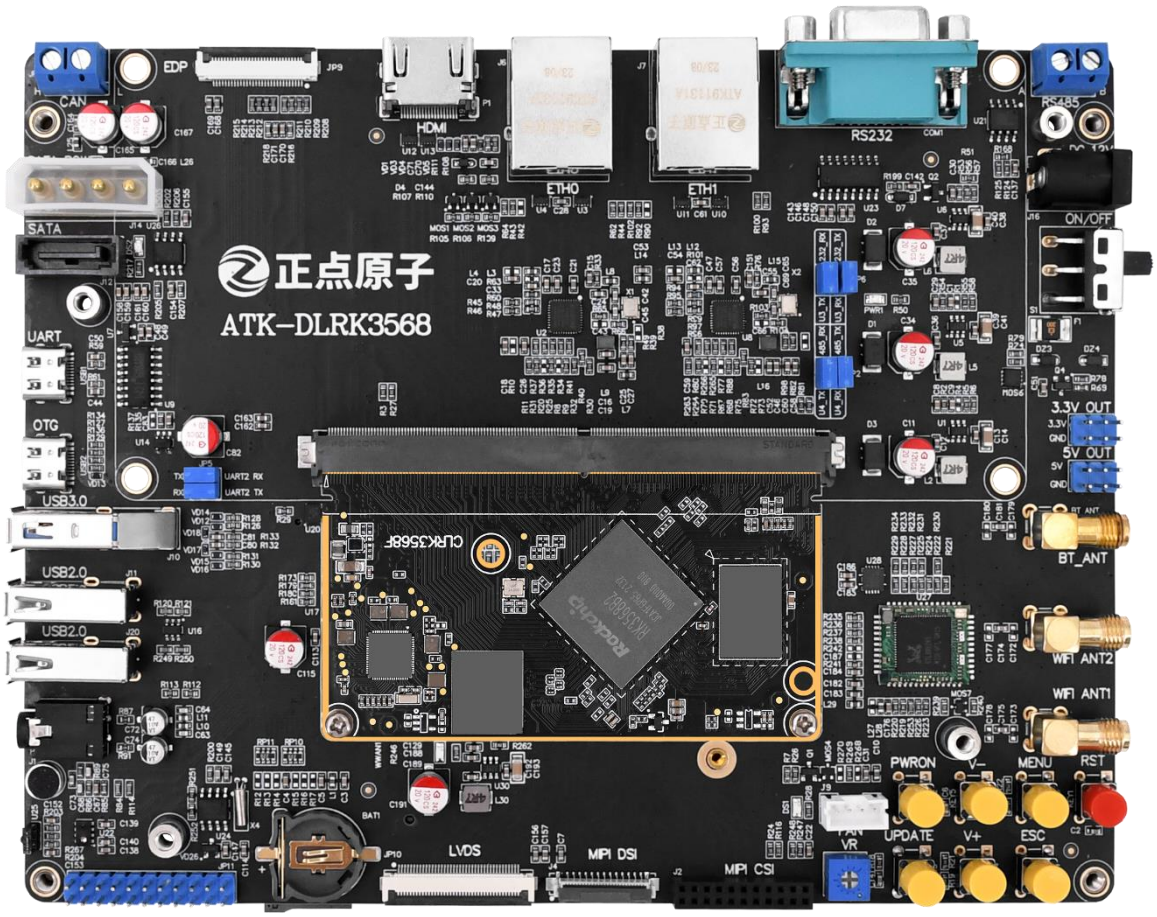
ATK-DLRK3568

Android13 SDK Compilation Instructions

V1.0



ALIENTEK

1. Shopping:

TMALL: https://zhengdianyuanzi.tmall.com

TAOBAO: https://openedv.taobao.com

2. Download

Address: http://www.openedv.com/docs/index.html

3. FAE

Website : www.alientek.com

Forum : http://www.openedv.com/forum.php

Videos : www.yuanzige.com

Fax : +86 - 20 - 36773971

Phone : +86 - 20 - 38271790

Disclaimer

The product specifications and instructions mentioned in this document are for reference only and subject to update without prior notice; Unless otherwise agreed, this document is intended as a product guide only, and none of the representations made herein constitutes a warranty of any kind.

In order to get the latest version of product information, please regularly visit the download center or contact the customer service of Taobao ALIENTEK flagship store. Thank you for your tolerance and support.

