



SIM8950 Series Smart Module Compilation and Burning

Smart Module

Shanghai SIMCom Wireless Solutions Ltd.
Building A, SIM Technology Building, No.633, Jinzhong Road
Changning District 200335
Tel: 86-21-31575100/31575200
support@simcom.com
www.simcom.com

Document Title:	SIM8950 Series Smart Module Compilation and Burning
Version:	1.01
Date:	2019-04-09
Status:	Release
SIM8950Series Smart Module Compilation and Burning_V1.00	SIM8950Series Smart Module Compilation and Burning

General Notes

SIMCom offers this information as a service to its customers, to support application and engineering efforts that use the products designed by SIMCom. The information provided is based upon requirements specifically provided to SIMCom by the customers. SIMCom has not undertaken any independent search for additional relevant information, including any information that may be in the customer's possession. Furthermore, system validation of this product designed by SIMCom within a larger electronic system remains the responsibility of the customer or the customer's system integrator. All specifications supplied herein are subject to change.

Copyright

This document contains proprietary technical information which is the property of SIMCom Limited., copying of this document and giving it to others and the using or communication of the contents thereof, are forbidden without express authority. Offenders are liable to the payment of damages. All rights reserved in the event of grant of a patent or the registration of a utility model or design. All specification supplied herein are subject to change without notice at any time.

Copyright © Shanghai SIMCom Wireless Solutions Ltd. 2018

Version History

Version	Date	Chapter	What is new
V1.00	2018-11-07		New Version
V1.01	2019-4-9		Revision

Contents

Version History	2
Contents	3
1 Purpose of this document	4
2 Compiling environment	4
2.1. Compiling environment (Recommended hardware or above)	4
2.2. How to compile the entire Android software	4
3 Sub-image compiler	6
4 Burning tool	7

1 Purpose of this document

Firstly, this document describes the establishment of environment that is required to compile firmware for SIMCom smart module SIM8950 series. Also in the document, it provides the steps and commands to compile firmware with examples. Finally, it illustrates how to burn images into SIM8950 series module with QFIL tool.

2 Compiling environment

2.1. Compiling environment (Recommended hardware or above)

The following is an example of the android compiling environment.

CPU: Intel (R) Core (TM) i7-4790 CPU @ 3.60GHz

Memory: 16G

Hard Disk: 1T

Ubuntu: Ubuntu 64bit 14.04

2.2. How to compile the entire Android software

1. Use “apt-get” command to install software packages as below:

The order is as follows:

```
sudo apt-get update
```

```
sudo apt-get install git-core gnupg flex bison gperf build-essential zip curl
```

```
sudo apt-get install zlib1g-dev libc6-dev lib32ncurses5-dev ia32-libs x11proto-core-dev libx11-dev
```

```
sudo apt-get install lib32readline-gplv2-dev lib32z-dev libgl1-mesa-dev g ++ - multilib mingw32
```

```
sudo apt-get install tofrodos python-markdown libxml2-utils xsltproc
```

