



Fibocom 广和通

完美无线体验

# Fibocom\_Linux 固件升级指南

V1.0

## 版权声明

版权所有©2022 深圳市广和通无线股份有限公司。保留一切权利。

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。

## 注意

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

## 商标申明

 为深圳市广和通无线股份有限公司的注册商标，由所有人拥有。

## 联系方式

公司网址：<https://www.fibocom.com/>

总部地址：深圳市南山区西丽街道西丽社区打石一路深圳国际创新谷六栋 A 座 10-14 层

总机：+86 755-26733555

## 目录

## 目录

修订记录 .....	2
1 引言 .....	3
1.1 适用型号 .....	3
2 编译 .....	4
2.1 工具包介绍 .....	4
2.2 编译 .....	6
2.2.1 Linux 环境编译 upgrade_tool.....	6
2.2.2 Android 环境编译 upgrade_tool.....	7
3 升级 .....	9
3.1 本地升级 .....	9
3.2 远程升级 .....	10
3.3 查看升级结果 .....	10
4 升级参数说明 .....	11
5 NV 还原说明.....	12
5.1 MDM9x07 和 MDM9205 模块还原 NV.....	12
5.2 其他型号模块还原 NV.....	12

# 修订记录

---

V1.0 (2022-08-28)      初始版本

# 1 引言

Upgrade\_tool 用于 Linux 和 Android 主机对广和通模块进行固件升级。

## 1.1 适用型号

序号	模块型号	说明
1	L716/L718	ZTE V3E/T
2	L610	UNISOC 8910
3	MC661-CN-19	UNISOC 8850
4	FG650,FG652,FG621	UNISOC UDX710
5	NL668/ MC116/LC116	QCOM MDM9x07
6	MA510/ MC109/MC100E	QCOM MDM9205
7	FG10x/FM10x	QCOM SDX12

表 1

## 2 编译

---

### 2.1 工具包介绍

- doc: 中英文下载指导文档
- misc\_code: 平台共用代码
- qcom\_code: 高通模块下载相关代码
- zte\_code: 中星微模块下载相关代码
- unisoc\_code: 展锐模块下载相关代码
- main.c: 下载主程序代码
- Makefile: Linux 环境编译配置文件
- Android.mk: Android 环境编译配置文件

详细目录结构如图 1 所示:

```
├── Android.mk
├── doc
│   ├── Fibocon_Linux_Firmware_Upgrade_Guide.pdf
│   └── Fibocon_Linux固件升级指南.pdf
├── main.c
├── Makefile
├── misc_code
│   ├── misc.c
│   ├── misc.h
│   ├── misc_usb.c
│   ├── misc_usb.h
│   ├── usb2tcp.c
│   └── usb2tcp.h
├── qcom_code
│   ├── firehose_download.c
│   ├── firehose_download.h
│   ├── md5sum.c
│   ├── md5sum.h
│   ├── pcie_download.c
│   ├── pcie_download.h
│   ├── qcom_devices_list.h
│   ├── qcom_main.c
│   ├── sahara_protocol.c
│   ├── sahara_protocol.h
│   ├── stream_download.c
│   └── stream_download.h
├── README
├── unisoc_code
│   ├── crc.c
│   ├── nv.c
│   ├── pac.c
│   ├── pac.h
│   ├── unisoc_devices_list.h
│   ├── unisoc_download.c
│   ├── unisoc_download.h
│   ├── unisoc_main.c
│   └── xml.c
├── zte_code
│   ├── zte_devices_list.h
│   ├── zte_download.c
│   ├── zte_download.h
│   └── zte_main.c
└── 5 directories, 37 files
```

图 1

## 2.2 编译

### 2.2.1 Linux 环境编译 upgrade\_tool

- 交叉编译工具

Makefile 默认使用系统 gcc，如果需要 arm gcc，需在 Makefile 设置 CROSS\_COMPILE 变量。

如下图所示：

```
# If use the arm gcc, please set the arm gcc path.
# ARM_GCC = ~/ARM_Linux_GCC/bin/arm-none-linux-gnueabi-

CC = $(ARM_GCC)gcc

CFLAGS = -g -Wall

LD_LIBRARY = -lpthread

INCLUDE = -I./zte_code -I./qcom_code -I./unisoc_code -I./misc_code
SOURCES = $(wildcard ./zte_code/*.c ./qcom_code/*.c ./unisoc_code/*.c ./misc_code/*.c ./*.c)

BIN_TARGET = upgrade_tool

linux: clean
    ${CC} ${CFLAGS} -s ${SOURCES} ${INCLUDE} -o ${BIN_TARGET} ${LD_LIBRARY}

.PHONY: clean

clean:
    @rm -rvf ${BIN_TARGET}

~
~
~
"Makefile" [dos] 22L, 515C                22,0-1                All
```