Revision History:

Version	Version Update Notes	Responsible person	Proofreading	Date
V1.0	release officially	ALIENTEK	ALIENTEK	2024.07.10

Catalogue

Brief

This document is the compilation instruction manual for the ATK-DLRK3568 Android 13 SDK. Before reading this document, please first refer to the contents of Chapter 1 and Chapter 2 in the document: <Development Board CD-ROM A Disk - Basic Materials→10_user_manual→02, Development Documents→ [ALIENTEK] ATK-DLRK3568_Android System Development Manual.pdf> which cover the installation of the Ubuntu system and the setup of the development environment. And follow the 4.1.1 section in this document to install the required software packages.

Chapter 1. Android 13 SDK compilation

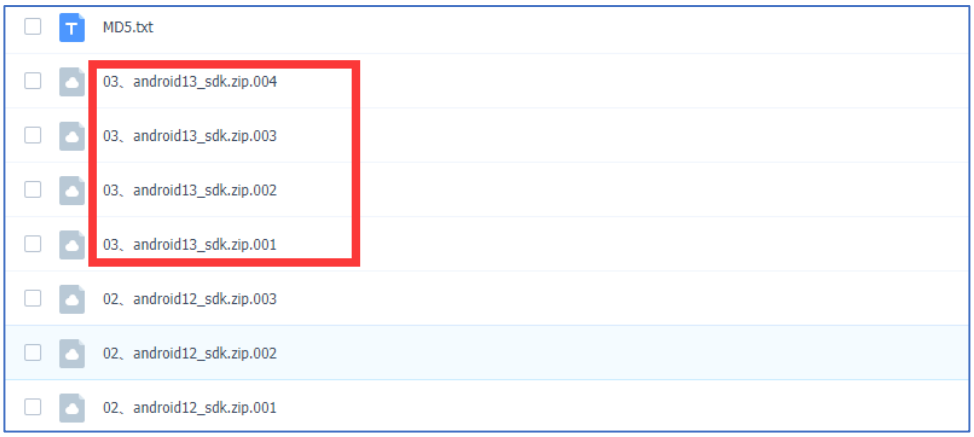
ATK-DLRK3568 Android 13 SDK Compilation Instructions.

1.1 Download SDK

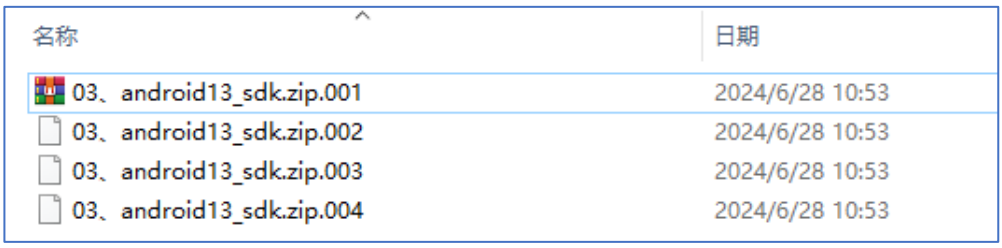
Since the RK3568 and RK3588 development boards of ALIENTEK use the same Android 13 SDK, the materials on the data storage disk of the RK3568 development board are not provided separately. They are only available on the data storage disk of the RK3588 development board. Therefore, everyone needs to download from the data storage disk of the RK3588 development board.

First, go to the following link: http://www.openedv.com/docs/boards/arm-linux/RK3588Linux.html. Then, follow the instructions below to download the Android 13 SDK:

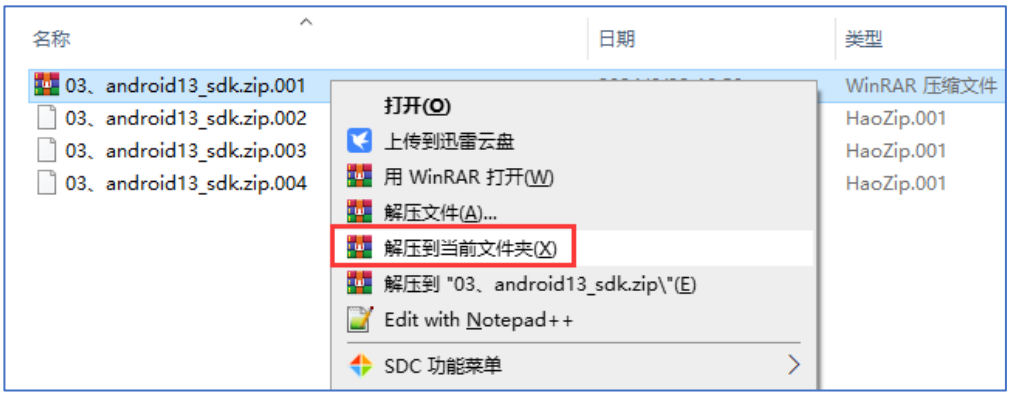
After entering the network drive, go to "Development Board Optical Disk B Drive - **Development Environment and SDK** -> **02, ATK-DLRK3588 Development Board SDK**", and then download the following four files: **03, android13_sdk.zip.001** ~ **03, android13_sdk.zip.004.**



