Rockchip RK625 RT-Thread SDK Release Note

ID: RK-FB-YF-901

Release Version: V1.0.0

Release Date: 2021-05-31

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 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. ©2021. 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

The document presents Rockchip RK625 RT-Thread SDK release notes, aiming to help engineers get started with RK625 RT-Thread SDK development and debugging faster.

Intended Audience

This document (this guide) is mainly intended for:

Technical support engineers

Software development engineers

Chipset and System Support

Chipset	Kernel Version
RK625	RT-Thread v3.1.x

Revision History

Version	Author	Date	Revision History
V0.1.0	Hans Yang	2021-03-12	Initial BETA V0.1.0 version
V1.0.0	Hans Yang	2021-05-08	Release V1.0.0 version

Contents

Rockchip RK625 RT-Thread SDK Release Note

- 1. Overview
- 2. Main Functions
- 3. How to Get the SDK
 - 3.1 SDK Download Command
 - 3.2 SDK Code Compression Package
 - 3.3 To Get the SDK Version
 - 3.4 SDK Code Update
- 4. RK625 RT-Thread Project Directory Introduction
- 5. SDK Building Instructions
 - 5.1 Set up the Building Environment
 - 5.2 Basic Building and Packaging Commands
- 6. SSH Public Key Operation Introduction
 - 6.1 Multi-device Use the Same SSH Public Key
 - 6.2 Switch Different SSH Public Keys on the Same Device
 - 6.3 Key Authority Management
 - 6.4 Reference Documents

1. Overview

This SDK is based on RT-Thread v3.1.3, which contains system source code, drivers, tools, and application software packages used for RT-Thread system development, and it also contains development documents and tool usage documents. Adapting to RK625 chip platform, it is suitable for RK625 EVB development board and all products developed based on RK625 platform.

2. Main Functions

Functions	Module Name
Data Communication	USB
Image Processing	ISP、JPEG encoder
Camera Interface	MIPI CSI
Application Demo	UVC

3. How to Get the SDK

Rockchip SDKs are released by Rockchip server. Please refer to Chapter 5 <u>SDK Building Introduction</u> to build a development environment.

To get RK625 RT-Thread SDK software package, customers need an account to access the source code repository provided by Rockchip. In order to be able to obtain code synchronization, please provide SSH public key for server authentication and authorization when apply for SDK from Rockchip technical window(e-Mail: fae@rock-chips.com). About Rockchip server SSH public key authorization, please refer to Chapter 6 SSH Public Key Operation Introduction.

3.1 SDK Download Command

Repo, a tool built on Python script by Google to help manage git repositories, is mainly used to download and manage software repository of projects. The download address is as follows:

git clone ssh://git@www.rockchip.com.cn/repo/rk/tools/repo

RK625 RT-Thread SDK download command:

```
repo init --repo-url ssh://git@www.rockchip.com.cn/repo/rk/tools/repo -u
ssh://git@www.rockchip.com.cn/rtos/rt-thread/rk/platform/release/manifests -b master -m
rk625_release.xml
```

After the code repository is initialized, you can use the following command to synchronize the code:

```
.repo/repo/repo sync
```

3.2 SDK Code Compression Package

For quick access to SDK source code, Rockchip Technical Window usually provides corresponding version of SDK initial compression package. In this way, developers can get SDK source code through decompressing the initial compression package, which is the same as the one downloaded by repo.

Take RK625_RT-Thread_SDK_Release_V1.0.0_20210531.tar.gz as an example. After getting an initialization package, you can get the source code by running the following command:

```
tar zxvf RK625_RT-Thread_SDK_Release_V1.0.0_20210531.tar.gz
cd RK625_RT-Thread_SDK_Release_V1.0.0_20210531
.repo/repo/repo sync -1
.repo/repo/repo sync
```

3.3 To Get the SDK Version

Please get the SDK release version through project xml file by the following command:

```
cd .repo/manifests
git log rk625_release.xml
```

Or check the current SDK version through RKDocs/RK625 RT-Thread Release Note.txt.