2. Set bash as the default shell mode

```
$ sudo dpkg-reconfigure dash
```

```
$ sudo rm /bin/sh
```

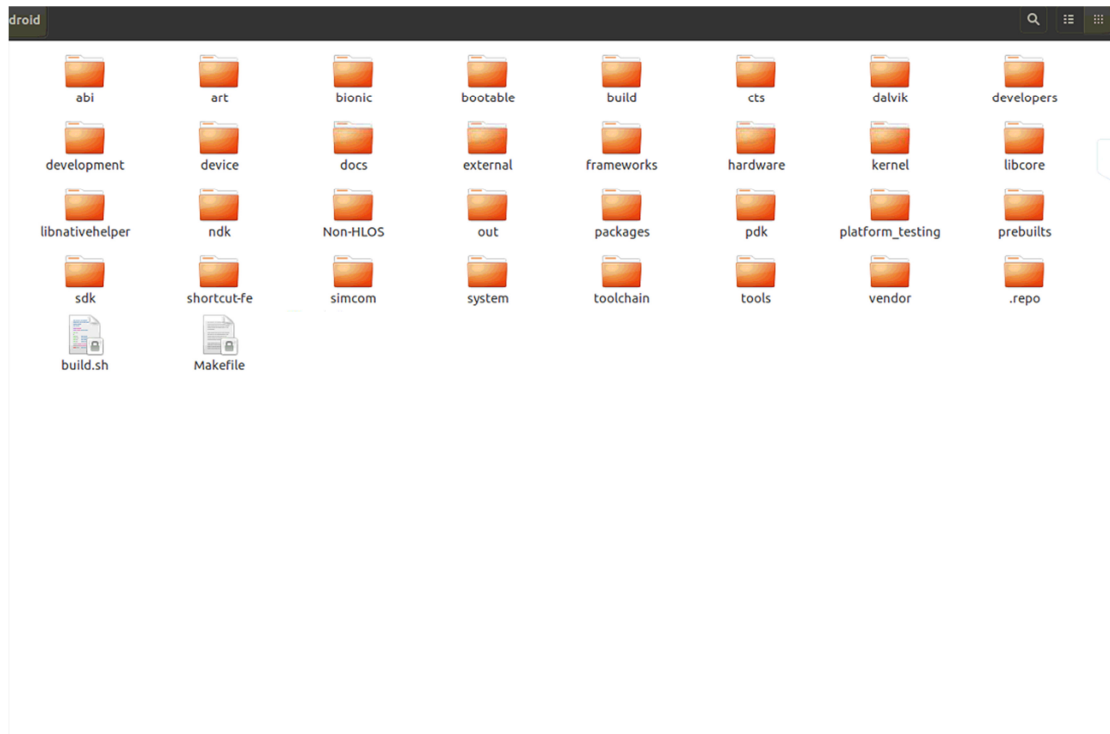
```
$ sudo ln -s /bin/bash /bin/sh
```

3. Use “apt-get” command to install JDK

```
sudo add-apt-repository ppa:openjdk-r/ppa
sudo apt-get update
sudo apt-get install openjdk-8-jdk
```

4. Compile android code

cd SIM8950



Then, in SIM8950 directory, run command below:

- 1 cd simcom, and sh sim8950.sh, and return "SIM8950" directory
- 2 source build/envsetup.sh
- 3 lunch and select "msm8953_64-userdebug"

If you want to build USER version, please add "msm8953_64-user" in device/qcom/common/vendorsetup.sh as following, and select "msm8953_64-user" in lunch command.

```
add_lunch_combo msm8953_32-userdebug
add_lunch_combo msm8953_64-user
add_lunch_combo msm8953_64-userdebug
add_lunch_combo msm8953_64-eng
```

- 4 make update-api
- 5 make -j8("8" means the thread numbers of CPU)

5. After compiling, it will generate many BIN files in directory of“out/target/product/msm8953_64”

Generate the files as the below:

```

android-info.txt  data  kernel  oem4.img  previous_build_config.mk  root  unsigned
boot.img  emmc_appsboot.mbn  module-info.json  OTA_Binary_Packs  ramdisk.img  secimage.log  userdata.img
build_fingerprint.txt  fake_packages  obj  OTA_Target_Files  ramdisk-recovery.img  signed
cache  gen  oem  ota.zip  recovery  symbols
cache.img  installed-files.json  oem2.img  persist  recovery.id  system
clean_steps.mk  installed-files.txt  oem3.img  persist.img  recovery.img  system.img
  
```

(Red marked file is required)

3 Sub-image compiler

1. Compile about:

Input Command:

```
<make about -j8>
```

Target Folder:

```
<SIM8950/out/target/product/msm8953_64>
```

Target File:

```
<emmc_appsboot.mbn>
```

2 Compile kernel:

Input Command:

```
<make bootimage -j8>
```

Target Folder:

```
<SIM8950/out/target/product/msm8953_64>
```

Target File:

```
<boot.img>
```

3. Compile system:

Input Command:

```
<make systemimage -j8>
```

Target Folder:

```
<SIM8950/out/target/product/msm8953_64>
```

Target File:

```
<system.img>
```

4.Compile userdata:

