



Panchip Microelectronics Co., Ltd.

Pan10xx ROM DFU 终端工具指南

当前版本：1.0

发布日期：2025.03

上海磐启微电子有限公司

地址：上海张江高科技园区盛夏路666号D栋3楼

联系电话：021-50802371

网址：<http://www.panchip.com>

工具更新版本查询下载地址

[下载](#)

文档说明

由于版本升级或存在其他原因，本文档内容会不定期进行更新。除非另有约定，本文档内容仅作为使用指导，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

磐启是磐启微电子公司的商标。本文档中提及的其他名称是其各自所有者的商标/注册商标。

免责声明

本文件中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，
磐启微电子有限公司对本文件容不做任何明示或暗示的声明或保证。

修订历史

1.概述

2.工具说明

- 2.1.直接调用工具 exe 加参数执行 DFU
- 2.2.运行工具 exe 之后执行命令运行
- 2.3.DFU 参数说明
- 2.4. DFU 执行流程

3.应用实例

- 3.1.直接调用工具 exe 加参数执行 DFU
- 3.2.运行工具 exe 之后执行 DFU 运行

1.概述

为 PAN107x/PAN101x 芯片 ROM 模式 USB 设备终端 DFU 升级工具。

2.工具说明

2.1.直接调用工具 exe 加参数执行 DFU

命令格式：Pan10xxRomDfuConsoleTool.exe [DFU 参数](#)

如果参数校验失败则进入直接运行 Pan10xxRomDfuConsoleTool.exe 格式

2.2.运行工具 exe 之后执行命令运行

直接双击运行 Pan10xxRomDfuConsoleTool.exe 工具

然后输入命令执行对应功能。

```
Pan10xx ROM DFU V0.0.1
=====
支持命令:
  help: 查看命令帮助
  v: 查看工具版本信息
  exit: 退出
  dfu: DFU 命令
      命令格式: dfu -w wait -s flash_size -m mode -p addr pg_path
      -w: 为设置查询设备连接等待标志
          wait: -w 后面紧跟的查询设备连接等待参数, 可设置值 0、1, 默认为 0
              0: 为查询设备连接不等待, 查询一次设备连接未查询到则失败退出
              1: 为查询设备连接等待, 一直在查询设备连接, 直到查询到设备连接为止
      -s: 为设置芯片程序区大小标志
          flash_size: -s 后面紧跟的芯片程序区大小参数, 范围 0 ~ 2, 默认为 1
              0: 为 256 KB, 程序区大小 252 KB
              1: 为 512 KB, 程序区大小 508 KB
              2: 为 1 MB, 程序区大小 1020 KB
      -m: 为设置 DFU 模式标志
          mode: -m 后面紧跟的 DFU 模式参数, 可设置值 0、1, 默认为 0
              0: 为 DFU 升级下载完成, 读取校验 (DFU 成功能保证升级成功程序的正确性, 但耗时较长)
              1: 为单独 DFU 升级下载 (DFU 成功不能完全保证升级成功程序的正确性, 但耗时较短)
      -p: 为设置 DFU 程序标志, 支持传入多个程序, 只要程序大小未超出芯片程序区范围, 以及程序之间没有重叠区域即可
          addr: -p 后面紧跟的程序 DFU 升级下载的起始地址参数, 十六进制格式 (例如: A000) 范围为芯片程序区范围
              范围为芯片程序区范围: 为指定程序起始地址, 对于 *.bin 程序文件必须指定有效地址
              FFFFFFFF: 对于 *.hex 格式程序文件, 以程序文件起始地址为准
          pg_path: -p 后面紧跟的在 addr 参数后面, 以英文空格隔开的 DFU 下载程序路径参数
=====
>>
```

图2-2-1 Pan10xxRomDfuConsoleTool.exe 直接运行显示

如图 2-2-1 所示, 为 Pan10xxRomDfuConsoleTool.exe 直接运行显示。

支持命令：

命令	说明
help	帮助命令，输入执行时会打印输出如图 2-2-1 所示支持命令显示。
v	查看版本命令，输入执行时，会打印输出工具版本信息。
exit	退出工具
dfu	执行 DFU 命令，输入执行时后面需要跟 DFU 参数

