



量产烧写 使用指南

文档版本 13

发布日期 2016-04-19

版权所有 © 深圳市海思半导体有限公司 2016。保留一切权利。

未经本公司书面许可，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。

商标声明



HISILICON、海思和其他海思商标均为深圳市海思半导体有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受海思公司商业合同和条款的约束，本文档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，海思公司对本文档内容不做任何明示或默示的声明或保证。

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。

深圳市海思半导体有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编：518129

网址： <http://www.hisilicon.com/cn/>

客户服务电话： 4008302118

客户服务邮箱： support@hisilicon.com



前 言

概述

本文主要介绍 Hi3716CV200 系列平台和 Hi379XX VX00 平台的量产烧录方案，包括如何制作量产烧录镜像、烧录方法及烧写注意事项等。

产品版本

与本文档相对应的产品版本如下。

产品名称	产品版本
Hi3716C	V2XX
Hi3719C	V1XX
Hi3718C	V1XX
Hi3719M	V1XX
Hi3718M	V1XX
Hi3716M	V4XX
Hi3798C	V1XX
Hi3798C	V2XX
Hi3796C	V1XX
Hi3798M	V1XX
Hi3716M	V31X
Hi3110E	V5XX
Hi3716M	V33X



读者对象

本文档（本指南）主要适用于以下工程师：

- 技术支持工程师
- 软件开发工程师

作者信息

章节号	章节名称	作者信息
1	概述	L00227819/Y00250993
2	HiPro 工具使用方法	L00227819/Y00250993
3	量产 MAC ID 烧录方法	L00227819/Y00250993
4	量产烧录方法	L00227819/Y00250993
5	量产问题定位指南	J00174387

修订记录

修订记录累积了每次文档更新的说明。最新版本的文档包含以前所有文档版本的更新内容。

修订日期	版本	修订说明
2013-11-5	00B01	第一次版本发布。
2014-01-02	01	2.2.1 章节分成 Linux SDK 和 Android SDK 两个小节描述。
2014-03-13	02	修改 2.3.3 章节。
2014-05-23	03	删除原来的 2.2 硬件注意事项和 Hipro-usb 小节。
2014-06-18	04	增加支持 Hi379XX 系列芯片。增加 HiProinfo 使用说明
2014-07-15	05	新增 4.1.2 章节。新增第 5 章“量产问题定位指南”。
2014-08-21	06	增加支持 Hi3716MV310 芯片的说明。
2014-11-25	07	增加案例五中的说明内容。
2014-12-30	08	修改 3.2.1 章节注意事项，添加了如何制作 ProgrammerforCA.bin 文件的方法。
2015-01-26	09	修改 4.1.2.1 章节的说明内容。



修订日期	版本	修订说明
2015-03-10	10	新增支持 Hi3110EV500 芯片。
2015-04-24	11	新增支持 Hi3798CV200、Hi3716MV420/410 芯片。
2016-03-10	12	新增支持 Hi3716MV330 芯片。
2016-04-19	13	修改 4.1.2 章节。



目 录

前 言.....	iii
1 概述.....	1-1
1.1 概述.....	1-1
1.2 量产烧录前的准备工作.....	1-2
2 HiPro 工具使用方法.....	2-1
2.1 HiPro 工具介绍	2-1
2.2 HiPro-serial 工具使用方法	2-2
2.2.1 编译 program 文件	2-2
2.2.2 制作 HiPro-serial 镜像 usb_update.bin	2-6
2.2.3 组网环境搭建及物料选择.....	2-8
2.2.4 HiPro-serial 烧录单板	2-9
2.3 HiPro-serial 常见问题	2-14
2.3.1 烧录失败是什么原因?	2-14
2.3.2 com 的编号如何对应?	2-15
3 量产 MAC ID 烧录方法.....	3-1
3.1 HiProinfo 烧录 MAC ID.....	3-1
3.1.1 HiProinfo 工具介绍.....	3-1
3.2 HiProinfo 工具使用方法.....	3-2
3.2.1 HiProinfo 工具配置说明.....	3-2
3.2.2 HiProinfo 工具烧写 MAC ID	3-4
3.3 HiProinfo 常见问题.....	3-7
4 量产烧录方法.....	4-1
4.1 eMMC 的方案	4-1
4.1.1 使用 HiPro-serial 工具烧录 boot 和 Android recovery + recovery 升级	4-1
4.1.2 使用 U 盘烧录 boot 和 Android recovery + recovery 升级	4-2
4.1.3 使用烧录器烧录整个 eMMC 镜像	4-3
4.1.4 使用烧录器烧录 boot 和 Android recovery + recovery 升级	4-11
4.2 SPI Flash+NAND Flash 或 NAND Flash 的方案	4-12
4.2.1 使用 HiPro-serial 工具烧录 boot 和 Android recovery + recovery 升级	4-12



4.2.2 使用烧录器烧录整个 Nand 镜像	4-13
4.2.3 使用烧录器烧录 boot 和 Android recovery + recovery 升级	4-18
5 量产问题定位指南	5-1
5.1 CPU 上电打印说明	5-1
5.2 DDR training 相关打印	5-8
5.2.1 Boot 下 ddr 时序窗口打印	5-8
5.2.2 DDR 报错问题打印	5-9



插图目录

图 2-1 HiPro-serial 工具 PC 端界面	2-1
图 2-2 SDK menuconfig 主界面	2-2
图 2-3 选择 HiLoader Support.....	2-3
图 2-4 选择 BootLoader	2-3
图 2-5 进入 Uboot	2-4
图 2-6 选择 Support Usb Drivers.....	2-4
图 2-7 选择 BootLoader Config	2-5
图 2-8 选择 Hisi File Protocol	2-5
图 2-9 保存配置。	2-6
图 2-10 制作 eMMC 单板的 HiPro-serial 镜像.....	2-7
图 2-11 制作 SPI Flash + NandFlash 或 Nand Flash 单板的 HiPro-serial 镜像	2-8
图 2-12 组网环境搭建示意图.....	2-9
图 2-13 HiPro-serial 工具主界面	2-10
图 2-14 等待输入 MAC ID	2-11
图 2-15 下载 program 文件	2-12
图 2-16 烧写单板	2-13
图 2-17 HiPro-serial 烧录成功	2-14
图 2-18 烧录失败	2-15
图 3-1 HiProinfo 工具 PC 端界面.....	3-2
图 3-2 HiProinfo 工具主界面.....	3-5
图 3-3 HiProinfo 工具选择串口.....	3-5
图 3-4 HiProinfo 工具选择变化规则.....	3-6
图 3-5 HiProinfo 烧录成功.....	3-6
图 4-1 制作 Android 的 HiPro-serial 镜像.....	4-2
图 4-2 HiTool 工具界面	4-4



图 4-3 配置各个分区后	4-4
图 4-4 eMMC 空白区填充配置	4-5
图 4-5 选择 eMMC 器件	4-7
图 4-6 定制烧录算法	4-8
图 4-7 设置扩展 CSD 寄存器	4-9
图 4-8 导入镜像文件	4-10
图 4-9 开始烧录	4-11
图 4-10 制作烧片器镜像。	4-12
图 4-11 制作 Android 的 HiPro-serial 镜像	4-13
图 4-12 HiTool 工具界面	4-14
图 4-13 配置各个分区后	4-15
图 4-14 制作 Nand 烧录器镜像的界面	4-15
图 4-15 设置 Nand 器件属性	4-16
图 4-16 NAND 器件上逻辑分区布局	4-17
图 5-1 写窗口	5-9
图 5-2 读窗口	5-9



表格目录

表 4-1 扩展 CSD 寄存器配置.....	4-6
表 4-2 文件烧写起始地址和烧写长度.....	4-18
表 5-1 正常启动流程	5-1
表 5-2 Hitools 烧写流程.....	5-4



1 概述

1.1 概述

Hi3716CV200 系列平台和 Hi379XX V100、Hi3716M V310、Hi3716MV330 平台支持三种主要的存储器件：SPI Flash、NAND Flash 和 eMMC，通常包含三种存储搭配方案：

- SPI Flash + NAND Flash
- NAND Flash
- eMMC



说明

- Hi3716CV200 系列平台指的是：Hi3716C V200、Hi3719C V100、Hi3718C V100、Hi3719M V100、Hi3718M V100；
- Hi379XX V100 平台指的是：Hi3798C V100、Hi3796C V100、Hi3798M V100。
- Hi3798MV100 不支持 SPI Flash，Hi3716MV310、Hi3716MV330 不支持 eMMC，不支持相关搭配方案。

对于上述三种存储搭配，Hi3716CV200 系列平台和 Hi379XX V100 平台支持以下几种量产烧录方案。



说明

以下以 Hi3716CV200 为例。其他芯片类似。

方案一

使用串口 HiPro-serial 工具，烧录 boot 和 Android recovery + recovery 升级。该方案可以省去烧录器的成本，通过单板自举来完成烧录，成本低，但烧写速度一般，适用于低成本的小批量生产。

烧录步骤如下：

步骤 1 使用 HiPro-serial 工具烧录 boot 和 Android 的 recovery 内核。

步骤 2 重启单板进入 recovery 模式，通过 recovery 程序读取 U 盘的 update 包升级单板。

----结束



方案二

使用烧录器烧录整个单板镜像。该方案需要购买 Nand 或 eMMC 烧录器，成本高，但烧录速度快，效率高，适用于大批量生产。

方案三

使用烧录器烧录 boot 和 Android recovery + recovery 升级。由于 SPI Flash 烧录器成本低，操作相对简单，因此，此方案通常在 SPI Flash + NAND Flash 的存储方案中采用，对于只有 Nand 或 eMMC 的存储方案，如果已经有烧录器，建议采用[方案二](#)。

烧录步骤如下：

步骤 1 使用烧录器烧录 boot 和 Android 的 recovery 内核。

步骤 2 重启单板进入 recovery 模式。

步骤 3 通过 Recovery 程序读取 U 盘的 update 包升级单板。

----结束

方案四

使用 HiPro-serial 烧录整个单板镜像。该方案可以省去烧录器的成本，通过单板自举来完成烧录，成本低，但烧录速度较慢，适用于 SPI Flash + NAND Flash 或 NAND Flash 的单板。

方案五



说明

本方案仅适用于 Hi3798MV100 芯片。

使用 U 盘烧录 boot 和 Android recovery + recovery 升级。

eMMC 使用的 Ext4 文件系统需要全片烧写，烧写效率较低，所以对于 eMMC 方案，建议采用[方案一](#)。

具体使用哪种方案，需要客户根据使用的存储器件、成本、量产规模等因素选取。

1.2 量产烧录前的准备工作

量产烧录前的准备工作如下：

- 准备待烧录的原始文件，包括：
 - boot 镜像
 - 参数区镜像
 - kernel 镜像
 - 文件系统镜像等。
- 准备 HiTool 工具。
- 准备 HiPro-serial 工具。



- 如果采用 HiPro-serial 工具，准备 U 盘、USB 转串口线和 USB HUB。



说明

如果采用烧录器，请联系厂家确认是否与当前使用的芯片平台兼容。

----结束



2 HiPro 工具使用方法

2.1 HiPro 工具介绍

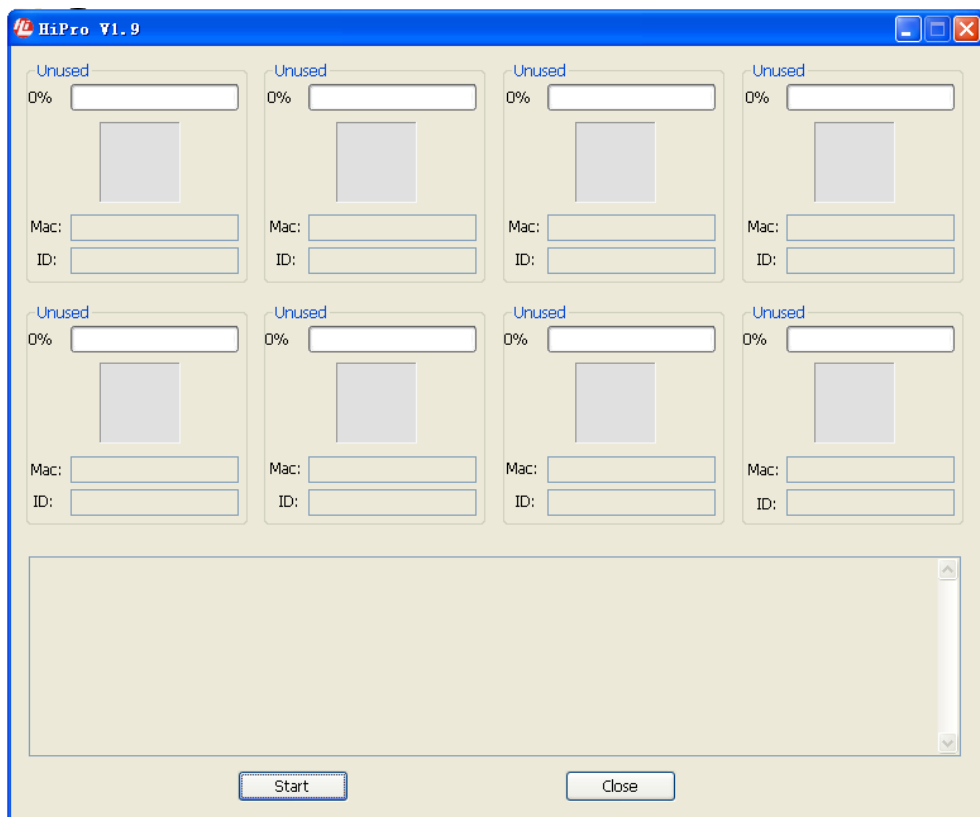


说明

目前，只有 Hi3716MV310、Hi3716MV330、Hi3716CV200 系列平台和 Hi379XX V100 平台支持 HiPro 工具。

HiPro 工具支持通过串口和 U 盘来烧录单板，支持裸片烧写，可以同时烧录 8 个单板。

图2-1 HiPro-serial 工具 PC 端界面





HiPro-serial 工具需要一个运行于单板的 program 文件来完成烧录，通过串口下载 program，然后由 program 读取放在 U 盘的 HiPro-serial 镜像来完成单板烧录。

2.2 HiPro-serial 工具使用方法

2.2.1 编译 program 文件

Program 文件需要在 SDK 环境编译，首先在 Linux 服务器上解压 SDK 软件包，进入 SDK 根目录，选择待烧录单板的配置。

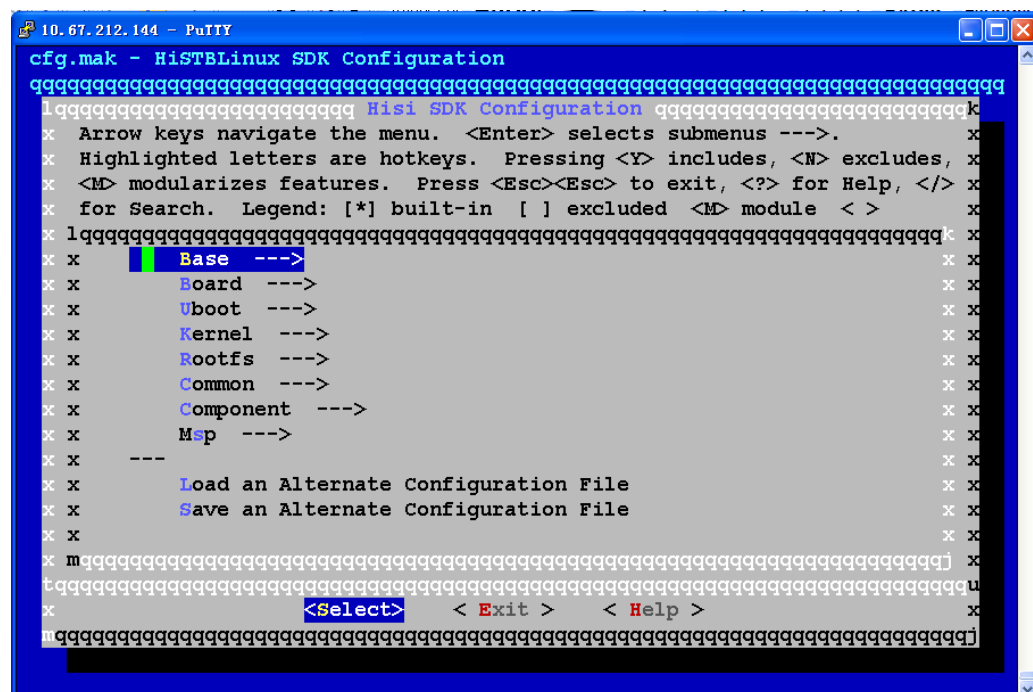
Linux SDK 和 Android SDK 的编译方法不同，以下分别说明。

2.2.1.1 Linux SDK

以 Hi3716CV200demo 板为例，对应单板配置为 hi3716cdmo2b_hi3716cv200_cfg.mak，执行 cp configs/hi3716cdmo2b_hi3716cv200_cfg.mak cfg.mak，可以将当前配置改为 Hi3716CV200demo 板的配置，建议在 cp 前先备份好原来才 cfg.mak 文件，以便生成 program 文件后，可以恢复回原来的配置。

