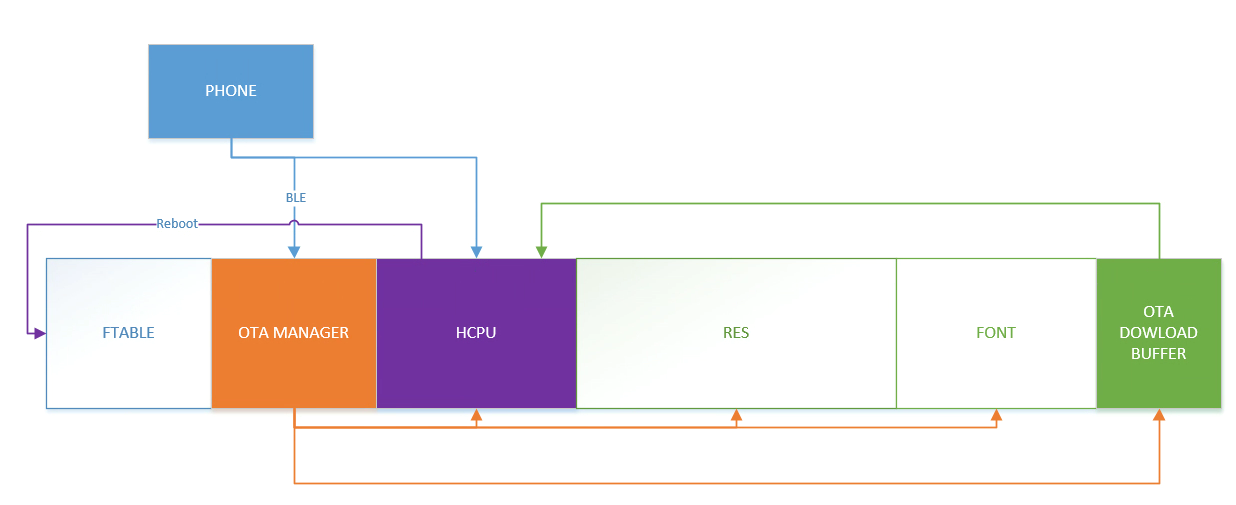
# NOR

## Overview



1. Use OTA Manager bin alone for the upgrade operation.

2. After HCPU receives the upgrade request, restart, after the code is executed to OTA Manager, it will stop at OTA Manager, and then download the corresponding image, flash erase, and only after the upgrade succeeds can it return to normal mode.

3. If there is enough flash space, OTA download buffer can also be configured, HCPU can download code in compressed form, even if the download fails in HCPU, it can also enter the normal system, and 0.7 times the HCPU size space is required to be allocated.

4. 56x NOR defaults to the override method, and no download buffer is configured.

5. To upgrade the OTA Manager, allocate at least 0.7 times the space of the OTA Manager size.

## Make bin

Example:

The meaning of the parameter app 0 0

* app is the name of bin, for example, if it is app.bin, fill in app, if it is ER\_IROM1.bin, fill in ER\_IROM1
* The first number represents the compression type, 0 is override, and 16 is compression
* The second number is Image ID, all IDs are as follows

DFU\_IMG\_ID\_HCPU = 0

DFU\_IMG\_ID\_LCPU = 1

DFU\_IMG\_ID\_LCPU\_ROM\_PATCH = 2

DFU\_IMG\_ID\_RES = 3

DFU\_IMG\_ID\_FONT = 4

DFU\_IMG\_ID\_EX = 5

DFU\_IMG\_ID\_OTA\_MANAGER = 6

DFU\_IMG\_ID\_TINY\_FONT = 7

* Generate HCPU only

gen\_dfu --img\_para app 0 0 --key=s01 --sigkey=sig --dfu\_id=1 --hw\_ver=51 --sdk\_ver=90006 --fw\_ver=1001001 --com\_type=0

* Generate HCPU, resources, fonts

gen\_dfu --img\_para app 0 0 res 0 3 font 0 4 --key=s01 --sigkey=sig --dfu\_id=1 --hw\_ver=51 --sdk\_ver=90006 --fw\_ver=1001001 --com\_type=0

* OTA Manager can only be upgraded separately, only compression is supported, and DFU\_ID = 2

run imgtool\_new.py gen\_dfu --img\_para ota 16 6 --key=s01 --sigkey=sig --dfu\_id=2 --hw\_ver=51 --sdk\_ver=7001 --fw\_ver=1001001 --com\_type=0

