

Rockchip RV1126/RV1109 Linux SDK 快速入门

文档标识: RK-JC-YF-360

发布版本: V0.0.1

日期: 2020-04-28

文件密级: ☐绝密 ☐秘密 ☐内部资料 ☒公开

免责声明

本文档按“现状”提供, 福州瑞芯微电子股份有限公司 (“本公司”, 下同) 不对本文档的任何陈述、信息和内容的准确性、可靠性、完整性、适销性、特定目的性和非侵权性提供任何明示或暗示的声明或保证。本文档仅作为使用指导的参考。

由于产品版本升级或其他原因, 本文档将可能在未经任何通知的情况下, 不定期进行更新或修改。

商标声明

“Rockchip”、“瑞芯微”、“瑞芯”均为本公司的注册商标, 归本公司所有。

本文档可能提及的其他所有注册商标或商标, 由其各自拥有者所有。

版权所有 © 2020 福州瑞芯微电子股份有限公司

超越合理使用范畴, 非经本公司书面许可, 任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部, 并不得以任何形式传播。

福州瑞芯微电子股份有限公司

Fuzhou Rockchip Electronics Co., Ltd.

地址: 福建省福州市铜盘路软件园A区18号

网址: www.rock-chips.com

客户服务电话: +86-4007-700-590

客户服务传真: +86-591-83951833

客户服务邮箱: fae@rock-chips.com

前言

概述

本文主要描述了RV1126/RV1109 Linux SDK的基本使用方法，旨在帮助开发者快速了解并使用RV1126/RV1109 SDK开发包。

产品版本

芯片名称	内核版本
RV1126/RV1109	Linux 4.19

读者对象

本文档（本指南）主要适用于以下工程师：

技术支持工程师

软件开发工程师

修订记录

版本号	作者	修改日期	修改说明
V0.0.1	CWW	2020-04-28	初始版本

目录

Rockchip RV1126/RV1109 Linux SDK 快速入门

前言

目录

1 开发环境搭建

2 SDK目录结构说明

3 SDK编译说明

3.1 查看编译命令

3.1 U-Boot编译

3.2 Kernel编译

3.4 Recovery编译

3.5 Rootfs编译

3.6 固件打包

3.7 全自动编译

4 刷机说明

4.1 EVB板正面示意图

4.2 EVB板背面示意图

4.3 硬件接口功能表

4.4 Windows 刷机说明

4.5 Linux 刷机说明

5 EVB板功能说明

5.1 如何访问3路RTSP和1路RTMP网络码流

5.1.1 使用串口或ADB连上EVB板子获取设备IP地址

5.1.2 使用RK IPCamera Tool获取设备IP地址

5.1.3 访问网络码流

5.2 如何通过网页访问设备信息

5.3 如何测试人脸识别功能

1 开发环境搭建

Ubuntu 16.04系统： 编译环境搭建所依赖的软件包以及安装命令如下：

```
sudo apt-get install repo git-core gitk git-gui gcc-arm-linux-gnueabi u-boot-  
tools device-tree-compiler gcc-aarch64-linux-gnu mtools parted libudev-dev  
libusb-1.0-0-dev python-linaro-image-tools linaro-image-tools autoconf  
autotools-dev libsigsegv2 m4 intltool libdrm-dev curl sed make binutils build-  
essential gcc g++ bash patch gzip gawk bzip2 perl tar cpio python unzip rsync  
file bc wget libncurses5 libqt4-dev libglib2.0-dev libgtk2.0-dev libglade2-dev  
cvs git mercurial rsync openssh-client subversion asciidoc w3m dlatex graphviz  
python-matplotlib libc6:i386
```

Ubuntu 17.04系统： 除了上述软件包外还需如下依赖包：

```
sudo apt-get install lib32gcc-7-dev g++-7 libstdc++-7-dev
```