2.3.DFU 参数说明

参数格式：-w wait -s flash_size -m mode -p addr pg_path

参数之间以英文空格隔开。

-w：为设置查询设备连接等待标志，**可选设置**。

wait: -w 后面紧跟的查询设备连接等待参数，可设置值 0、1，默认为 0。

0: 为查询设备连接不等待，查询一次设备连接未查询到则失败退出。

1: 为查询设备连接等待，一直在查询设备连接，直到查询到设备连接为止。

-s: 为设置芯片程序区大小标志，**可选设置**。

flash_size: -s 后面紧跟的芯片程序区大小参数，范围 0 ~ 2，默认为 1。

0: 为 256 KB，程序区大小 252 KB。

1: 为 512 KB，程序区大小 508 KB。

2: 为 1 MB，程序区大小 1020 KB。

-m: 为设置 DFU 模式标志，**可选设置**。

mode: -m 后面紧跟的 DFU 模式参数，可设置值 0、1，默认为 0。

0: 0: 为 DFU 升级下载完成，读取校验（DFU 成功能保证升级成功程序的正确性，但耗时较长）。

1: 为单独 DFU 升级下载（DFU 成功不能完全保证升级成功程序的正确性，但耗时较短）。

-p: 为设置 DFU 程序标志，支持传入多个程序，只要程序大小未超出芯片程序区范围，以及程序之间没有重叠区域即可，**必须设置**。

addr: -p 后面紧跟的程序 DFU 升级下载的起始地址参数，十六进制格式（例如：A000）范围为芯片程序区范围。

范围为芯片程序区范围: 为指定程序起始地址，对于 *.bin 程序文件必须指定有效地址。

FFFFFFFF: 对于 *.hex 格式程序文件，以程序文件起始地址为准。

pg_path: -p 后面紧跟的在 addr 参数后面，以英文空格隔开的 DFU 下载程序路径参数。

2.4. DFU 执行流程

1、检测 DFU 执行传入参数

检测成功，进入下一步执行。

检测失败，如果是直接调用工具 exe 加参数模式，则直接运行工具 exe，如果是运行工具 exe 执行 DFU 则失败退出。

2、载入 DFU 程序，并校验载入的程序是否越界或重叠

所有程序载入成功，并且程序未超出指定 flash 大小区域，以及多个程序之间未有重叠区域，进入下一步执行。

载入程序失败，或者程序超出指定 flash 大小区域，或者多个程序之间有重叠区域，如果是直接调用工具 exe 加参数模式，则直接运行工具 exe，如果是运行工具 exe 执行 DFU 则失败退出。

3、查询 PAN10xx ROM 模式 USB 设备连接。

如果设置查询连接不等待，则会查询一次设备连接，未查询到则失败退出，查询到则进入下一步执行。

如果设置查询连接等待，则会一直在查询设备连接，直到查询到设备连接为止，进入下一步执行。

4、按照载入程序大小以扇区擦除预下载程序区。

擦除成功，进入下一步执行。

擦除失败，DFU 则失败退出。

5、下载程序到芯片 Flash 对应区域。

全部执行下载成功，进入下一步执行。

执行下载失败，DFU 则失败退出。

6、读取校验，**注**：只有设置 DFU 模式下载完成校验才会执行。

读取芯片下载写入的所有程序数据与下载的程序数据进行对比校验。

校验成功，进入下一步执行；校验失败，DFU 则失败退出。

7、发送重启芯片命令，DFU 成功。

注：直接调用工具 exe 加参数执行 DFU 时，只刷新输出 DFU 百分比进度(>>：下载进度: x%)，以及输出最终 DFU 结果以及总耗时。

运行工具 exe 之后执行 DFU 运行时，会输出 DFU 所有流程 Log 信息。

3.应用实例

3.1.直接调用工具 exe 加参数执行 DFU

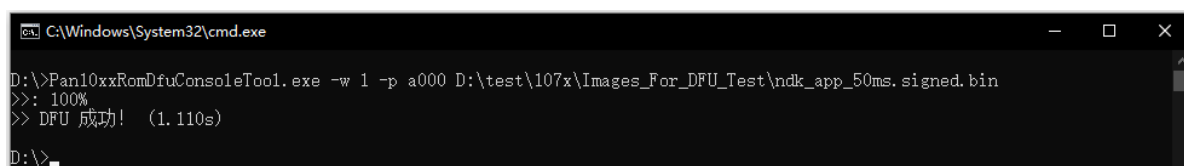
命令： Pan10xxRomDfuConsoleTool.exe -w 1 -p a000
D:\test\107x\Images_For_DFU_Test\ndk_app_50ms.signed.bin