图 2

- 编译

将下载工具代码放到 Linux 主机，进入代码目录，执行 make 编译，编译成功生成 upgrade\_tool。

如图 3 所示：

```
ght@ght-pc:~/Fibocom_Linux_Firmware_Upgrade_V1.1.0.0
$ ls
Android.mk  main.c      misc_code  README     zte_code
doc         Makefile   qcom_code  unisoc_code
ght@ght-pc:~/Fibocom_Linux_Firmware_Upgrade_V1.1.0.0
$ make
gcc -g -Wall -s ./zte_code/zte_download.c ./zte_code/zte_main.c ./qcom_code/saha
ra_protocol.c ./qcom_code/firehose_download.c ./qcom_code/pcie_download.c ./qcom
_code/stream_download.c ./qcom_code/qcom_main.c ./qcom_code/md5sum.c ./unisoc_co
de/xml.c ./unisoc_code/unisoc_download.c ./unisoc_code/crc.c ./unisoc_code/uniso
c_main.c ./unisoc_code/nv.c ./unisoc_code/pac.c ./misc_code/usb2tcp.c ./misc_cod
e/misc_usb.c ./misc_code/misc.c ./main.c -I./zte_code -I./qcom_code -I./unisoc_c
ode -I./misc_code -o upgrade_tool -lpthread
ght@ght-pc:~/Fibocom_Linux_Firmware_Upgrade_V1.1.0.0
$ ls
Android.mk  main.c      misc_code  README     upgrade_tool
doc         Makefile   qcom_code  unisoc_code  zte_code
ght@ght-pc:~/Fibocom_Linux_Firmware_Upgrade_V1.1.0.0
$
```

图 3

## 2.2.2 Android 环境编译 upgrade\_tool

1. 将解压后的下载工具代码放到 Android 代码目录。
2. 执行 `source build/envsetup.sh`
3. 执行 `Lunch` 选择 Android 主机的编译选项。
4. 执行 `mmm Fibocom_MultiPlatform_Upgrade_Tool_XXX`
5. 编译成功，会生成 upgrade\_tool，编译 log 会打印 upgrade\_tool 生成路径。

例如：`out/target/product/msm8953_64/system/bin/upgrade_tool`。

```
ght@ubuntu-230:~/android_work
$ source build/envsetup.sh
=====BSP_SELECT_ID=====
including device/qcom/common/vendorsetup.sh
including vendor/qcom/proprietary/common/vendorsetup.sh
including sdk/bash_completion/adb.bash
ght@ubuntu-230:~/android_work
$ lunch msm8953_64-user

=====
PLATFORM_VERSION_CODENAME=REL
PLATFORM_VERSION=7.1.2
TARGET_PRODUCT=msm8953_64
TARGET_BUILD_VARIANT=user
TARGET_BUILD_TYPE=release
TARGET_BUILD_APPS=
TARGET_ARCH=arm64
TARGET_ARCH_VARIANT=armv8-a
TARGET_CPU_VARIANT=generic
TARGET_2ND_ARCH=arm
TARGET_2ND_ARCH_VARIANT=armv7-a-neon
TARGET_2ND_CPU_VARIANT=cortex-a53
HOST_ARCH=x86_64
HOST_2ND_ARCH=x86
HOST_OS=linux
HOST_OS_EXTRA=Linux-4.4.0-31-generic-x86_64-with-Ubuntu-14.04-trusty
HOST_CROSS_OS=windows
HOST_CROSS_ARCH=x86
HOST_CROSS_2ND_ARCH=x86_64
HOST_BUILD_TYPE=release
BUILD_ID=N2G47H
OUT_DIR=out
=====
ght@ubuntu-230:~/android_work
$ mmm Fibocom_Linux_Firmware_Upgrade_V1.1.0.0

=====
PRODUCT_COPY_FILES device/qcom/common/media/media_profiles.xml:system/etc/media_profiles.xml ignored.
PRODUCT_COPY_FILES device/qcom/common/media/media_codecs.xml:system/etc/media_codecs.xml ignored.
No private recovery resources for TARGET_DEVICE msm8953_64
616+0 records in
616+0 records out
630784 bytes (631 kB) copied, 0.000854025 s, 739 MB/s
Starting build with ninja
ninja: Entering directory `.'
[100% 25/25] Install: out/target/product/msm8953_64/system/bin/upgrade_tool
make: Leaving directory `~/work/zhoumatiyun/android_work'

#### make completed successfully (7 seconds) ####
```