2 SDK目录结构说明

进入工程目录下有buildroot、app、kernel、u-boot、device、docs、external等目录。每个目录或其子目录会对应一个git工程，提交需要在各自的目录下进行。

- buildroot：定制根文件系统。
- app：存放上层应用程序。
- external：相关库，包括音频、视频等。
- kernel：kernel代码。
- device/rockchip：存放每个平台的一些编译和打包固件的脚本和预备文件。
- docs：存放开发指导文件、平台支持列表、工具使用文档、Linux 开发指南等。
- prebuilts：存放交叉编译工具链。
- rkbin：存放固件和工具。
- rockdev：存放编译输出固件。
- tools：存放一些常用工具。
- u-boot：U-Boot代码。

3 SDK编译说明

3.1 查看编译命令

在根目录执行命令：./build.sh -h|help

```
./build.sh help  
Usage: build.sh [OPTIONS]  
Available options:  
BoardConfig*.mk  -switch to specified board config  
uboot             -build uboot  
spl              -build spl  
kernel           -build kernel  
modules          -build kernel modules  
toolchain        -build toolchain  
rootfs           -build default rootfs, currently build buildroot as default  
buildroot        -build buildroot rootfs  
ramboot          -build ramboot image
```

```

multi-npu_boot    -build boot image for multi-npu board
yocto             -build yocto rootfs
debian           -build debian9 stretch rootfs
distro           -build debian10 buster rootfs
pcba             -build pcba
recovery         -build recovery
all              -build uboot, kernel, rootfs, recovery image
cleanall         -clean uboot, kernel, rootfs, recovery
firmware         -pack all the image we need to boot up system
updateimg        -pack update image
otapackage       -pack ab update otapackage image
save            -save images, patches, commands used to debug
allsave         -build all & firmware & updateimg & save

Default option is 'allsave'.

```

查看部分模块详细编译命令，例如：./build.sh -h kernel

```

./build.sh -h kernel
###Current SDK Default [ kernel ] Build Command###
cd kernel
make ARCH=arm rv1109_defconfig
make ARCH=arm rv1126-evb-ddr3-v10.img -j12

```

3.1 U-Boot编译

U-Boot编译命令：./build.sh uboot

```

### 查看U-Boot详细编译命令
./build.sh -h uboot
###Current SDK Default [ uboot ] Build Command###
cd u-boot
./make.sh rv1126

```

3.2 Kernel编译

Kernel编译命令：./build.sh kernel

```

### 查看Kernel详细编译命令
./build.sh -h kernel
###Current SDK Default [ kernel ] Build Command###
cd kernel
make ARCH=arm rv1109_defconfig
make ARCH=arm rv1126-evb-ddr3-v10.img -j12

```

3.4 Recovery编译

Recovery编译命令：./build.sh recovery

```
### 查看Recovery详细编译命令
./build.sh -h recovery
###Current SDK Default [ recovery ] Build Command###
source envsetup.sh rockchip_rv1126_rv1109_recovery
/home/user/sdk/device/rockchip/common/mk-ramdisk.sh recovery.img
rockchip_rv1126_rv1109_recovery
```

3.5 Rootfs编译

Rootfs编译命令：./build.sh rootfs

```
### 查看Roofs详细编译命令
./build.sh -h rootfs
###Current SDK Default [ rootfs ] Build Command###
source envsetup.sh rockchip_rv1126_rv1109
make
```

