

Rockchip RK2118 RT-Thread SDK 发布说明

文档标识: RK-FB-YF-592

发布版本: V0.1.0

日期: 2024-03-15

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，由其各自拥有者所有。

版权所有 © 2024 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

文档主要介绍 Rockchip RK2118 RT-Thread SDK发布说明，旨在帮助工程师更快上手RK2118 RT-Thread SDK开发及相关调试方法。

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

各芯片系统支持状态

芯片名称	内核版本
RK2118	RT-Thread v4.1.x

修订记录

版本	日期	作者	修改说明
V0.0.1	2024-03-15	Roger Hu	初始版本
V0.1.0	2024-04-15	Roger Hu	Beta版本

目录

Rockchip RK2118 RT-Thread SDK 发布说明

- 概述
- 主要支持功能
- SDK 获取说明
 - SDK下载命令
 - SDK代码压缩包
 - SDK版本查看
 - SDK代码更新
- SDK目录介绍
- SDK编译说明
 - 开发环境搭建
 - 基础编译打包命令
- SSH 公钥操作说明
 - 多台机器使用相同 SSH 公钥
 - 一台机器切换不同 SSH 公钥
- 密钥权限管理
- 参考文档

概述

本SDK是基于RT-Thread v4.1.1 的软件开发包，包含RT-Thread系统开发用到的系统源码、驱动、工具、应用软件包。本SDK还包含开发文档，工具使用文档。适配RK2118芯片平台，适用于RK2118 EVB开发板及基于RK2118平台开发的所有产品。

主要支持功能

功能	模块名
数据通信	Wi-Fi、BT、USB、以太网
音频接口	SAI(PCM、I2S、TDM)、SPDIF TX/RX、PDM
显示接口	MCU panel、SPI panel、LED Segmenet display
产品形态	Soundbar、Partbox、车载音频、拾音器等

SDK 获取说明

SDK通过瑞芯微代码服务器对外发布。其编译开发环境，参考第5节 [SDK编译说明](#)。

获取RK2118 RT-Thread软件包，需要有一个帐户访问Rockchip提供的源代码仓库。客户向瑞芯微技术窗口（邮箱地址fae@rock-chips.com）申请SDK，同步提供SSH公钥进行服务器认证授权，获得授权后即可同步代码。关于瑞芯微代码服务器SSH公钥授权，请参考第6节 [SSH公钥操作说明](#)。

SDK下载命令

repo 是Google用 Python 脚本写的调用 git 的一个脚本，主要是用来下载、管理项目的软件仓库，其下载地址如下：

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

RK2118 RT-Thread SDK下载命令如下：

```
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 rk2118.xml
```

代码仓库初始化完成后，可用如下命令进行代码的同步：

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

SDK代码压缩包

为方便客户快速获取SDK源码，瑞芯微技术窗口通常会提供对应版本的SDK初始压缩包，开发者可以通过这种方式，获得SDK代码的初始压缩包，该压缩包解压得到的源码，与通过repo下载的源码是一致的。

以RK2118_RT-Thead_SDK_Alpha_V0.0.1_20240315.tar.gz为例，获取到该初始压缩包后，通过如下命令可检出源码：

```
tar zxvf RK2118_RT-Thead_SDK_Alpha_V0.0.1_20240315.tar.gz  
cd RK2118_RT-Thead_SDK_Alpha_V0.0.1_20240315  
.repo/repo/repo sync -l  
.repo/repo/repo sync
```

SDK版本查看

SDK每次版本更新都会同步对应的版本xml，可通过以下命令查看SDK软件版本：

```
cd .repo/manifests  
git log rk2118.xml
```