步骤 1 执行 make menuconfig，进入 SDK menuconfig 主界面。

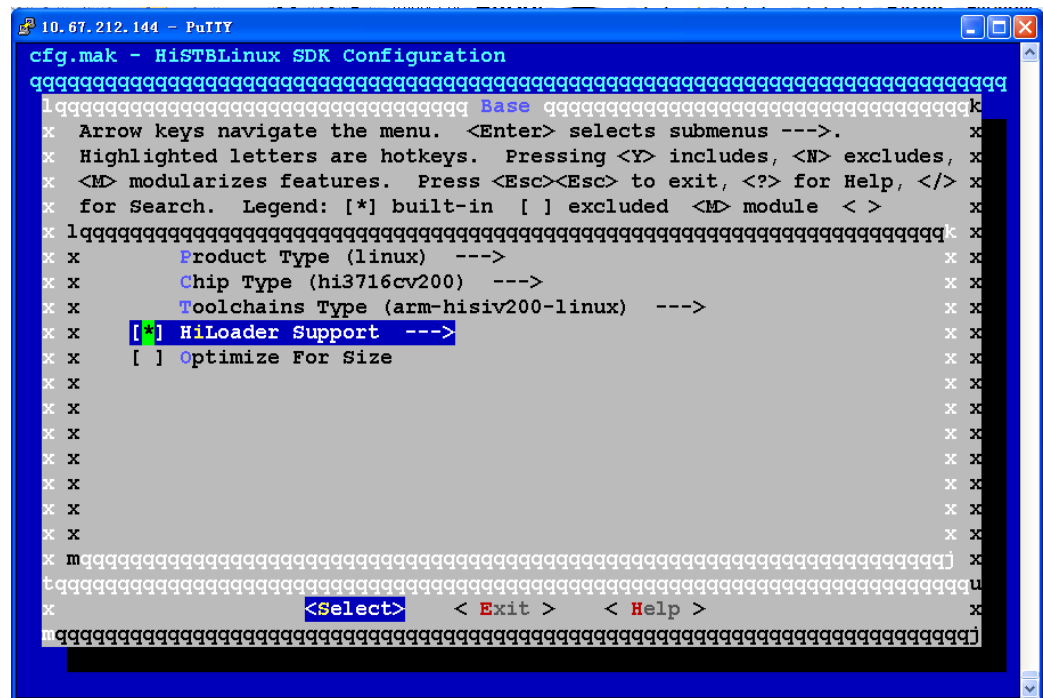
图2-2 SDK menuconfig 主界面



步骤 2 进入 Base，打开 HiLoader Support。

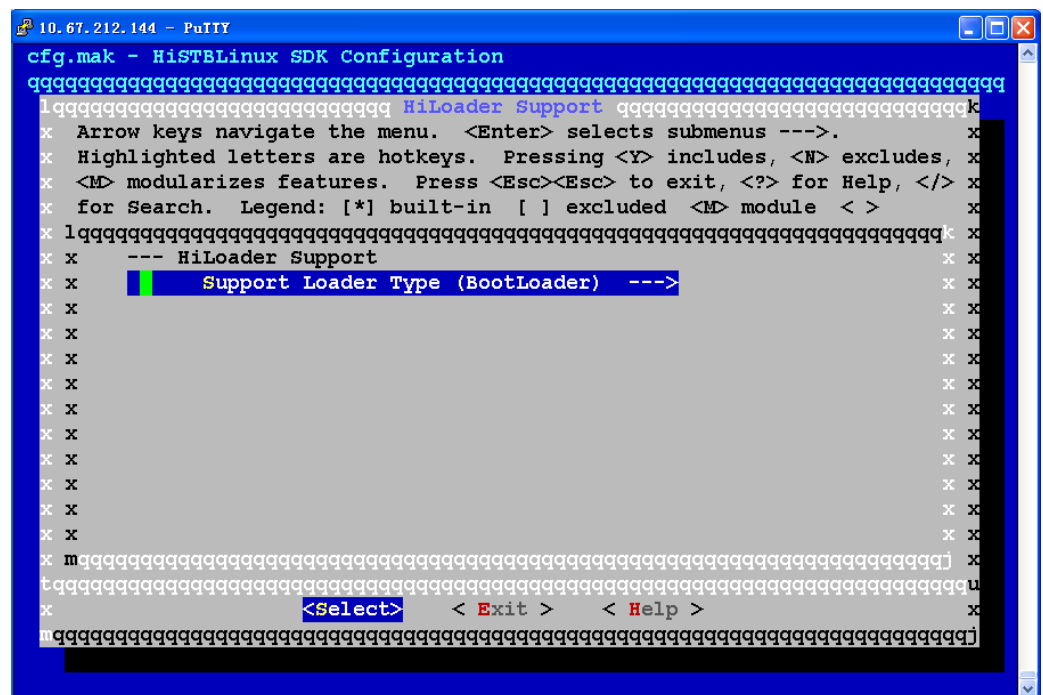


图2-3 选择 HiLoader Support



步骤 3 进入 HiLoader Support，Support Loader Type 选择 BootLoader。

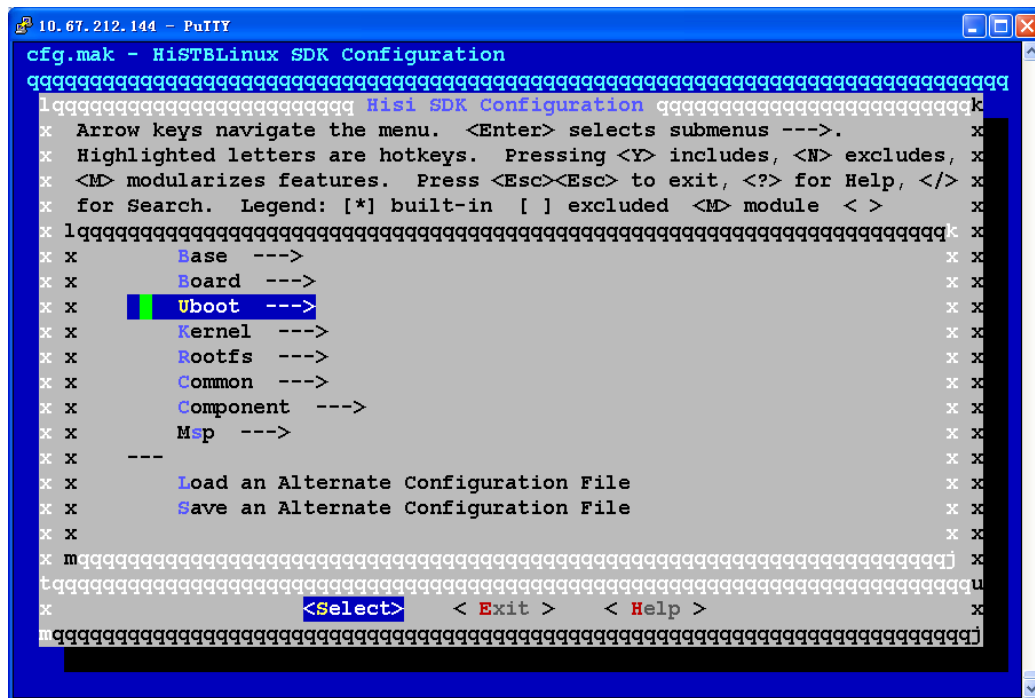
图2-4 选择 BootLoader



步骤 4 回到主界面，进入 Uboot。

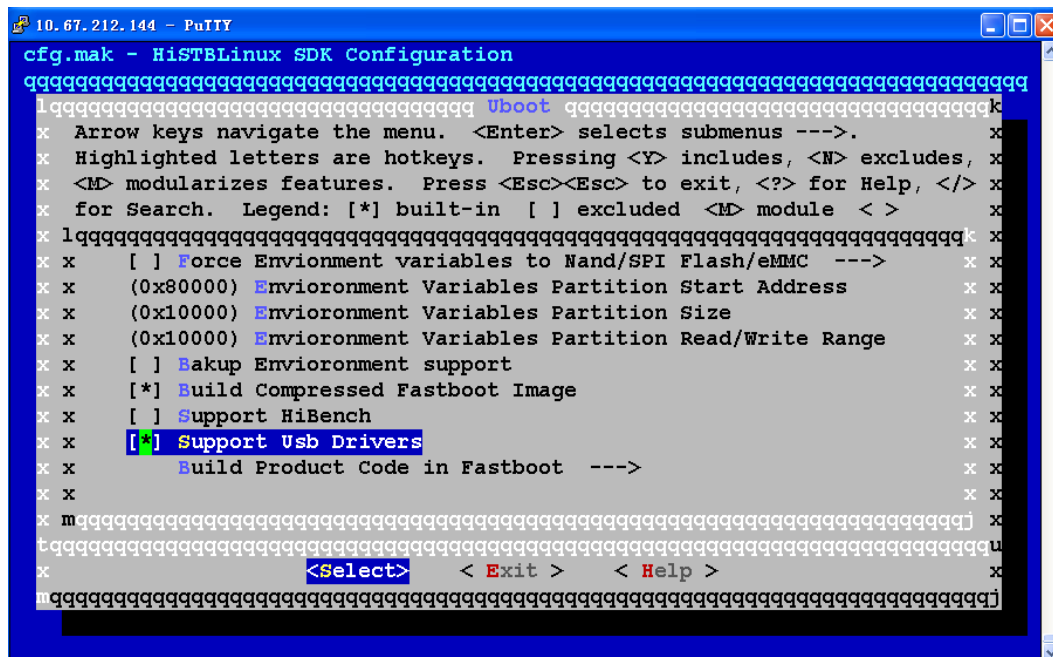


图2-5 进入 Uboot



步骤 5 选择 Support Usb Driver, 如果单板 boot 较大, 可以选择 Build Compressed Fastboot Image, 这样生成的 program 文件较小, 可以节省下载时间。

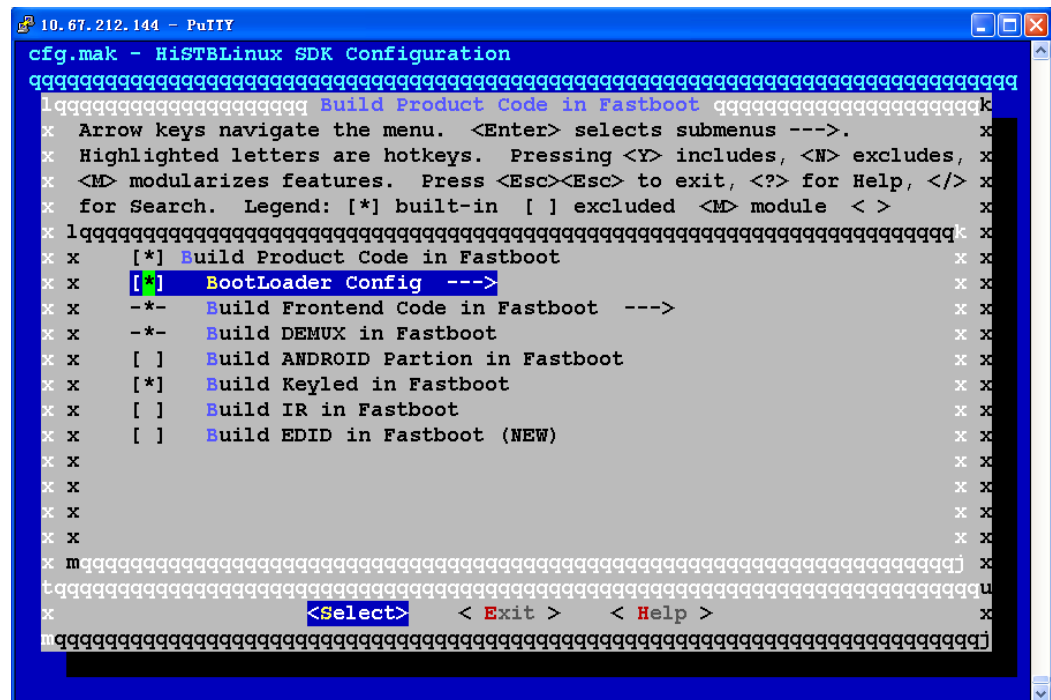
图2-6 选择 Support Usb Drivers



步骤 6 进入 Build Product Code in Fastboot, 选择 BootLoader Config。

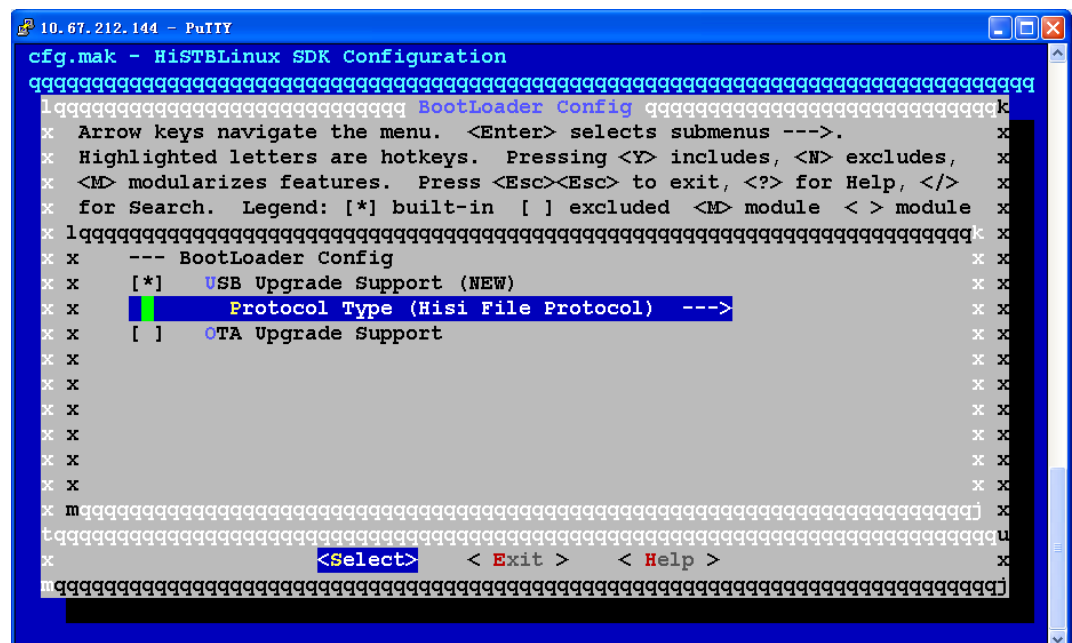


图2-7 选择 BootLoader Config



步骤 7 进入 BootLoader Config，选择 Protocol Type -> Hisi File Protocol。

图2-8 选择 Hisi File Protocol



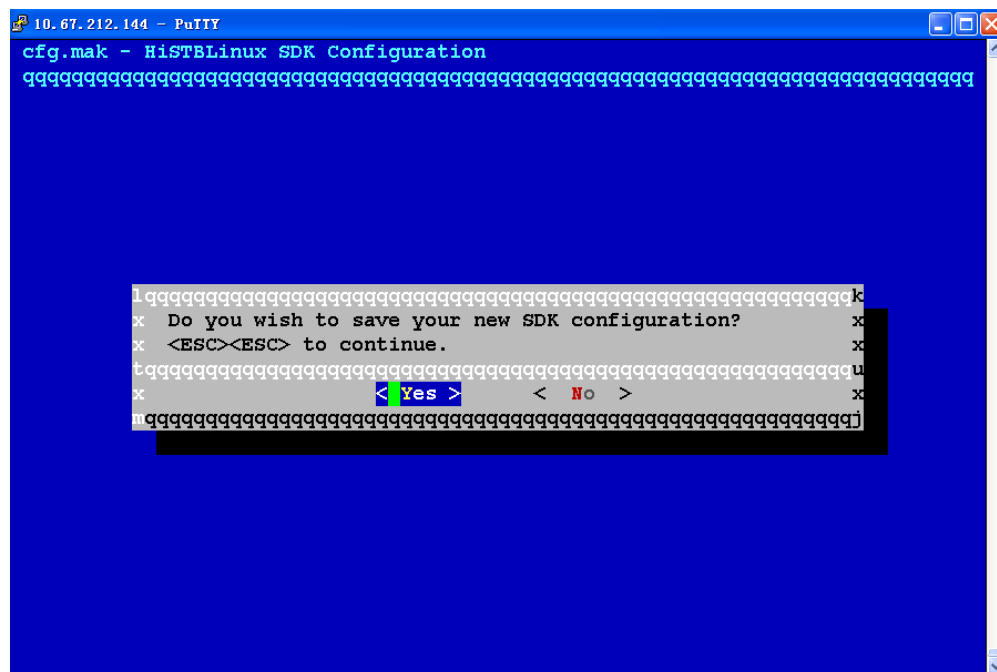


注意

不能选中 OTA Upgrade Support, 如果所选的 SDK 配置文件已经默认开启此项, 需要关闭。

步骤 8 退出 menuconfig, 并保存配置。

图2-9 保存配置。



步骤 9 执行 make hiboot_clean;make hiboot_install, 编译 program, 成功后, 在 SDK 的 tools/windows/HiPro-serial 目录下生成对应的 program.bin。

----结束

2.2.1.2 Android SDK

在 Android 根目录下执行 make hipro, 编译完成后, 会在 out 目录下生成 program.bin 和 HiPro-serial 工具, 以 Hi3716CV200 为例, 在 out/target/product/Hi3716CV200/Emmc/和 out/target/product/Hi3716CV200/Nand 下生成。

2.2.2 制作 HiPro-serial 镜像 usb_update.bin

HiPro-serial 镜像就是按一定格式打包的单板烧录镜像, 其中包含了待烧录的 boot、内核和文件系统的等。

HiPro-serial 镜像需要使用 HiTool 工具制作, 不同的单板制作过程有差别, 分两种单板:



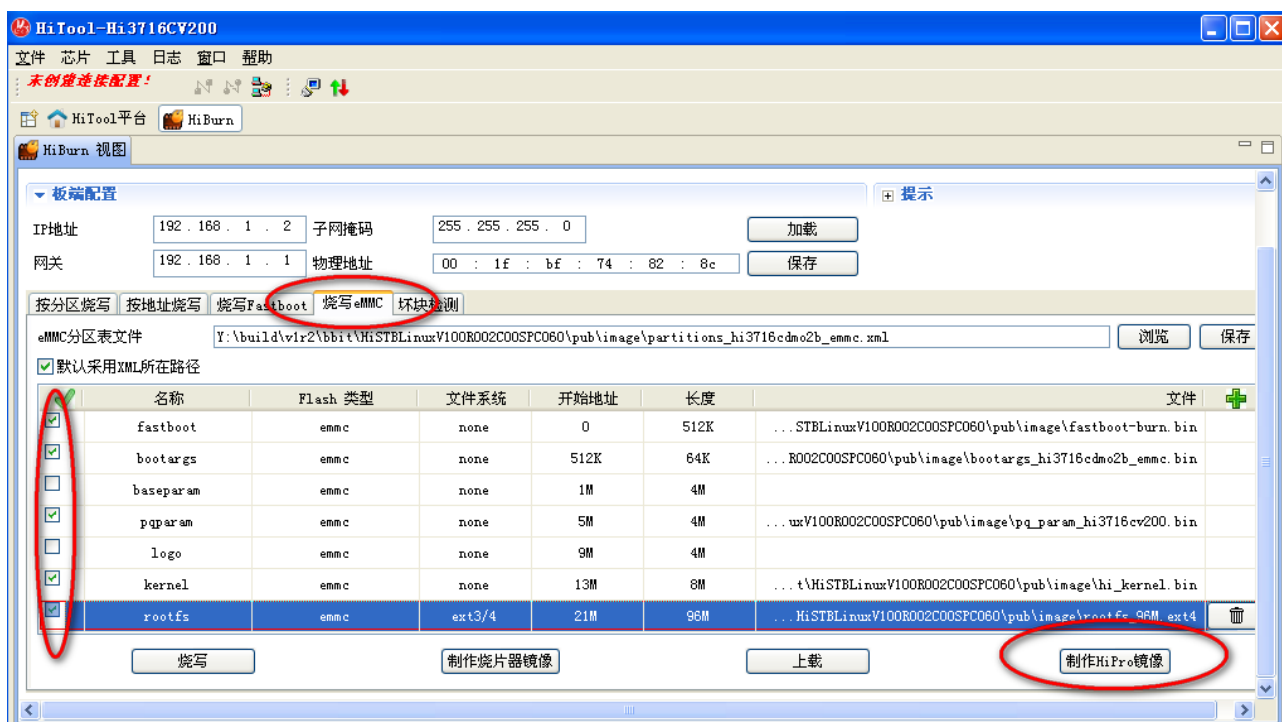
- eMMC 单板
- SPI Flash+NAND Flash（或 NAND Flash）单板。

2.2.2.1 eMMC 单板

eMMC 单板 HiPro-serial 镜像制作过程如下：

- 步骤 1 启动 HiTool，进入 HiBurn 视图。
- 步骤 2 选择烧写 eMMC 栏，配置需要烧录的分区，也可以导入 xml 格式的分区表。
- 步骤 3 点击制作 HiPro-serial 镜像，如图 2-10 所示。

