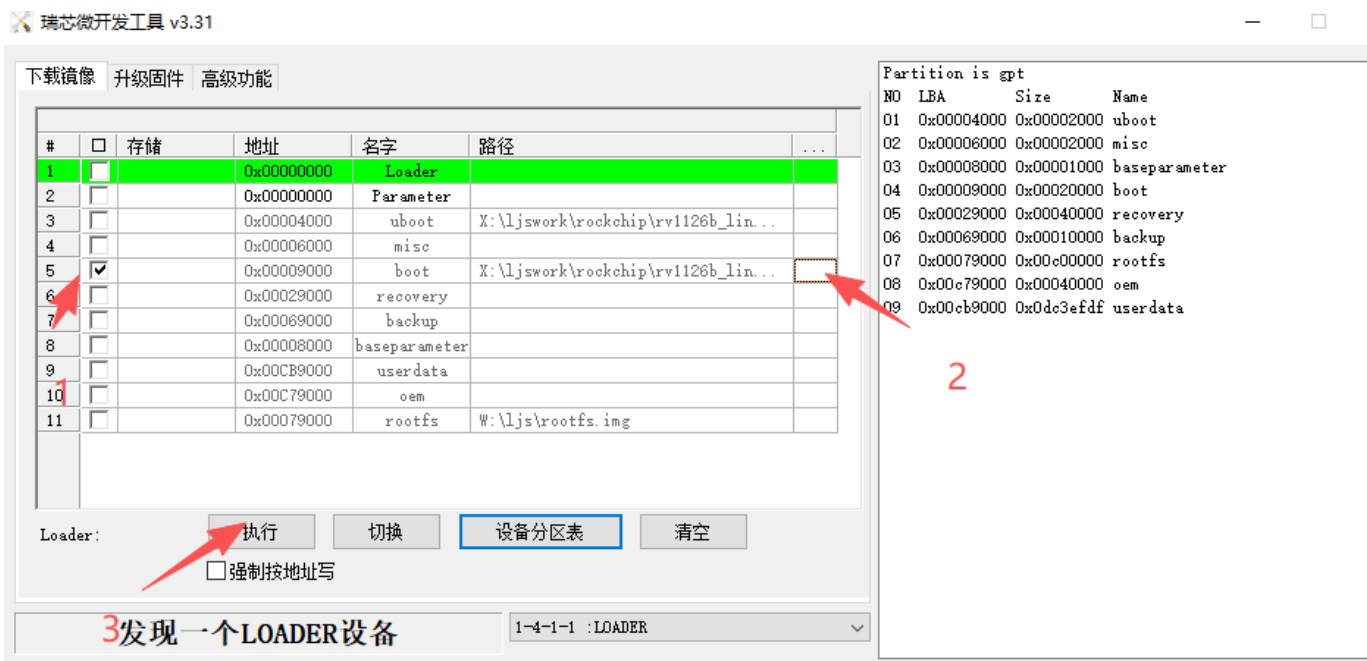
Partition is gpt

| NO | LBA        | Size       | Name          |
|----|------------|------------|---------------|
| 01 | 0x00004000 | 0x00002000 | uboot         |
| 02 | 0x00006000 | 0x00002000 | misc          |
| 03 | 0x00008000 | 0x00001000 | baseparameter |
| 04 | 0x00009000 | 0x00020000 | boot          |
|    |            | 040000     | recovery      |
|    |            | 010000     | backup        |
|    |            | 000000     | rootfs        |
|    |            | 040000     | oem           |
|    |            | 03efdf     | userdata      |