Input Command:

```
<make userdataimage -j8>
```

Target Folder:

```
<SIM8950/out/target/product/msm8953_64>
```

Target File:

```
<userdata.img>
```

5. Compile recovery:

Input Command:

```
<make recoveryimage -j8>
```

Target Folder:

```
<SIM8950/out/target/product/msm8953_64>
```

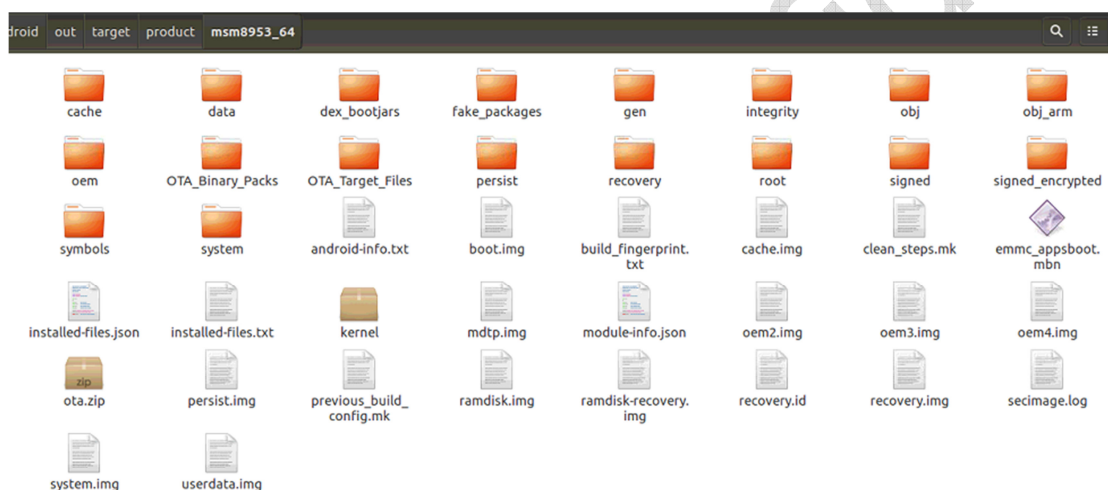
Target File:<recovery.img>

4 Burning tool

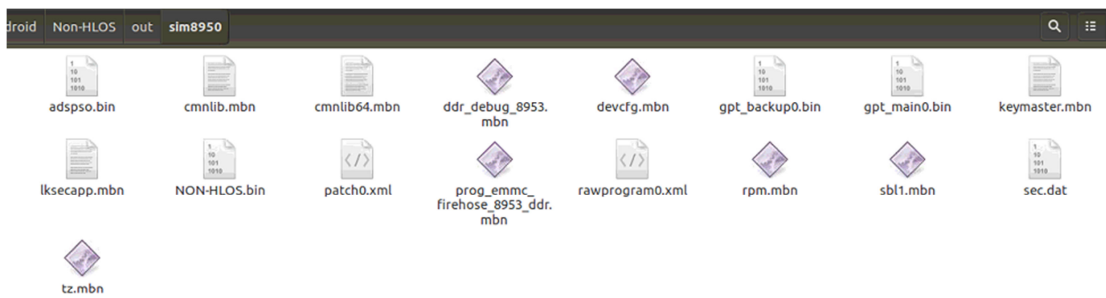
Put compiled files (the following directories) into the same folder.

For example, SIM8950 (the directory as below)

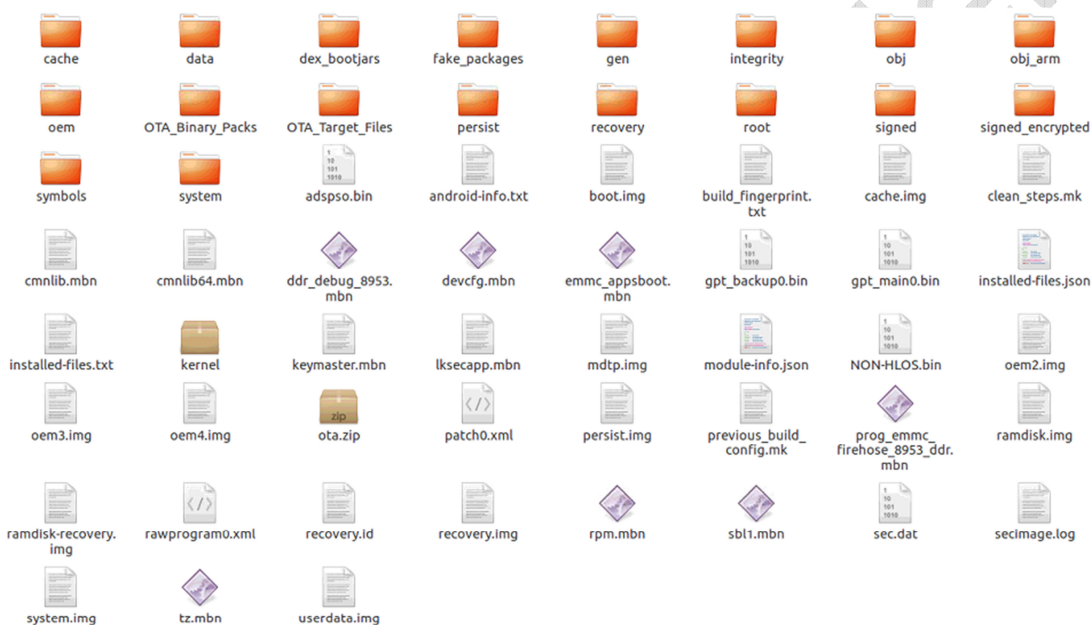
out/target/product/msm8953_64/



Non-HLOS/out/msm8950/



Move above files into the same folder:



Download images with QFIL tool

1. First, please make sure that QPST software and USB driver have been installed on your PC.
2. The device should be power on.
3. Connect USB cable to the computer.