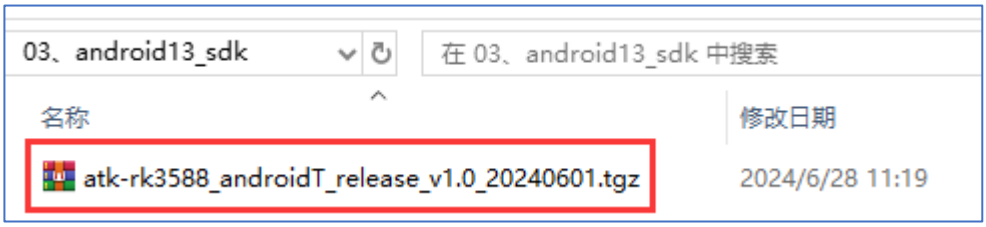
After the download is completed, it will look like this (make sure it is a complete download and you can compare the MD5 value yourself):



Then right-click on **03, android13_sdk.zip.001** to extract it:



After decompression, a folder named "**03" and "android13_sdk**" will be obtained. Enter this folder as follows:



This is the compressed package file of Android 13 SDK. As the version is updated, the name of the SDK compressed package file will also change, but they will all be named as **atk-rk3588_androidT_release_ version_ release date.tgz**!

1.2 Copy the SDK to the Ubuntu system

Note: If the current terminal has already compiled other SDKs, you need to open a new terminal first, and then compile the Android 13 SDK in this new terminal. Otherwise, it may cause compilation errors!!!

Copy the **atk-rk3588_androidT_release_v1.0_20240601.tgz** compressed file to the home directory of the user on the Ubuntu system, and execute the following command to decompress it:

```
mkdir ~/android13_SDK
tar -xzf ~/atk-rk3588_androidT_release_v1.0_20240601.tgz -C ~/android13_SDK
```

```
alientek@alientek-virtual-machine:~$
alientek@alientek-virtual-machine:~$ ls
公共的  模板  视频  图片  文档  下载  音乐  桌面  apache-tomcat-7.0.29  atk-rk3588_androidT_release_v1.0_20240601.tgz  jdk1.6.0_45
alientek@alientek-virtual-machine:~$
alientek@alientek-virtual-machine:~$ mkdir ~/android13_SDK
alientek@alientek-virtual-machine:~$
alientek@alientek-virtual-machine:~$ tar -xzf ~/atk-rk3588_androidT_release_v1.0_20240601.tgz -C ~/android13_SDK
alientek@alientek-virtual-machine:~$
alientek@alientek-virtual-machine:~$ ls
公共的  模板  视频  图片  文档  下载  音乐  桌面  android13_SDK  apache-tomcat-7.0.29  atk-rk3588_androidT_release_v1.0_20240601.tgz
alientek@alientek-virtual-machine:~$
alientek@alientek-virtual-machine:~$
```

The decompression process will take a long time. Please be patient and wait!

After decompression is completed, execute the following command to check out the source code:

```
cd ~/android13_SDK/
.repo/repo/repo sync -l -j10
```

```
alientek@alientek-virtual-machine:~$
alientek@alientek-virtual-machine:~$ cd ~/android13_SDK/
alientek@alientek-virtual-machine:~/android13_SDK$
alientek@alientek-virtual-machine:~/android13_SDK$ .repo/repo/repo sync -l -j10
正在更新文件: 100% (1222/1222), 完成.
正在更新文件: 100% (1133/1133), 完成.
正在更新文件: 100% (9543/9543), 完成.
正在更新文件: 100% (344/344), 完成.
正在更新文件: 100% (1558/1558), 完成.
正在更新文件: 100% (737/737), 完成.
正在更新文件: 100% (2695/2695), 完成.
正在更新文件: 100% (16/16), 完成.
正在更新文件: 100% (2794/2794), 完成.
正在更新文件: 100% (4595/4595), 完成.
正在更新文件: 100% (3440/3440), 完成.
正在更新文件: 100% (78/78), 完成.
正在更新文件: 100% (87/87), 完成.
正在更新文件: 100% (7408/7408), 完成.
正在更新文件: 100% (818/818), 完成.
正在更新文件: 100% (1787/1787), 完成.
正在更新文件: 100% (708/708), 完成.
```

```
正在更新文件: 100% (11975/11975), 完成.
正在更新文件: 100% (590/590), 完成.
正在更新文件: 100% (9218/9218), 完成.
正在更新文件: 100% (9260/9260), 完成.
正在更新文件: 100% (34798/34798), 完成.
正在更新文件: 100% (8370/8370), 完成.
正在更新文件: 100% (1523/1523), 完成.
正在更新文件: 100% (1366/1366), 完成.
正在更新文件: 100% (8703/8703), 完成.m/prebuilts/jdk/jdk17正在更新文件:  48% (4178/8703)
正在更新文件: 100% (1904/1904), 完成.
Checking out: 100% (1169/1169), done in 1h50m51.820s
repo sync has finished successfully.
alientek@alientek-virtual-machine:~/android13_SDK$
```