### 3.2.2. 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



## 4. 驱动开发

EVB1126B dts 的层级结构关系

ido-evb1126b-v1a-hdmi.dts

ido-evb1126b-v1a.dtsi

rv1126b.dtsi

ido-som1126b.dtsi

ido-evb1126b-cam-csi0.dtsi

ido-evb1126b-cam-csi1.dtsi

dts使用的一般用法，举例

```
1 &i2c5 {
2     pinctrl-0 = <&i2c5m2_pins>;
3     status = "okay";
4
5     // i2c5 下的设备
6     hym8563: hym8563@51 {
7         status = "okay";
8         compatible = "haoyu,hym8563";
9         reg = <0x51>;
10        #clock-cells = <0>;
11        clock-frequency = <32768>;
12        clock-output-names = "hym8563";
13        wakeup-source;
14    };
15 };
16
17 i2c5 对应cpu 的第五个i2c控制器
18 pinctrl-0 = <&i2c5m2_pins>;表示默认状态下的引脚控制配置
19 status :okay 为启动 disabled: 禁止
20
21 hym8563: hym8563@51 : 定义一个挂载在 I2C5 上的子设备，标签为 hym8563，I2C 地址为 0x51
22 compatible = "haoyu,hym8563"; 与驱动匹配的字符串
23 reg = <0x51>; 地址为0x51
24 clock-frequency = <32768>; 时钟频率为32768
25 clock-output-names = "hym8563"; 输出时钟的名字
26 wakeup-source; 允许该设备作为系统的 唤醒源
```

## 4.1. RTC

### 4.1.1. dts 配置

```
1 ▾ &i2c5 {  
2     pinctrl-0 = <&i2c5m2_pins>;  
3     status = "okay";  
4  
5 ▾ hym8563: hym8563@51 {  
6     status = "okay";  
7     compatible = "haoyu,hym8563";  
8     reg = <0x51>;  
9     #clock-cells = <0>;  
10    clock-frequency = <32768>;  
11    clock-output-names = "hym8563";  
12    wakeup-source;  
13 };  
14 };
```

#### 4.1.2. 测试指令

```
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  2025-05-27
7
8  $ cat /sys/class/rtc/rtc0/time
9  10:10:30
10
11 $ 查看系统时间
12 $ date
13 Tue May 27 18:10:41 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
```

## 4.2. LCD

### 4.2.1. 打开MIPI

```

1  #include <dt-bindings/display/drm_mipi_dsi.h>
2  #include <dt-bindings/display/rockchip_vop.h>
3
4  /{
5
6      board = "RV1126B";
7      model = "Industio EVB1126B Board";
8      compatible = "rockchip,ido-evb1126b-vs1.0.0-20251022", "rockchip,rv11
26b";
9
10     backlight: backlight {
11         compatible = "pwm-backlight";
12         pwms = <&pwm0_8ch_2 0 25000 0>;
13         brightness-levels = <
14             0 20 20 21 21 22 22 23
15             23 24 24 25 25 26 26 27
16             27 28 28 29 29 30 30 31
17             31 32 32 33 33 34 34 35
18             35 36 36 37 37 38 38 39
19             40 41 42 43 44 45 46 47
20             48 49 50 51 52 53 54 55
21             56 57 58 59 60 61 62 63
22             64 65 66 67 68 69 70 71
23             72 73 74 75 76 77 78 79
24             80 81 82 83 84 85 86 87
25             88 89 90 91 92 93 94 95
26             96 97 98 99 100 101 102 103
27             104 105 106 107 108 109 110 111
28             112 113 114 115 116 117 118 119
29             120 121 122 123 124 125 126 127
30             128 129 130 131 132 133 134 135
31             136 137 138 139 140 141 142 143
32             144 145 146 147 148 149 150 151
33             152 153 154 155 156 157 158 159
34             160 161 162 163 164 165 166 167
35             168 169 170 171 172 173 174 175
36             176 177 178 179 180 181 182 183
37             184 185 186 187 188 189 190 191
38             192 193 194 195 196 197 198 199
39             200 201 202 203 204 205 206 207
40             208 209 210 211 212 213 214 215
41             216 217 218 219 220 221 222 223
42             224 225 226 227 228 229 230 231
43             232 233 234 235 236 237 238 239
44             240 241 242 243 244 245 246 247