3.6 固件打包

固件打包命令：./mkfirmware.sh

固件目录：rockdev

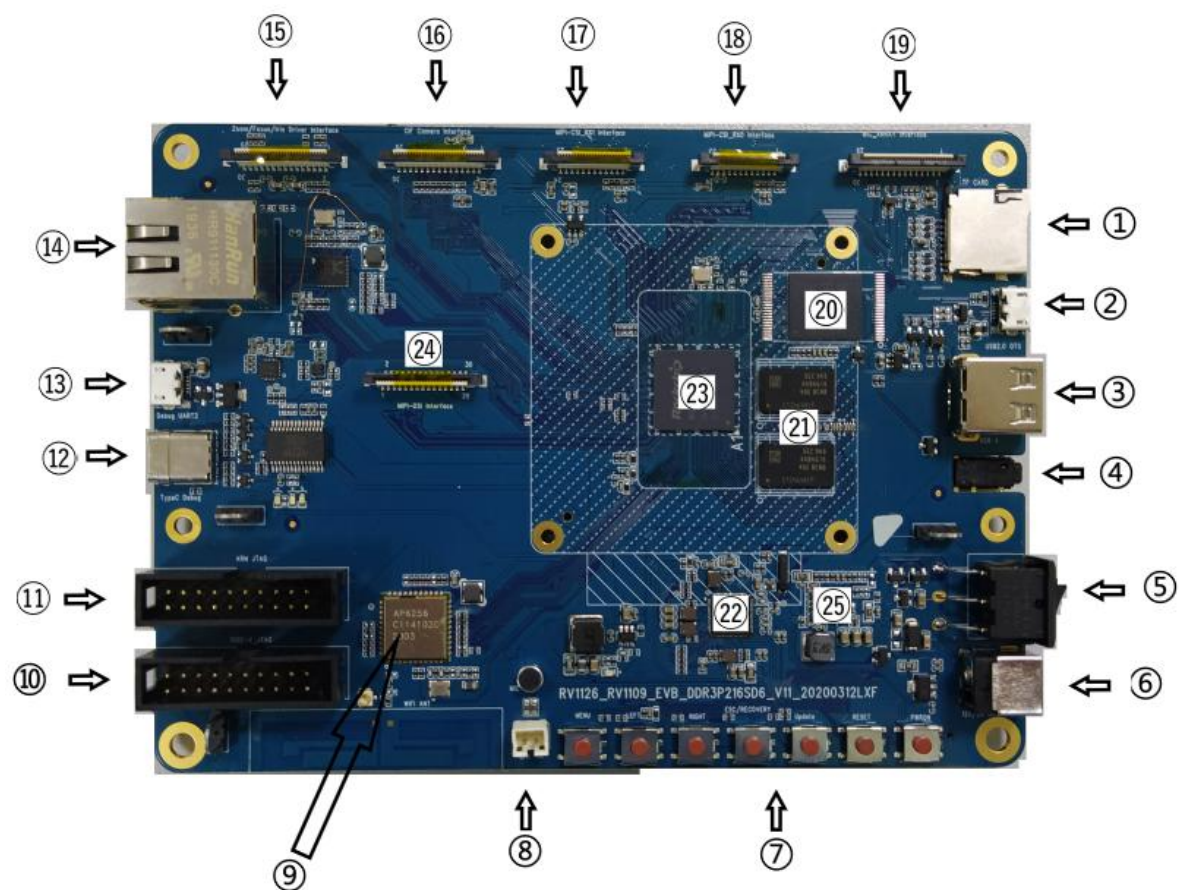
3.7 全自动编译

进入工程根目录执行以下命令自动完成所有的编译：

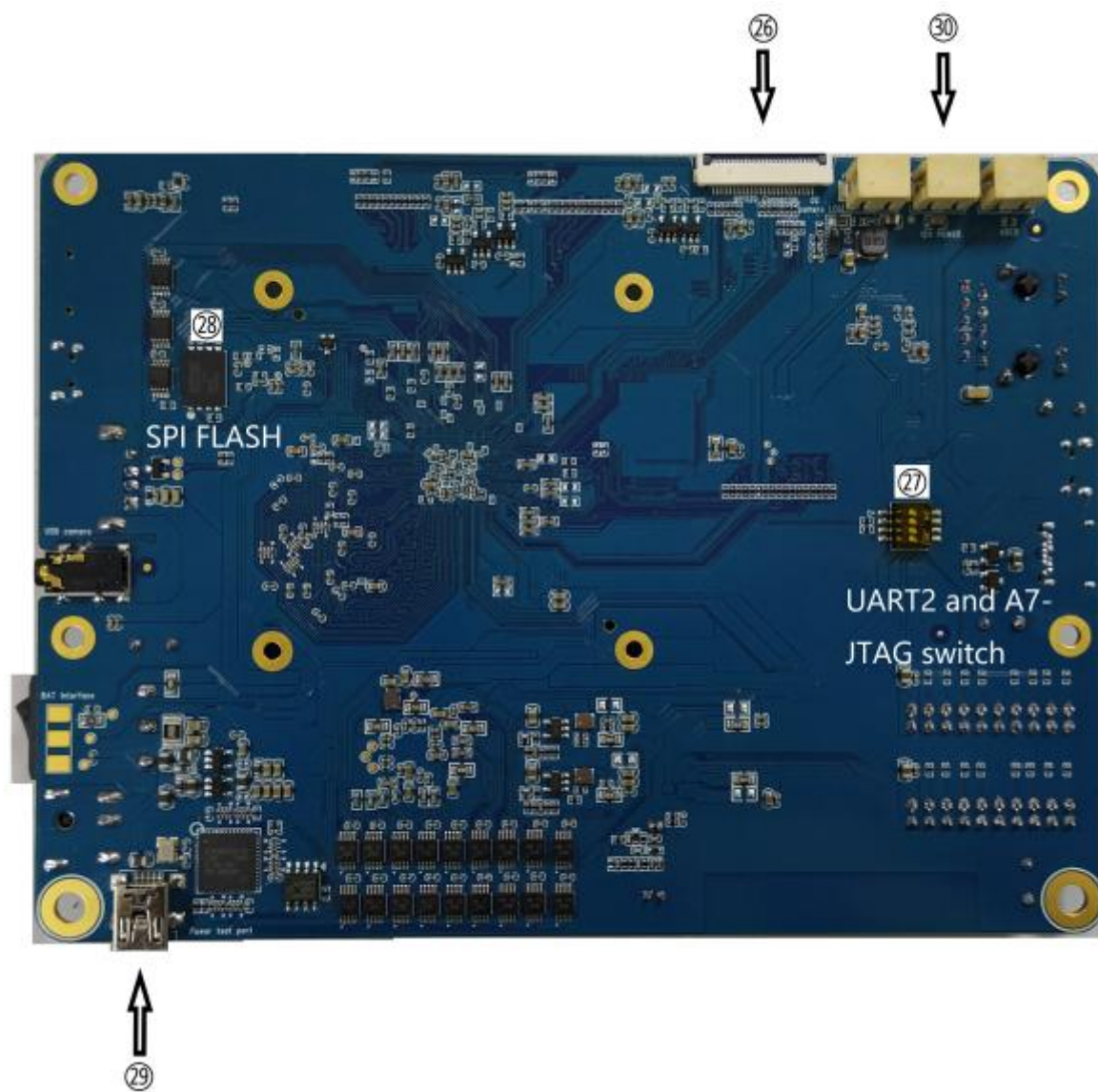
```
./build.sh all
```

4 刷机说明

4.1 EVB板正面示意图



4.2 EVB板背面示意图



4.3 硬件接口功能表

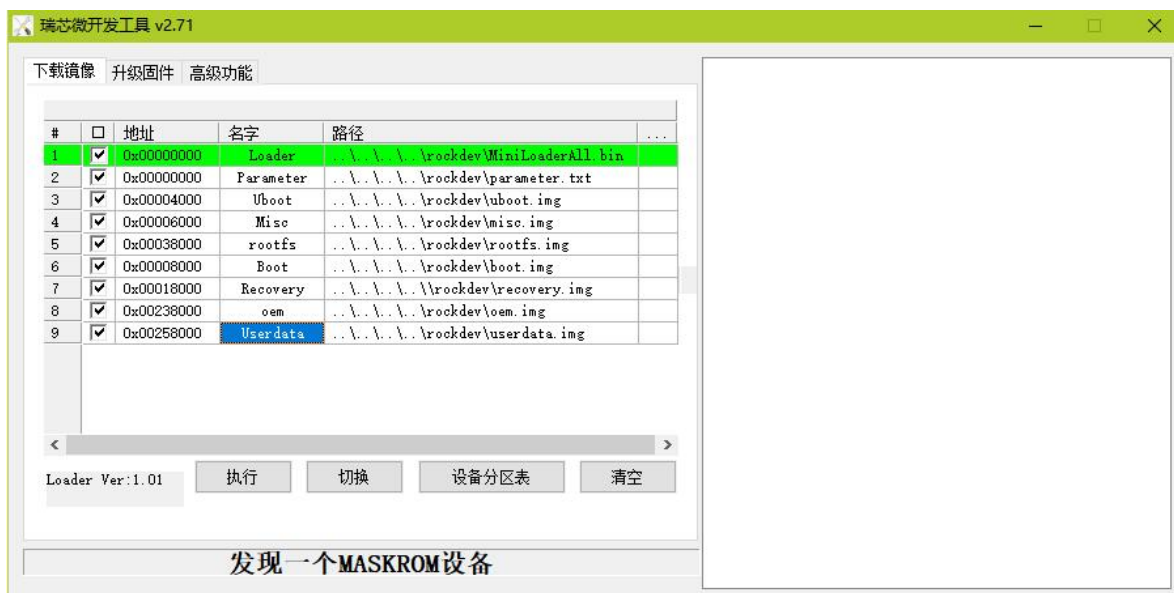
序号 Item	功能部分 Function Part	要求 Requirement
1	TF Card	正常识别TF Card
2	USB Micro-B Port	可以认到ADB设备，可以下载固件
3	USB Type-A Port	可以识别device设备，且功能正常
4	USB camera input	正常识别USB camera