This process will also last for a long time. Please be patient and wait!

After the checkout is completed, the entire source code directory of Android 13 SDK is obtained, as shown below:

```
alientek@alientek-virtual-machine:~/android13_SDK$
alientek@alientek-virtual-machine:~/android13_SDK$ ls
Android.bp      BUILD         device      kernel-5.10      packages            RKDocs     toolchain
art             build.sh      external    libcore          pdk                 rkst       tools
bionic          cts           frameworks  libnativehelper  platform_testing    RKTools    u-boot
bootable        dalvik        hardware    mkcombinedroot   prebuilts           sdk        vendor
bootstrap.bash  developers    javaenv.sh  mkimage_ab.sh    restore_patches.sh  system     WORKSPACE
build           development   kernel      mkimage.sh       rkbin               test
alientek@alientek-virtual-machine:~/android13_SDK$
```

1.3 Selection of screens

The ALIENTEK ATK-DLRK3568 development board supports the following types of screens:

- ALIENTEK 5.5-inch 1080x1920 (vertical screen) MIPI display screen
- ALIENTEK 5.5-inch 720x1280 (vertical screen) MIPI display screen
- ALIENTEK 10.1-inch 800x1280 (vertical screen) MIPI display screen
- ALIENTEK 10.1-inch 1280x800(Vertical screen) LVDS screen
- Standard HDMI display screen (connected via HDMI cable)
- eDP display screen (a separate adapter board needs to be made to connect the eDP screen. ALIENTEK does not provide eDP screens. Users can purchase them online.)

The ATK-DLRK3568 development board has one MIPI screen interface (MIPI DSI), which can be connected to a 5.5-inch or 10.1-inch MIPI screen from ALIENTEK; one LVDS screen interface, which can be connected to a 10.1-inch 1280x800 LVDS screen from ALIENTEK; one HDMI output interface, which can be connected to a standard HDMI display; one eDP screen interface, which can be connected to an eDP screen; these display interfaces can be found on the board, so no pictures are provided here!

Users need to enable the corresponding display interface based on the screen they purchased and the configuration.

Open the device tree file **kernel-5.10/arch/arm64/boot/dts/rockchip/rk3568-screen_choose.dtsi**:

```
vi kernel-5.10/arch/arm64/boot/dts/rockchip/rk3568-screen_choose.dtsi
```

```
/*
 * 屏幕选择
 * ATK_LCD_TYPE_MIPI_720P:   正点原子5.5寸 720*1280 MIPI屏
 * ATK_LCD_TYPE_MIPI_1080P:  正点原子5.5寸 1080*1920 MIPI屏
 * ATK_LCD_TYPE_MIPI_10P1_800X1280: 正点原子10.1寸 800*1280 MIPI屏
 * ATK_LCD_TYPE_LVDS:        正点原子10.1寸 1280*800 LVDS屏
 * ATK_LCD_TYPE_HDMI:        HDMI显示器
 * ATK_LCD_TYPE_EDP_VGA:     eDP屏或者VGA显示器（硬件默认使能的是VGA接口，若用户需要使用eDP屏，则需修改硬件
 *                           具体情况可以看正点原子RK3568底板原理图！）
 */
/*
 * RK3568可支持三屏显示，也就是三路显示 VP0 VP1 VP2
 * 但是三屏显示需要注意一些问题，具体情况可以看正点原子提供的文档<RK3568三屏显示参考手册>！
 */

/*
 * ATK_LCD_TYPE_MIPI_720P/ATK_LCD_TYPE_MIPI_1080P/ATK_LCD_TYPE_MIPI_10P1_800X1280 三选一
 */
#define ATK_LCD_TYPE_MIPI_720P          // 从VP1输入
//#define ATK_LCD_TYPE_MIPI_1080P       // 从VP1输入
//#define ATK_LCD_TYPE_MIPI_10P1_800X1280       // 从VP1输入

/*
 * ATK_LCD_TYPE_HDMI 和 ATK_LCD_TYPE_EDP_VGA 二选一
 */
#define ATK_LCD_TYPE_HDMI               // 从VP0输入
//#define ATK_LCD_TYPE_EDP_VGA          // 从VP0输入

//#define ATK_LCD_TYPE_LVDS             // 从VP2输入
```