SDK代码更新

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

SDK目录介绍

以下是SDK主要目录对应的说明：

— applications	# Rockchip应用demo源码
— AUTHORS	
— bsp	# 所有芯片相关代码
— rockchip	
— common	
— drivers	# Rockchip OS适配层通用驱动
— hal	# Rockchip HAL(硬件抽象层)实现
— tests	# Rockchip 驱动测试代码
— rk2118	# RK2118 主目录
— board	# 板级配置
— build	# 编译主目录，存放中间文件
— drivers	# RK2118 私有驱动目录
— Image	# 存放固件
— tools	# Rockchip 通用工具
— ChangeLog.md	
— components	# 系统各个组件，包括文件系统，shell和框架层等驱动
— hifi4	
— dsp	# dsp代码
— rtt	# 运行在mcu上dsp相关代码
— shared	# mcu/dsp 公共代码
— tools	# dsp固件生成工具
— documentation	# RT-Thread官方文档
— examples	# RT-Thread例子程序和测试代码
— include	# RT-Thread官方头文件目录
— Kconfig	
— libcpu	
— LICENSE	
— README.md	
— README_zh.md	
— RKDocs	# Rockchip 文档
— src	# RT-Thread内核源码
— third_party	# Rockchip增加的第三方代码的目录
— tools	# RT-Thread官方工具目录，包括menuconfig和编译脚本

SDK编译说明

开发环境搭建

本SDK推荐的编译环境是64位的 Ubuntu 20.04 或 Ubuntu18.04，在其它 Linux 上尚未测试过, 所以推荐安装与RK开发者一致的发行版。

编译工具选用的是RT-Thread官方推荐的 SCons + GCC，SCons 是一套由 Python 语言编写的开源构建系统，GCC 交叉编译器由ARM官方提供，可直接使用以下命令安装所需的所有工具：

```
sudo apt-get install gcc-arm-embedded scons clang-format astyle libncurses5-dev  
build-essential python-configparser
```

从 ARM 官网下载编译器，通过环境变量指定 toolchain 的路径即可，具体如下：

```
wget https://developer.arm.com/-/media/Files/downloads/gnu/13.2.rel1/binrel/arm-gnu-toolchain-13.2.rel1-x86_64-arm-none-eabi.tar.xz
tar xvf arm-gnu-toolchain-13.2.rel1-x86_64-arm-none-eabi.tar.xz
export RTT_EXEC_PATH=/path/to/toolchain/arm-gnu-toolchain-13.2.Rel1-x86_64-arm-none-eabi/bin
```

或者使用SDK初始发布包中的编译器：arm-gnu-toolchain-13.2.rel1-x86_64-arm-none-eabi.tar.xz，具体如下：

```
tar -xvf arm-gnu-toolchain-13.2.rel1-x86_64-arm-none-eabi.tar.xz
export RTT_EXEC_PATH=/path/to/toolchain/arm-gnu-toolchain-13.2.Rel1-x86_64-arm-none-eabi/bin
```

基础编译打包命令

编译打包命令如下：

```
cd RK2118_RT-Thread_SDK_Alpha_V0.0.1_20240315
cd bsp/rockchip/rk2118
# 选择一个板级如adsp_demo
cp board/adsp_demo/defconfig .config
# 用.config生成rtconfig.h
scons --useconfig=.config
# optional：根据自己需要配置模块开关
scons --menuconfig
# 编译
scons -j8
# 打包固件
./mkimage.sh board/adsp_demo/setting.ini
```

或者使用编译打包脚本build.sh来编译替换上面步骤：

```
# 注意：即使配置没有修改，退出menuconfig时选择YES保存new configuration
./build.sh adsp_demo
```

生成的固件在：

```
Image/Firmware.img
```

RK2118 RT-Thread SDK更详细编译、调试以及刷机说明，请参考文档：

<SDK>/RKDocs/manuals/rk2118/Rockchip_RK2118_Quick_Start_RT-Thread_CN.pdf

SSH 公钥操作说明

请根据《Rockchip SDK 申请及同步指南》文档说明操作，生成 SSH 公钥，发邮件至fae@rock-chips.com，申请开通 SDK 代码。该文档会在申请开通权限流程中，释放给客户使用。