3.4 SDK Code Update

```
.repo/repo/repo sync
```

4. RK625 RT-Thread Project Directory Introduction

The following is the main SDK directory:

```
- AUTHORS
|-- bsp
                        # Chip related source code
   - rockchip
       - common
          — drivers # Universal driver of rockchip OS adaptation layer
            — hal # Rockchip HAL (hardware Abstraction Layer) implementation
                      # Rockchip driver test code
          L_ tests
         - swallow # RK625 main directory
       - applications # RK625 application code
          board # Board level configuration
          - build
                      # Build main directory and store the intermediate files
          build.sh # RK625 build script
            - drivers # RK625 Private driver directory
          └── Image # Firmware
                      # Rockchip commonly used tools
       L- tools
- ChangeLog.md
|-- components
                       # Various components of the system, including file system, shell
and framework layer and other drivers
|-- examples
                       # RT-Thread example program and test code
|-- include
                       # RT-Thread official header file directory
- Kconfig
|-- libcpu
- LICENSE
- README.md
--- README zh.md
|-- RKDocs
                       # Rockchip documents
|-- src
                       # RT-Thread kernel source code
|-- third party
                       # Directory of third-party code added by Rockchip
|-- tools
                        # RT-Thread official tool directory, including menuconfig and
building scripts
```

5. SDK Building Instructions

5.1 Set up the Building Environment

It is recommended to take 64-bit Ubuntu 16.04 or Ubuntu 18.04 system as an building environment, for other Linux systems have not been tested yet, it is recommended to install the release version consistent with Rockchip developers.

The building tool are SCons + GCC officially recommended by RT-Thread. SCons is an open source build system written in Python language. And GCC cross building tool is officially provided by ARM. You can directly install all the required tools by the following commands:

```
sudo add-apt-repository ppa:team-gcc-arm-embedded/ppa
sudo apt-get update
sudo apt-get install gcc-arm-embedded scons clang-format astyle libncurses5-dev build-
essential python-configparser
```

If the toolchain can not be installed, you can also download them from ARM official website and specify their path through environment variables as follows:

```
wget https://developer.arm.com/-/media/Files/downloads/gnu-rm/7-2018q2/gcc-arm-none-eabi-7-2018-q2-update-linux.tar.bz2
tar xvf gcc-arm-none-eabi-7-2018-q2-update-linux.tar.bz2
export RTT_EXEC_PATH=/path/to/toolchain/gcc-arm-none-eabi-7-2018-q2-update/bin
```

5.2 Basic Building and Packaging Commands

The building command is as follows:

```
cd RK625_RT-Thread_SDK_Release_V1.0.0_20210531
cd bsp/rockchip/swallow
cp board/usb_camera/defconfig .config
scons --menuconfig #Modify the switch of building modules, after exiting, the rtconfig.h
file will be generated.
./build.sh
```

The generated firmware is located in:

```
Image/Firmware.img
```

For more detailed buildings, debugs, and flashes instructions about RK625 RT-Thread SDK, please refer to the following document:

/RKDocs/manuals/Rockchip_RK625_Quick_Start_RT-Thread_CN.pdf

6. SSH Public Key Operation Introduction

Please follow the introduction in the "Rockchip SDK Application and Synchronization Guide" to generate an SSH public key and send the email to fae@rock-chips.com, applying for permission to download SDK code. This document will be released to customers during the process of applying for permission.

6.1 Multi-device Use the Same SSH Public Key

If the same SSH public key should be used in different devices, you can copy the SSH private key file id_rsa to "~/.ssh/id_rsa" of the device you want to use.