图2-10 制作 eMMC 单板的 HiPro-serial 镜像



- 步骤 4 在弹出窗口中，保存 HiPro-serial 镜像，并命名为 usb_update.bin。



说明

HiPro-serial 镜像必须命名为 usb_update.bin。

----结束

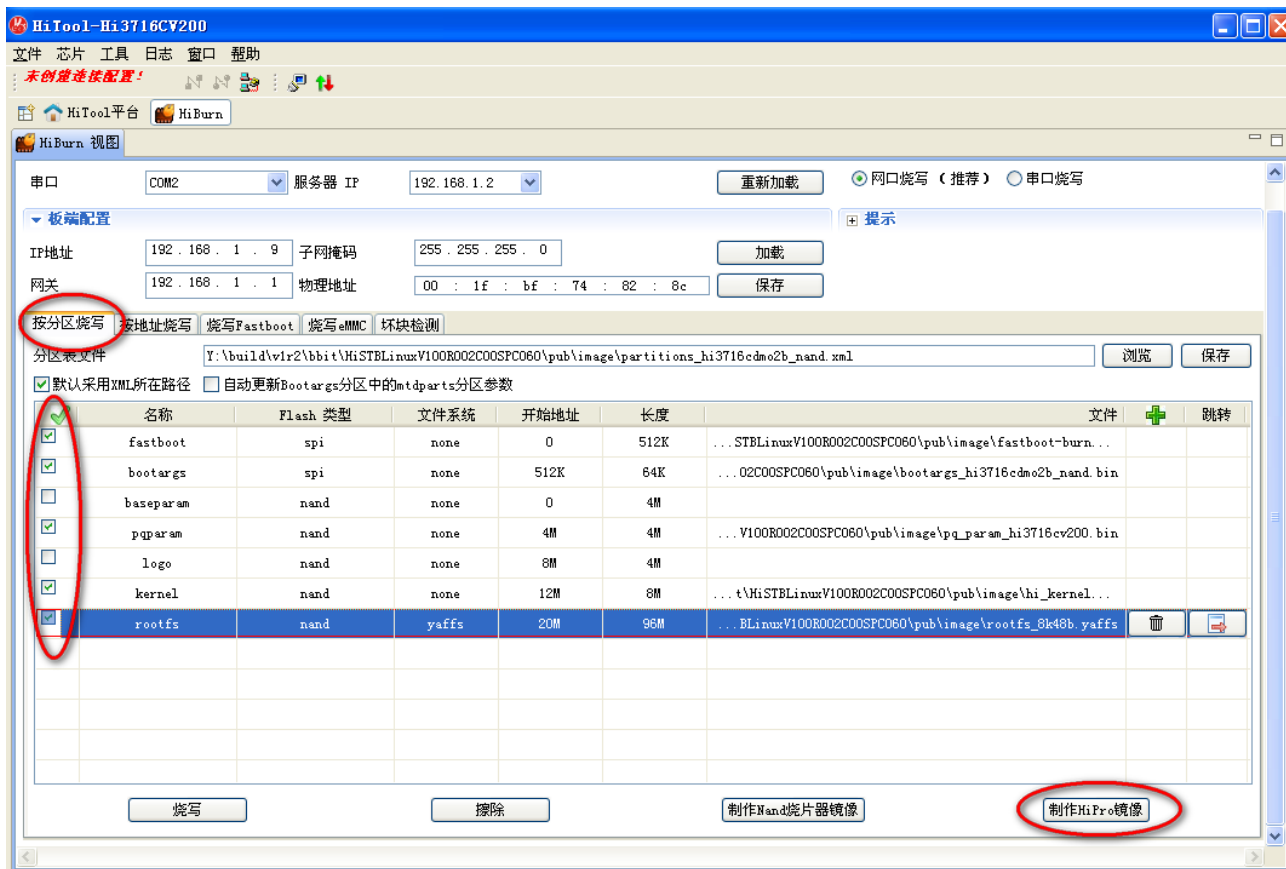
2.2.2.2 SPI Flash +NAND Flash 或 NAND Flash 的单板

SPI Flash +NAND Flash 或 NAND Flash 单板 HiPro-serial 镜像制作过程如下：

- 步骤 1 启动 HiTool，进入 HiBurn 视图。
- 步骤 2 选择按分区烧写栏，配置需要烧录的分区，也可以导入 xml 格式的分区表。
- 步骤 3 点击制作 HiPro-serial 镜像，如图 2-11 所示。



图2-11 制作 SPI Flash + NandFlash 或 Nand Flash 单板的 HiPro-serial 镜像



步骤 4 在弹出窗口中，保存 HiPro-serial 镜像，并命名为 usb_update.bin。



注意

HiPro-serial 镜像必须命名为 usb_update.bin。

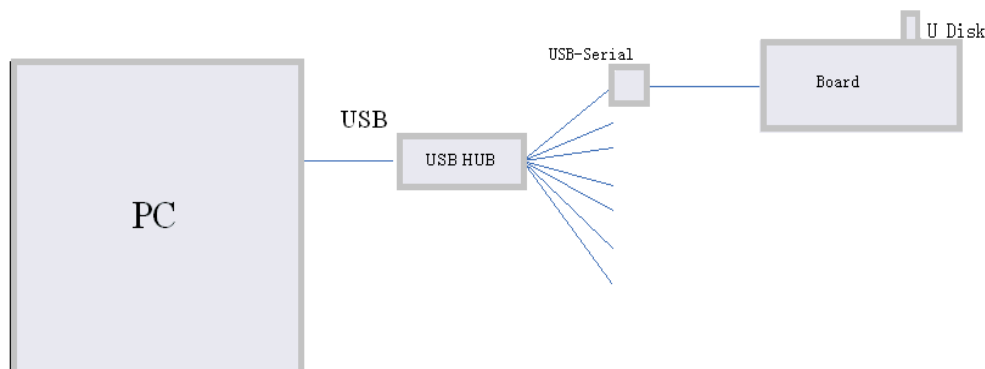
-----结束

2.2.3 组网环境搭建及物料选择

PC 通过 usb hub 接多条 usb 转串口线，然后接待量产的盒子的串口，达到一台电脑接多台设备的目的。通过串口下载 program 文件，再通过 program 把镜像从 U 盘烧写到单板的 flash 上，组网环境搭建如图 2-12。如果是 dongle 产品，使用 mini usb 同时提供电源和串口连接，可通过制作简单的线材或转接板可以达到复用的目的。



图2-12 组网环境搭建示意图



环境的搭建强烈推荐使用如下测试过比较稳定的物料：

- U 盘 指定型号：Kingston DataTraveler 4G USB 2.0，如果用 4G 以上的盘，需格式化成 4G 分区来使用。U 盘只放升级包文件。
<http://item.taobao.com/item.htm?spm=a1z10.3.0.72.uPOvap&id=9009181350&>
- USB HUB 使用带电源的 HUB，推荐型号：SSK，[魔王 SSK 积木 UBS 分线器](#)
[USB 集线器 USB HUB 带电源 SHU01](#)
<http://detail.tmall.com/item.htm?spm=a230r.1.10.88&id=14491968034>
- USB 转串口线（USB） 推荐型号：力特（Z-TEK）

2.2.4 HiPro-serial 烧录单板

烧录单板的步骤如下：

- 步骤 1 在 Windows 平台的 PC 机上，将 U 盘所有分区删除，只保留一个分区，并格式化为 FAT32 文件系统，然后拷贝 HiPro-serial 镜像（usb_update.bin）拷贝到 U 盘根目录，将 program.bin 文件拷贝到 HiPro-serial 工具所在目录。



注意

U 盘只能有一个分区，且必须格式化成 FAT32 文件系统。

- 步骤 2 将 PC 机的串口与单板串口连接，并将 U 盘插到单板的 USB 端口。



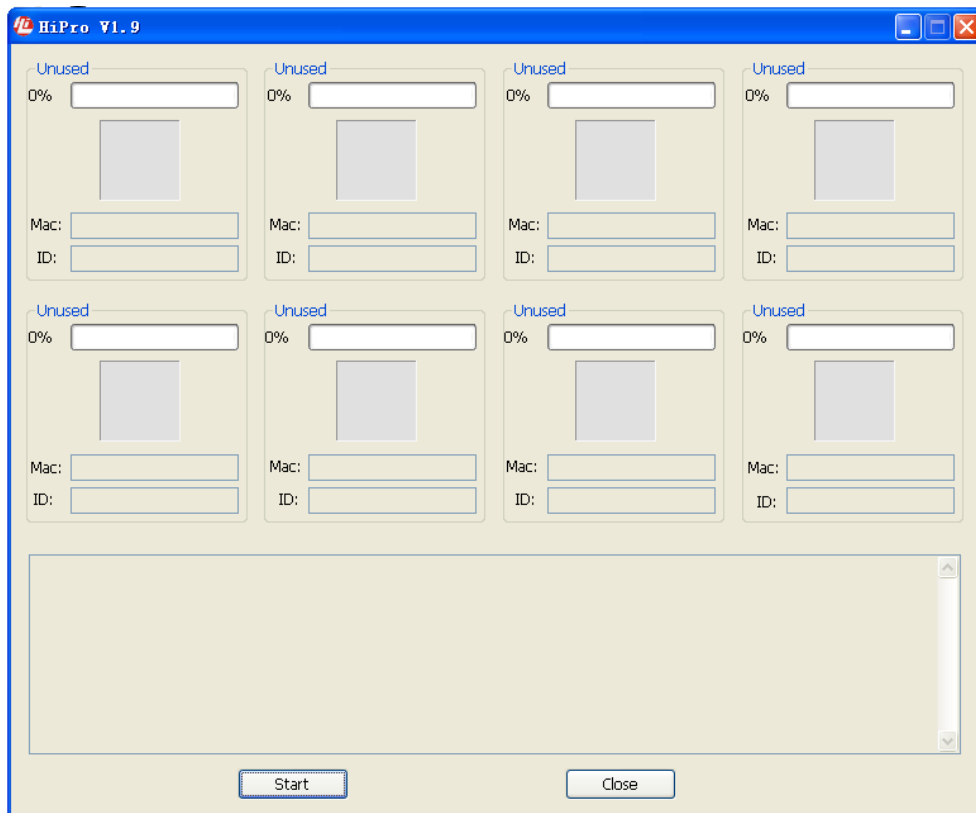
注意

每个单板需要一个 U 盘和一根串口线，PC 机可以通过 USB Hub+USB 转串口线来外扩多个串口。



步骤 3 在 PC 机上运行 SDK 的 tools\windows\HiPro-serial\HiPro-serial_XX 目录下的 HiPro-serial.exe 程序。

图2-13 HiPro-serial 工具主界面



步骤 4 点击 Start 按钮，然后单板上电。HiPro-serial 工具会自动检测到上电的单板，如果用户在配置文件内配置输入 MAC 和 ID，则会等待用户输入 MAC 和 ID，如图 2-15 所示，如没有配置则跳过等待输入过程，直接开始下载 program 文件，如图 2-16 所示。



图2-14 等待输入 MAC ID

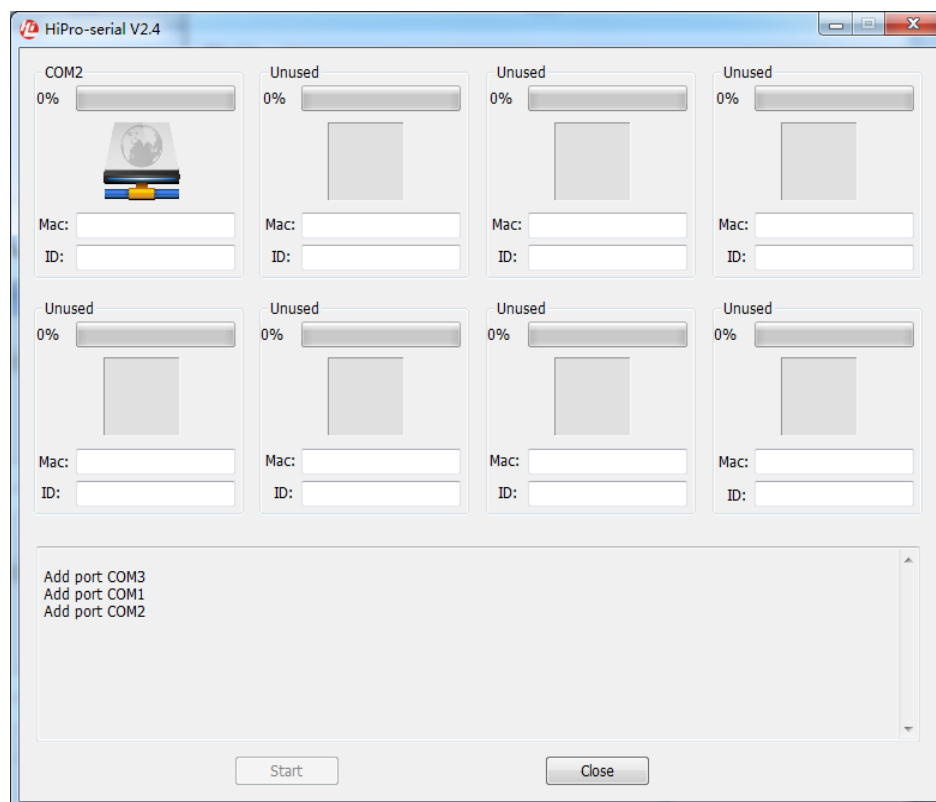
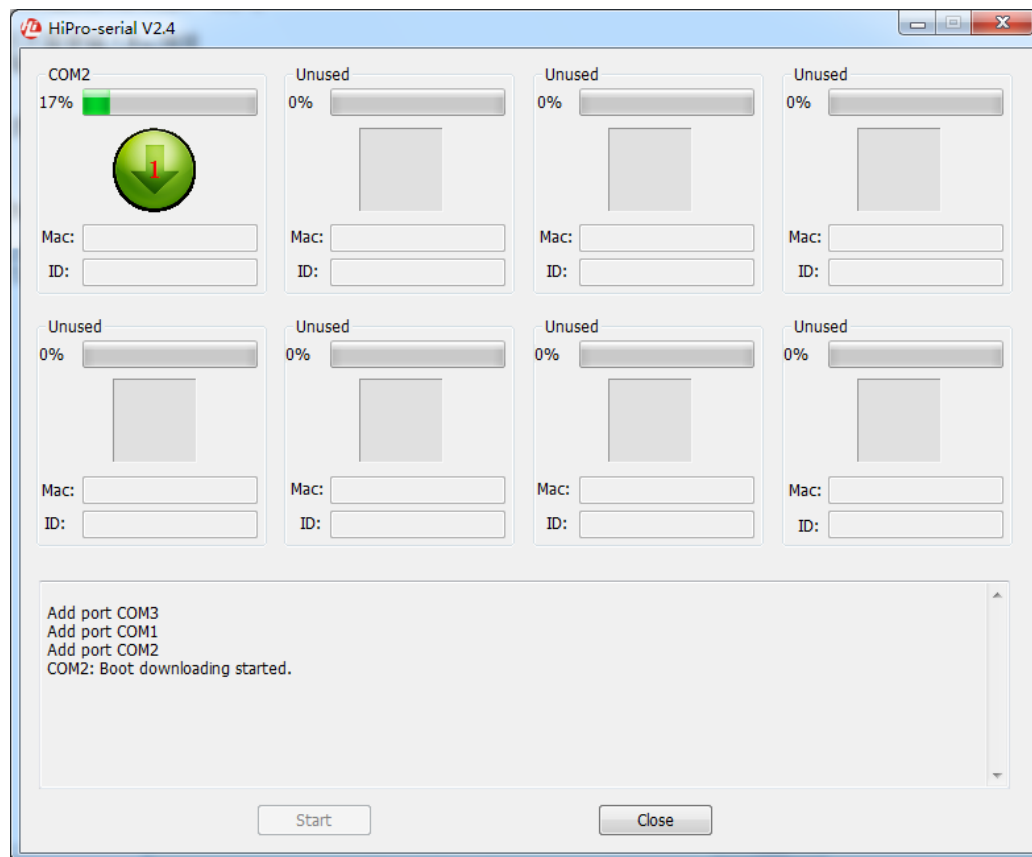