```

```

45         248 249 250 251 252 253 254 255
46         >;
47         default-brightness-level = <200>;
48     };
49 };
50
51 &pwm0_8ch_2 {
52     status = "okay";
53     pinctrl-0 = <&pwm0m2_ch2_pins>;
54 };
55
56 /*
57  * mipidcphy0 needs to be enabled
58  * when dsi is enabled
59  */
60 &dsi {
61     status = "okay";
62
63     dsi_panel: panel@0 {
64         status = "okay";
65         compatible = "simple-panel-dsi";
66         reg = <0>;
67         //power-supply = <&vcc3v3_lcd_n>;
68         backlight = <&backlight>;
69         reset-gpios = <&gpio5 RK_PA2 GPIO_ACTIVE_LOW>;
70         reset-delay-ms = <10>;
71         enable-delay-ms = <10>;
72         prepare-delay-ms = <10>;
73         unprepare-delay-ms = <10>;
74         disable-delay-ms = <60>;
75         init-delay-ms = <60>;
76
77         width-mm = <68>;
78         height-mm = <121>;
79         dsi,flags = <(MIPI_DSI_MODE_VIDEO | MIPI_DSI_MODE_VIDEO_BURST |
80                     MIPI_DSI_MODE_LPM | MIPI_DSI_MODE_NO_EOT_PACKET)>;
81         dsi,format = <MIPI_DSI_FMT_RGB888>;
82         dsi,lanes = <4>;
83         panel-init-sequence = [
84             05 96 01 11
85             05 14 01 29
86         ];
87
88         disp_timings0: display-timings {
89             native-mode = <&dsi_timing0>;
90             dsi_timing0: timing0 {
91                 clock-frequency = <42000000>;
92                 hactive = <1024>;

```

```

93         vactive = <600>;
94         hfront-porch = <23>;
95         hsync-len = <5>;
96         hback-porch = <50>;
97         vfront-porch = <12>;
98         vsync-len = <1>;
99         vback-porch = <23>;
100        hsync-active = <0>;
101        vsync-active = <0>;
102        de-active = <0>;
103        pixelclk-active = <0>;
104    };
105
106
107    };
108
109    ports {
110        #address-cells = <1>;
111        #size-cells = <0>;
112
113        port@0 {
114            reg = <0>;
115            panel_in_dsi: endpoint {
116                remote-endpoint = <&dsi_out_panel>;
117            };
118        };
119    };
120
121 };
122
123 ports {
124     #address-cells = <1>;
125     #size-cells = <0>;
126
127     port@1 {
128         reg = <1>;
129         dsi_out_panel: endpoint {
130             remote-endpoint = <&panel_in_dsi>;
131         };
132     };
133 };
134 };
135
136 &i2c2 {
137     status = "okay";
138     pinctrl-0 = <&i2c2m0_pins>;
139
140     gt911@14 {

```

```

141         compatible = "goodix,gt9xx";
142         reg = <0x14>;
143         touch-gpio = <&gpio0 RK_PA3 IRQ_TYPE_LEVEL_LOW>;
144         reset-gpio = <&gpio0 RK_PC7 GPIO_ACTIVE_HIGH>;
145         pinctrl-names = "default";
146         pinctrl-0 = <&touch_gpio>;
147         max-x = <1200>;
148         max-y = <1900>;
149         tp-size = <9110>;
150         status = "okay";
151     };
152
153 };
154
155 &display_subsystem {
156     status = "okay";
157 };
158
159 &dsi {
160     status = "okay";
161 };
162
163 &dsi_in_vop {
164     status = "okay";
165 };
166
167 &mipi_dphy {
168     status = "okay";
169 };
170
171 &route_dsi {
172     status = "okay";
173     connect = <&vop_out_dsi>;
174 };
175
176 &pinctrl {
177     touch {
178         touch_gpio: touch-gpio {
179             rockchip,pins =
180                 <0 RK_PA3 RK_FUNC_GPIO &pcfg_pull_up>,
181                 <0 RK_PC7 RK_FUNC_GPIO &pcfg_pull_up>;
182         };
183     };
184 };

```

### 4.2.2. 关闭MIPI

```
1 ▾ &display_subsystem {  
2     status = "disabled";  
3 };  
4  
5 ▾ &dsi {  
6     status = "disabled";  
7 };  
8  
9 ▾ &dsi_in_vop {  
10     status = "disabled";  
11 };
```

