

Rockchip Linux Edge SDK Debian 开发指导

文件标识: RK-KF-YF-862

发布版本: V0.1.1

日期: 2022-03-25

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, 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, 由其各自所有者所有。

版权所有 © 2022 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。

瑞芯微电子股份有限公司

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

本文为基于边缘计算SDK Debian11，提供开机启动、紧急修复和系统软件包使用和开发指导。

产品版本

芯片名称	内核版本
RK3588	linux-5.10

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
V0.1.0	Addy Ke	2022-01-27	初始版本
V0.1.1	Addy Ke	2022-03-25	增加更多系统软件包帮助

目录

Rockchip Linux Edge SDK Debian 开发指导

1. 系统要求
2. 开机启动
 - 2.1 登录账号
3. 远程登录
 - 3.1 adb登录
 - 3.2 ssh登录
4. 紧急模式
 - 4.1 进入紧急模式
 - 4.2 命令行操作
5. 添加APT源
6. 添加PIP源
7. 系统软件包
 - 7.1 Toybrick Property
 - 7.1.1 软件包说明
 - 7.1.2 软件包
 - 7.1.2.1 toybrick-prop
 - 7.1.2.2 toybrick-prop-bin
 - 7.1.2.3 toybrick-prop-dev
 - 7.2 Toybrick Usbconfig
 - 7.2.1 软件包说明
 - 7.2.2 软件包
 - 7.2.2.1 toybrick-usbd
 - 7.3 Rockchip ISP
 - 7.3.1 软件包说明
 - 7.3.2 软件包
 - 7.3.2.1 rockchip-isp
 - 7.3.2.2 rockchip-isp-sample
 - 7.4 Edge Utils
 - 7.4.1 软件包说明
 - 7.4.2 软件包
 - 7.4.2.1 edge-utils
 - 7.5 Toybrick Vendor
 - 7.5.1 软件包说明
 - 7.5.2 软件包
 - 7.5.2.1 toybrick-vendor-bin
 - 7.5.2.2 toybrick-vendor-dev
 - 7.6 Vendor Firmware
 - 7.6.1 软件包说明
 - 7.6.2 软件包
 - 7.6.2.1 vendor-firmware
 - 7.7 Toybrick Server
 - 7.7.1 软件包说明
 - 7.7.2 软件包
 - 7.7.2.1 toybrick-server
 - 7.8 Rockchip MPP
 - 7.8.1 软件包说明
 - 7.8.2 软件包
 - 7.8.2.1 rockchip-mpp
 - 7.8.2.2 rockchip-mpp-sample
 - 7.8.2.3 rockchip-mpp-dev
 - 7.9 Rockchip RGA
 - 7.9.1 软件包说明
 - 7.9.2 软件包
 - 7.9.2.1 rockchip-rga

7.9.2.2 [rockchip-rga-sample](#)

7.9.2.3 [rockchip-rga-dev](#)

1. 系统要求

Rockchip边缘计算SDK Debian11

2. 开机启动

2.1 登录账号

Debian11默认的登录账号是：rockchip，登录密码是：rockchip

说明：可以通过修改vendor目录下的config.json中rootfs子集的user字段修改登录账号。

3. 远程登录

RK3588开发板支持两种远程登录：adb和ssh

3.1 adb登录

Linux主机通过USB（主机的USB Host连接开发板的USB OTG口），执行如下命令远程登录RK3588开发板的debian系统：

```
adb shell
```