4. Open the QFIL tool.

5. The configuration steps as Figure 1.

Click “Browse” to select “*prog_emmc_firehose_8953_ddr.mbn*” file

Click “Load XML” to select “*rawprogram0.xml*” file

Click the download button.

6. When completion, a prompt is shown as Figure 2.

7. Restart the device.

Figure 1

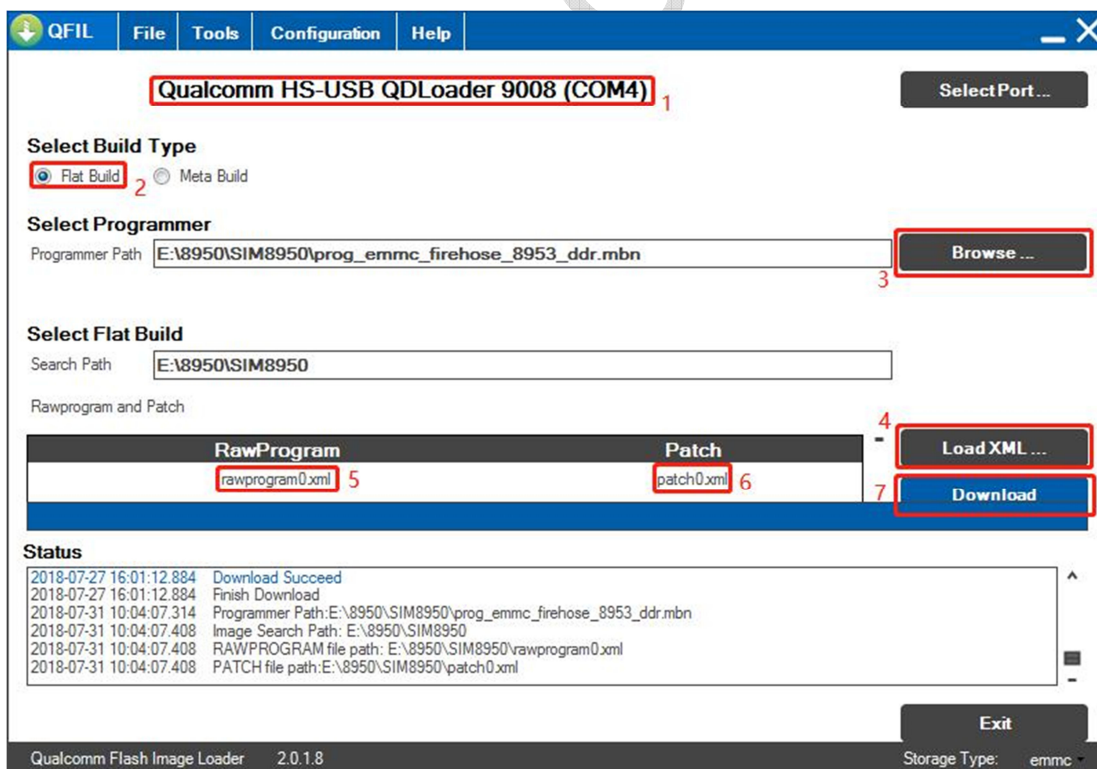



Figure 2


QFIL

File

Tools

Configuration

Help

No Port Available

SelectPort...

Select Build Type

☒ Flat Build
 ☐ Meta Build

Select Programmer

Programmer Path

Browse ...

Select Flat Build

Search Path

Rawprogram and Patch

RawProgram	Patch
rawprogram0.xml	patch0.xml

Load XML ...

Download

Status

2018-07-31 10:14:18.507
 2018-07-31 10:14:18.507 Log is "C:\Users\Administrator\AppData\Roaming\Qualcomm\QFIL\COMPORT_4\port_trace.txt"
 2018-07-31 10:14:18.507
 2018-07-31 10:14:18.528 Waiting for reset done...
 2018-07-31 10:14:19.557 Download Succeed
 2018-07-31 10:14:19.565 Finish Download

Exit

Qualcomm Flash Image Loader 2.0.1.8

Storage Type: emmc