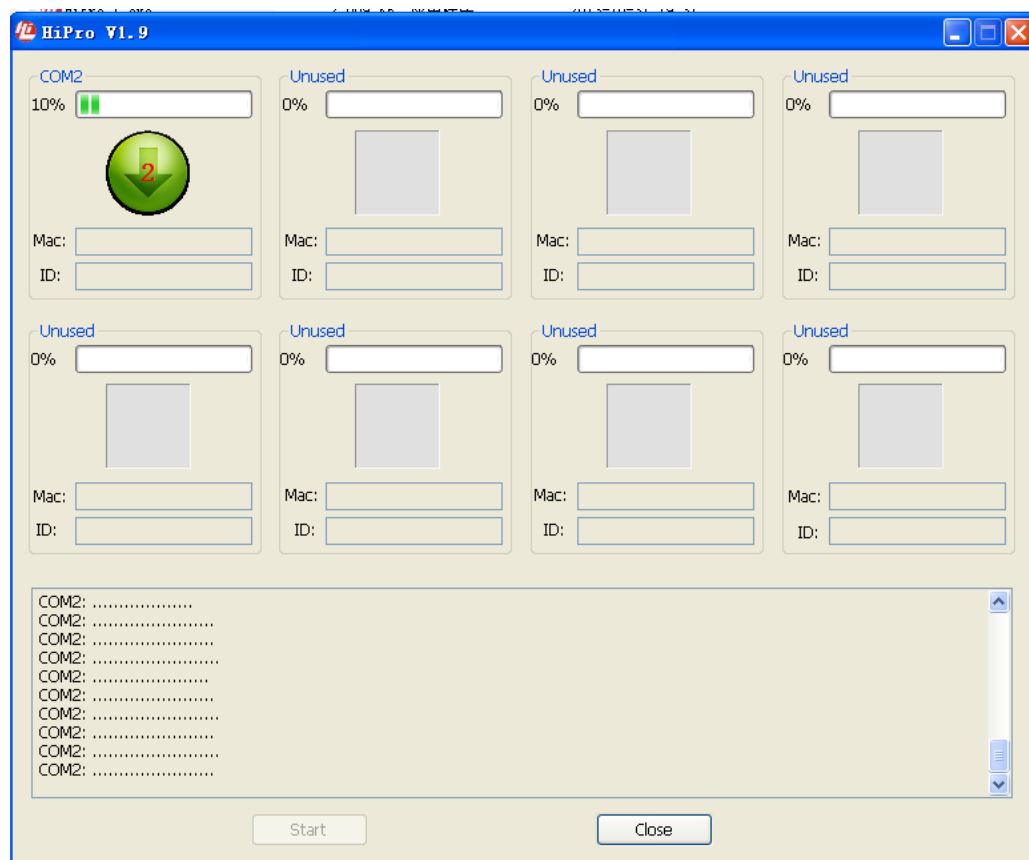
图2-15 下载 program 文件



下载完 program 文件后，HiPro-serial 会扫描 U 盘，并读取 usb_update.bin，烧写单板，如图 2-16 所示。



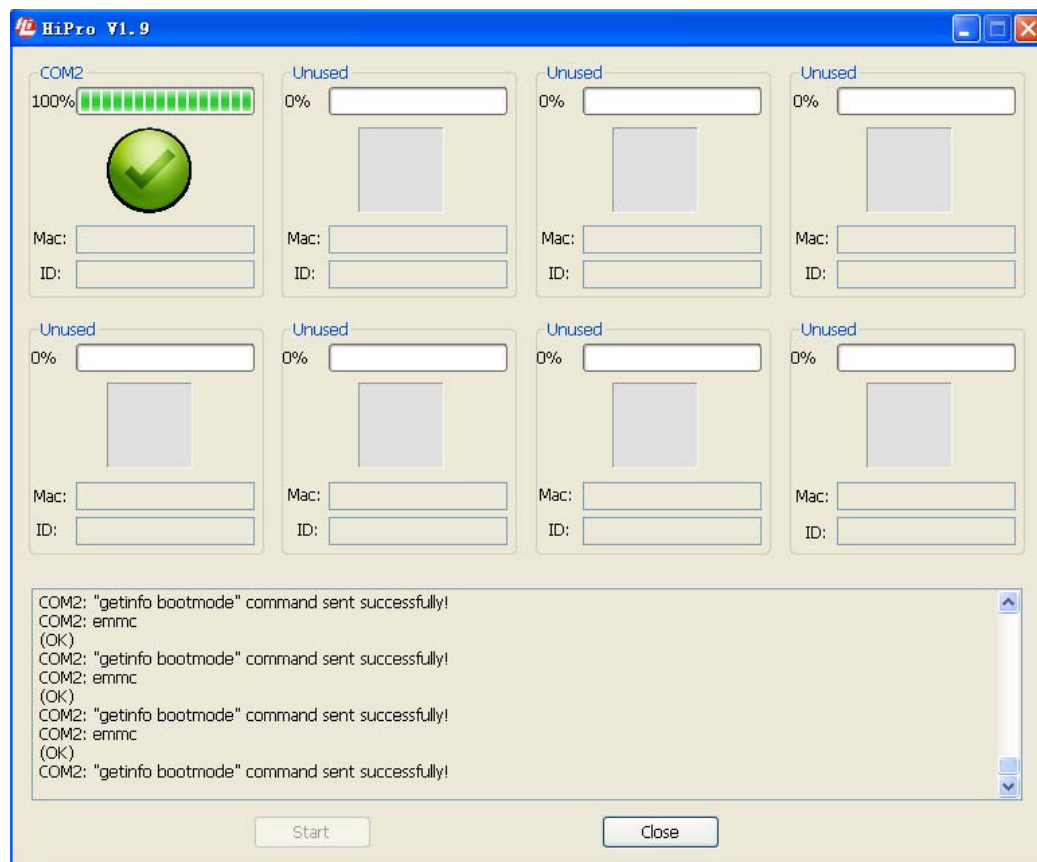
图2-16 烧写单板



烧录完成后如图 2-17 所示。



图2-17 HiPro-serial 烧录成功



----结束

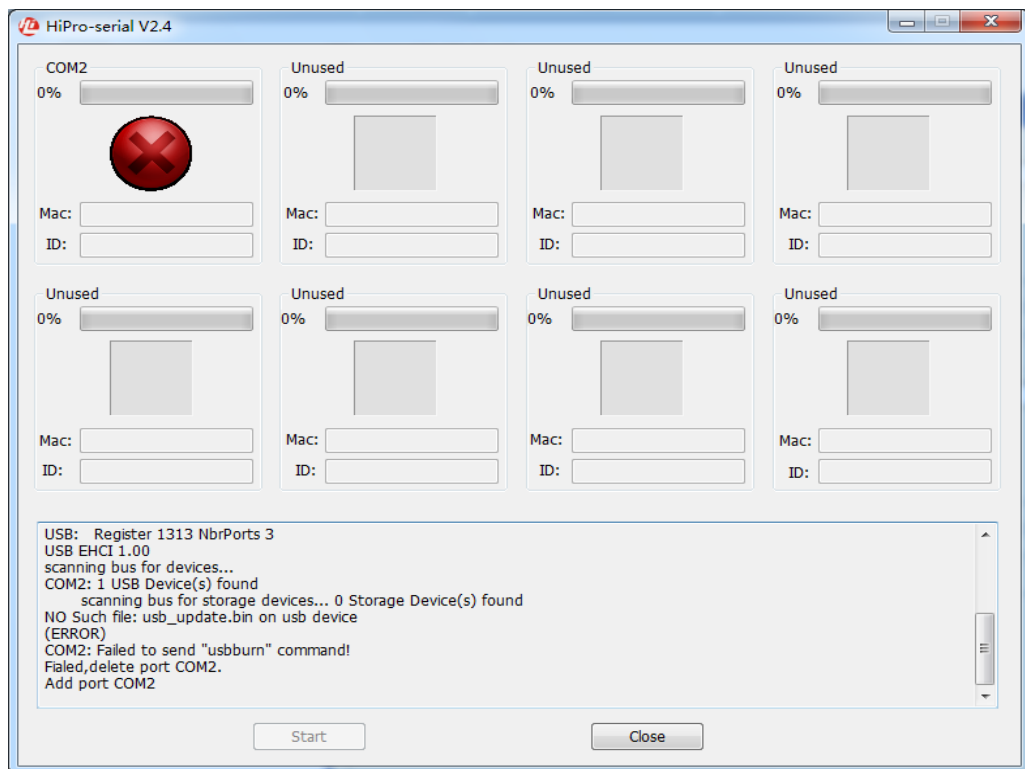
2.3 HiPro-serial 常见问题

2.3.1 烧录失败是什么原因？

如果烧录失败，HiPro-serial 工具提示如[图 2-18](#)所示。



图2-18 烧录失败



烧录失败，通常有以下几种原因：

- U 盘不是 FAT32 文件系统。
- U 盘没有 usb_update.bin。
- U 盘有多个分区。
- 串口线接触不好。

2.3.2 com 的编号如何对应？

工具启动后，后台自动识别并占用所有未被占用的串口，无需用户指定。与单板建立连接的串口号，从左至右，从上至下，依次排列在界面上，最多建立 8 个连接。

操作员将被烧写的单板与 pc 相连，工具自动识别并显示该连接使用的串口号后，建议在串口线上进行标识，这样量产时我们可以通过工具显示上的串口号与之前在串口线上的标识一一对应，从而知道每块单板的状态。



3 量产 MAC ID 烧录方法

3.1 HiProinfo 烧录 MAC ID

3.1.1 HiProinfo 工具介绍



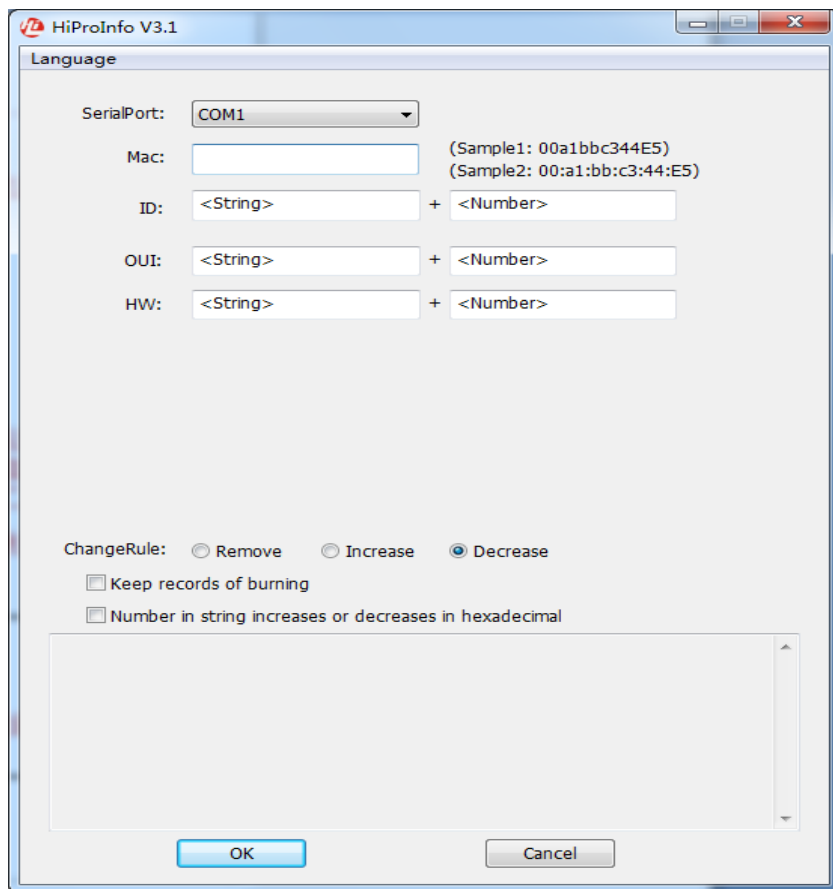
说明

目前，只有 Hi3716MV310、Hi3716MV330、Hi3716CV200 系列平台和 Hi379XX V100 平台支持 HiProinfo 工具。

HiProinfo 工具是一个量产 MAC、ID 烧录工具，支持通过串口烧录 MAC、ID。



图3-1 HiProinfo 工具 PC 端界面



HiProinfo 工具要求单板为非裸板，工具通过串口下载分组数据，然后等待 fastboot 运行，通过与 fastboot 交互完成单板 MAC、ID 烧录。

3.2 HiProinfo 工具使用方法

3.2.1 HiProinfo 工具配置说明

HiProinfo 工具会因为用户配置的不同打开或关闭部分功能。在工具所在目录下存在一个名为 BurnConfig.ini 的配置文件，用户在打开工具之前必须按照自己的需求修改配置文件的各个配置选项,如想关闭某项配置只需在对应配置前使用“;”将其注释即可。

- IsCA: 是否高安单板。普通单板需配置 IsCA=0，高安即 CA 单板需配置 IsCA=1



注意

目前只有部分芯片平台支持 CA，具体请咨询 FAE。烧录 CA 单板时工具目录下必须存在 ProgrammerforCA.bin 文件。ProgrammerforCA.bin 就是安全版本生成的 fastboot.bin，制作方法参考《海思智能机顶盒 2 级安全方案使用指南》第 2 章节，以 Hi3798MV100 单板为例，生成目录为 out/target/product/Hi3798MV100/Security_L2/MAINTAIN/fastboot.bin，将其改名为 ProgrammerforCA.bin 放到工具目录下即可。

- **BoardType:** 单板类型
当前支持的 BoardType 类型如下（后续如果有更新，请咨询 FAE）：
 - Hi3716CV100 单板配置 BoardType=0;
 - Hi3716MV300 单板配置 BoardType=1;
 - Hi3716CV200ES 单板需配置 BoardType=2;
 - Hi3712V100 单板配置 BoardType=3;
 - Hi3716CV200 /Hi3719CV100 /Hi3718CV100 /Hi3719MV100 /Hi3798CV100/Hi3798MV100 /Hi3796CV100 单板需配置 BoardType=4。
- **MacBurnFlashType:** Flash 类型
 - Nand 需配置 MacBurnFlashType=1;
 - Spi 需配置 MacBurnFlashType=2;
 - eMMC 需配置 MacBurnFlashType=3。
- **MacBurnAddress:** 烧写地址
 - Nand 和 Spi 默认配置 MacBurnAddress=0x3f0000;
 - eMMC 默认配置 MacBurnAddress=0xc00000;
 - 高安即 CA 默认配置 MacBurnAddress=0x80000。



注意

MacBurnAddress 配置错误会导致烧写失败或者烧错位置；地址配置必须块对齐，否则会导致烧写失败。

- **CfgName:** 配置自定义烧写信息。
例如配置 Loader 升级里面 OUI、HW 相关序列号等烧写，每一条记录之间用半角竖线“|”隔开，每一条记录可以支持两种格式：
 - 只有自定义信息；此时，工具会在界面上给出自定义配置框，且 ID 与 MAC 一样支持递增与递减。最多支持 3 条这样的记录。
 - 自定义输入固定字符串：以<自定义名称：字符串>的形式配置，例如 CfgName=OUI:123456789|HW，工具会直接烧写配置的字符串 123456789 不再在界面上提供输入框。
- **ExtraInfo:** 配置额外烧写信息。



例如可以进行共享分区文件烧写，或者其他需要版本信息等烧写，每一条记录之间用半角竖线“|”隔开，每一条记录可以支持三种格式：

- 只配置偏移地址：此时，工具会在界面上给出前缀字符与 ID 的配置框，且 ID 与 MAC 一样支持递增与递减。最多支持 4 条这样的记录。
- 偏移地址与烧写字符串：以<地址:字符串的形式>给出。工具会直接将字符串写入对应的偏移地址，例如可以直接写入软件的版本号，例如配置 ExtraInfo=0x1000=versionA|0x2000|0x3000。支持的条目数没有限制。
- 将文件烧写到偏移地址需以下格式配置文件路径 ExtraInfo=0x1000:<file:绝对路径>，例如 ExtraInfo=0x1000:<file:D:\LoaderDB.bin>，文件不宜过大



注意

ExtraInfo 中的地址配置也必须块对齐，否则会导致烧写失败。

- AppLanguage: 默认语言配置
 - “1” 表示中文；
 - “2” 表示英文。

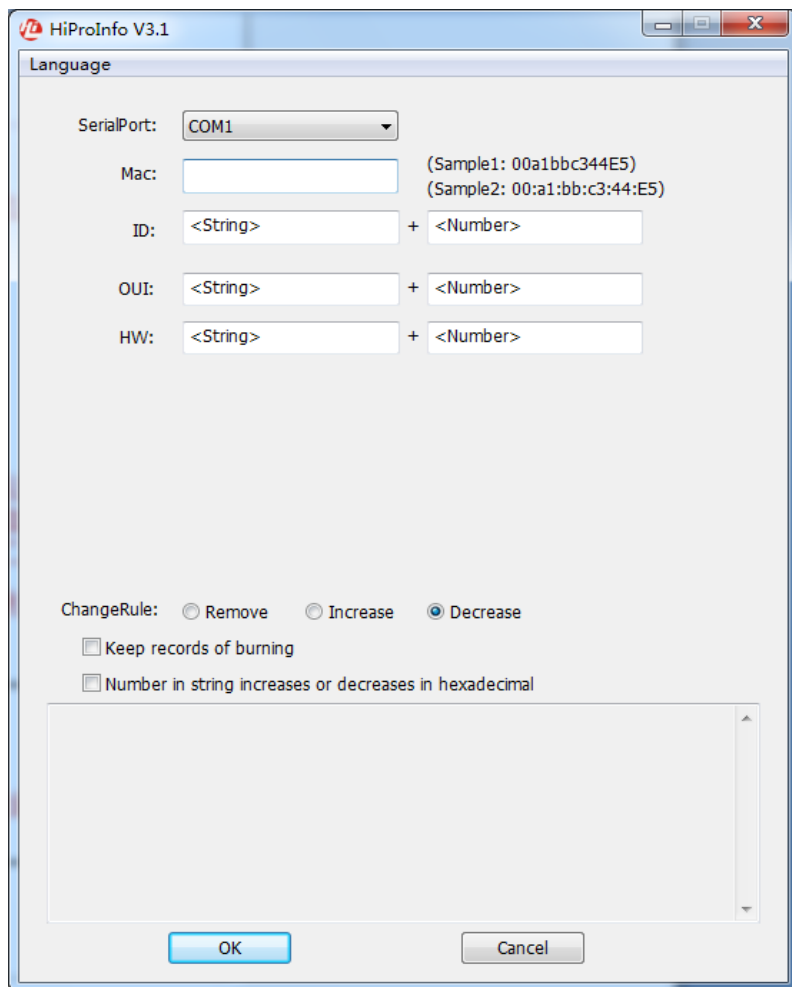
3.2.2 HiProinfo 工具烧写 MAC ID

烧录 MAC、ID 的步骤如下：

- 步骤 1 用户按照需求修改工具所在目录中的 BurnConfig.ini 配置文件，配置文件必须和工具放在同一目录；
- 步骤 2 将 PC 机的串口与单板串口连接。
- 步骤 3 在 PC 机上运行 SDK 的 tools\windows\HiProinfo\HiProinfo_XX 目录下的 HiProinfo.exe 程序。

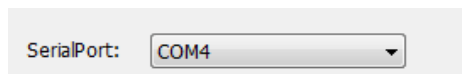


图3-2 HiProinfo 工具主界面



步骤 4 选择串口，选择已连接到单板的串口，确保该串口没有被其他工具占用，否则会报错。

图3-3 HiProinfo 工具选择串口

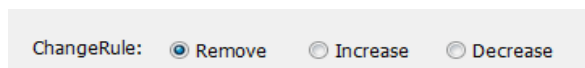


步骤 5 输入要烧写的 Mac 地址和 ID 以及其他信息：第一次输入之后 MAC 会自动保存此配置，之后均会自动加载，输入格式见工具界面的例子。

步骤 6 选择变化规则，烧写成功时会遵照此规则对上面填写的 Mac 地址及序列号进行自动更新，以免去手动修改的麻烦。此选项采用单选框形式。



图3-4 HiProinfo 工具选择变化规则



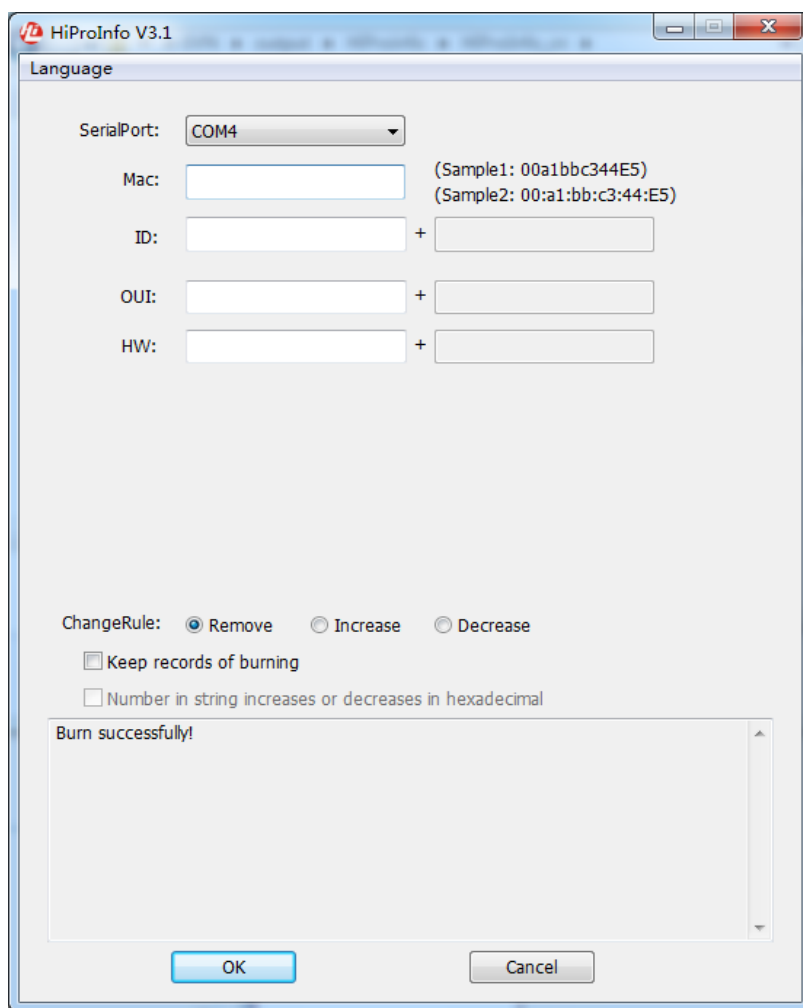
步骤 7 选择是否保存烧写记录，点击确定按钮，单板上电开始烧写。



注意

HiProinfo 工具支持按回车键切换输入框，在最后一个输入框内按回车将直接开始烧写，因此在使用回车快速输入的方式时请确认界面上其他配置（如串口，变化规则等）是否已经正确选择。串口、变化规则和保存烧写记录这 3 项会在退出时被保存，下次使用工具时将沿用上一次的选择。

