

Rockchip Third-Party Systems Adaptation Development Guide

ID: RK-KF-YF-917

Release Version: V1.0.1

Release Date: 2022-02-17

Security Level: ☐Top-Secret ☐Secret ☐Internal ☒Public

DISCLAIMER

THIS DOCUMENT IS PROVIDED “AS IS”. ROCKCHIP ELECTRONICS CO., LTD.(“ROCKCHIP”)DOES NOT PROVIDE ANY WARRANTY OF ANY KIND, EXPRESSED, IMPLIED OR OTHERWISE, WITH RESPECT TO THE ACCURACY, RELIABILITY, COMPLETENESS, MERCHANTABILITY, FITNESS FOR ANY PARTICULAR PURPOSE OR NON-INFRINGEMENT OF ANY REPRESENTATION, INFORMATION AND CONTENT IN THIS DOCUMENT. THIS DOCUMENT IS FOR REFERENCE ONLY. THIS DOCUMENT MAY BE UPDATED OR CHANGED WITHOUT ANY NOTICE AT ANY TIME DUE TO THE UPGRADES OF THE PRODUCT OR ANY OTHER REASONS.

Trademark Statement

"Rockchip", "瑞芯微", "瑞芯" shall be Rockchip's registered trademarks and owned by Rockchip. All the other trademarks or registered trademarks mentioned in this document shall be owned by their respective owners.

All rights reserved. ©2022. Rockchip Electronics Co., Ltd.

Beyond the scope of fair use, neither any entity nor individual shall extract, copy, or distribute this document in any form in whole or in part without the written approval of Rockchip.

Rockchip Electronics Co., Ltd.

No.18 Building, A District, No.89, software Boulevard Fuzhou, Fujian, PRC

Website: www.rock-chips.com

Customer service Tel: +86-4007-700-590

Customer service Fax: +86-591-83951833

Customer service e-Mail: fae@rock-chips.com

Preface

Overview

This document introduces the adaptation process of third-party systems, like Debian and similar systems.

Product Version

Chipset	Kernel Version
RK3568	Linux 4.19
RK3588	Linux 5.10

Intended Audience

This document (this guide) is mainly intended for:

Technical support engineers

Software development engineers

Revision History

Version	Author	Date	Change Description
V1.0.0	Zack Huang	2021-12-21	Initial version
V1.0.1	Ruby Zhang	2022-02-17	Fix some expressions

Contents

Rockchip Third-Party Systems Adaptation Development Guide

1. Hardware Adaptation Required on RK Platform
2. Software Adaptation on RK platform
3. Special Notes of Software Development Adaptation
4. Display Services Adaptation
 - 4.1 GPU Adaptation
 - 4.2 RGA Adaptation
 - 4.3 drm-cursor Adaptation
 - 4.4 xserver Adaptation
 - 4.5 How to Verify If the Adaptation Is Successful
5. Codec Service Adaptation
 - 5.1 mpp Adaptation
 - 5.2 Multimedia Software Adaptation
 - 5.3 GStreamer Adaptation
 - 5.4 How to Verify If the Adaptation Is Successful
6. Camera Service Adaptation
 - 6.1 camera_engine_rkaiq Adaptation
 - 6.2 How to Verify If the Adaptation Is Successful
7. NPU Service Adaptation
 - 7.1 rknn_runtime Adaptation
 - 7.2 rknn_server Adaptation
 - 7.3 rknn Demo
8. Wi-Fi/BT Service Adaptation

1. Hardware Adaptation Required on RK Platform

GPU: Graphics Processing Unit, it is also known as display core, vision processor, mainly used to accelerate geometric transformations and light processing, cubic mapping and vertex blending, texture compression and bump mapping, dual-texture quad-pixel 256-bit rendering engine, etc. At present, the GPU hardware of RK platform has been integrated into the SOC.