说明：开发板的OTG口通常是标有"TYPE_C"或"DOWNLOAD"的丝印，接口类似是Type-C。

3.2 ssh登录

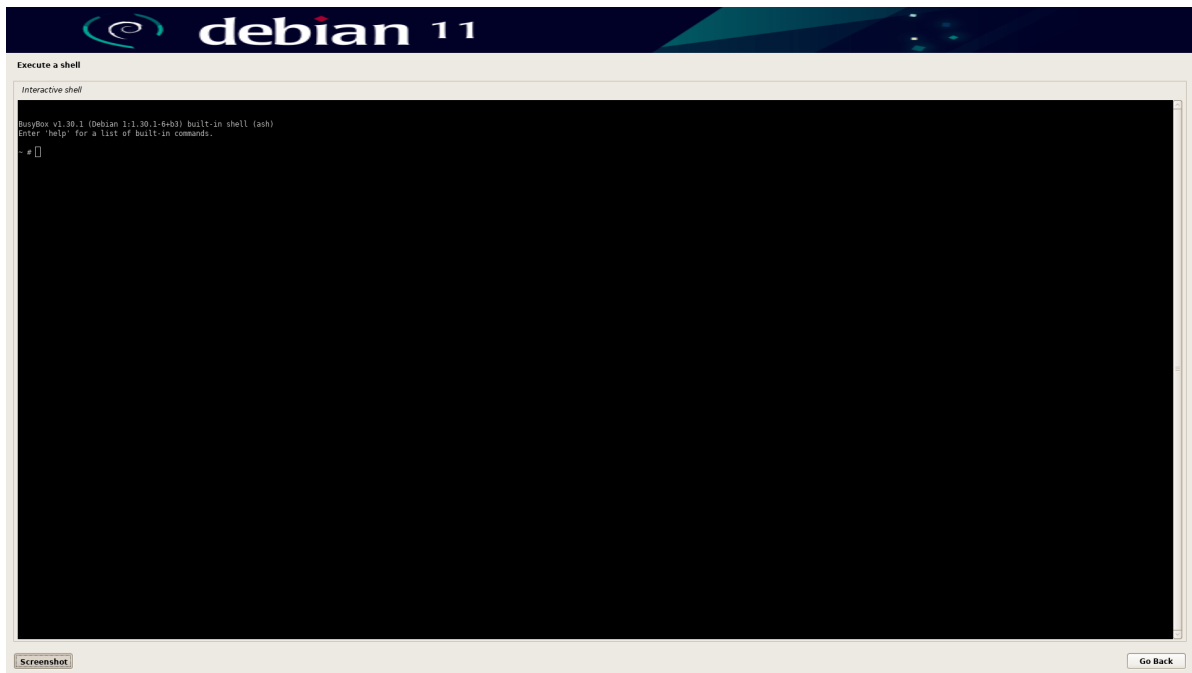
Linux主机通过网络，执行如下命令远程登录RK3588开发板的debian系统：

```
ssh rockchip@xxx.xxx.xxx.xxx //xxx.xxx.xxx.xxx是开发板的IP地址
```

4. 紧急模式

4.1 进入紧急模式

拔出设备OTG口的Type-C的线，长按recovery按键后重启设备，系统将进入紧急模式的命令行，如图所示：



4.2 命令行操作

- 网络连接
 - 插入网线
 - 查看以太网接口名

```
ip a
```

- 动态分配IP地址(假设以太网接口为eth1)

```
dhclient eth1
```

- 挂载U盘(假设U盘为： /dev/sda1)

```
mount /dev/sda1 /mnt
```

- 远程拷贝

```
scp $LOCAL_FILE $USER@$IP:/$REMOTE_PATH
```

- 重要文件备份

- 挂载rootfs分区到/sysroot目录：进入紧急模式后系统自动挂载rootfs分区到/sysroot，用户无需重复操作。
 - 将重要文件拷贝到U盘或拷贝的远程主机上：

```
cp $FILE /mnt/  
scp $LOCAL_FILE $USER@$IP:/$REMOTE_PATH
```

- 系统还原

- 将待还原的镜像rootfs.img拷贝到U盘上，并将U盘挂载到/mnt目录。

```
$CP rootfs.img $Udisk  
mount /dev/sda1 /mnt
```

