



启动介质烧写指南

文档版本 00B01

发布日期 2018-02-10

版权所有 © 深圳市海思半导体有限公司 2018。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基地华为电气生产中心 邮编：518129

网址：<http://www.hisilicon.com>

客户服务电话：+86-755-28788858

客户服务传真：+86-755-28357515

客户服务邮箱：support@hisilicon.com



前言

概述

本文档主要介绍在单板上烧写镜像及设置启动参数的相关操作。

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
Hi35xx	Vxx

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

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
2018-02-10	00B01	第 1 次临时版本发布。



目 录

前 言.....	i
1 概述.....	5
1.1 概述.....	5
1.2 网络配置.....	5
2 烧写镜像到 SPI Nor Flash	7
2.1 地址空间说明.....	7
2.2 烧写 u-boot	7
2.3 烧写 ulmage.....	8
2.4 烧写文件系统.....	8
2.5 SPI Nor Flash 启动参数设置	8
3 烧写镜像到 SPI NAND 或 NAND Flash	9
3.1 地址空间说明.....	9
3.2 烧写 u-boot	9
3.3 烧写 ulmage.....	9
3.4 烧写文件系统.....	10
3.4.1 烧写 yaffs2 文件系统.....	10
3.4.2 烧写 ubifs 文件系统.....	10
3.5 NAND Flash 启动参数设置.....	10
3.5.1 yaffs2 文件系统启动参数设置.....	10
3.5.2 ubifs 文件系统启动参数设置.....	11
4 烧写镜像到 EMMC	12
4.1 地址空间说明.....	12
4.2 烧写 u-boot	12
4.3 烧写 ulmage.....	13
4.4 烧写文件系统.....	13
4.5 EMMC 启动参数设置.....	13
5 烧写镜像到 UFS.....	14
5.1 地址空间说明.....	14
5.2 烧写 u-boot	14



5.3 烧写 ulmage.....	14
5.4 烧写文件系统.....	15
5.5 UFS 启动参数设置	15



表格目录

表 1-1 <ddr_addr>平台差异对照表.....	5
------------------------------	---



1 概述

1.1 概述

- 如果待烧写单板中已有 U-boot 运行，则可以通过串口或网口与服务器连接，直接更新 U-boot。
- 如果是第一次烧写，则需要使用 Hitoool 工具或者 DS-5 工具进行烧写。具体操作方法请参考《HiBurn 工具使用指南》及《Hi35xxVxx U-boot 移植应用开发指南》。



说明

在 Hi3521A、Hi3531A、Hi3516A、Hi3536 等平台使用 Hitoool 工具烧写时，具体操作方法请参考《HiTool 工具使用说明》。

不同平台在镜像烧写过程中涉及到的内存起始地址可能不同，具体使用过程中请参考表 1-1，后续不再复述。

表 1-1 <ddr_addr>平台差异对照表

内存起始地址<ddr_addr>	平台
0x42000000	Hi3531A、Hi3531DV100、Hi3536、Hi3559AV100、Hi3559CV100
0x82000000	Others

1.2 网络配置

在串口终端烧写镜像时，使用 tftp 服务软件下载镜像文件需要保证 u-boot 下的网络处于连接状态。

U-boot 下的网络配置如下：

- 配置子网掩码、物理地址和 ip 地址（板端 ip 和本地 PC 端 ip）
setenv netmask xxx.xxx.xxx.xxx;
setenv ethaddr xx:xx:xx:xx:xx:xx;
setenv ipaddr xxx.xxx.xxx.xxx;



```
setenv serverip xxx.xxx. xxx. xxx;
```