图3-5 HiProinfo 烧录成功





3.3 HiProinfo 常见问题

烧录失败，通常有以下几种原因：

- 串口线接触不好。
- 串口被占用。
- 单板类型配置错误。
- Flash 类型配置错误。
- 单板为裸板。



4 量产烧录方法

由于使用 SPI Flash 方案通常需要搭配 NAND Flash，因此以下按两种存储方案来说明如何完成量产烧录：

- eMMC
- SPI Flash+NAND Flash 或 NAND Flash

4.1 eMMC 的方案



说明

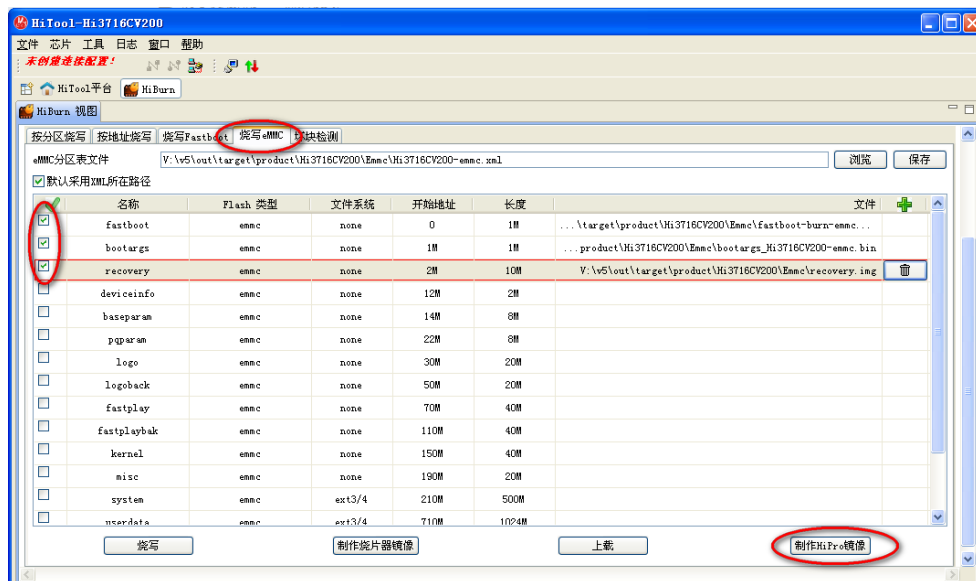
目前，部分芯片平台不支持 eMMC。下述方案只针对支持 eMMC 的芯片平台。

4.1.1 使用 HiPro-serial 工具烧录 boot 和 Android recovery + recovery 升级

步骤 1 制作 Android recovery 的 HiPro-serial 镜像，只勾选 fastboot、bootargs 和 recovery 三个分区，然后点击制作 HiPro-serial 镜像按钮，如图 4-1 所示。



图4-1 制作 Android 的 HiPro-serial 镜像



步骤 2 拷贝 usb_update.bin 到 U 盘根目录。

步骤 3 拷贝 Android 的 update.zip 包，放到 U 盘根目录。



说明

Update.zip 包是 Android 平台的 recovery 升级包，需要在 Android 平台下编译生成，编译方法请遵循 Android 的编译说明。

步骤 4 使用 HiPro-serial 工具烧录 boot 和 Android 的 recovery 内核。

步骤 5 重启单板，通过按键进入 recovery 模式。recovery 程序会读取 U 盘的 update.zip 包升级单板。

----结束

4.1.2 使用 U 盘烧录 boot 和 Android recovery + recovery 升级



说明

本小节适用于 Hi3798MV100、Hi3798CV200 芯片，以 Hi3798CV200 为例。

4.1.2.1 准备工作

- 准备 U 盘，格式化为 FAT32 文件系统。



说明

不支持 USB 读卡器、SD 卡转 USB、USB 硬盘、扩展分区 U 盘。

在 U 盘中有多分区情况下，请保证只有一个分区有 update.zip 升级包。

U 盘型号可使用金士顿品牌，推荐使用如下测试过比较稳定的型号：Kingston DataTraveler 4G USB 2.0。

- 将编译好的 fastboot.bin, bootargs.bin, recovery.img, update.zip 四个镜像文件拷贝至 U 盘根目录

镜像文件位置：



- eMMC 方案: out/target/product/Hi3798CV200/Emmc

4.1.2.2 操作流程

步骤 1 将存有镜像文件的 U 盘插入单板 **USB2.0** 口。

步骤 2 按下 USB 升级按键后，再给单板上电。

如果是空板，可以不用按下按键。

步骤 3 约 2~5 秒后指示灯闪烁，进入 USB 升级流程。

步骤 4 约几分钟后，指示灯常亮，升级完成。

----结束

4.1.3 使用烧录器烧录整个 eMMC 镜像

使用烧录器烧录，主要包括三部分：

- 生成 eMMC 镜像文件
- 配置 eMMC 的扩展 CSD 寄存器
- 操作烧录器完成烧录

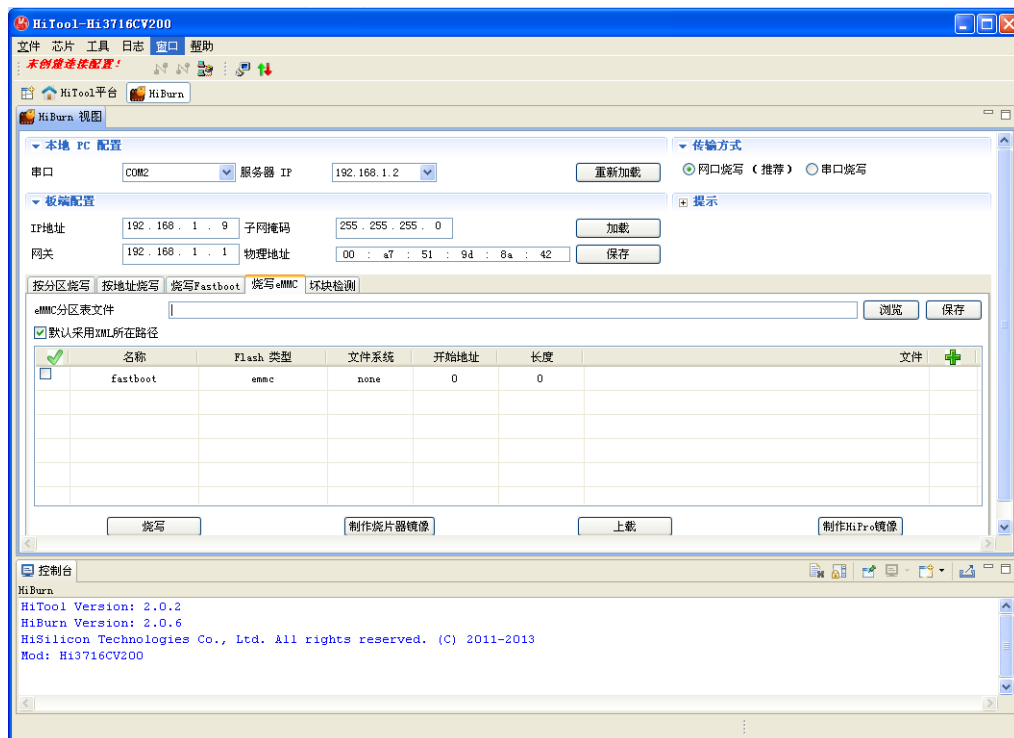
4.1.3.1 生成 eMMC 镜像文件

烧录器需要使用的 eMMC 镜像文件可以通过 HiTool 制作。

步骤 1 打开 HiTool 工具，选择 HiBurn，然后选择烧写 eMMC 栏，如图 4-2 所示。

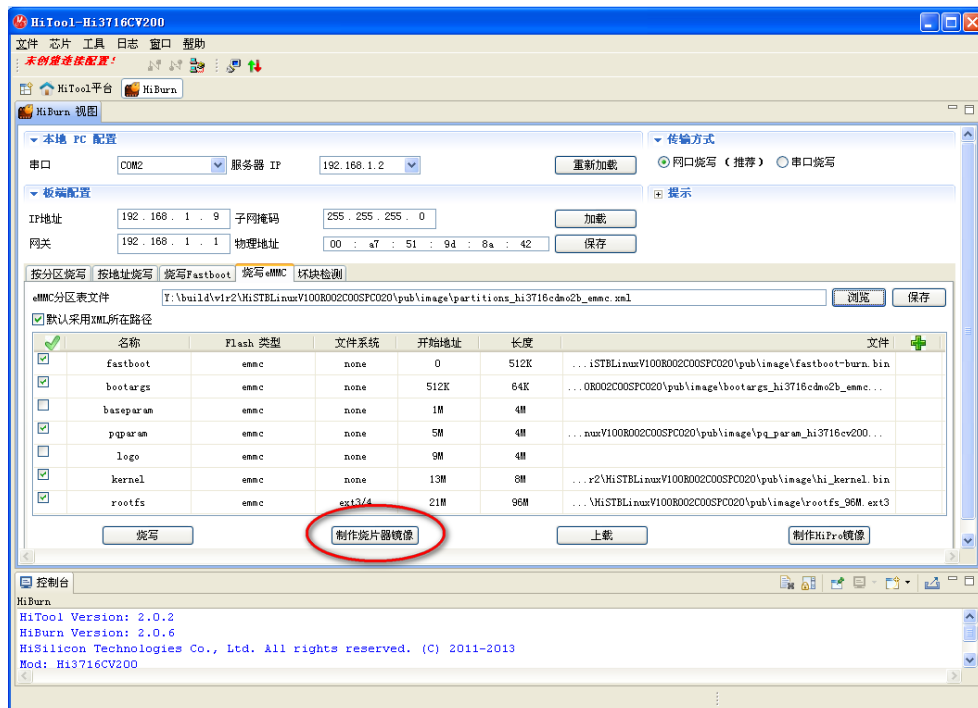


图4-2 HiTool 工具界面



步骤 2 配置各个分区，或者直接导入 xml 格式的分表，如图 4-3 所示。

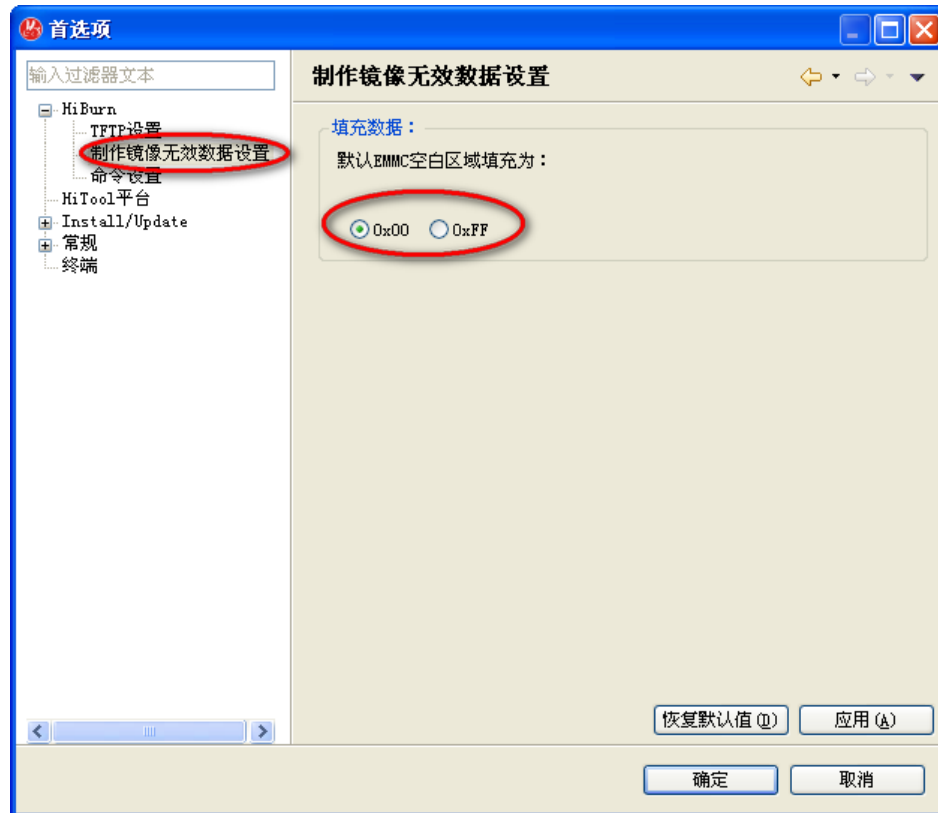
图4-3 配置各个分区后





步骤3 由于不同的器件对于空白区域填充 0 或者 F 设置不同，用户可以自行定义，最好与 flash 厂家或烧片器厂家确认后设置，保证烧片器时的性能。选择菜单栏->窗口->首选项，点击 HiBurn，进行设置。

图4-4 eMMC 空白区填充配置



步骤4 点击制作烧片器镜像按钮，并保存镜像文件，例如生成的镜像名称为 Hi3716Cv200_eMMC.img。

----结束

4.1.3.2 设置 eMMC 扩展 CSD 寄存器

此部分主要说明使用烧录器烧录前，在烧录器操作界面上，需要配置 eMMC 的哪些寄存器及寄存器的值。

Hi3716CV200 系列平台和 Hi379XX V100 平台支持 eMMC 4.41 标准，支持该标准的 eMMC 器件包含有 BOOT1、BOOT2 和 USER DATA 分区，同时支持 n_RST 管脚和下电复位。Hi3716CV200 系列平台从 USER DATA 区启动，所有镜像数据都烧录到 USER DATA 分区，同时只支持 n_RST 管脚复位器件，因此，烧录器必须按表 3-1 配置寄存器的值，否则单板无法启动。"

表4-1 扩展 CSD 寄存器配置

扩展 CSD 寄存器编号	寄存器值	说明
179	0x38	此寄存器用于配置 boot 分区，Hi3716CV200 系列平台和 Hi379XX V100 平台默认从 USER DATA 区
177	0x1	此寄存器用于配置 eMMC 在 boot 模式下的总线宽度，Hi3716CV200 系列平台和 Hi379XX V100 平台在 boot 模式下总线需配置成 4bit
162	0x1	此寄存器用于配置 eMMC 器件的 n_RST 管脚是否有效。Hi3716CV200 系列平台和 Hi379XX V100 平台的默认使用 n_RST 管脚，该寄存器必须配置成 0x1。



说明

- 必须在烧录之前完成 eMMC 扩展寄存器的配置。
- 部分烧录器可能不支持设置扩展 CSD 寄存器的功能，需烧录器厂家支持。
- 具体的设置由于各厂家的 eMMC 烧录器不同而存在差异，请参考烧录器手册来配置。