- 执行如下命令还原：

```
umount /sysroot  
dd if=/mnt/rootfs.img of=/dev/disk/by-partlabel/rootfs
```

- 制作根文件系统

当用户完成产品化系统配置，执行如下步骤制作根文件系统：

- 挂载rootfs分区到sysroot目录：进入紧急模式后系统自动挂载rootfs分区到sysroot，用户无需重复操作。
- 挂载U盘到/mnt目录：

```
mount /dev/sda1 /mnt
```

- 打包根文件系统所有文件：

```
cd /sysroot  
tar cvpfJ ../mnt/rootfs.tar.xz *  
cd -
```

- 创建空文件系统并格式化为EXT4文件系统：

```
dd if=/dev/zero of=/mnt/rootfs.img bs=2K count=3M // 创建文件系统大小为6G  
          (2K * 3M)，用户可以修改count的大小设置文件系统大小  
mkfs.ext4 /mnt/rootfs.img
```

- 挂载空文件系统到/rootfs目录：

```
mkdir /rootfs  
mount /mnt/rootfs.img /rootfs
```

- 解压打包文件到/rootfs目录

```
tar xvpfJ /mnt/rootfs.tar.xz /rootfs
```

- 卸载/rootfs目录和U盘：

```
umount /rootfs  
umount /mnt
```

5. 添加APT源

添加边缘计算SDK的DEB源（边缘计算SDK编译默认添加）：

```
echo "deb http://repo.rock-chips.com/edge/debian bullseye main" | sudo tee
/etc/apt/sources.list.d/edge.list
```

6. 添加PIP源

添加边缘计算SDK的Python软件包的PIP源（边缘计算SDK编译默认添加）：

```
echo "[global]" | sudo tee /etc/pip.conf
echo "extra-index-url = http://repo.rock-chips.com/edge/pypi/simple
http://pypi.douban.com/simple" | sudo tee -a /etc/pip.conf
echo "trusted-host = repo.rock-chips.com pypi.douban.com" | sudo tee -a
/etc/pip.conf
```

7. 系统软件包

7.1 Toybrick Property

7.1.1 软件包说明

Toybrick Property基于D-BUS实现安卓系统属性的相关接口：getprop、setprop和C/C++库函数。

7.1.2 软件包

7.1.2.1 toybrick-prop

1. 包含运行时链接库（路径：/usr/lib/aarch64-linux-gnu）：libtoybrick_prop.so.0和libtoybrick_prop.so.0.0.0
2. 执行如下命令安装：

```
sudo -y install toybrick-prop toybrick-log
```

7.1.2.2 toybrick-prop-bin

1. 包含可执行文件和启动服务：
 - 可执行文件（路径：/usr/bin）：toybrick_propd, getprop, setprop, toybrick-prop.sh
 - 启动服务（路径：/usr/lib/systemd/system）：toybrick-prop.service

2. 执行如下命令安装:

```
sudo -y install toybrick-prop-bin
```

3. 开机启动服务:

```
sudo systemctl enable toybrick-prop.service
```

4. Prop命令使用帮助:

- 列出所有系统属性

```
getprop

[persist.dev.model]: [TB-RK3588X0]
[persist.sys.usb.config]: [adb]
[dev.model]: [TB-RK3588X0]
[sys.usb.config]: [adb]
[sys.usb.touch.width]: [1920]
[sys.usb.touch.height]: [1080]
[sys.usb.touch.points]: [10]
[sys.usb.touch.in]: [/dev/hidg2]
[sys.usb.touch.report.id]: [4]
[sys.usb.touch.report.size]: [64]
[sys.usb.touch.feature.id]: [2]
[sys.usb.keyboard.in]: [/dev/hidg0]
[sys.usb.mouse.in]: [/dev/hidg1]
```

