

## 目录

1.bootanimation.zip.....	2
1.1 The content of desc.txt is as follows.....	2
1.2 part1 & part2.....	3
1.3. Compress files.....	4
2.How to verify.....	4
3.How to add in AOSP.....	4

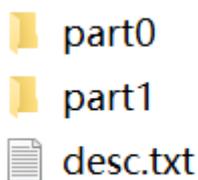
# How to change Boot Animation on SC200R Android10

Created by Great Zheng

The boot animation consists of some JPG/PNG files, these files are included in a zip file named bootanimation.zip. If you want to make your own boot animation, you must prepare some JPG files, the size of the file should be the same with the Screen resolution.

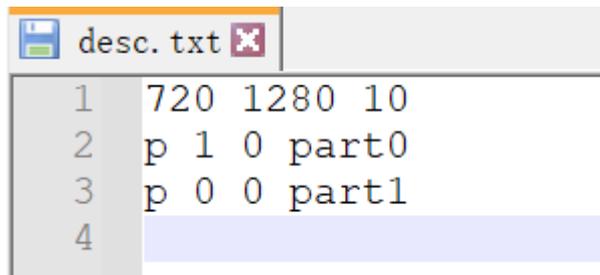
You can check the following steps.

## 1.bootanimation.zip



If you unzip the file, you can see two folders and one desc.txt file in bootanimation.zip, this means the animation consists of two parts, if you want to add more parts, you can create new folder named part2, part3 and so on, two parts are enough.

### 1.1 The content of desc.txt is as follows.



- **720 1280 10:**
  - 720 1280 Resolution of the images(width height),you can modify as per your screen resolution
  - 10 The frame rate at which the images will be displayed
- **p 1 0 part0:**
  - p A tag
  - 1 Defines the number of times this part will b played(1 means one time)
  - 0 Defines the switching interval of this part
  - part0 Folder part0
- **p 0 0 part1:**
  - p A tag
  - 0 Defines the number of times this part will be played(0 means this part will loop

until the system starts)

0 Defines the switching interval of this part

part1 Folder part1

So from the above desc.txt, we divide the animation into two parts ,the animation of part0 will play only one time, the animation of part1 will loop play till Android system starts completely.

## 1.2 part1 & part2

We take a bootanimation.zip as example, we can unzip the zip file.

logo&bootanimation > BootAnimation > bootanimation > part0

---

 00000.jpg

 00001.jpg

 00002.jpg

 00003.jpg

 00004.jpg

We can see part0 contains 5 jpg files.

logo&bootanimation > BootAnimation > bootanimation > part1

---

 00021.jpg

 00022.jpg

 00023.jpg

 00024.jpg

 00025.jpg

 00026.jpg

 00027.jpg

 00028.jpg

 00029.jpg

 00030.jpg

 00031.jpg

 00032.jpg

 00033.jpg

 00034.jpg

 00035.jpg

 00036.jpg

 00037.jpg

 00038.jpg

 00039.jpg

 00040.jpg

 00041.jpg

We can see part1 contains 21 jpg files.

We can put images into part0 and part1. In general, the file name in part0 is from 00000.jpg to 00020.jpg, the file name in part1 is from 00021.jpg to 00040.jpg.

## 1.3. Compress files

We can use WinRAR to compress the files in Windows or zip command in Linux.

zip command: `zip -0 -r ../bootanimation.zip ./*`

Note: **Compression method must be storage mode.**

## 2.How to verify

We can push bootanimation.zip into device to verify the animation effect.

You can do as follows:

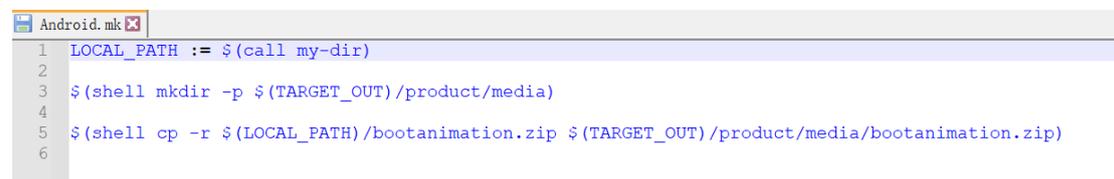
```
adb root
adb disable-verity
adb reboot
adb root
adb remount
adb push bootanimation.zip /system/product/media
adb reboot
```

If all is ok, you can see new boot animation.

## 3.How to add in AOSP

You can create a folder named BootAnimation in **AOSP\vendor\qcom\proprietary\commons\qrdplus\Extension\apps**.

Then you can put your BootAnimation.zip in this folder and create a file named Android.mk, the content is as follows:



```
1 LOCAL_PATH := $(call my-dir)
2
3 $(shell mkdir -p $(TARGET_OUT)/product/media)
4
5 $(shell cp -r $(LOCAL_PATH)/bootanimation.zip $(TARGET_OUT)/product/media/bootanimation.zip)
6
```

In order to make sure this folder can be compiled, you must add the following content to the Android.mk file which is in the up one level directory.

```
Android.mk x
1 # Disable include all subdir makefiles, If any one module can normal compile, please include module makefile in here.
2 #include $(call all-subdir-makefiles)
3
4 BASE_PATH := $(call my-dir)
5
6 include $(BASE_PATH)/PowerOffAlarm/Android.mk
7 include $(BASE_PATH)/SimContacts/Android.mk
8 include $(BASE_PATH)/PerformanceMode/Android.mk
9 include $(BASE_PATH)/QColor/Android.mk
10 include $(BASE_PATH)/CameraExtension/Android.mk
11 include $(BASE_PATH)/BootAnimation/Android.mk
12
```

If you want to burn system.img to verify, you can do as follows:

**make vendorimage -jn**

**make systemimage -jn**

Then you can burn system.img via fastboot and restart the device to check if it can take effect.

**adb reboot bootloader**

**fastboot flash system system.img**

**fastboot reboot**