4.1.3.3 操作烧录器完成烧录

此部分内容与烧录器型号相关，不同烧录器烧录方法不一样，请先咨询烧录器厂家。



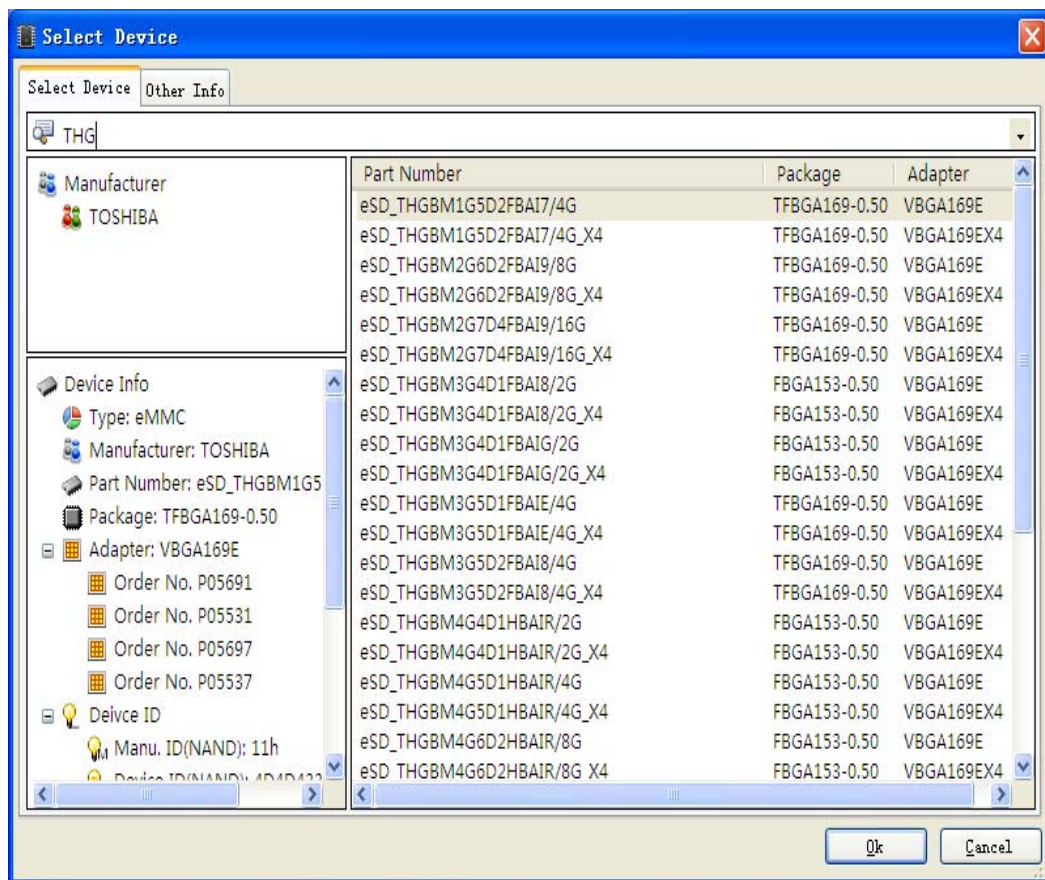
说明

本文仅以浦洛电子（www.prosystems.com.cn）的烧录器为例说明如何烧录。其中，图片来自浦洛电子烧录工具截图。

步骤 1 选择器件。该器件必须是 Hi3716CV200 系列平台和 Hi379XX V100 平台兼容器件列表中推荐器件。如图 4-5 所示。



图4-5 选择 eMMC 器件

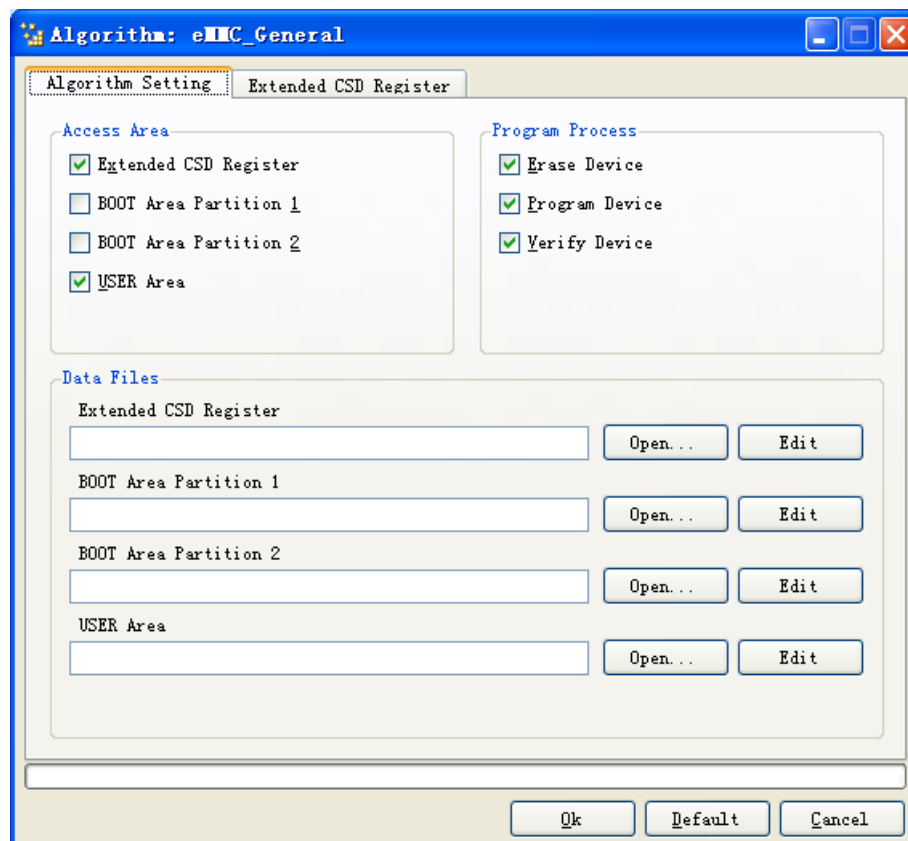


步骤 2 定制烧录算法，如图 4-6 所示。

按照前面的原则，定制将镜像烧录至 USER Area，烧录过程包括擦除、编程、校验。



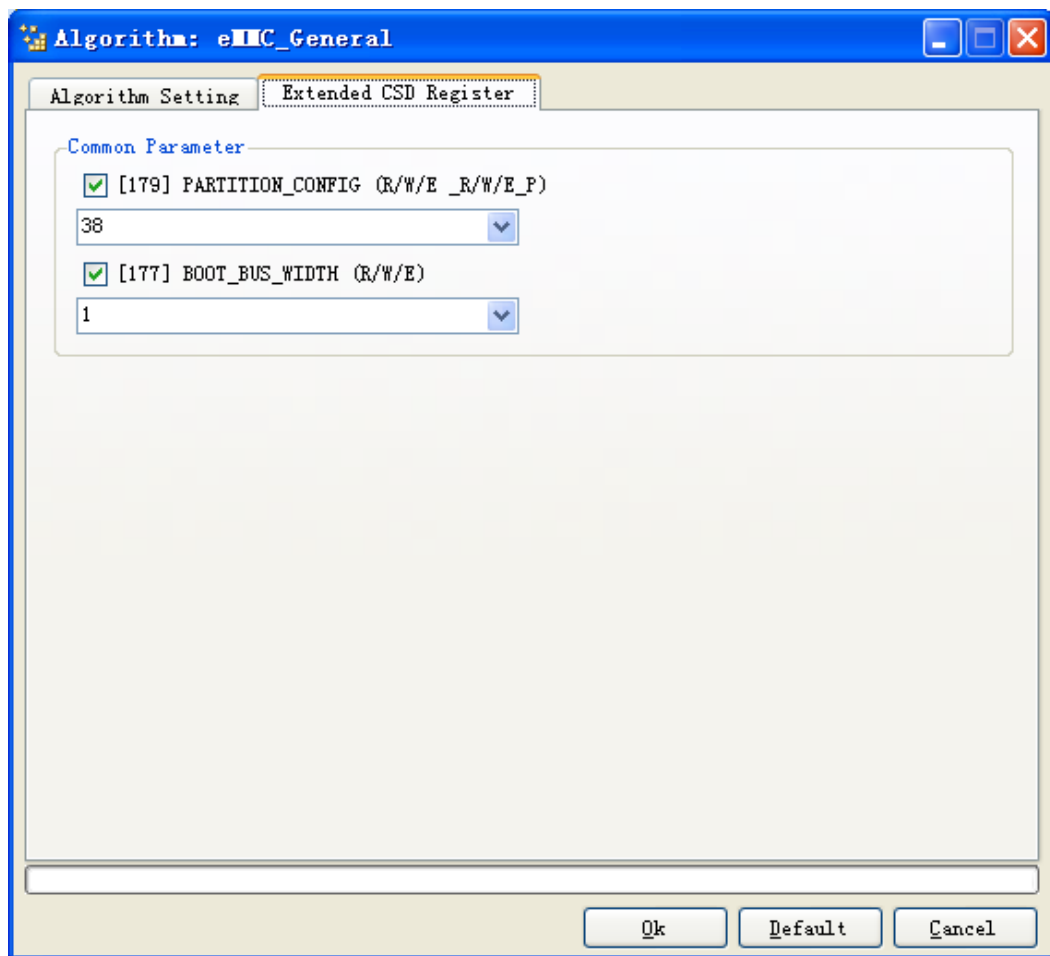
图4-6 定制烧录算法



步骤3 如图4-7所示。按照前面的原则，设置扩展CSD寄存器。



图4-7 设置扩展 CSD 寄存器



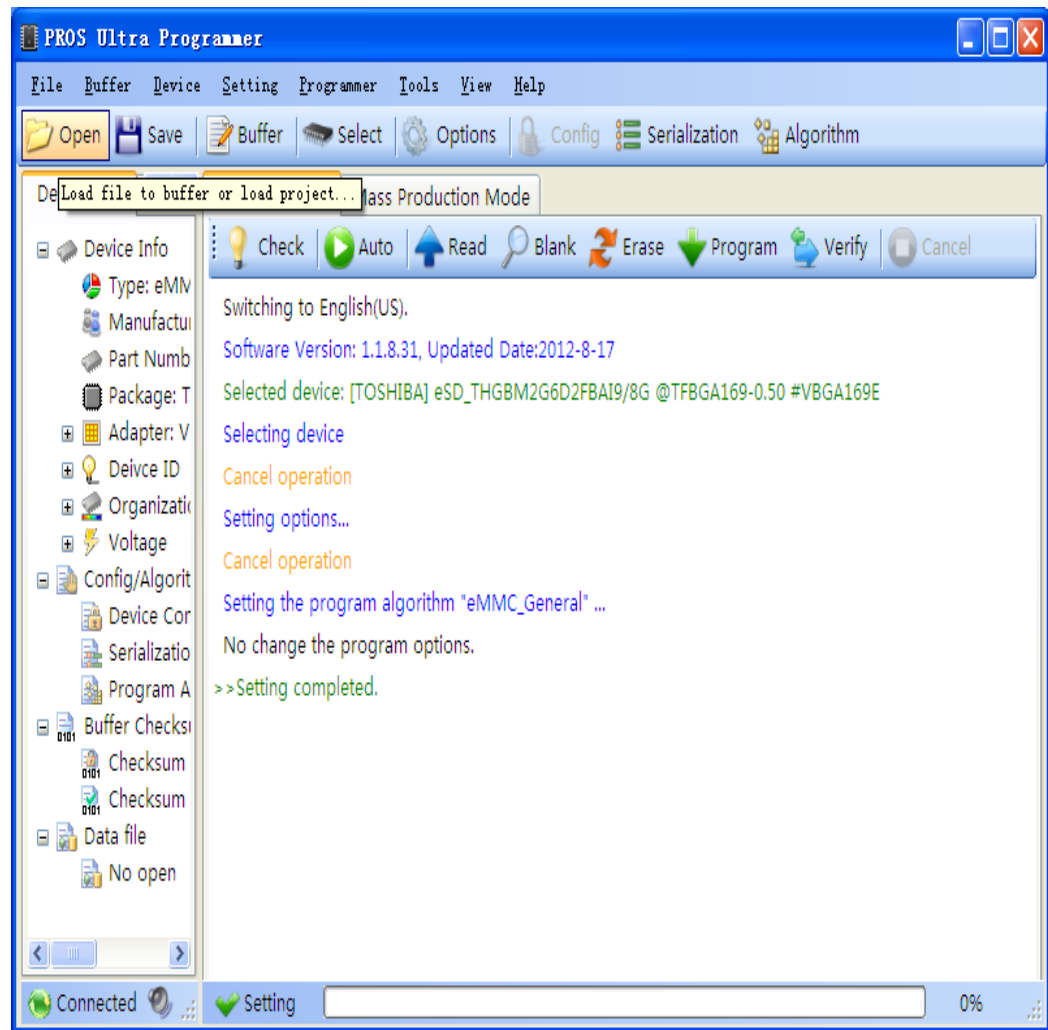
说明

本文使用的烧录器不支持 162 寄存器的配置。

步骤 4 导入镜像文件。如图 4-8 所示，点击打开按钮，导入镜像文件至缓存，镜像文件即为“4.1.3.1 生成 eMMC 镜像文件”节步骤中生成的 Hi3716Cv200_eMMC.img。



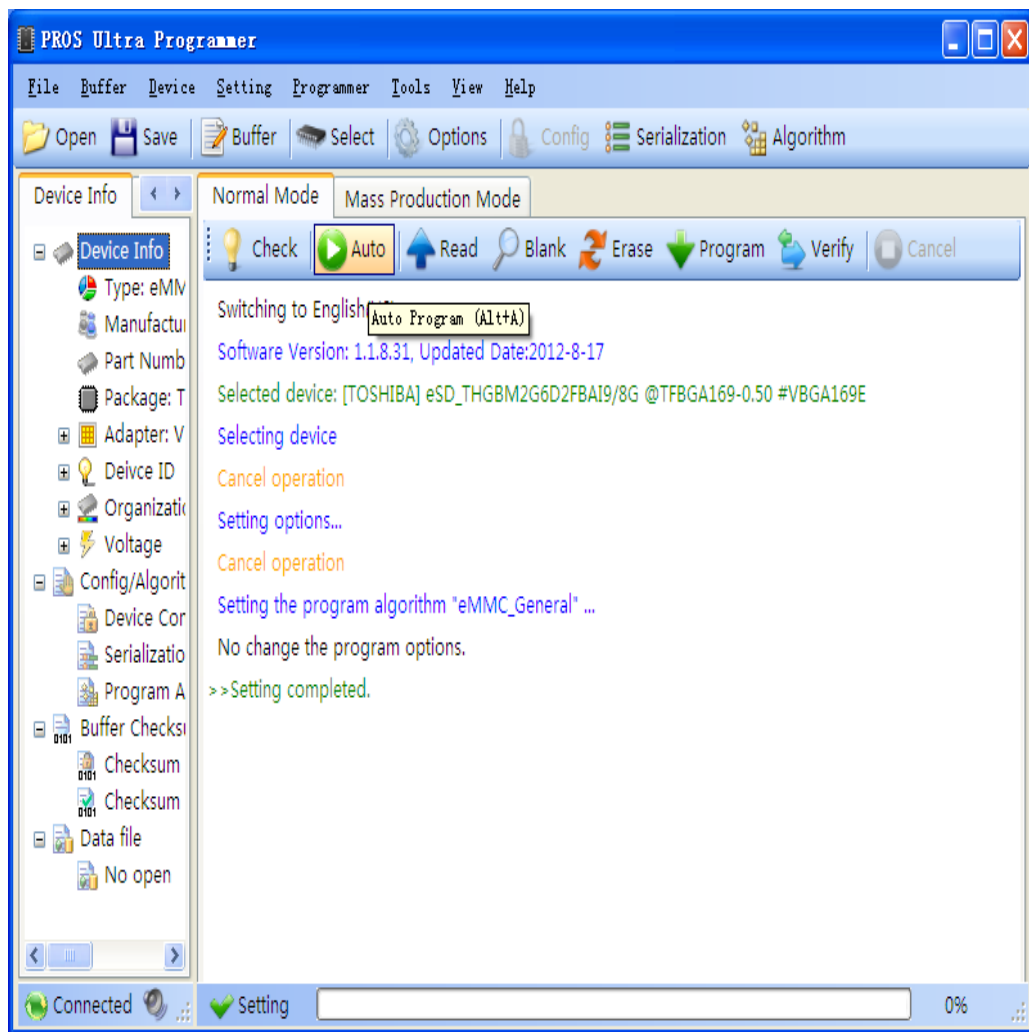
图4-8 导入镜像文件



步骤 5 开始烧录。如图 4-9 所示。



图4-9 开始烧录



-----结束

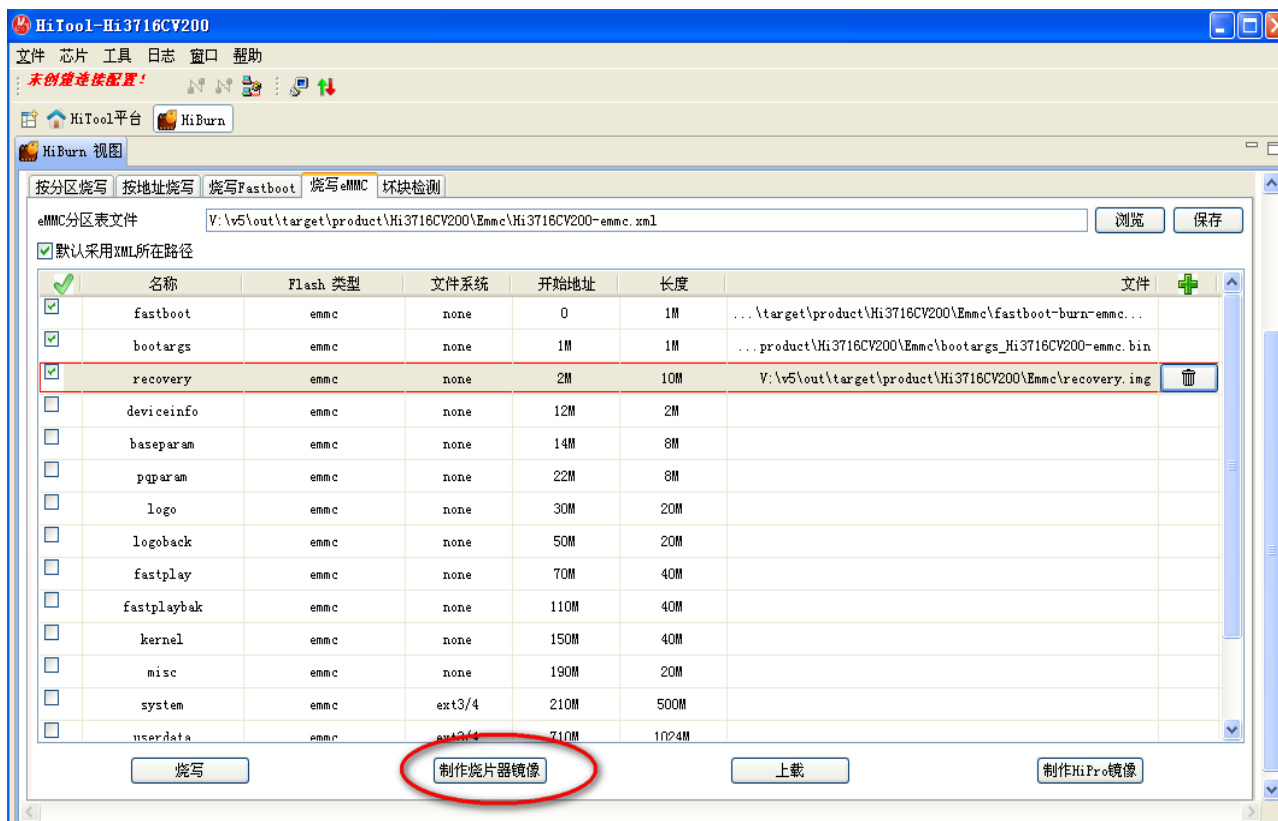
4.1.4 使用烧录器烧录 boot 和 Android recovery + recovery 升级

步骤如下：

- 步骤 1 制作 Android recovery 的烧录器镜像，只勾选 fastboot、bootargs 和 recovery 分区，点击制作烧片器镜像，并保存镜像文件，例如生成的镜像名称为 Hi3716Cv200_eMMC_recovery.img，如图 4-10 所示。



图4-10 制作烧片器镜像。



步骤 2 使用烧录器烧录 Hi3716CV200_eMMC_recovery.img，注意，参考“4.1.3.2 设置 eMMC 扩展 CSD 寄存器”节在烧录器软件界面上设置好 eMMC 的扩展 CSD 寄存器，烧录器的使用方法请联系烧录器厂家支持！

步骤 3 拷贝 Android 的 update.zip 包，放到 U 盘根目录；

步骤 4 重启单板，通过按键进入 recovery 模式。Recovery 程序会读取 U 盘的 update.zip 包升级单板。

----结束

4.2 SPI Flash+NAND Flash 或 NAND Flash 的方案

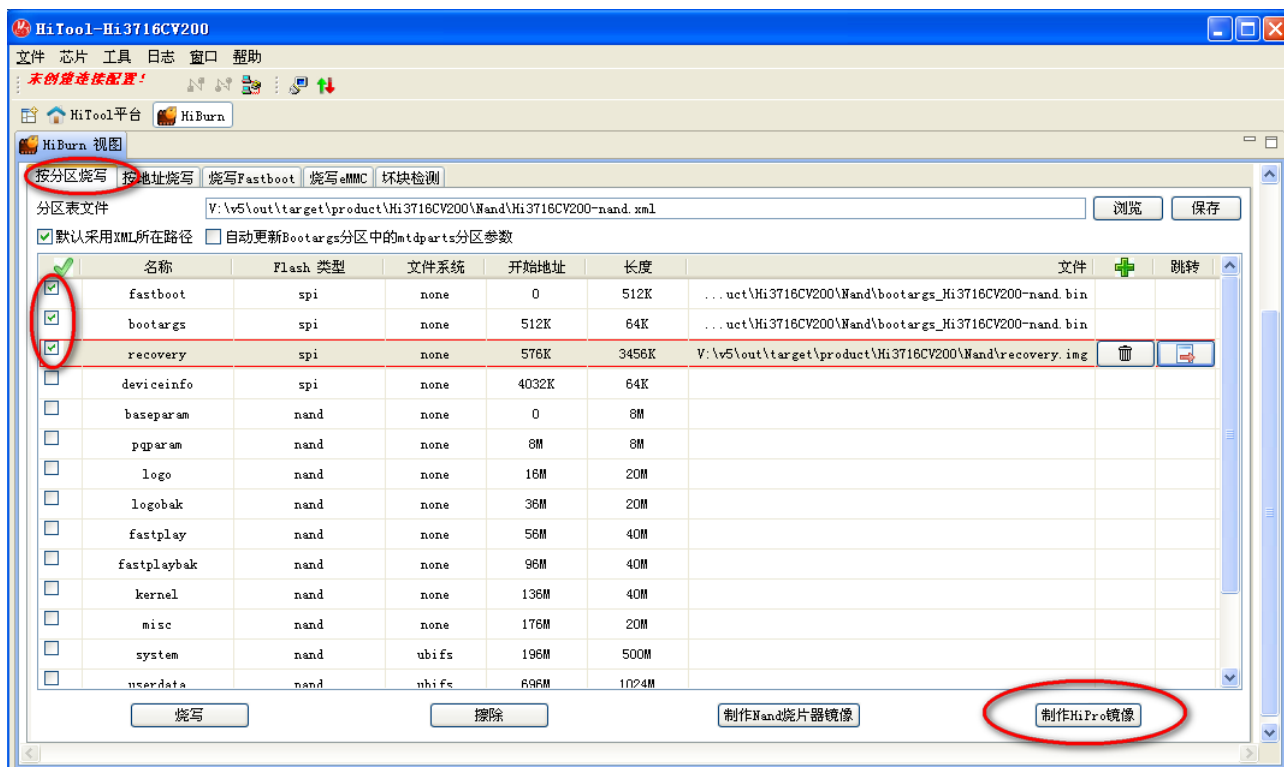
4.2.1 使用 HiPro-serial 工具烧录 boot 和 Android recovery + recovery 升级

步骤如下：

步骤 1 制作 Android recovery 的 HiPro-serial 镜像，只勾选 fastboot、bootargs 和 recovery 三个分区，然后点击制作 HiPro-serial 镜像按钮，如图 4-11 所示。



图4-11 制作 Android 的 HiPro-serial 镜像



步骤 2 拷贝 usb_update.bin 到 U 盘根目录。

步骤 3 拷贝 Android 的 update.zip 包，放到 U 盘根目录。

步骤 4 使用 HiPro-serial 工具烧录 boot 和 Android 的 recovery 内核/

步骤 5 重启单板，通过按键进入 recovery 模式。Recovery 程序会读取 U 盘的 update.zip 包升级单板。

----结束

4.2.2 使用烧录器烧录整个 Nand 镜像

Nand 镜像需要使用 SDK 提供的 Hitool 工具进行转换，生成烧录器使用的镜像文件。

对于 SPI Flash 的镜像，直接使用原始的编译生成文件即可烧录，请联系 SPI Flash 烧录器厂家支持。



说明

本文仅介绍如何制作 Nand 烧片器镜像。

使用烧录器烧录 Nand，主要包括以下步骤：

步骤 1 准备原始文件，包括 boot 映像、参数区映像、kernel 映像、文件系统映像等。

步骤 2 生成 Nand 镜像文件。



- 步骤 3 计算每个烧录文件的烧写起始地址，获得烧录文件的烧写长度，具体的计算方法，请见 4.2.2.4 节。
- 步骤 4 使用烧写器按照每个烧录文件的烧写起始地址和烧写长度，对 Nand Flash 中的区域进行单独烧写。具体烧写器的使用方法，需要与烧写器厂家联系。