说明: 以

persist

开头的系统属性会保存在

/etc/prop

目录下, 重启系统系统会自动加载导入配置。

- 列出所有前缀为指定字段的系统属性

```
getprop list sys.usb

[sys.usb.config]: [adb]
[sys.usb.touch.width]: [1920]
[sys.usb.touch.height]: [1080]
[sys.usb.touch.points]: [10]
[sys.usb.touch.in]: [/dev/hidg2]
[sys.usb.touch.report.id]: [4]
[sys.usb.touch.report.size]: [64]
[sys.usb.touch.feature.id]: [2]
[sys.usb.keyboard.in]: [/dev/hidg0]
[sys.usb.mouse.in]: [/dev/hidg1]
```

- 查看单个系统属性

```
getprop sys.usb.config

adb
```

- 设置系统属性 (需要root权限)

```
sudo setprop dev.version v0.1    // 设置dev.version属性，重启后丢失
sudo setprop persist.dev.version v0.1    //设置persist.dev.version属性，保存在/etc/prop/dev.json
```

- 删除指定的property

```
sudo setprop dev.version    //删除dev.version属性
sudo setprop persist.dev.version //删除dev.version属性，同时删除/etc/prop/dev.json里的相关配置
```

7.1.2.3 toybrick-prop-dev

1. 包含编译连接所需的相关文件：

- 头文件（路径：/usr/include/toybrick）：toybrick_properties.h, properties.h和system_properties.h
- 编译时链接库（路径：/usr/lib/aarch64-linux-gnu）：libtoybrick_prop.so
- pkgconfig（路径：/usr/lib/aarch64-linux-gnu/pkgconfig）：toybrick_prop.pc

2. 执行如下命令安装：

```
sudo -y install toybrick-prop-dev toybrick-log
```

3. 编译链接选项：

```
LDDFLAG=`pkg-config --libs toybrick_prop`
CFLAG=`pkg-config --cflags toybrick_prop`
```

4. 包含头文件：

```
#include <toybrick/toybrick_properties.h>
#include <toybrick/properties.h>
#include <toybrick/system_properties.h>
```

5. 示例代码：test.c

```
#include <toybrick/toybrick_properties.h>
#include <stdio.h>

int main(int argc, char **argv)
{
    const char *key = "dev.model";
    const char *default_value = "TB-RK3588X0";
    char value[PROPERTY_VALUE_MAX];
    int ret;

    ret = property_get(key, value, default_value);
    printf("key %s, value %s, len %d\n", key, value, ret);
    return ret;
}
```

```
编译命令: gcc test.c `pkg-config --libs toybrick_prop` `pkg-config --cflags toybrick_prop` -o test
```

7.2 Toybrick Usbconfig

7.2.1 软件包说明

Toybrick Usbconfig基于Type-C OTG口实现虚拟USB设备如：adb ,ntb, rndis网卡，虚拟摄像头，虚拟声卡，虚拟键盘鼠标和触摸屏等。

7.2.2 软件包

7.2.2.1 toybrick-usbd

- 包含可执行文件和启动服务
 - 可执行文件（路径：/usr/bin）：toybrick_usbd, toybrick_adbd等
 - 启动服务（路径：/usr/lib/systemd/system）：toybrick-usb.service
- 执行如下命令安装：

```
sudo apt -y install toybrick-usbd
```

- 开机启动服务：

```
sudo systemctl enable toybrick-usb.service
```

- USB Config配置

```
sudo setprop sys.usb.config adb //当前支持adb功能，下个版本计划支持  
adb,ntb,rndis,key,touch,keyboard,uvic,uvic等  
sudo setprop persist.sys.usb.config adb //写入磁盘，永久生效。
```

7.3 Rockchip ISP

7.3.1 软件包说明

Rockchip ISP运行RKISP 3A tuning服务和camera示例代码，默认支持imx327，imx415，imx464，os04a10hk，ov3855和ov50c40模组。