RGA: 2D graphics accelerator, used to accelerate the rotation, cropping, scaling and other operations of 2D graphics. At present, the RGA hardware of RK platform has been integrated into the SOC.

VPU: Video Processing Unit, used to accelerate video codec, supports multiple formats. At present, the VPU hardware of RK platform has been integrated into the SOC.

ISP: The key function of ISP is to do post-processing on the signal output from front-end image sensor. At present, the ISP hardware of RK platform has been integrated into the SOC.

NPU: The embedded Neural-Network Processing Unit, adopts the "data-driven parallel computing" architecture, and it is especially good at processing massive multimedia data such as video and images. The NPU processor is specially designed for the artificial intelligence of the Internet of Things, which is used to accelerate the operation of the neural network and solve the problem of low efficiency of traditional chips in the operation of the neural network. At present, the NPU hardware of RK platform has been integrated into the SOC.

Wi-Fi/BT: A short distance communication module adapted to the development board.

2. Software Adaptation on RK platform

Source Code Path	Description
SDK/external/libmali	User mode function interfaces for hardware gpu which is currently provided by RK platform, mainly opengles general interface, the current mainstream debian-like display frameworks include xserver and wayland, the software interface used by APPs that adapted to the system should make corresponding changes.
SDK/external/xserver	Currently, RK platform has a patch for xserver hardware acceleration. If your displays framework using xserver, please refer to the patch adaptation of this repository.
SDK/external/linux-rga	User mode function interface for hardware RGA currently provided by RK platform.
SDK/external/mpp	User mode function interface for hardware VPU currently provided by RK platform.
SDK/external/rknpu2	User mode function interface for hardware NPU currently provided by RK platform.
SDK/external/camera_engine_rkaiq	User mode function interface for hardware ISP user mode currently provided by RK platform.
SDK/external/drm-cursor	User mode function interface for hardware mouse layer currently provided by RK platform.

3. Special Notes of Software Development Adaptation

When adapting software, debian-like systems may have many version dependencies. If the installation of the following packages fails, you need to re-arrange the source code to your target device for compilation.

Compile command:

```
cd 'Source code directory'
sudo apt build-dep .
sudo DEB_BUILD_OPTIONS=nocheck dpkg-buildpackage -rfakeroot -b -d -us
```

4. Display Services Adaptation

4.1 GPU Adaptation

Firstly, check the display framework used in your system, xserver or wayland(currently only these two frameworks are supported) and then install SDK/debian/packages/"ARCH"/libmali/libmali-***.deb

For example, if the system uses x11 on the rk3588 platform, and the GPU model used on the rk3588 platform is valhall-g610-g6p0, then you should install:

```
SDK/debian/packages/arm64/libmali/libmali-valhall-g610-g6p0-x11_1.9-1_arm64.deb
```

4.2 RGA Adaptation

Installation package:

```
SDK/debian/packages/'ARCH'/rga/*.deb
```

4.3 drm-cursor Adaptation

Installation package:

```
SDK/debian/packages/arm64/libdrm-cursor/*.deb
```

4.4 xserver Adaptation

Because there are many versions of xserver, you have to check the version of xserver used in your system first, and then porting to your version of xserver based on our source code patch, and then rebuild.

Generated patch:

```
cd 'source directory'

git format-patch e4f4521ca
```

The following is the compiled installation package of version 1.20.11:

```
SDK/debian/packages/'ARCH'/xserver/xserver-common_1.20.11-1_all.deb

SDK/debian/packages/'ARCH'/xserver/xserver-xorg-core_1.20.11-1_'ARCH'.deb

SDK/debian/packages/'ARCH'/xserver/xserver-xorg-legacy_1.20.11-1_'ARCH'.deb
```

xserver also needs a boot script to do some initialization operations. If it is a debian-like system, you can try to copy the boot script into the system. If not, please refer to the script to configure it yourself:

```
SDK/debian/overlay/etc/X11
```

4.5 How to Verify If the Adaptation Is Successful

Display service acceleration uses the standard interface of opengles. If your system uses opengl, please porting it to opengles. If the adaptation is successful, when rendering the screen or dragging the window, check the platform-related gpu node, there will be changes in gpu utilization , such as RK3588 platform:

```
cat /sys/devices/platform/fb000000.gpu/utilisation
```

Note: The gpu nodes of different platforms are different.