If the following prompt appears when using a wrong private key, please be careful to replace it with the correct private key.

```
~/tmp$ git clone git@172.16.10.211:rk292x/mid/4.1.1_rl
Initialized empty Git repository in /home/cody/tmp/4.1.1_rl/.git/
The authenticity of host '172.16.10.211 (172.16.10.211)' can't be established.
RSA key fingerprint is fe:36:dd:30:bb:83:73:e1:0b:df:90:e2:73:e4:61:46.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '172.16.10.211' (RSA) to the list of known hosts.
git@172.16.10.211's password:
```

After adding the correct private key, you can use git to clone code, as shown below.

```
~$ cd tmp/
~/tmp$ git clone git@172.16.10.211:rk292x/mid/4.1.1_r1
Initialized empty Git repository in /home/cody/tmp/4.1.1_r1/.git/
The authenticity of host '172.16.10.211 (172.16.10.211)' can't be established.
RSA key fingerprint is fe:36:dd:30:bb:83:73:e1:0b:df:90:e2:73:e4:61:46.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '172.16.10.211' (RSA) to the list of known hosts.
remote: Counting objects: 237923, done.
remote: Compressing objects: 100% (168382/168382), done.
Receiving objects: 9% (21570/237923), 61.52 MiB | 11.14 MiB/s
```

Adding SSH private key may result in the following error.

```
Agent admitted failture to sign using the key
```

Please enter the following command in console to solve:

```
ssh-add ~/.ssh/id_rsa
```

6.2 Switch Different SSH Public Keys on the Same Device

You can configure SSH according to the ssh_config documentation.

```
~$ man ssh_config
```

```
🔞 😔 🔗 🏻 Terminal
文件(F) 编辑(E) 查看(V) 终端(T) 帮助(H)
SSH CONFIG(5)
                                       BSD File Formats Manual
                                                                                            SSH CONFIG(5)
NAME
       ssh_config - OpenSSH SSH client configuration files
SYNOPSIS
       ~/.ssh/config
       /etc/ssh/ssh_config
       ssh(1) obtains configuration data from the following sources in the fol-
       lowing order:
               1.
                     command-line options
                     user's configuration file (~/.ssh/config)
                     system-wide configuration file (/etc/ssh/ssh config)
      For each parameter, the first obtained value will be used. The configuration files contain sections separated by "Host" specifications, and that section is only applied for hosts that match one of the patterns given in the specification. The matched host name is the one given on
       the command line.
 Manual page ssh config(5) line 1
```

Run the following command to configure SSH configuration of current user.

```
~$ cp /etc/ssh/ssh_config ~/.ssh/config
~$ vi .ssh/config
```

As shown in the figure, SSH uses the file "~/.ssh1/id_rsa" of another directory as an authentication private key. In this way, different keys can be switched.

```
文件(F) 编辑(E) 查看(V) 终端(T) 帮助(H)

# ForwardXllTrusted yes
# RhostsRSAAuthentication no
# RSAAuthentication yes
# HostbasedAuthentication no
# GSSAPIAuthentication no
# GSSAPIAuthentication no
# GSSAPIAuthentication no
# GSSAPITouthentication no
# GSSAPITouthentication no
# GSSAPITouthentication no
# GSSAPITouthons no
# BatchMode no
# CheckHostIP yes
# AddressFamily any
# ConnectTimeout 0
# StrictHostKeyChecking ask
# IdentityFile ~/.ssh/id_rsa
IdentityFile ~/.ssh/id_dsa
# Port 22
# Protocol 2,1
# Cipher 3des
# Ciphers aes128-ctr,aes192-ctr,aes256-ctr,arcfour256,arcfour128,aes128-cbc,3d
es-cbc

MACS hmac-md5,hmac-shal,umac-64@openssh.com,hmac-ripemd160

43,1 70%
```

6.3 Key Authority Management

Server can monitor download times and IP information of a key in real time. If an abnormality is found, download permission of the corresponding key will be disabled.

Keep the private key file properly. Do not grant second authorization to third parties.

6.4 Reference Documents

For more details, please refer to the document:

 $/RKDocs/Others/Rockchip_User_Guide_SDK_Application_And_Synchronization_CN.pdf$