7.3.2 软件包

7.3.2.1 rockchip-isp

- 包含可执行文件，iqfile文件和启动服务
 - 可执行文件（路径：/usr/bin）：rkaiq_3A_server
 - 启动服务（路径：/usr/lib/systemd/system）：rockchip-isp.service
- 执行如下命令安装：

```
sudo apt -y install rockchip-isp
```

- 开机服务器:

```
sudo systemctl enable rockchip-isp.service
```

- 新增iqfile: 将iqfile文件拷贝到/etc/iqfiles目录下, 重启服务。

```
cp ${IQFILE} /etc/iqfiles/  
sudo systemctl restart rockchip-isp.service
```

7.3.2.2 rockchip-isp-sample

- 包含camera示例代码(路径: /usr/share/camera)
- 编译示例代码

```
cp -r /usr/share/camera ./  
cd camera  
make
```

7.4 Edge Utils

7.4.1 软件包说明

Edge Utils包含蓝牙、WIFI、IO、网络配置、显示、camera等实用工具。

7.4.2 软件包

7.4.2.1 edge-utils

- 包含系统相关工具(路径: /usr/local/bin)
- 执行如下命令安装:

```
sudo apt -y install edge-utils
```

7.5 Toybrick Vendor

7.5.1 软件包说明

Toybrick vendor提供/dev/vendor-storage的SN、MAC等的读写命令、C语言调用接口和示例代码。

7.5.2 软件包

7.5.2.1 toybrick-vendor-bin

- 包含/dev/vendor-storage的SN、MAC等的读写命令（/usr/bin/toybrick_vendor）
- 执行如下命令安装：

```
sudo apt -y install toybrick-vendor-bin
```

- 命令帮助：

```
toybrick_vendor --help
Usage:
  toybrick_vendor get {sn | product | id } len
  toybrick_vendor set {sn | product | id } data
  toybrick_vendor get mac count
  toybrick_vendor set mac mac0 mac1 ...
e.g.
  toybrick_vendor set mac 329bb75e915e 329bb75e915f //写入两个mac地址
  toybrick_vendor get mac 2 //读取两个mac地址
  toybrick_vendor set sn xxxxxxxx //写入SN号
  toybrick_vendor get sn 16 //读取SN号，count值
                           必须大于实际SN号的长度
```

7.5.2.2 toybrick-vendor-dev

1. 包含编译连接所需的相关文件：
 - 头文件（路径：/usr/include/toybrick）：toybrick-vendor.h
 - 编译时链接库（路径：/usr/lib/aarch64-linux-gnu）：libtoybrick_vendor.so
 - pkgconfig（路径：/usr/lib/aarch64-linux-gnu/pkgconfig）：toybrick_vendor.pc
2. 执行如下命令安装：

```
sudo -y install toybrick-vendor-dev
```

3. 编译链接选项：

```
LDDFLAG=`pkg-config --libs toybrick_vendor`
CFLAG=`pkg-config --cflags toybrick_vendor`
```

4. 包含头文件：

```
#include <toybrick/toybrick-vendor.h>
```

5. 示例代码：test.c

```
#include <toybrick/toybrick-vendor.h>
#include <stdio.h>

int main(int argc, char **argv)
{
    const char *sn = "xxxxxxxx";

    return vendor_set_sn(sn, strlen(sn));
}

编译命令: gcc test.c `pkg-config --libs toybrick_vendor` `pkg-config --cflags
toybrick_vendor` -o test
```

7.6 Vendor Firmware

7.6.1 软件包说明

Vendor Firmware包含WIFI，蓝牙模组的firmware固件。

7.6.2 软件包

7.6.2.1 vendor-firmware

- 包含系统相关工具(路径: /vendor, /system)
- 执行如下命令安装:

```
sudo apt -y install vendor-firmware
```

7.7 Toybrick Server

7.7.1 软件包说明

Toybrick Server集成启动服务，开启启动执行toybrick-server.sh脚本，启动蓝牙服务，并调用/usr/local/bin/toybrick-custom.sh