Pan10xxRomDfuConsoleTool.exe： Pan10xxRomDfuConsoleTool.exe 程序名，**注：**
Pan10xxRomDfuConsoleTool.exe 必须在当前目录，否则需要该参数为工具文件路径

-w 1： 为查询连接等待。

-p a000 D:\test\107x\Images_For_DFU_Test\ndk_app_50ms.signed.bin： 设置传入程序下载起始地址为: 0xA000， 程序文件路径: D:\test\107x\Images_For_DFU_Test\ndk_app_50ms.signed.bin。

执行命令时：则进入 [2.4. DFU 执行流程](#)。



```
C:\Windows\System32\cmd.exe
D:\>Pan10xxRomDfuConsoleTool.exe -w 1 -p a000 D:\test\107x\Images_For_DFU_Test\ndk_app_50ms.signed.bin
>>: 100%
>> DFU 成功! (1.110s)
D:\>_
```

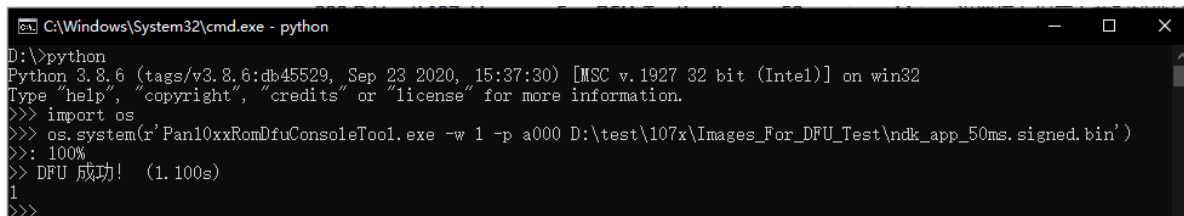
图3-1-1 直接调用 Pan10xxRomDfuConsoleTool.exe 执行 DFU 显示

如图3-1-1所示，直接在终端调用 Pan10xxRomDfuConsoleTool.exe + 参数执行 DFU。

DFU 过程会刷新显示 DFU 进度百分比显示。

DFU 成功：输出 **DFU 成功!** 以及总耗时。

自动结束运行。



```
C:\Windows\System32\cmd.exe - python
D:\>python
Python 3.8.6 (tags/v3.8.6:db45529, Sep 23 2020, 15:37:30) [MSC v.1927 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import os
>>> os.system(r'Pan10xxRomDfuConsoleTool.exe -w 1 -p a000 D:\test\107x\Images_For_DFU_Test\ndk_app_50ms.signed.bin')
>>>: 100%
>> DFU 成功! (1.100s)
1
>>>
```

图3-1-2 使用 python 脚本调用 Pan10xxRomDfuConsoleTool.exe 执行 DFU 显示

如图3-1-2所示，使用 python 脚本调用 Pan10xxRomDfuConsoleTool.exe + 参数执行 DFU。

DFU 过程会刷新显示 DFU 进度百分比显示。

DFU 成功：输出 **DFU 成功!** 以及总耗时。

成功返回：1；失败返回：0

自动结束运行。

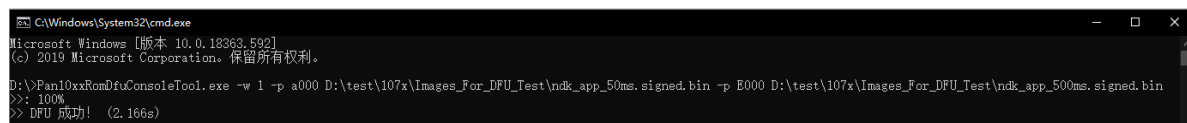
命令： Pan10xxRomDfuConsoleTool.exe -w 1 -p a000
D:\test\107x\Images_For_DFU_Test\ndk_app_50ms.signed.bin -p E000
D:\test\107x\Images_For_DFU_Test\ndk_app_500ms.signed.bin