----结束

下面以某公司的 Nand Flash 器件为例，说明如何制作 Nand 镜像文件，以及如何计算烧写地址和长度。

从 Boot 打印找到如下 Nand 特性：

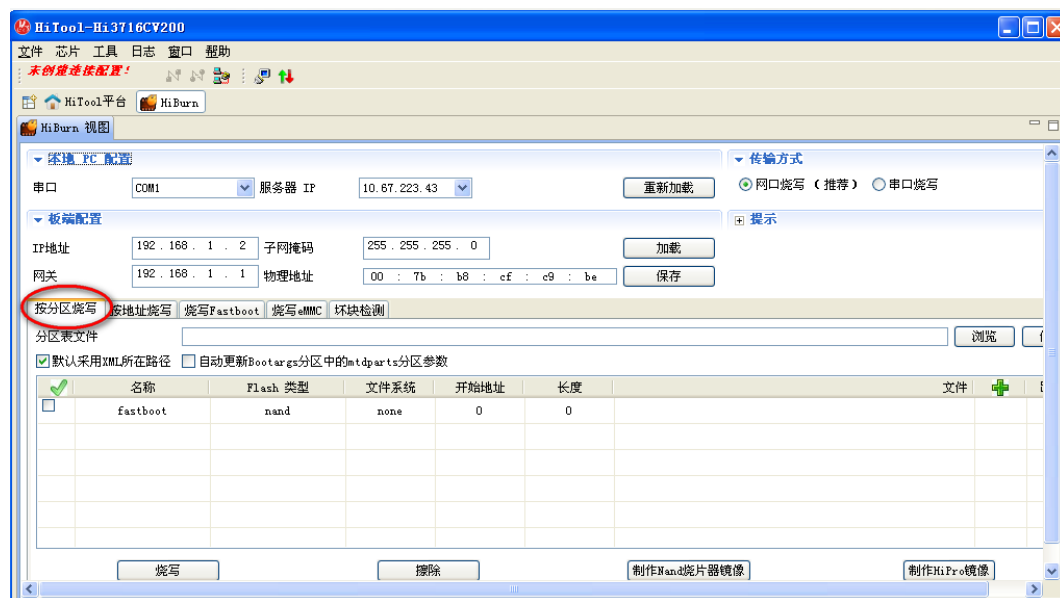
- 容量：512MByte
- 块大小：128KByte
- 页大小：2KByte
- SpareArea 区大小：64Byte (以下称 OOB 区)
- ECC 类型：8bit

4.2.2.2 生成 Nand 镜像文件

烧录器需要使用的 Nand 镜像文件可以通过 HiTool 制作。下面以 Hi3716CV200 芯片平台为例，介绍烧录器镜像的制作方法。

- 步骤 1 打开 HiTool 工具，选择 HiBurn，然后选择烧写按分区烧写栏，如图 4-12 所示。

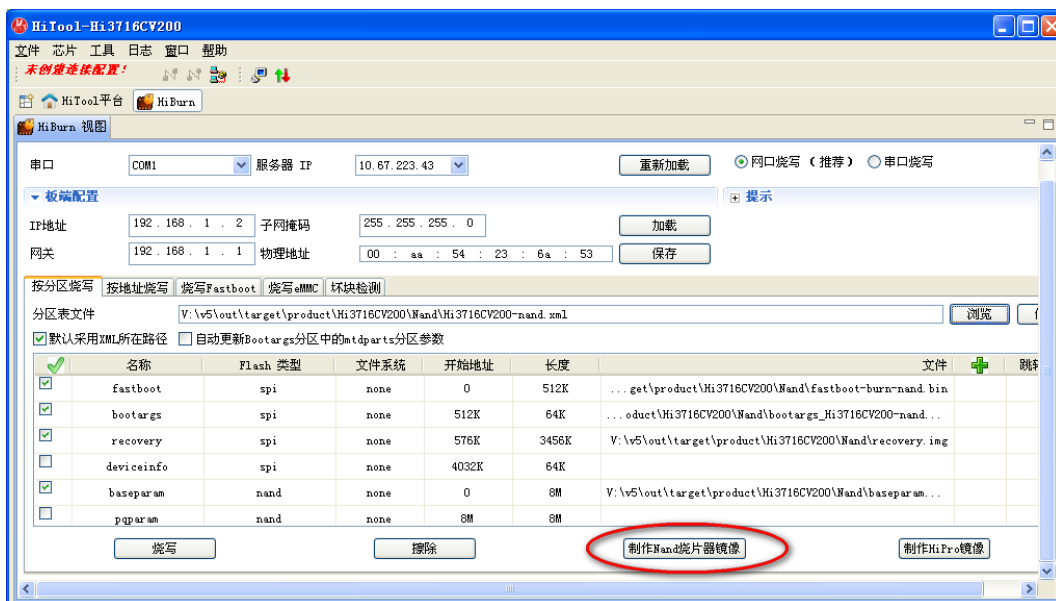
图4-12 HiTool 工具界面



- 步骤 2 配置各个分区，或者直接导入 xml 格式的分表，如图 4-3 所示。

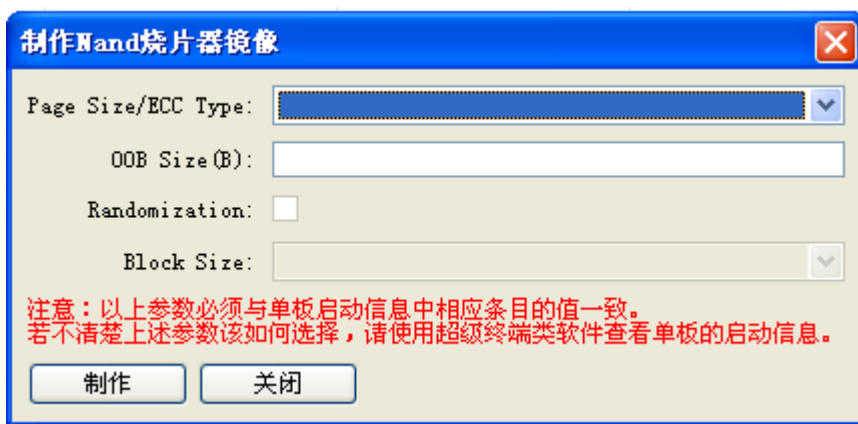


图4-13 配置各个分区后



步骤 3 点击制作烧片器镜像按钮，弹出制作 Nand 烧片器镜像界面，如图 4-14 所示。

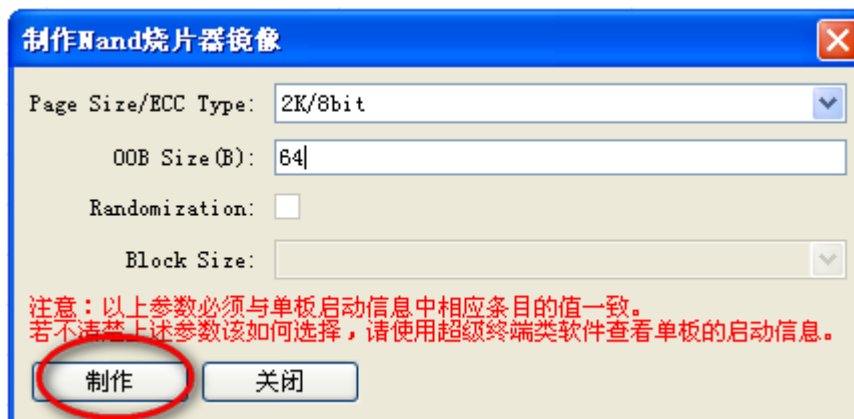
图4-14 制作 Nand 烧录器镜像的界面



步骤 4 根据 Boot 中打印的 Nand 器件信息，配置好页大小和 ECC 类型，OOB 区的大小，如果器件支持 Randomization，需要勾选，并配置好块大小，然后点击制作按钮。



图4-15 设置 Nand 器件属性



步骤 5 选择好保存路径后，HiTool 工具会自动遍历每个 Nand 分区文件生成对应的镜像，每个镜像均以.burn 作为扩展名。

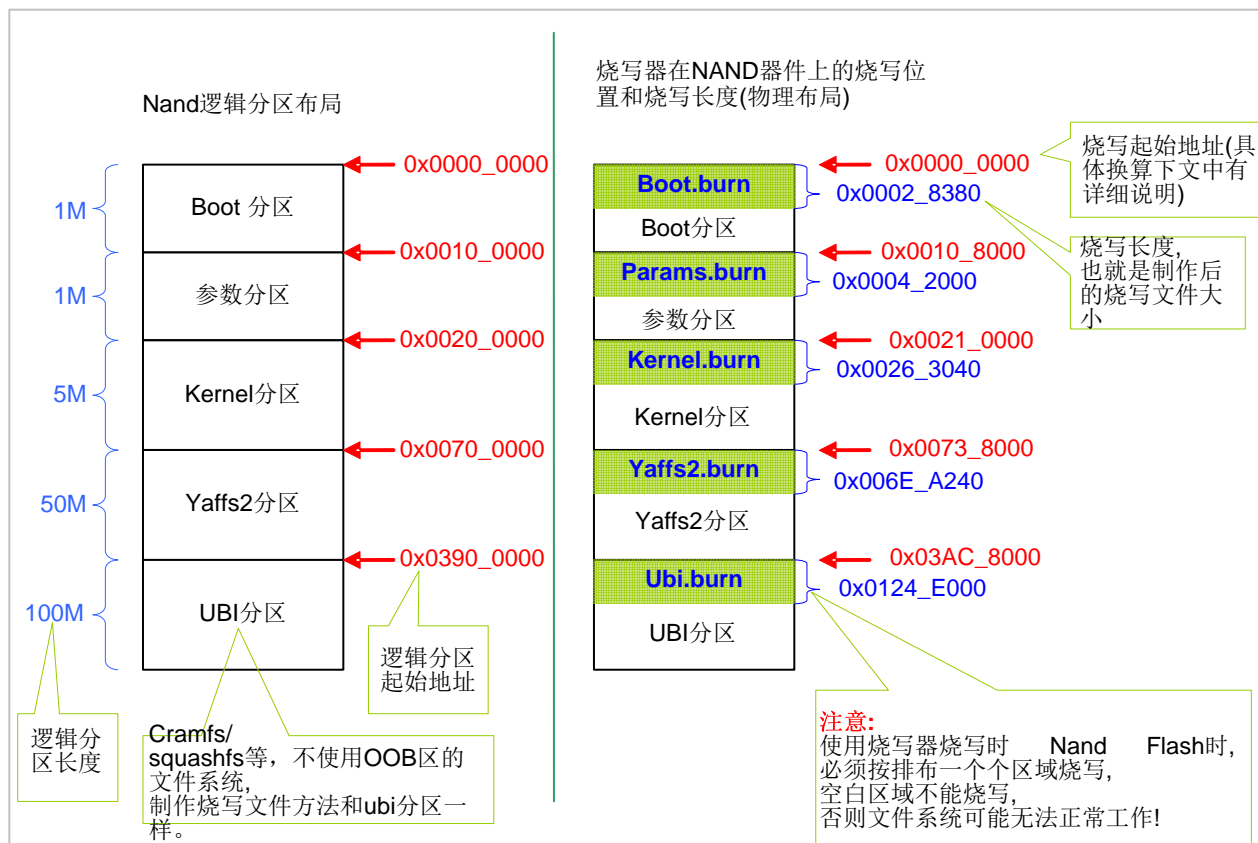
----结束

4.2.2.3 烧写布局

图 4-16 是 NAND 器件上逻辑分区布局，以及转换后的烧写文件在 NAND 器件上布局。



图4-16 NAND 器件上逻辑分区布局



说明

- 以上分区只是一个示例, 用户应该按自己的实际情况来决定分区起始位置和分区大小。
- 分区之间不能重叠。
- 分区起始地址, 长度应该是 NAND Flash 块大小整数倍。
- 分区大小要大于需要烧录的数据大小, 并为坏块保留余量。
- 所有分区的大小之和要小于 NAND Flash 器件容量。
- Cramfs/Squashfs 等不用 OOB 区的文件系统, 制作烧写文件的方法和 Kernel 一样。
- 以上示例的物理布局中, 未标示坏块。用户应选择有跳坏块功能的烧写器, 烧写器遇坏块, 跳到下一个块烧写。

4.2.2.4 烧写地址计算方法

烧写器看到的是物理地址, 所以:

$$\text{烧写起始地址} = \frac{\text{逻辑分区起始地址}}{\text{芯片页大小}} \times (\text{芯片页大小} + \text{芯片OOB大小})$$

例如, 对于本文给出的 Nand Flash 芯片:

$$\text{烧写起始地址} = \frac{\text{逻辑分区起始地址}}{2K} \times (2K + 64)$$



表4-2 文件烧写起始地址和烧写长度

分区	原始文件	逻辑分区起始地址	逻辑分区大小	转换后的烧写器文件	烧写起始地址	烧写长度
1	Boot.bin	0x0000_0000	1M	Boot.burn	0x0000_0000	0x0002_8380
2	Params.bin	0x0010_0000	1M	Params.burn	0x0010_8000	0x0004_2000
3	Kernel.img	0x0020_0000	5M	Kernel.burn	0x0021_0000	0x0026_3040
4	rootfs.yaffs2	0x0070_0000	50M	Yaffs2.burn	0x0073_8000	0x006E_A240
5	Ubi.img	0x0390_0000	100M	Ubi.burn	0x03AC_8000	0x0124_E000

**注意**

- 使用烧写器烧写 Nand Flash 时，必须按排布一个个区域烧写，空白区域不要烧写，否则文件系统可能无法正常工作。
- 烧写器必须能控制对 Nand Flash 中的某个区域进行单独烧写。

用户分区时，通常以“页大小”为单位，烧写器是以“页大小+OOB 区”为单位，从用户分区到烧写器需要进行换算。对于 NAND Flash 而言，页区域和 OOB 区域是同等，没有任何区别。

4.2.3 使用烧录器烧录 boot 和 Android recovery + recovery 升级

SPI Flash 烧录器的成本较低，如果考虑到成本，可以采用 SPI Flash 烧录 boot 和 Android recovery，然后单板上电进入 recovery，通过 recovery 程序完成升级。因此，在 SPI + Nand 的方案中，可以考虑此烧录方案，对于只有 Nand 的方案，如果已经有 Nand 烧录器，建议用烧录器全烧 Nand 镜像，烧录效率更高。

- 步骤 1** 使用烧录器烧录 boot、bootargs 和 recovery 三个分区，烧录器的使用方法请联系烧录器厂家支持。
- 步骤 2** 拷贝 Android 的 update.zip 包，放到 U 盘根目录。
- 步骤 3** 重启单板，通过按键进入 recovery 模式。Recovery 程序会读取 U 盘的 update.zip 包升级单板。

----结束



5 量产问题定位指南



注意

本章节仅适用于 Hi3798M V100。

5.1 CPU 上电打印说明

以下按打印在代码流程中的先后顺序，说明各个打印含义。

表5-1 正常启动流程

序号	阶段	打印	说明
1	Bootrom	Bootrom start	<p>这是 CPU 上电第一条打印。</p> <p>如果没有此打印，需要检查硬件：</p> <ul style="list-style-type: none">• CPU 供电是否正常； <p>要求各个供电电源（3.3V_standby、3.3V_MOS、1.1V_CORE、1.1V_CPU）供电正常；</p> <p>注意：CPU 上电后是否有过热的现象，如果存在该现象则立即关闭电源，测量 4 路供电对地的阻抗，检查电源问题。</p> <ul style="list-style-type: none">• 系统 24M 时钟是否起振； <p>使用万用表 DC 测量晶体的两端，电压应该为 1.65V 左右，两端略有差异；使用示波器观测晶体两端波形应看到完整的、幅度约 3.3Vp-p 的 24M 正弦波；</p> <ul style="list-style-type: none">• 检查串口电路和串口连接是否正常。芯片串口是 CMOS 电平（3.3V）、PC 端是 RS232 电平，注意电平转换电路设计；串口收发工具应设置为：115200bps，数据位 8，停止位 1，奇偶位和校验位为 0；• 检查 CPU 是否焊接良好，不能出现焊反、虚焊等情况。



序号	阶段	打印	说明
2		Boot from NAND 或者 Boot from eMMC 或者 Boot from SD	如果系统识别的启动方式和板级不符，请检查 BOOT_SEL0 和 BOO_SEL1 信号，具体管脚请参考硬件用户手册： <ul style="list-style-type: none"> • BOOT_SEL0 和 BOO_SEL1 这两个启动配置管脚的外置上下拉电阻是否配置正确，详细请参考硬件用户指南； • 请检查 BOOT_SEL0 和 BOO_SEL1 这两个启动配置管脚是否对接了输出器件，上电采集时受对接器件影响。
3		Read eMMC error 或者 Read SD error	出现此打印，表示系统从 eMMC/SD 卡启动，但软件读 eMMC/SD 状态出错。需要检查硬件： <ul style="list-style-type: none"> • eMMC/SD 卡座焊接是否有问题、连锡情况，SD 卡是否插上； • 检查 eMMC/SD 供电是否正常，要求供电 $3.3V \pm 5\%$； • 检查 eMMC/SD 卡接口串联电阻、上拉电阻是否正常上件，电阻/排阻阻值是否正确； • 更换一块新 eMMC/SD 器件（烧录好 boot 后）。
4		Starting fastboot ...	出现此打印，表明 bootrom 代码已经成功执行完成，即将跳转到 boot 代码执行。
5	Fastboot	System startup	这个表示 CPU 已开始执行 flash 上的内容。请注意，如果没有此打印： <ol style="list-style-type: none"> 1. 检查软件： <p>NAND/eMMC/SD 上镜像是否正确烧录，版本是否正确；</p> 2. 检查硬件： <p>eMMC/SD 启动：</p> <p>则前面打印已经确认硬件没有问题，请检查软件烧写镜像。</p> <p>NAND 启动：</p> <p>NAND 供电是否正常，要求供电 $3.3V \pm 5\%$；</p> <p>检查 NAND 焊接是否正常，NAND 管脚比较密，容易连锡，或者更换一块 NAND 器件；</p>
6		*** Not support current start mode	硬件板级启动方式配置在第 2 步已经完成确认，应该不会有问 题。 因此，出现这个打印，表示 Flash 中烧录的 fastboot 镜像与 Flash 不匹配，如板级焊接的是 eMMC，从 eMMC 启动，但是 eMMC 中却烧写了 SPI 或者 NAND 的 fastboot 镜像。
7		没有任何打印，且也不 往下执行，串口没有任 何打印	如果没有任何打印，则表示 DDR 完全就没有通，DDR 初始化 失败，请考虑如下检查： <ul style="list-style-type: none"> • 检查编写 boot 时，实际使用的*.reg 文件和当前单板是否 完全匹配，表格对应关系的详细说明请参考： <p>linux 版本目录/source/boot/sysreg/read_cn.xls;</p> <p>android 版本目录</p>