注意: *toybrick-server.sh*会因为软件包升级而被覆盖，如果用有需要添加开机启动脚本请新建/修改*/usr/local/bin/toybrick-custom.sh*。

7.7.2 软件包

7.7.2.1 toybrick-server

- 包含可执行文件和启动服务
 - 可执行文件（路径: /usr/bin）: toybrick-server.sh
 - 启动服务（路径: /usr/lib/systemd/system）: toybrick.service

- 执行如下命令安装：

```
sudo apt -y install toybrick-server
```

- 开机启动服务：

```
sudo systemctl enable toybrick.service
```

7.8 Rockchip MPP

7.8.1 软件包说明

Rockchip MPP集成VPU视频编解码库、开发头文件和测试命令。

开发指导详见：边缘计算SDK工程的docs/common/MPP/目录下文档。

7.8.2 软件包

7.8.2.1 rockchip-mpp

1. 包含运行时链接库（路径：/usr/lib/aarch64-linux-gnu）：librockchip_mpp.so.
2. 执行如下命令安装：

```
sudo -y install rockchip-mpp
```

7.8.2.2 rockchip-mpp-sample

1. 包含MPP 示例代码：
 - 示例代码（路径：/usr/share/rockchip-mpp/sample）
2. 执行如下命令编译示例代码：

```
sudo apt -y install rockchip-mpp-dev  
cp -r /usr/share/rockchip-mpp/sample ~/  
cd ~/sample/build  
./make.sh
```

3. 执行如下命令安装：

```
sudo -y install rockchip-mpp-sample
```

7.8.2.3 rockchip-mpp-dev

1. 包含编译连接所需的相关文件：

- 头文件（路径：/usr/include/rockchip）：MPP相关头文件
- 编译时链接库（路径：/usr/lib/aarch64-linux-gnu）：librockchip_mpp.so
- pkgconfig（路径：/usr/lib/aarch64-linux-gnu/pkgconfig）：rockchip_mpp.pc

2. 执行如下命令安装：

```
sudo -y install rockchip-mpp-dev
```

3. 编译链接选项：

```
LDDFLAG=`pkg-config --libs rockchip_mpp`  
CFLAG=`pkg-config --cflags rockchip_mpp`
```

7.9 Rockchip RGA

7.9.1 软件包说明

Rockchip RGA集成RGA 2D图像加速库、开发头文件和测试命令。

开发指导详见：边缘计算SDK工程的docs/edge/rga/目录下文档。

7.9.2 软件包

7.9.2.1 rockchip-rga

1. 包含运行时链接库（路径：/usr/lib/aarch64-linux-gnu）：librga.so.

2. 执行如下命令安装：

```
sudo -y install rockchip-rga
```

7.9.2.2 rockchip-rga-sample

1. 包含RGA示例代码：

- 示例代码（路径：/usr/share/rockchip-rga/sample）

2. 执行如下命令编译示例代码：

```
sudo apt -y install rockchip-rga-dev libdrm-dev  
cp -r /usr/share/rockchip-rga/sample ~/  
cd ~/sample/build  
./make.sh
```

3. 执行如下命令安装：


```
sudo -y install rockchip-rga-bin
```

7.9.2.3 rockchip-rga-dev

1. 包含编译连接所需的相关文件：

- 头文件（路径：/usr/include/rockchip）：RGA相关头文件
- 编译时链接库（路径：/usr/lib/aarch64-linux-gnu）：librga.so
- pkgconfig（路径：/usr/lib/aarch64-linux-gnu/pkgconfig）：rockchip_rga.pc

2. 执行如下命令安装：

```
sudo -y install rockchip-rga-dev
```

3. 编译链接选项：

```
LDDFLAG=`pkg-config --libs rockchip_rga`  
CFLAG=`pkg-config --cflags rockchip_rga`
```