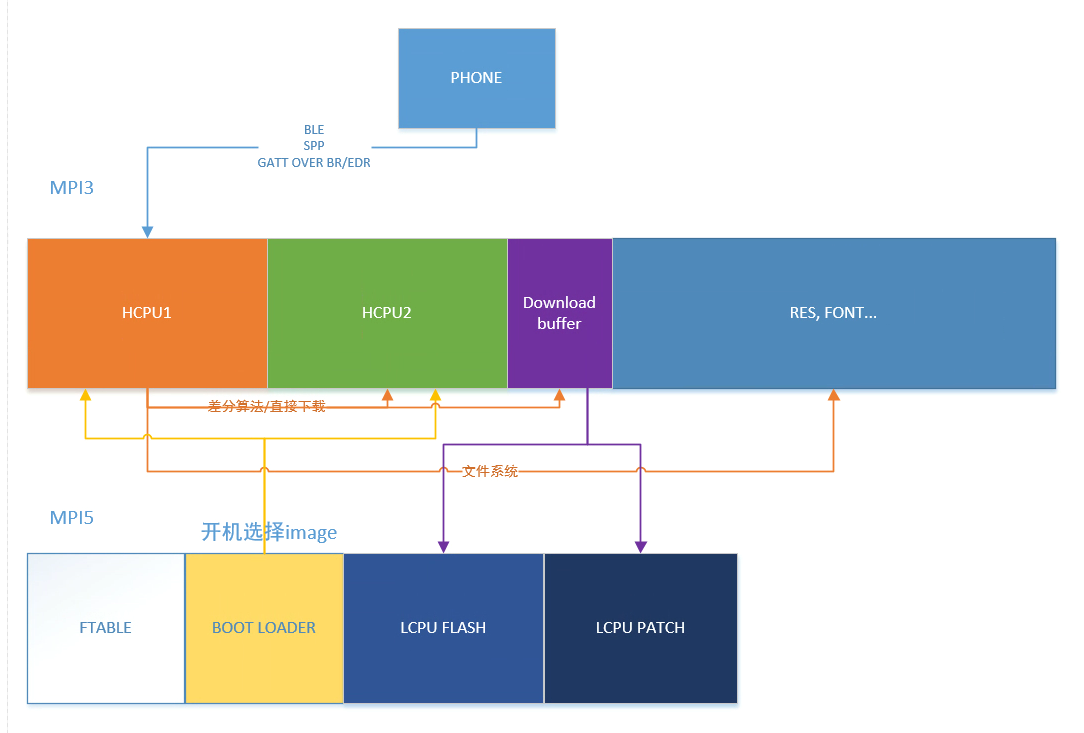
* On NOR, the LCPU ROM PATCH must be upgraded in the form of compression, parameter 16 2

For example, make an upgrade package that includes HCPU, LCPU FLASH, LCPU ROM PATCH

gen\_dfu --img\_para app 0 0 lcpu 0 1 lcpu\_rom\_patch 16 2 --key=s01 --sigkey=sig --dfu\_id=1 --hw\_ver=51 --sdk\_ver=90006 --fw\_ver=1001001 --com\_type=0