- 设置缺省网关

```
setenv gatewayip xxx.xxx.xxx.xxx;
```

- ping 通本地 PC 端 ip

```
ping xxx.xxx.xxx.xxx;
```



说明

设置的地址只是一个举例说明，具体的地址设置要根据具体使用的 ip 来配置。



2 烧写镜像到 SPI Nor Flash

2.1 地址空间说明

本文以 32M SPI Nor Flash 为例。以下的操作均基于图示的地址空间分配，您也可以根据实际情况进行调整。

1M	4M	15M
u-boot	kernel	rootfs

2.2 烧写 u-boot

步骤 1. 在内存中运行起来之后在串口终端中输入：

```
mw.b <ddr_addr> 0xff 0x100000 /* 对内存初始化*/  
tftp <ddr_addr> u-boot-hi35xx.bin /*U-boot 下载到内存*/  
sf probe 0 /*探测并初始化 SPI flash*/  
sf erase 0x0 0x100000 /*擦除 1M 大小*/  
sf write <ddr_addr> 0x0 0x100000 /*从内存写入 SPI NOR Flash*/
```



注意

SPI Nor Flash 烧写时必须先对内存初始化。

步骤 2. 重启系统可以看到 U-boot 烧写成功。

----结束



2.3 烧写 ulmage

在内存中运行起来之后在串口终端中输入：

```
mw.b <ddr_addr> 0xff 0x400000    /* 对内存初始化*/
tftp <ddr_addr> ulmage_hi35xx    /*ulmage 下载到内存*/
sf probe 0                      /*探测并初始化 SPI flash*/
sf erase 0x100000 0x400000      /*擦除 4M 大小*/
sf write <ddr_addr> 0x100000 0x400000 /*从内存写入 SPI NOR Flash*/
```

2.4 烧写文件系统

在内存中运行起来之后在串口终端中输入：

```
mw.b <ddr_addr> 0xff 0xf00000    /* 对内存初始化*/
tftp <ddr_addr> rootfs_hi35xx_xk.jffs2 /*文件系统下载到内存*/
sf probe 0                      /*探测并初始化 SPI flash*/
sf erase 0x500000 0xf00000      /*擦除 15M 大小*/
sf write <ddr_addr> 0x500000 0xf00000 /*从内存写入 SPI NOR Flash*/
```



说明

文件系统 rootfs_hi35xx_xk.jffs2 中 xk 表示 Flash 器件的 blocksize，在实际使用过程中请根据 Flash 器件的 blocksize 选择对应的文件系统。

2.5 SPI Nor Flash 启动参数设置

步骤 1. 设置 bootargs：

```
setenv bootargs 'mem=64M console=ttyAMA0,115200 root=/dev/mtdblock2
rootfstype=jffs2 rw mtdparts=hi_sfc:1M(boot),4M(kernel),15M(rootfs)';
saveenv
```

步骤 2. 设置 bootcmd:

```
setenv bootcmd 'sf probe 0;sf read <ddr_addr> 100000 400000;bootm
<ddr_addr>'; saveenv
```

----结束



3 烧写镜像到 SPI NAND 或 NAND Flash

3.1 地址空间说明

SPI Nand Flash 与 NAND Flash 烧写方法及启动参数设置方法一致。

本文以 64M NAND Flash 为例。以下的操作均基于图示的地址空间分配，您也可以根据实际情况进行调整。

1M	4M	32M	16M
u-boot	kernel	rootfs	other

3.2 烧写 u-boot

步骤 1. 在内存中运行起来之后在串口终端中输入：

```
nand erase 0x0 0x100000      /*擦除 1M 大小*/  
mw.b <ddr_addr> 0xff 0x100000 /* 对内存初始化*/  
tftp <ddr_addr> u-boot-hi35xx.bin /*U-boot 下载到内存*/  
nand write <ddr_addr> 0x0 0x100000 /*从内存写入 Flash*/
```

步骤 2. 重启系统可以看到 U-boot 烧写成功。

----结束

3.3 烧写 ulmage

在内存中运行起来之后在串口终端中输入：

```
mw.b <ddr_addr> 0xff 0x400000 /* 对内存初始化*/ tftp  
<ddr_addr> ulmage_hi35xx /*ulmage 下载到内存*/  
nand erase 0x100000 0x400000 /*擦除 4M 大小*/
```



```
nand write <ddr_addr> 0x100000 0x400000 /*从内存写入 Flash*/
```