5. Codec Service Adaptation

5.1 mpp Adaptation

Installation package:

```
SDK/debian/packages/'ARCH'/mpp/*.deb
```

Note: For permission settings of some nodes, please refer to

```
SDK/debian/overlay/etc/udev/rules.d/99-rockchip-permissions.rules
```

To verify whether mpp is configured successfully:

```
# mpi_enc_test -w 1920 -h 1080 -t 7 -o /tmp/test.h264
# mpi_dec_test -w 1920 -h 1080 -t 7 -i /tmp/test.h264
```

5.2 Multimedia Software Adaptation

There are many multimedia frameworks for customers. It is recommended that customers' player can directly call the interface of mpp to adapt the video codec. Please refer to the interface document of mpp "Rockchip_Developer_Guide_MPP_CN.pdf" for details.

5.3 GStreamer Adaptation

A precompiled package for gst-base is currently adapted by RK platform.:

```
SDK/debian/packages/'ARCH'/gst-plugins-base1.0/*.deb
SDK/debian/packages/'ARCH'/gst-plugins-bad1.0/*.deb
SDK/debian/packages/'ARCH'/gst-plugins-good1.0/*.deb
```

RK platform also supports GStreamer general multimedia framework. You need to install the official GStreamer components in advance (you can also install the gst-base precompiled package mentioned above, if your system version is compatible), and then install the gst-rockchip plugin:

```
SDK/debian/packages/'ARCH'/gst-rkmp/*.deb
```

5.4 How to Verify If the Adaptation Is Successful

```
sudo GST_DEBUG=2 gst-launch-1.0 playbin uri=file 'absolute video path' video-sink="show sink" audio-sink=fakesink
```

If there is tips for mpp called, it means that the hardware decoding is adapted successfully.

6. Camera Service Adaptation

6.1 camera_engine_rkaiq Adaptation

Installation package:

```
SDK/debian/packages/'ARCH'/rkaiq/camera_engine_rkaiq_'ARCH'.deb
```

6.2 How to Verify If the Adaptation Is Successful

Check whether images capture is successful by the v4l2 command, and check whether the captured image is correct: (Note: The nodes, formats, resolutions, etc. of different cameras on different platforms may be different, please change the command according to the actual situation)

```
v4l2-ctl -d /dev/video0 --set-fmt-video=width=1920,height=1080,pixelformat=NV12 -  
-stream-mmap=4 --stream-skip=3 --stream-to=/tmp /0.yuv --stream-count=1 --stream-  
poll
```

7. NPU Service Adaptation

7.1 rknn_runtime Adaptation

The rknn_runtime is mainly a user mode library, just copy it to the your file system.

```
SDK/external/rknpu2/Linux/librknn_api/'ARCH'/librknn_api.so
```

```
SDK/external/rknpu2/Linux/librknn_api/'ARCH'/librknnrt.so
```


7.2 rknn_server Adaptation

rknn_server is mainly a service used for on-board communication when the rknn-tool2 tool is used on the PC. If the model is trained on the PC, converted to rknn and imported on the board, in this case, the service is not required.

```
SDK/external/rknpu2/Linux/rknn_server/'ARCH'/usr/bin/*
```

7.3 rknn Demo

Please refer to:

```
SDK/external/rknpu2/examples
```

8. Wi-Fi/BT Service Adaptation

Please refer to "Rockchip_Developer_Guide_Linux_WIFI_BT_CN.pdf" For details.

There are precompiled packages in the SDK:

```
SDK\debian\packages\'ARCH'\rkwifi\*.deb
```