# NAND

## Overview

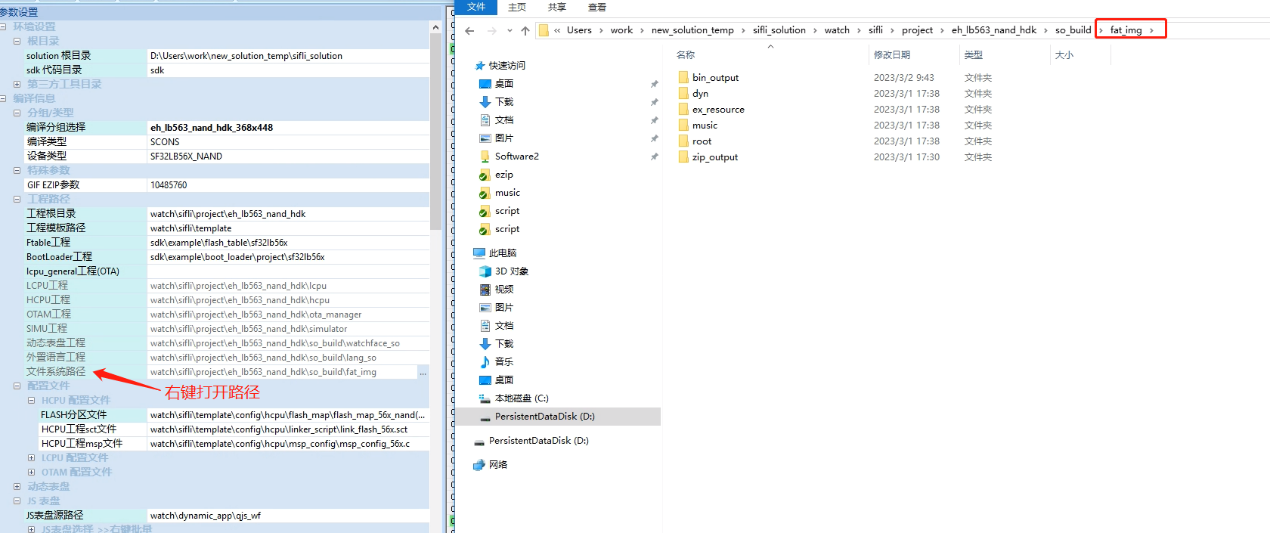


* It mainly includes upgrade resources (file part) and upgrade HCPU (image part).
* The upgrade of HCPU is basically the same as the NOR upgrade.
* The resource part will only upgrade the specific files to be upgraded, which can be downloaded through the current HCPU, and use the file system to add, delete, and modify them accordingly.
* The HCPU part will be upgraded by pingpong, the current working HCPU will be used to download the spare HCPU.
* At present, the full HCPU bin is downloaded directly, and in the future, the download of differential patches will be considered, and the new HCPU bin will be obtained through the differential algorithm, which can reduce the transfer content.
* For LCPU FLASH, LCPU ROM, if an upgrade is required, it will be downloaded to the download cache area, and then in the final stage of OTA, updated to MPI5.
* OTA will be rebooted later, and if the running bin of HCUP is switched, the desired HCPU will be selected in the boot loader for jumping