-w 1： 为查询连接等待。

-p a000 D:\test\107x\Images_For_DFU_Test\ndk_app_50ms.signed.bin：设置传入程序下载起始地址为: 0xA000，程序文件路径: D:\test\107x\Images_For_DFU_Test\ndk_app_50ms.signed.bin。

-p E000 D:\test\107x\Images_For_DFU_Test\ndk_app_500ms.signed.bin：设置传入程序下载起始地址为: 0xE000，程序文件路径:
D:\test\107x\Images_For_DFU_Test\ndk_app_500ms.signed.bin。

执行命令时：则进入 [2.4. DFU 执行流程](#)。



```
C:\Windows\System32\cmd.exe
Microsoft Windows [版本 10.0.18363.592]
(c) 2019 Microsoft Corporation. 保留所有权利。

D:\>Pan10xxRomDfuConsoleTool.exe -w 1 -p a000 D:\test\107x\Images_For_DFU_Test\ndk_app_50ms.signed.bin -p E000 D:\test\107x\Images_For_DFU_Test\ndk_app_500ms.signed.bin
>>: 100%
>> DFU 成功! (2.166s)
```

图3-1-3 直接调用 Pan10xxRomDfuConsoleTool.exe 执行 DFU 下载多个程序显示

如图3-1-3所示，直接在终端调用 Pan10xxRomDfuConsoleTool.exe + 参数执行 DFU。

DFU 过程会刷新显示 DFU 进度百分比显示。

DFU 成功：输出 **DFU 成功!** 以及总耗时。

自动结束运行。

3.2.运行工具 exe 之后执行 DFU 运行

直接双击运行 Pan10xxRomDfuConsoleTool.exe 工具，然后在工具的终端窗口中输入 DFU 命令。

命令：dfu -w 1 -p a000 D:\test\107x\Images_For_DFU_Test\ndk_app_50ms.signed.bin

dfu：DFU 命令字符串。

-w 1：为查询连接等待。

-p a000 D:\test\107x\Images_For_DFU_Test\ndk_app_50ms.signed.bin：设置传入程序下载起始地址为: 0xA000，程序文件路径: D:\test\107x\Images_For_DFU_Test\ndk_app_50ms.signed.bin。