序号	阶段	打印	说明
			<p>device/hisilicon/bigfish/sdk/source/boot/sysreg/read_cn.xls;</p> <ul style="list-style-type: none"> • 测量 DDR 1V5 是否供电，测量 DDR 1V5 电容对地阻抗，看电源是否短路； • DDR 的参考电压是否正常，要求 $0.5VDD \pm 1\%$； • 查看 DDR 颗粒是否焊接正常，颗粒是否焊接错误； • 测量主芯片 CORE 电源是否供电正常，要求 $VDD \pm 5\%$。 • 若存在 VTT 匹配，则需测量 VTT 电压是否正常，要求 $0.5VDD \pm 5\%$； • 请检查所使用的 DDR 颗粒不在兼容器件列表中；
8		SWL DDR training failed 或者 SGA DDR training failed 或者 HGA DDR training failed 或者 HRD DDR training failed 或者 SRD DDR training failed 或者 SWD DDR training failed	<p>出现此类打印，表示 DDR 初始化流程已经完成，但部分信号还是存在问题，可能是通断问题，也有可能是信号质量问题，需要通过打印排查：</p> <ul style="list-style-type: none"> • 打印 HGA + 00000000，表示 DDR 的 CK 或者 DQS 有问题： • 这个打印无法判断具体是哪个信号问题，需要看后面的打印，可以从 DDR 供电、焊接等方面作简单的排查； • SWL + 0000000X，X 表示 PCB 板走线网络命名对应的 DQS_0/1/2/3，打印只有一条，优先打印低位 DQS，例如 DQS0 和 DQS1 都有问题，则只打印 DQS0； • 出现此打印，基本可以说明 CK、DQSx 或者 DQSx 对应的 DQ0 三者必然有一个通路有问题，建议通过 DQSx 打印判断哪个 DDR 颗粒有问题，可能是 DDR 或者主芯片焊接问题、PCB 设计问题、PCB 加工质量问题，可以通过换 DDR 颗粒或者换主芯片的方法定位； • SWD + 0000000X，X 表示 PCB 板走线网络命名对应的 DQ 信号； • SRD + 0000000X，X 表示 PCB 板走线网络命名对应的 DQ 信号； • 如果有多个 DQ 信号错误，则只打印 1 个 DQ 错误，优先打印低 bit，如 DQ6 和 DQ10 同时报错，则实际是打印 DQ6，DQ6 修复后，才会报 DQ10 错误。 • 出现此打印，则表示对应的 DQ（与主芯片管脚名称为准）通路有问题或者信号质量有问题。可以通过降频作初步排查，降频方法参考附一： • 降频 OK，则表示信号质量有问题，需要排查 PCB 走线是否完全 copy 海思 demo 板，两层板包地线是否漏掉等； • 降频依然 fail，则表示对应 DQ 通断问题，则需要排查 PCB 质量问题和 DDR 或者主芯片的焊接问题 • 如果在产线需要快速定位，推荐通过打印判断出问题的 DDR 颗粒，直接更换 DDR 颗粒。 <p>原则上，系统打印 DDR 问题，定位方法为：</p>



序号	阶段	打印	说明
			<ul style="list-style-type: none"> • 检查单板硬件，包括 CORE 和 DDR 电源、DDR 和主芯片的焊接、BOM 物料； • 确认串口打印，确认出问题的 DDR 颗粒； • 降频排查通断问题，还是信号质量问题； • 更换 DDR 和主芯片，排查焊接、PCB 板设计或者加工质量问题。 DDR 错误打印含义详细信息见 5.2.2
9		Reg Version: xxxx Reg Time: xxxxx Reg Name: xxxxx	此打印表示表格初始化完成，DDR 初始化完成，DDR training 正常，理论上 DDR 可以正常工作。
10		No NAND device found!!!	出现此打印，请检查板级启动方式： <ul style="list-style-type: none"> • 如果是 eMMC/SD 启动，则是正常打印；
11		MMC/SD controller initialization. No MMC/SD card detect when read `SDIO_CARD_DETECT` pin.	出现此打印，请检查板级启动方式： <ul style="list-style-type: none"> • 如果是 NAND 启动，则是正常打印； • 如果是 eMMC/SD 启动，则表示硬件 SDIO_CARD_DETECT 管脚连接有问题； • 检查 SDIO_CARD_DETECT 是否低电平（要求为低电平），下拉电阻是否上件； • SDIO_CARD_DETECT 是否由于连锡的缘故和 3.3V 供电短路。
12		*** irq: data abort *** irq: undefined instruction *** irq: prefetch abort	出现此打印，可能原因： <ul style="list-style-type: none"> • Flash 上读出来的数据不正确，可能要重烧软件，确认 boot 版本是否正确； • DDR 异常，DDR 上的数据不正确，此种问题极少出现，但比较复杂，可以参考前文 DDR 定位方法，从 DDR 降频、电源（包括 CORE、VDD_DDR、VTT）、焊接、更换 DDR 颗粒、PCB 板质量等方面排查定位。

表5-2 Hitools 烧写流程

序号	阶段	打印	说明
1	Bootrom	Bootrom start	这是 CPU 上电第一条打印。 如果没有此打印，需要检查硬件： <ul style="list-style-type: none"> • CPU 供电是否正常； 要求各个供电电源（3.3V_standby、3.3V_MOS、1.1V_CORE、1.1V_CPU）供电正常； 注意：CPU 上电后是否有过热的现象，如果存在该现象则



序号	阶段	打印	说明
			<p>立即关闭电源，测量 4 路供电对地的阻抗，检查电源问题。</p> <ul style="list-style-type: none"> 系统 24M 时钟是否起振； <p>使用万用表 DC 测量晶体的两端，电压应该为 1.65V 左右，两端略有差异；使用示波器观测晶体两端波形应看到完整的、幅度约 3.3Vp-p 的 24M 正弦波；</p> <ul style="list-style-type: none"> 检查串口电路和串口连接是否正常。芯片串口是 CMOS 电平（3.3V）、PC 端是 RS232 电平，注意电平转换电路设计；串口收发工具应设置为：115200bps，数据位 8，停止位 1，奇偶位和校验位为 0； 检查 CPU 是否焊接良好，不能出现焊反、虚焊等情况。
2		Boot from NAND 或者 Boot from eMMC 或者 Boot from SD	<p>如果系统识别的启动方式和板级不符，请检查 BOOT_SEL0 和 BOO_SEL1 信号，具体管脚请参考硬件用户手册：</p> <ul style="list-style-type: none"> BOOT_SEL0 和 BOO_SEL1 这两个启动配置管脚的外置上下拉电阻是否配置正确，详细请参考硬件用户指南； 请检查 BOOT_SEL0 和 BOO_SEL1 这两个启动配置管脚是否对接了输出器件，上电采集时受对接器件影响。
3		Read eMMC error 或者 Read SD error	<p>出现此打印，表示系统从 eMMC/SD 卡启动，但软件读 eMMC/SD 状态出错。需要检查硬件：</p> <ul style="list-style-type: none"> eMMC/SD 卡座焊接是否有问题、连锡情况，SD 卡是否插上； 检查 eMMC/SD 供电是否正常，要求供电 $3.3V \pm 5\%$； 检查 eMMC/SD 卡接口串联电阻、上拉电阻是否正常上件，电阻/排阻阻值是否正确； 更换一块新 eMMC/SD 器件（烧录好 boot 后）；
4		#####	<p>打印#说明串口开始下载，如果打印“###”不到 10%（正常有百分比打印），说明 DDR 通路有问题，请考虑如下检查：</p> <ul style="list-style-type: none"> 检查编写 boot 时，实际使用的*.reg 文件和当前单板是否完全匹配，表格对应关系的详细说明请参考： linux 版本目录/source/boot/sysreg/read_cn.xls； android 版本目录 device/hisilicon/bigfish/sdk/source/boot/sysreg/read_cn.xls； 测量 DDR 1V5 是否供电，测量 DDR 1V5 电容对地阻抗，看电源是否短路； DDR 的参考电压是否正常，要求 $0.5VDD \pm 1\%$； 查看 DDR 颗粒是否焊接正常，颗粒是否焊接错误； 测量主芯片 CORE 电源是否供电正常，要求 $VDD \pm 5\%$。



序号	阶段	打印	说明
			<ul style="list-style-type: none"> • 若存在 VTT 匹配, 则需测量 VTT 电压是否正常, 要求 $0.5VDD \pm 5\%$; • 请检查所使用的 DDR 颗粒不在兼容器件列表中。
5		SWL DDR training failed 或者 SGA DDR training failed 或者 HGA DDR training failed 或者 HRD DDR training failed 或者 SRD DDR training failed 或者 SWD DDR training failed 或者	<p>出现此类打印, 表示 DDR 初始化流程已经完成, 但部分信号还是存在问题, 可能是通断问题, 也有可能是信号质量问题, 需要进一步通过打印排查:</p> <ul style="list-style-type: none"> • 打印 HGA + 00000000, 表示 DDR 的 CK 或者 DQS 有问题; • 这个打印无法判断具体是哪个信号问题, 需要看后面的打印, 可以从 DDR 供电、焊接等方面作简单的排查; • SWL + 0000000X, X 表示 PCB 板走线网络命名对应的 DQS_0/1/2/3, 打印只有一条, 优先打印低位 DQS, 例如 DQS0 和 DQS1 都有问题, 则只打印 DQS0; • 出现此打印, 基本可以说明 CK、DQSx 或者 DQSx 对应的 DQ0 三者必然有一个通路有问题, 建议通过 DQSx 打印判断哪个 DDR 颗粒有问题, 可能是 DDR 或者主芯片焊接问题、PCB 设计问题、PCB 加工质量问题, 可以通过换 DDR 颗粒或者换主芯片的方法定位; • SWD + 0000000X, X 表示 PCB 板走线网络命名对应的 DQ 信号; • SRD + 0000000X, X 表示 PCB 板走线网络命名对应的 DQ 信号; • 如果有多个 DQ 信号错误, 则只打印 1 个 DQ 错误, 优先打印低 bit, 如 DQ6 和 DQ10 同时报错, 则实际是打印 DQ6, DQ6 修复后, 才会报 DQ10 错误。 • 出现此打印, 则表示对应的 DQ (与主芯片管脚名称为准) 通路有问题或者信号质量有问题。可以通过降频作初步排查, 降频方法参考附一: • 降频 OK, 则表示信号质量有问题, 需要排查 PCB 走线是否完全 copy 海思 demo 板, 两层板包地线是否漏掉等; • 降频依然 fail, 则表示对应 DQ 通断问题, 则需要排查 PCB 质量问题和 DDR 或者主芯片的焊接问题; • 如果在产线需要快速定位, 推荐通过打印判断出问题的 DDR 颗粒, 直接更换 DDR 颗粒。 <p>原则上, 系统打印 DDR 问题, 定位方法为:</p> <ul style="list-style-type: none"> • 检查单板硬件, 包括 CORE 和 DDR 电源、DDR 和主芯片的焊接、BOM 物料; • 确认串口打印, 确认出问题的 DDR 颗粒; • 降频排查通断问题, 还是信号质量问题; • 更换 DDR 和主芯片, 排查焊接、PCB 板设计或者加工



序号	阶段	打印	说明
			质量问题。 • DDR 错误打印含义详细信息见 附二。
6	Fastboot	System startup	此打印说明 fastboot 已经下载到 DDR，并开始执行。如果 download 完成 100%，但没有此打印。理论上这种情况比较少见，出现属于一种异常，从以下几点排查： • 检查 Fastboot 版本是否和单板匹配，*.reg 文件和板级匹配，包括 PCB 板层、DDR 颗粒数等； • 检查 VDD_CPU、VDD_CORE 的电压值是否正常； • 下载 DDR 降频版本，检查是否和 DDR 相关，如果相关则从第 4 步开始检查。
7		*** Not support current start mode	硬件板级启动方式配置在第 2 步已经完成确认，应该不会有问题。 因此，出现这个打印，表示 Flash 中烧录的 fastboot 镜像与 Flash 不匹配，如板级焊接的是 eMMC，从 eMMC 启动，但是 eMMC 中却烧写了 SPI 或者 NAND 的 fastboot 镜像。
8		没有任何打印，且也不往下执行，串口没有任何打印	如果没有任何打印，则表示 DDR 完全就没有通，DDR 初始化失败，请考虑如下检查： • 检查编写 boot 时，实际使用的*.reg 文件和当前单板是否完全匹配，表格对应关系的详细说明请参考 sdk 版本目录/source/boot/sysreg/read_cn.xls； • 测量 DDR 1V5 是否供电，测量 DDR 1V5 电容对地阻抗，看电源是否短路； • 查看 DDR 颗粒是否焊接正常，颗粒是否焊接错误； • 测量主芯片 CORE 电源是否供电正常，要求 VDD±5%。
9		Reg Version: xxxx Reg Time: xxxxx Reg Name: xxxxx	HiTool 烧写，此打印没有实际意义。
10		No NAND device found!!!	出现此打印，请检查板级启动方式： • 如果是 eMMC/SD 启动，则是正常打印； • 如果是 NAND 启动，则请检查板上 NAND FLASH 电路。
11		MMC/SD controller initialization. No MMC/SD card detect when read 'SDIO_CARD_DETECT' pin.	出现此打印，请检查板级启动方式： • 如果是 NAND 启动，则是正常打印； • 如果是 eMMC/SD 启动，则表示硬件 SDIO_CARD_DETECT 管脚连接有问题； • 检查 SDIO_CARD_DETECT 是否低电平（要求为低电平），下拉电阻是否上件；



序号	阶段	打印	说明
			<ul style="list-style-type: none"> SDIO_CARD_DETECT 是否由于连锡的缘故和 VDD 供电短路。
12		*** irq: data abort *** irq: undefined instruction *** irq: prefetch abort	出现此打印，可能原因： <ul style="list-style-type: none"> Flash 上读出来的数据不正确，可能要重烧软件，确认 boot 版本是否正确； DDR 异常，DDR 上的数据不正确，此种问题极少出现，但比较复杂，可以参考前文 DDR 定位方法，从 DDR 降频、电源（包括 CORE、VDD_DDR、VTT）、焊接、更换 DDR 颗粒、PCB 板质量等方面排查定位。

5.2 DDR training 相关打印

DDR training 是海思平台内嵌的 DDR 时序自适应算法，主要有三方面的功能：

- 单板上电 DDR 最佳时序窗口自适应；
- Boot 下 ddr 时序窗口打印；
- DDR 报错问题打印。

附件简要介绍第 2、3 两种用法。

5.2.1 Boot 下 ddr 时序窗口打印

在 fastboot 命令行下，输入“ddr training”命令，则会出现如图打印，主要包含以下几点信息：

- “The write window”和“The read window”：表示 DDR 写时序窗口和 DDR 读的时序窗口；
- DQ 列：从 0-31 表示 DQ0-DQ31，表示 DDR 的 32bit，和主芯片出线管脚名称对应，主芯片 DDR 的出线管脚名称区分两层板和四层板，请参考实际发布包原理图 Symbol；
- WIN 列：表示对应 DQ 信号的总时序窗口，越大越好，要求窗口大于等于 14 级；
- “-”：表示 DDR 时序窗口 OK；
“X”：表示 DDR 时序窗口 Fail；

每个 DQ 对应的“-”越多，则表示这个 DQ 的时序窗口越好，但如果整体往一边偏的很多，则需要检查下 fastboot 版本。



图5-1 写窗口

DDR training terminated.

Boot Training result. The write window of prebit-deskew

DQ	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	HEX VALUE	DQphDQ	WIN		
0	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	0xfffffe0	18	16	23	
1	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	0x1fffffe0	18	17	24
2	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	0xfffffff0	18	16	24
3	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	0xffffffe0	18	16	23
4	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	0x7fffff0	18	15	23
5	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	0x3fffff8	18	14	23	
6	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	0xfffffe0	18	16	23
7	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	0x7fffff8	18	15	24	
8	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	0x1fffffe0	18	17	24
9	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	0x1fffffe0	18	17	24
10	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	0x3fffff8	18	14	23	
11	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	0x1fffffe0	18	17	24
12	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	0x7fffff8	18	15	24
13	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0xfffffff0	18	20	24
14	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	0x7fffff8	18	15	24
15	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	0xfffffc0	18	17	22
16	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	0x7ffffe	17	12	22	
17	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	0x3fffff	17	11	22	
18	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	0x7ffffe	17	12	22	
19	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	0xfffffe	17	12	23	
20	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	0xfffffe	17	12	23
21	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	0x1fffff	17	10	21
22	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	0x1fffff8	17	14	22
23	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	0xfffff	17	10	20
24	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	0x3fffff0	15	15	22
25	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	0x7fffff0	15	15	23
26	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	0x7ffffe0	15	16	22
27	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	0x7fffff0	15	15	23
28	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	0x3fffff8	15	14	23
29	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-														

图5-2 读窗口

Boot Training result	The read window of prebit-deskew																																								
DQ	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	HEX VALUE	DQS	DQ	WIN					
0	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	0x7fffff8	26	13	20					
1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	X	X	X	0x7ffff	26	9	19						
2	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	0x3fffff8	26	12	19					
3	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	X	X	0x1fffffc	26	11	19					
4	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	0x7ffffc	26	12	21					
5	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	X	0x1fffffc	26	11	19						
6	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	-	-	-	X	X	X	X	X	0x3fffff0	26	15	22					
7	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	X	0x7fffff8	26	13	20					
8	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	X	0x3ffffc	26	12	20					
9	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	X	0xfffff0	26	14	20					
10	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	X	0x1fffffe	26	11	20						
11	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	0x3fffff0	26	15	22					
12	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	X	0x1fffff	26	10	21						
13	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	0x1fffff0	26	14	21					
14	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	0x7fffff8	26	13	20					
15	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	0x1fffff0	26	14	21					
16	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	0xfffffe0	26	16	23					
17	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	0x7ffffc	26	12	21					
18	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	0x7ffffe0	26	16	22					
19	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	X	X	0x1ffffc	26	11	19					
20	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	0x3ffffe0	26	15	21					
21	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	X	0x3fffff8	26	12	19					
22	X	X	X	X	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0x3ffffe00	26	19	21					
23	X	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	0x1ffffe0	26	15	20					
24	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	0x3fffff0	26	15	22					
25	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	0x3fffff0	26	15	22					
26	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	0xfffffc	26	13	22					
27	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	0x1fffff0	26	14	21					
28	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	0x7ffffe	26	12	22					
29	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	0x7ffffc	26	12	21					
30	X	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	X	X	0x7ffff0	26	13	19					
31	X	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X	X	X	X	0xfffff8	26	13	21					



- 软件 write leveling 错误打印: SWL + 0000000X, X 表示 对应的 byte;
- 软件 dqs gating 错误打印: SGA + 0000000X, X 表示 对应的 byte;
- 硬件 dqs gating 错误打印: HGA + 00000000, 无法确认具体哪个 DQS;
- 硬件 read dataeye 错误打印: HRD + 00000000, 无法确认具体哪个 DQ;
- 软件 read dataeye 错误打印: SRD + 0000000X, X 表示 第一个错误的 DQ;
- 软件 write dataeye 错误打印: SWD + 0000000X, X 表示 第一个错误的 DQ;

出现以上错误打印信息, 基本是因为 DDR 与主芯片之间 DQS/DQ 等信号的通断或者信号质量有问题, 导致单板无法正常启动。

海思方案为了保证系统正常启动后, 能稳定的运行, 在 DDR 自检时, 对系统正常启动的 DDR 时序窗口进行了限制。从实验数据显示, DDR 时序窗口大于等于 12 级, 则基本能保证 DDR 系统稳定运行, 因此, DDR training 算法, 在系统上电 DDR 自检时, 如果发现部分 DQ 时序窗口小于 12 级, 则会打印: DDR training failed, result: 00000004, 防止量产时漏筛。

可以通过在 fastboot 表格中配置寄存器: 0xf80000d0 的 bit31=1 来 disable 掉功能, 重新编译 fastboot 烧写后, 系统能正常启动, 从而通过在 fastboot 命令行输入: ddr training 命令, 查看详细的 DDR 时序窗口, 协助定位 DQ 时序偏小的原因。