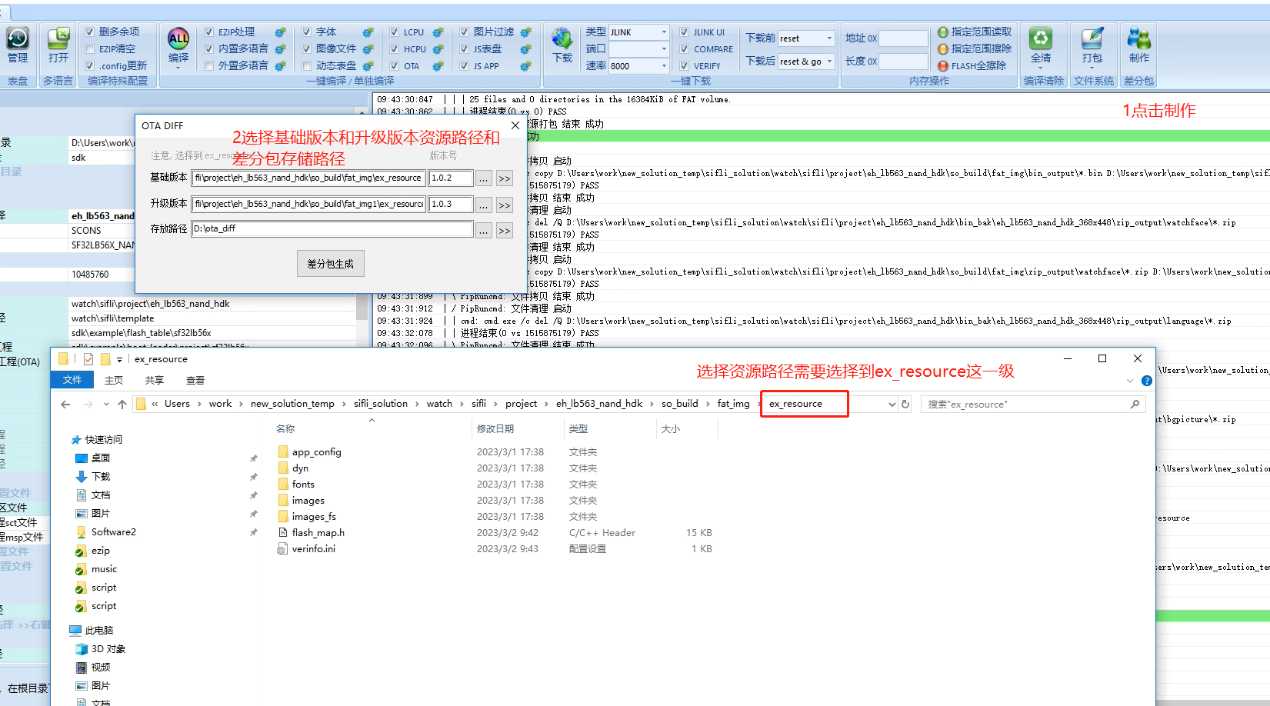
## Make file upgrade package

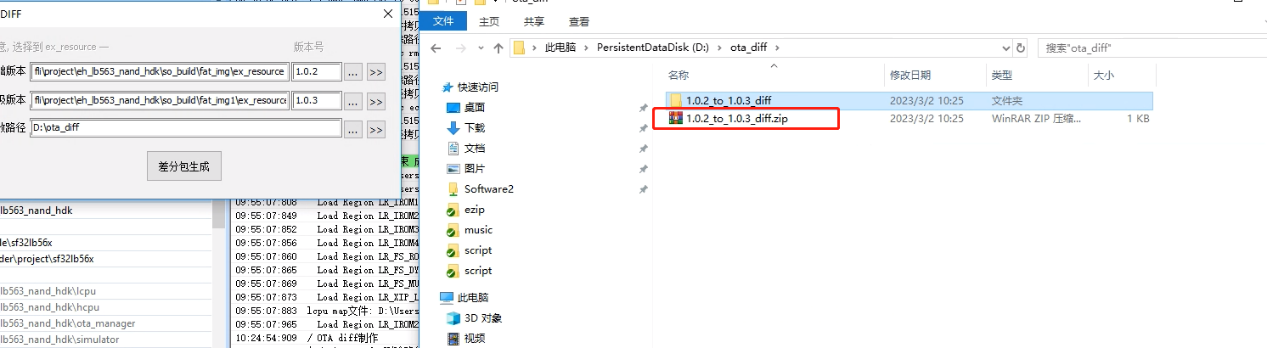
1: Compile the basic version, save the fat\_img folder, if the upgrade project and the basic version are not the same project, copy to another path or rename it to avoid overwriting the compiled and upgraded version.

2: Compile the upgrade version and save the fat\_img folder.



3: Differential package production, the generated zip package is the resource package from the basic version to the upgrade version





## Make bin

Example:

The meaning of the parameter app 0 0

* app is the name of bin, for example, if it is hcpu.bin, fill in hcpu, if it is ER\_IROM1.bin, fill in ER\_IROM1
* The first number represents the compression type, 0 is override, and 16 is compression
* The second number is Image ID, all IDs are as follows

IMAGE\_ID\_HCPU = 0

IMAGE\_ID\_LCPU = 1

IMAGE\_ID\_HCPU\_PATCH = 2

IMAGE\_ID\_FS\_ROOT = 3

IMAGE\_ID\_LCPU\_PATCH = 4

IMAGE\_ID\_FS\_DYN = 5

IMAGE\_ID\_FS\_MUSIC = 6

To make upgrade files containing the following types

MAGE\_ID\_FS\_ROOT

IMAGE\_ID\_FS\_DYN

IMAGE\_ID\_FS\_MUSIC

Completely erase and upgrade the bin of the file system, the bin of the file system is relatively large, if it is all erased, the upgrade would be slow, therefore it is not recommended for users to completely upgrade such files. If there is a need to upgrade, it is recommended to use the aforementioned differential to make file upgrade package.

HCPU\_PATCH is not used yet

* Compared to the execution parameters of nor, --bksize=2048 has been added

gen\_dfu --img\_para hcpu 0 0 --key=s01 --sigkey=sig --dfu\_id=1 --hw\_ver=51 --sdk\_ver=7001 --fw\_ver=1001001 --com\_type=0 --bksize=2048

gen\_dfu --img\_para ER\_IROM1 0 0 lcpu\_flash 0 1 lcpu\_rom\_patch 0 4 --key=s01 --sigkey=sig --dfu\_id=1 --hw\_ver=51 --sdk\_ver=7001 --fw\_ver=1001001 --com\_type=0 --bksize=2048

# Demo app

## Nor:

CTRL: Select the ctrl file

HCPU: Select the created HCPU upgrade package with the name outER\_IROM1.bin, if the name itself is ER\_IROM1.bin

RES: Same as above

Font: Same as above

Nor old is for 55x and early 56x

Nor optimized part of the OTA start commands

## Nand:

CTRL: Select the ctrl file

HCPU: Select the created HCPU upgrade package with the name outER\_IROM1.bin, if the name itself is ER\_IROM1.bin

LCPU and LCPU PATCH can be selected in the same way.

NAND RES: Resource upgrade package zip

Start: Start the upgrade

resume: Try to renew the transfer