图 4

## 3 升级

### 3.1 本地升级

1. 查看 USB 连接正常。

```
lsusb
```

**注意：高通型号：如果模块在正常开机模式，需要先发送 at+disk=0,0,0 解锁 diag 端口。**

2. 将固件包和 upgrade\_tool 工具拷贝到主机上面。

3. 进入 upgrade\_tool 所在目录，执行下载命令

```
./upgrade_tool -f firmware image or dir
```

例如：

```
高通型号： ./upgrade_tool -f 19010.1000.00.02.73.15/Maincode -r 1
```

```
展锐型号： ./upgrade_tool -f 16000.1000.00.06.01.05.pac
```

```
中星微型号： ./upgrade_tool -f 17016.1000.00.38.01.21.bin
```

## 3.2 远程升级

如果设备主机没有足够的存储空间，upgrade\_tool 可以用远程主机通过网络连接设备主机升级固件。

操作步骤如下：

1. 将 upgrade\_tool 工具放到设备主机上。
2. 将固件包和 upgrade\_tool 放到远程主机上。
3. 在设备主机运行 upgrade\_tool 启动 USB TCP 服务。

```
./upgrade_tool -d 9008
```

4. 查看设备主机 IP 地址，通过远程主机可以 ping 通设备主机。
5. 在远端主机执行下载命令：

```
./upgrade_tool -f firmware image dir -d The client IP:9008
```

## 3.3 查看升级结果

- 升级成功会打印

```
Upgrade module successfully
```

- 升级失败后，可以从下载日志获取升级失败原因。
- 如果下载的时候添加了 -l 参数，upgrade\_tool 工具会保存升级结果到 update\_result.txt 文件。

升级中：UPGRADING

升级失败：ERROR

升级成功：OK

## 4 升级参数说明

序号	参数	是否必须	描述
1	-f firmware image or dir	是	升级固件包名称或者固件包目录
2	-l log dir	否	设置 log 路径，保存下载失败 log
3	-r 0/1	否	此参数只针对 NL668 项目。当设置-r 1，下载后开机自动还原 NV。不设置-r 1，下载后需要手动执行 AT 命令还原 NV。
4	-d /sys/bus/usb/devices/xx	否	当多个模块连接到主机时，该参数用于指定需要升级的模块。当只有一个模块连接主机时，无需设置该参数
5	-d portname	否	网络远程升级: 模块主机端运行 upgrade_tool 添加-d 9008，远程主机运行 upgrade_tool 添加 -d IP:9008
6	-z 0/1	否	发送 0 包，默认为 0，没有特别说明无需设置参数。
7	-e	否	升级前全擦分区，默认为 0，通常无需设置该参数

## 5 NV 还原说明

### 5.1 MDM9x07 和 MDM9205 模块还原 NV

- 执行 upgrade\_tool 的时候添加 -r 1 参数， 模块第一次开机可以自动还原 NV。

例如：

```
./upgrade_tool -f 19010.1000.00.02.73.15/Maincode -r 1
```

- 如果升级固件的时候未添加-r 1， 可以执行 AT 命令还原 NV。

1. 发送 at 命令: at+efserrfatal
2. 如果模块没有自动重启， 需要发送重启命令: at+cfun=15
3. 模块自动重启进行 NV 还原。
4. 等待开机完成后， 查看 IMEI 和 SN 是否正常。

**注意：MDM9205 TX 项目无法升级 efs， 如果需要升级 EFS 静态 NV， 需要用 Windows 工具 FIBOCOM\_MDM\_MultiUpdater 下载。**

### 5.2 其他型号模块还原 NV

下载的时候无需加-r 参数， 下载完成后， 模块会自动还原 NV。