## 4.3. Ethernet

```

1  &mdio {
2      phy1: phy1@1 {
3          compatible = "ethernet-phy-ieee802.3-c22";
4          reg = <0x1>;
5          clocks = <&cru CLK_MAC_OUT2I0>;
6      };
7  };
8
9  &gmac {
10     phy-mode = "rgmii";
11     clock_in_out = "output";
12
13     snps,reset-gpio = <&gpio5 RK_PD0 GPIO_ACTIVE_LOW>;
14     snps,reset-active-low;
15     /* Reset time is 20ms, 100ms for rtl8211f */
16     snps,reset-delays-us = <0 20000 100000>;
17     assigned-clocks = <&cru CLK_MAC_OUT2I0>;
18     assigned-clock-rates = <25000000>;
19
20
21     pinctrl-names = "default";
22     pinctrl-0 = <&ethm1_miim_pins
23         &ethm1_tx_bus2_pins
24         &ethm1_rx_bus2_pins
25         &ethm1_rgmii_clk_pins
26         &ethm1_rgmii_bus_pins
27         &eth_clk_25mm1_out_pins
28         &gmac_reset>;
29
30     tx_delay = <0x36>;
31     rx_delay = <0x2c>;
32
33     phy-handle = <&phy1>;
34     status = "okay";
35 };

```

## 4.4. CAN

### 4.4.1. CAN dts 配置

```

1 ▾ &can0 {
2     compatible = "rockchip,rv1126b-canfd";
3     assigned-clocks = <&cru CLK_CAN0>;
4     assigned-clock-rates = <200000000>;
5     pinctrl-names = "default";
6     pinctrl-0 = <&can0m0_pins>;
7     status = "okay";
8 };
9
10 ▾ &can1 {
11     compatible = "rockchip,rv1126b-canfd";
12     assigned-clocks = <&cru CLK_CAN1>;
13     assigned-clock-rates = <200000000>;
14     pinctrl-names = "default";
15     pinctrl-0 = <&can1m0_pins>;
16     status = "okay";
17 };

```

#### 4.4.2. CAN 测试

```

1  #关闭can0设备
2  ip link set can0 down
3
4  #普通can协议 (1.0)
5  ip link set can0 type can bitrate 1000000 dbitrate 1000000 fd on
6
7  #查看can信息, 波特率等等
8  ip -details link show can0
9
10 #启动can0
11 ip link set can0 up
12
13 #执行candump, 阻塞等待can0接收
14 candump can0
15
16 #can格式发送
17 cansend can0 123#DEADBEEF12345679

```

## 4.5. UART

### 4.5.1. Uart dts 配置

```
1  /* ttl */
2  &uart4 {
3      status = "okay";
4      pinctrl-names = "default";
5      pinctrl-0 = <&uart4m2_xfer_pins>;
6  };
7
8  /* 485 */
9  &uart5 {
10     status = "okay";
11     pinctrl-names = "default";
12     pinctrl-0 = <&uart5m2_xfer_pins &uart5m2_rtsn_pins>;
13     linux,rs485-enabled-at-boot-time;
14 };
15 /* ttl */
16 &uart6 {
17     status = "okay";
18     pinctrl-names = "default";
19     pinctrl-0 = <&uart6m1_xfer_pins>;
20 };
21 /* ttl */
22 &uart7 {
23     status = "okay";
24     pinctrl-names = "default";
25     pinctrl-0 = <&uart7m1_xfer_pins>;
26 };
```

### 4.5.2. uart 测试

```
1  microcom -s 115200 -p /dev/ttyS4
```

### 4.5.3. 应用开发参考

应用demo获取：百度网盘/EVB1126B/EVB1126B-Linux/4.软件资料/工具/test\_serial.zip

代码解压

```
-rwxrw-r-- 1 lijs lijs 12159 6月 27 17:10 main.c
-rw-rw-r-- 1 lijs lijs 1817 9月 18 16:14 Makefile
drwxrwxr-x 2 lijs lijs 4096 6月 12 13:43 src
-rwxrwxr-x 1 lijs lijs 29968 9月 18 16:14 test_serial
```