多台机器使用相同 SSH 公钥

在不同机器使用，可以将你的 SSH 私钥文件 id_rsa 拷贝到要使用的机器的“~/ssh/id_rsa”即可。在使用错误的私钥会出现如下提示，请注意替换成正确的私钥。

```
~/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.
git@172.16.10.211's password: █
```

添加正确的私钥后，就可以使用 git 克隆代码，如下图。

```
~$ 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
```

添加 ssh 私钥可能出现如下提示错误。

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

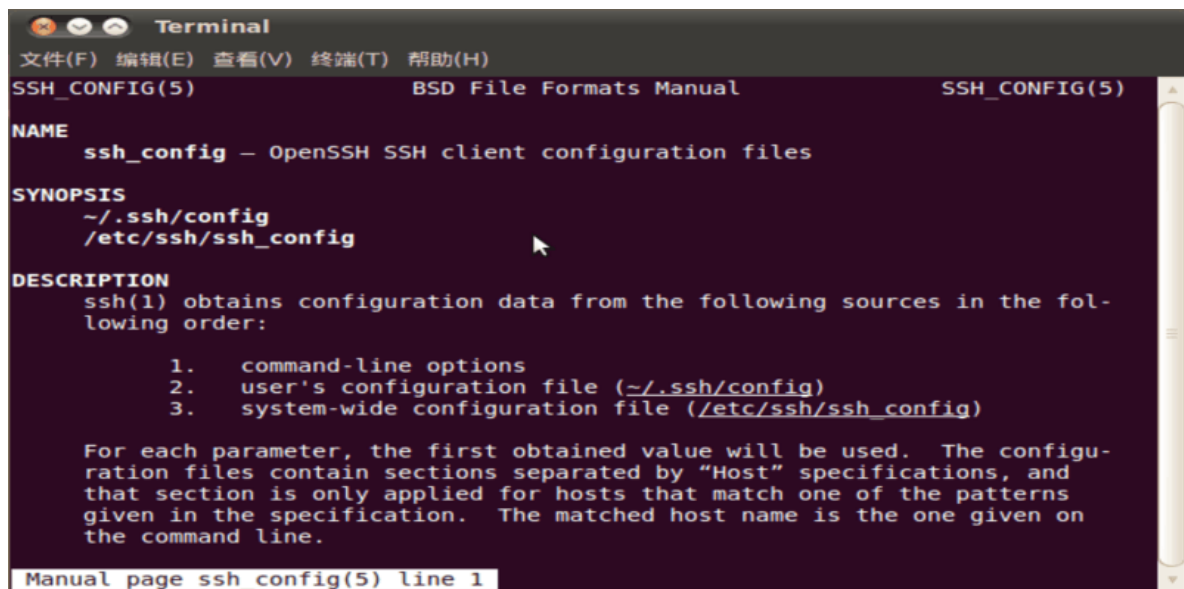
在 console 输入如下命令即可解决。

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

一台机器切换不同 SSH 公钥

可以参考 ssh_config 文档配置 SSH。

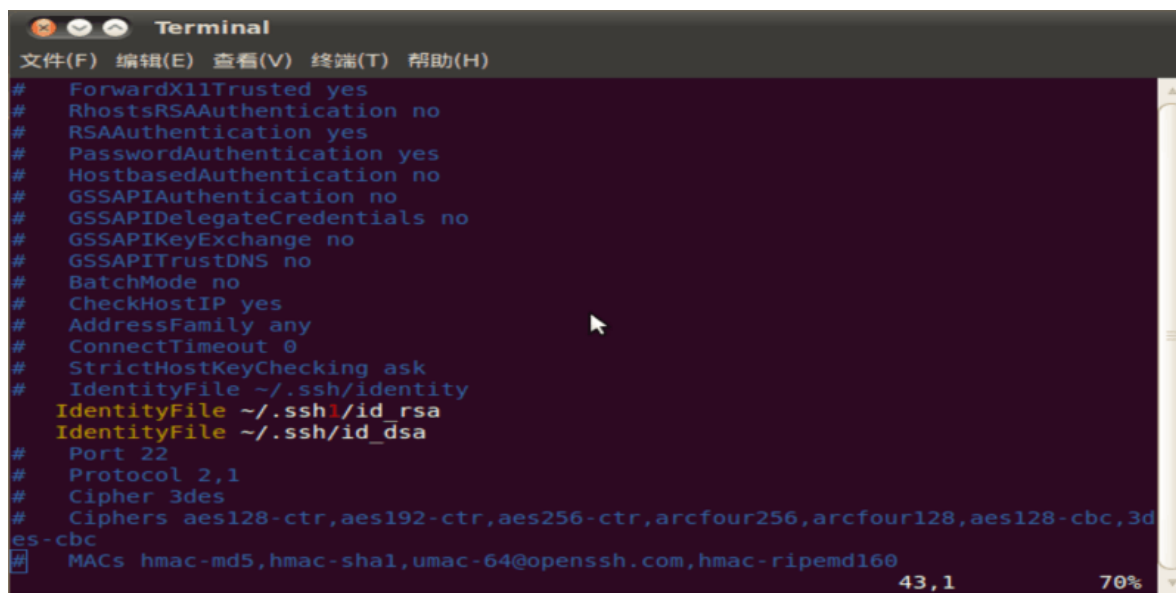
```
~$ man ssh_config
```



通过如下命令，配置当前用户的 SSH 配置。

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

如图，将 SSH 使用另一个目录的文件“~/.ssh1/id_rsa”作为认证私钥。通过这种方法，可以切换不同的密钥。

A screenshot of a macOS Terminal window titled "Terminal". The window has a menu bar with "文件(F)", "编辑(E)", "查看(V)", "终端(T)", and "帮助(H)". The terminal content shows SSH configuration parameters: ForwardX11Trusted yes, RhostsRSAAuthentication no, RSAAuthentication yes, PasswordAuthentication yes, HostbasedAuthentication no, GSSAPIAuthentication no, GSSAPIDelegateCredentials no, GSSAPIKeyExchange no, GSSAPITrustDNS no, BatchMode