5	the boat switch	直流适配器输入的12v电源，可以通过船型开关来控制打开或关闭
6	12V power supply input	
7	KEY BAORD	所有按键功能正常
8	CLASS D output	喇叭功能正常
9	WI-FI/BT	AP6256模组功能正常
10	RISC-V JTAG	芯片验证调试
11	V7-JTAG	
12	TPYEC	
13	USB Micro-B Port	串口可以正常输入和输出
14	以太网Ethernet	网络连接正常
15	Zoom/ Iris Driver Interface	开发板预留Zoom/Focus/Iris连接座，方便客户进行CAMERA设备调试开发。
16	CIF camera	摄像头功能正常，CIF摄像头输入
17	MIPI Camera 1	摄像头功能正常，默认MIPI摄像头输入
18	MIPI Camera 2	摄像头功能正常，默认MIPI摄像头输入
19	MIC-ARRAY	MIC陈列输入
20	eMMC Flash	可以正常识别容量16GByte
21	DDR DDR3	可识别到总容量8Gbit
22	PMIC RK809-2	各路电源正常输出，电池电量检测准确
23	CPU	RV1126_RV1109
24	MIPI屏 MIPI panel	屏幕图像显示正常
25	BQ24171	双节电池充放电正常
Bottom Layer		
26	BT1120 Camera	摄像头功能正常，BT1120 摄像头输入
27	功能切换SWITCH	UART2和A7-JTAG功能切换
28	SPI flash	验证SPI flash功能
29	USB Micro-B Port	用于功耗测试
30	Camera_LED 驱动输出	Warm up lamp drive

4.4 Windows 刷机说明

SDK 提供 Windows 烧写工具(工具版本需要 V2.71 或以上)，工具位于工程根目录：

```
tools/
└─ windows/AndroidTool
```