▼ Makefile

C |

```
1 CC=/mnt/work2/lijs/ljswork/rockchip/rv1126b_linux6.1/linux6.1/prebuilts/gcc/linux-x86/aarch64/gcc-arm-10.3-2021.07-x86_64-aarch64-none-linux-gnu/bin/aarch64-none-linux-gnu-gcc
2 LIBS = -lpthread
3
4
5 ifeq ($(CC), )
6     CC = gcc
7 endif
8
9 serial_test: main.c src/*.h src/*.c
10     #$(CC) $(CFLAGS) -o test_serial main.c src/*.h src/*.c ${LIBS}
11     $(CC) -o test_serial main.c src/*.h src/*.c ${LIBS}
12 clean:
13     rm serial_test -rf
14
```

其中CC 指定evb1126b 的sdk目录下的交叉编译器，执行make 编译

```
1  int main(int argc, char *argv[])
2  {
3      ...
4      signal(SIGINT, stop_loop); // 监听ctrl + c
5      if (rxplen == 0 && txplen == 0) {
6          usage();
7          return EXIT_FAILURE;
8      } else if (rxplen != 0 && txplen == 0) { // 单个串口rx测试
9          strcpy(rx.port, rxpath);
10         rx.loop = -1;
11         singel_rx(&rx);
12     } else if (rxplen == 0 && txplen != 0) { // 单个串口tx测试
13         strcpy(tx.port, txpath);
14         tx.loop = -1;
15         singel_tx(&tx);
16     } else { // 两个串口 rx tx 测试。
17         strcpy(rx.port, rxpath);
18         strcpy(tx.port, txpath);
19         tx2rx(&tx, &rx);
20         if (txdone == 1 && rx.result == 0) {
21             DEV_PRINT("tx:%s, rx:%s  check succ\n", tx.port, rx.port);
22             return EXIT_SUCCESS;
23         } else {
24             DEV_PRINT("tx:%s, rx:%s  check fail\n", tx.port, rx.port);
25             return EXIT_FAILURE;
26         }
27     }
28     ...
29 }
```

```
1
2 // 设置波特率, 用户可以在这里添加自己想要用的波特率
3 void set_baudrate(struct termios * opt, unsigned int baudrate)
4 {
5     switch(baudrate)
6     {
7
8     case 300:
9         cfsetispeed(opt, B300);
10        cfsetospeed(opt, B300);
11        break;
12    case 600:
13        cfsetispeed(opt, B600);
14        cfsetospeed(opt, B600);
15        break;
16    case 1200:
17        cfsetispeed(opt, B1200);
18        cfsetospeed(opt, B1200);
19        break;
20
21    case 2400:
22        cfsetispeed(opt, B2400);
23        cfsetospeed(opt, B2400);
24        break;
25    case 4800:
26        cfsetispeed(opt, B4800);
27        cfsetospeed(opt, B4800);
28        break;
29    case 9600:
30        cfsetispeed(opt, B9600);
31        cfsetospeed(opt, B9600);
32        break;
33    case 115200:
34        cfsetispeed(opt, B115200);
35        cfsetospeed(opt, B115200);
36        break;
37    case 460800:
38        cfsetispeed(opt, B460800);
39        cfsetospeed(opt, B460800);
40        break;
41    case 500000:
42        cfsetispeed(opt, B500000);
43        cfsetospeed(opt, B500000);
44        break;
45    case 1000000:
```

```

46         cfsetispeed(opt, B1000000);
47         cfsetospeed(opt, B1000000);
48         break;
49     case 1500000:
50         cfsetispeed(opt, B1500000);
51         cfsetospeed(opt, B1500000);
52         break;
53     case 2000000:
54         cfsetispeed(opt, B2000000);
55         cfsetospeed(opt, B2000000);
56         break;
57     case 4000000:
58         cfsetispeed(opt, B4000000);
59         cfsetospeed(opt, B4000000);
60         break;
61     default:
62         cfsetispeed(opt, B9600);
63         cfsetospeed(opt, B9600);
64         break;
65 }
66 }
67
68 void set_data_bit(struct termios *opt, unsigned int databit)
69 {
70     opt->c_cflag &= ~CSIZE;
71     switch (databit) {
72         case 8:
73             opt->c_cflag |= CS8;
74             break;
75         case 7:
76             opt->c_cflag |= CS7;
77             break;
78         case 6:
79             opt->c_cflag |= CS6;
80             break;
81         case 5:
82             opt->c_cflag |= CS5;
83             break;
84         default:
85             opt->c_cflag |= CS8;
86             break;
87     }
88 }
89
90 //设置数据长度
91 void set_parity(struct termios *opt, char parity)
92 {
93     switch (parity) {