By default, the 5.5-inch 720x1280 MIPI screen and HDMI screen are enabled; if the user does not need to connect the MIPI screen on the board and only uses the HDMI display screen, the definition is as follows:

```
//#define ATK_LCD_TYPE_MIPI_720P                // 从VP1输入
//#define ATK_LCD_TYPE_MIPI_1080P       // 从VP1输入
//#define ATK_LCD_TYPE_MIPI_10P1_800X1280       // 从VP1输入

/*
 * ATK_LCD_TYPE_HDMI 和 ATK_LCD_TYPE_EDP_VGA 二选一
 */
#define ATK_LCD_TYPE_HDMI               // 从VP0输入
//#define ATK_LCD_TYPE_EDP_VGA          // 从VP0输入

//#define ATK_LCD_TYPE_LVDS             // 从VP2输入
```

If the user is using a 10.1-inch 800x1280 MIPI screen, the following definitions apply:

```
//#define ATK_LCD_TYPE_MIPI_720P                  // 从VP1输入
//#define ATK_LCD_TYPE_MIPI_1080P         // 从VP1输入
#define ATK_LCD_TYPE_MIPI_10P1_800X1280 // 从VP1输入

/*
 * ATK_LCD_TYPE_HDMI 和 ATK_LCD_TYPE_EDP_VGA 二选一
 */
#define ATK_LCD_TYPE_HDMI                 // 从VP0输入
//#define ATK_LCD_TYPE_EDP_VGA            // 从VP0输入

//#define ATK_LCD_TYPE_LVDS               // 从VP2输入
```

If the user is using a 10.1-inch LVDS screen, the following definitions apply:

```
//#define ATK_LCD_TYPE_MIPI_720P                  // 从VP1输入
//#define ATK_LCD_TYPE_MIPI_1080P         // 从VP1输入
//#define ATK_LCD_TYPE_MIPI_10P1_800X1280         // 从VP1输入

/*
 * ATK_LCD_TYPE_HDMI 和 ATK_LCD_TYPE_EDP_VGA 二选一
 */
#define ATK_LCD_TYPE_HDMI                 // 从VP0输入
//#define ATK_LCD_TYPE_EDP_VGA            // 从VP0输入

#define ATK_LCD_TYPE_LVDS               // 从VP2输入
```

In summary, users need to configure according to the screen they are using. After the configuration is completed, save and exit!

1.4 Compile SDK

In the android13_SDK directory, execute the following command for compilation:

```
source build/envsetup.sh
lunch ATK_DLRK3568-userdebug
./build.sh -UCKA -J12
```

-U：Indicates the compilation of U-Boot
-K：Indicates the compilation of the Linux Kernel
-C：Compile using the clang compiler.
-A：Indicating the compilation of Android
-J12：Specify the number of compilation threads as 12

The entire compilation process will take a considerable amount of time, approximately 3 to 4 hours or longer, depending on the configuration of the personal computer (CPU and memory).

If there are no unexpected issues, the compilation will be successful. After the compilation is completed, the generated image file will be packaged into the **rockdev/Image-ATK_DLRK3568/** directory, as shown below:

```
Android13_SDK$ cd rockdev/Image-ATK_DLRK3568/
Android13_SDK/rockdev/Image-ATK_DLRK3568$
Android13_SDK/rockdev/Image-ATK_DLRK3568$ ls
baseparameter.img  boot.img     dtbo.img          misc.img       pcba_small_misc.img  recovery.img  super.img  vbmeta.img
boot-debug.img     config.cfg   MiniLoaderAll.bin  parameter.txt  pcba_whole_misc.img  resource.img  uboot.img
Android13_SDK/rockdev/Image-ATK_DLRK3568$
```

1.5 Package into update.img

Execute the following command in the root directory of the SDK to package the individual images in the rockdev/Image-ATK_DLRK3568/ directory into a single update.img firmware, making it convenient for users to flash:

```
./build.sh -u
```

The generated update.img firmware is also stored in the "rockdev/Image-ATK_DLRK3568/" directory.