no, CheckHostIP yes, AddressFamily any, ConnectTimeout 0, StrictHostKeyChecking ask, IdentityFile ~/.ssh/identity, IdentityFile ~/.ssh/id_rsa (highlighted in yellow), IdentityFile ~/.ssh/id_dsa (highlighted in yellow), Port 22, Protocol 2,1, Cipher 3des, Ciphers aes128-ctr,aes192-ctr,aes256-ctr,arcfour256,arcfour128,aes128-cbc,3des-cbc, and MACs hmac-md5,hmac-sha1,umac-64@openssh.com,hmac-ripemd160. The bottom right corner shows "43,1" and "70%".

```
Terminal
文件(F) 编辑(E) 查看(V) 终端(T) 帮助(H)
# ForwardX11Trusted yes
# RhostsRSAAuthentication no
# RSAAuthentication yes
# PasswordAuthentication yes
# HostbasedAuthentication no
# GSSAPIAuthentication no
# GSSAPIDelegateCredentials no
# GSSAPIKeyExchange no
# GSSAPITrustDNS no
# BatchMode no
# CheckHostIP yes
# AddressFamily any
# ConnectTimeout 0
# StrictHostKeyChecking ask
# IdentityFile ~/.ssh/identity
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,3des-cbc
# MACs hmac-md5,hmac-sha1,umac-64@openssh.com,hmac-ripemd160
43,1 70%
```

密钥权限管理

服务器可以实时监控某个 key 的下载次数、IP 等信息，如果发现异常将禁用相应的 key 的下载权限。请妥善保管私钥文件。并不要二次授权与第三方使用。

参考文档

更多详细说明，可参考文档：

<SDK>/RKDocs/Others/Rockchip_User_Guide_SDK_Application_And_Synchronization_CN.pdf