```

```

94         case 'n':
95         case 'N':                                /*no parity check */
96             opt->c_cflag &= ~PARENB;
97         opt->c_iflag &= ~INPCK;
98             break;
99         case 'e':
100         case 'E':                                /* 偶效验 */
101             opt->c_cflag |= PARENB;
102             opt->c_cflag &= ~PARODD;
103             opt->c_iflag |= INPCK;
104             break;
105         case 'o':
106         case 'O':                                /* 奇效验 */
107             opt->c_cflag |= (PARODD | PARENB);
108             opt->c_iflag |= INPCK;
109             break;
110         default:                                /* no parity check */
111             opt->c_cflag &= ~PARENB;
112         opt->c_iflag &= ~INPCK;
113             break;
114     }
115 }
116
117 //设置停止位
118 void set_stopbit(struct termios *opt, char stopbit)
119 {
120     if (stopbit == 1) {
121         opt->c_cflag &= ~CSTOPB; /* 1 stop bit */
122     } else if (stopbit == 2) {
123         opt->c_cflag |= CSTOPB; /* 2 stop bits */
124     } else {
125         opt->c_cflag &= ~CSTOPB; /* 1 stop bit */
126     }
127 }
128
129 int set_port_attr(int fd, unsigned int baudrate, int databit, int stopbi
130 t, char parity, int vtime, int vmin)
131 {
132     struct termios opt;
133     tcgetattr(fd, &opt);
134     //设置波特率
135     set_baudrate(&opt, baudrate);
136     opt.c_cflag |= CLOCAL | CREAD;
137     //设置数据位
138     set_data_bit(&opt, databit);
139     //设置校验位
140     set_parity(&opt, parity);
    //设置停止位

```

```

141     set_stopbit(&opt, stopbit);
142     //设置无硬件流控
143     opt.c_cflag &= ~CRTSCTS;
144     //设置无软件流控
145     opt.c_iflag &= ~(IXON | IXOFF | IXANY);
146     //选择原始输入输出
147     opt.c_lflag &= ~(ICANON | ECHO | ECHOE | ISIG);
148     opt.c_oflag &= ~OPOST;
149     // 防止0x0D变0x0A
150     opt.c_iflag &= ~(ICRNL|IGNCR);
151
152     //其他设置
153     opt.c_cc[VTIME] = vtime;
154     opt.c_cc[VMIN] = vmin;
155     tcflush(fd, TCIFLUSH);
156     if (tcsetattr(fd, TCSANOW, &opt) != 0)
157         return EXIT_FAILURE;
158     return EXIT_SUCCESS;
159 }

```

#### 4.5.3.1. 测试

```

1  root@rv1126b-buildroot:/# test_serial
2  serial test:
3      -h print this help
4      -t <str>  tx device path
5      -r <str>  rx device path
6      -b <int>  set baudrate, default:115200, support: 600,1200,2400, 4800,
9600, 115200, 460800,1500000,2000000,4000000
7      -d <int>  set databit, default:8, support:7, 8
8      -s <int>  set stopbit, default:1, support:1, 2
9      -p <int>  set parity, default:0, 0:no parity check, 1:even, 2:odd
10     -l <int>  tx loop count, default:5 only for double serials test
11     -o <int>  set rx out time, default:3 second, only for double serials test
12     Example: ./test_serial -t /dev/ttyS1 -r /dev/ttyS2
13

```

Example: `./test_serial -t /dev/ttyS1 -r /dev/ttyS2`

表示 ttyS1 发送数据，ttyS2 接受数据 如若不指定波特率，则默认设置为115200

也可以执行



Shell |

```
1  ./test_serial -t /dev/ttyS1 -r /dev/ttyS2 -b 9600
```

ttyS1循环发送数据，并配置波特率为1500000



Shell |

```
1  ./test_serial -t /dev/ttyS1 -b 1500000
```

ttyS1接收数据，并配置波特率为1500000



Shell |

```
1  ./test_serial -r /dev/ttyS1 -b 1500000
```