执行命令时：则进入 [2.4. DFU 执行流程](#)。

```
C:\Windows\System32\cmd.exe
(c) 2019 Microsoft Corporation. 保留所有权利。

D:\>Pan10xxRomDfuConsoleTool.exe
Pan10xx ROM Dfu V0.0.1
=====
支持命令:
help: 查看命令帮助
v: 查看工具版本信息
exit: 退出
dfu: Dfu 命令
命令格式: dfu -w wait -s flash_size -m mode -p addr pg_path
-w: 为设置查询设备连接等待标志
wait: -w 后面紧跟的查询设备连接等待参数, 可设置值 0、1, 默认为 0
0: 为查询设备连接不等待, 查询一次设备连接未查询到则失败退出
1: 为查询设备连接等待, 一直在查询设备连接, 直到查询到设备连接为止
-s: 为设置芯片程序区大小标志
flash_size: -s 后面紧跟的芯片程序区大小参数, 范围 0 ~ 2, 默认为 1
0: 为 256 KB, 程序区大小 252 KB
1: 为 512 KB, 程序区大小 508 KB
2: 为 1 MB, 程序区大小 1020 KB
-m: 为设置 Dfu 模式标志
mode: -m 后面紧跟的 Dfu 模式参数, 可设置值 0、1, 默认为 0
0: 为 Dfu 升级下载完成, 读取校验 (Dfu 成功能保证升级成功程序的正确性, 但耗时较长)
1: 为单独 Dfu 升级下载 (Dfu 成功不能完全保证升级成功程序的正确性, 但耗时较短)
-p: 为设置 Dfu 程序标志, 支持传入多个程序, 只要程序大小未超出芯片程序区范围, 以及程序之间没有重叠区域即可
addr: -p 后面紧跟的程序 Dfu 升级下载的起始地址参数, 十六进制格式 (例如: A000) 范围为芯片程序区范围
范围: 为芯片程序区范围: 为指定程序起始地址, 对于 *.bin 程序文件必须指定有效地址
FFFFFFFF: 对于 *.hex 格式程序文件, 以程序文件起始地址为准
pg_path: -p 后面紧跟的在 addr 参数后面, 以英文空格隔开的 Dfu 下载程序路径参数
=====
>> dfu -w 1 -p a000 D:\test\107x\Images_For_DFU_Test\ndk_app_50ms.signed.bin
>> 进入 Dfu
>> 查询 USB 设备连接
>> , 等待连接到设备: "Panchip USB Dfu(HID\VID_0000&PID_C081&REV_0100)"
>> 擦除设备 Flash
>> : 擦除 Flash 区域 (0x0000A000 - 0x000034E8) >>> : 执行成功! (0.069s, 0.070s)
>> 下载程序
>> : 下载 (0x0000A000 - 0x000034E8 )
>> : 下载进度: 100%
>> : 执行成功! (0.603s, 0.678s)
>> 校验
>> : 读取校验 (0x0000A000 - 0x000034E8 )
>> : 校验进度: 100%
>> : 执行成功! (0.496s, 1.174s)
>> 重启设备
>> : 执行成功! (0.001s, 1.177s)
>> Dfu 成功! (1.177s)
```

图3-2-1 先运行 Pan10xxRomDfuConsoleTool.exe 之后, 执行 Dfu 命令显示

如图3-2-1所示, 先运行 Pan10xxRomDfuConsoleTool.exe 然后输入 Dfu 命令执行 Dfu 的过程显示。

命令: dfu -w 1 -p a000 D:\test\107x\Images_For_DFU_Test\ndk_app_50ms.signed.bin

Dfu 过程会显示每一个步骤的日志信息。

Dfu 成功: 最终输出 **Dfu 成功!** 以及总耗时。

执行完 Dfu 自动结束运行。

```
>> dfu -w 1 -p a000 D:\test\107x\Images_For_DFU_Test\ndk_app_50ms.signed.bin -p E000 D:\test\107x\Images_For_DFU_Test\ndk_app_500ms.signed.bin
>> 进入 Dfu
>> 查询 USB 设备连接
>> , 等待连接到设备: "Panchip USB Dfu(HID\VID_0000&PID_C081&REV_0100)"
>> 擦除设备 Flash
>> : 擦除 Flash 区域 (0x0000A000 - 0x000074EC) >>> : 执行成功! (0.139s, 0.139s)
>> 下载程序
>> : 下载 (0x0000A000 - 0x000034E8 )
>> : 下载进度: 100%
>> : 下载 (0x0000E000 - 0x000034EC )
>> : 下载进度: 100%
>> : 执行成功! (0.595s, 1.294s)
>> 校验
>> : 读取校验 (0x0000A000 - 0x000034E8 )
>> : 校验进度: 100%
>> : 读取校验 (0x0000E000 - 0x000034EC )
>> : 校验进度: 100%
>> : 执行成功! (0.500s, 2.294s)
>> 重启设备
>> : 执行成功! (0.002s, 2.296s)
>> Dfu 成功! (2.296s)
```

图3-2-2 先运行 Pan10xxRomDfuConsoleTool.exe 之和, 执行 Dfu 命令添加多个程序显示

如图3-2-2所示, 先运行 Pan10xxRomDfuConsoleTool.exe 然后输入 Dfu 命令执行 Dfu 的过程显示。

命令: dfu -w 1 -p a000 D:\test\107x\Images_For_DFU_Test\ndk_app_50ms.signed.bin -p E000 D:\test\107x\Images_For_DFU_Test\ndk_app_500ms.signed.bin

DFU 过程会显示每一个步骤的日志信息。

DFU 成功：最终输出 **DFU 成功!** 以及总耗时。

执行完 DFU 自动结束运行。