如下图，编译生成相应的固件后，设备烧写需要进入 MASKROM 或 BootROM 烧写模式，连接好 USB 下载线后，按住按键“Update”不放并按下复位键“RESET”后松手，就能进入 MASKROM 模式，加载编译生成固件的相应路径后，点击“执行”进行烧写，也可以按“recovery”按键不放并按下复位键“RESET”后松手进入 loader 模式进行烧写，下面是 MASKROM 模式的分区偏移及烧写文件。(注意：Windows PC 需要在管理员权限运行工具才可执行)



注：烧写前，需安装最新 USB 驱动，驱动详见：

<SDK>/tools/windows/DriverAssitant_v4.91.zip

4.5 Linux 刷机说明

Linux 下的烧写工具位于 tools/linux 目录下(Linux_Upgrade_Tool 工具版本需要 V1.49 或以上)，请确认你的板子连接到 MASKROM/loader rockusb。比如编译生成的固件在 rockdev 目录下，升级命令如下：

```
sudo ./upgrade_tool ul rockdev/MiniLoaderAll.bin
sudo ./upgrade_tool di -p rockdev/parameter.txt
sudo ./upgrade_tool di -u rockdev/uboot.img
sudo ./upgrade_tool di -misc rockdev/misc.img
sudo ./upgrade_tool di -b rockdev/boot.img
sudo ./upgrade_tool di -recovery rockdev/recovery.img
sudo ./upgrade_tool di -oem rockdev/oem.img
sudo ./upgrade_tool di -rootfs rockdev/rootfs.img
sudo ./upgrade_tool di -userdata rockdev/userdata.img
sudo ./upgrade_tool rd
```

或升级整个 firmware 的 update.img 固件：

```
sudo ./upgrade_tool uf rockdev/update.img
```

或在根目录，机器在 MASKROM 状态运行如下升级：

```
./rkflash.sh
```

5 EVB板功能说明

EVB板支持如下功能：

- 支持3路RTSP和1路RTMP网络码流
- 支持本地屏幕1280x720显示
- 支持保存主码流到设备
- 支持网页端访问设备
- 支持人脸识别

5.1 如何访问3路RTSP和1路RTMP网络码流

使用网线接到EVB板的网口，上电开机。默认会自动获取IP地址。

5.1.1 使用串口或ADB连上EVB板子获取设备IP地址

```
ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 02:E0:F9:16:7E:E9
          inet addr:172.16.21.218  Bcast:172.16.21.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:199225 errors:0 dropped:2231 overruns:0 frame:0
          TX packets:372371 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:20874811 (19.9 MiB)  TX bytes:522220899 (498.0 MiB)
          Interrupt:56
```

使用串口连接EVB板子的PC端配置如下：

波特率：1500000
数据位：8
停止位：1
奇偶校验：none
流控：none

5.1.2 使用RK IPCamera Tool获取设备IP地址

安装SDK目录tools/windows/RK_IPCamera_Tool-V1.1.zip工具。打开工具，通过EVB板网口连接到电脑所在局域网，查看RK IPCamera Tool工具设备总数列表获取设备IP地址。



5.1.3 访问网络码流

使用支持RTSP或RTMP的播放器访问，例如（VLC播放器）。

RTSP访问地址：

- rtsp://设备IP地址/live/mainstream
- rtsp://设备IP地址/live/substream
- rtsp://设备IP地址/live/thirdstream

RTMP访问地址：

- rtmp://设备IP地址:1935/live/substream

5.2 如何通过网页访问设备信息

打开Web浏览器（推荐Chrome浏览器）访问地址：

```
http://设备IP地址
```

网页端详细的操作说明请参考SDK目录docs下的文档。

5.3 如何测试人脸识别功能

使用播放器访问RTSP主码流：rtsp://设备IP地址/live/mainstream

SDK的人脸识别功能默认授权的测试时间是30~60分钟，授权失效后主码流预览会有“人脸算法软件未授权”提示，需要重启才能再测试。