3.4 烧写文件系统

3.4.1 烧写 yaffs2 文件系统

在内存中运行起来之后在串口终端中输入：

```
mw.b <ddr_addr> 0xff 0x2000000 /* 对内存初始化*/
tftp <ddr_addr> rootfs_hi35xx_xk_xxbit.yaffs2 /*文件系统下载到内存*/
nand erase 0x500000 0x2000000 /*擦除 32M 大小*/
nand write.yaffs <ddr_addr> 0x500000 $(filesize) /*从内存写入 Flash*/
```



说明

- 文件系统 rootfs_hi35xx_xk_xxbit.yaffs2 中 xk 表示 pagesize，xxbit 表示 ecc，在实际使用过程中请根据 Flash 器件选择对应的文件系统。
- 在烧写 yaffs 文件系统时，从内存写入 flash 必须使用 nand write.yaffs，而不能使用 nand write。

3.4.2 烧写 ubifs 文件系统

在内存中运行起来之后在串口终端中输入：

```
mw.b <ddr_addr> 0xff 0x2000000 /* 对内存初始化*/
tftp <ddr_addr> rootfs_hi35xx_xk_xxk_32M.ubifs /*文件系统下载到内存*/
nand erase 0x500000 0x2000000 /*擦除 32M 大小*/
nand write <ddr_addr> 0x500000 $(filesize) /*从内存写入 Flash*/
```



说明

文件系统 rootfs_hi35xx_xk_xxk_32M.ubifs 中 xk 表示 Flash 器件的 Pagesize，xxk 表示 Flash 器件的 Blocksize，32M 表示 ubifs 文件系统分区的大小，在实际使用过程中请根据 Flash 器件选择对应的文件系统。

3.5 NAND Flash 启动参数设置

3.5.1 yaffs2 文件系统启动参数设置

步骤 1. 设置 bootargs：

```
setenv bootargs 'mem=128M console=ttyAMA0,115200 root=/dev/mtdblock2 rw
rootfstype=yaffs2
mtdparts=hinand:1M(boot),4M(kernel),32M(rootfs)';saveenv
```

步骤 2. 设置 bootcmd:

```
setenv bootcmd 'nand read <ddr_addr> 0x100000 0x400000;bootm
```



```
<ddr_addr>;saveenv
```

3.5.2 ubifs 文件系统启动参数设置

步骤 1. 设置 bootargs :

```
setenv bootargs 'mem=128M console=ttyAMA0,115200 ubi.mtd=2  
root=ubi0:ubifs rootfstype=ubifs rw  
mtdparts=hinand:1M(boot),4M(kernel),32M(ubifs)'; saveenv
```

步骤 2. 设置 bootcmd:

```
setenv bootcmd 'nand read <ddr_addr> 0x100000 0x400000;bootm  
<ddr_addr>;saveenv
```



4 烧写镜像到 eMMC

4.1 地址空间说明

本文以 32M EMMC 为例。以下的操作均基于图示的地址空间分配，您也可以根据实际情况进行调整。

1M	4M	32M
u-boot	kernel	rootfs

4.2 烧写 u-boot

步骤 1. 在内存中运行起来之后在串口终端中输入：

```
mw.b <ddr_addr> 0xff 0x100000      /* 对内存初始化*/  
tftp <ddr_addr> u-boot-hi35xx.bin    /*U-boot 下载到内存*/  
mmc write 0 <ddr_addr> 0x0 0x800    /*从内存写入 EMMC*/
```



注意

- EMMC 烧写命令中的地址和长度是以 Block 为单位的，每个 block 长度为 512Byte，所以烧写 1MB 的 u-boot 时，0x0-0x800 表示从地址 0 开始写，长度为 1MB。
- mmc write 0 中 0 表示片选，在 EMMC 启动时，默认 0 片选为 eMMC。

步骤 2. 重启系统可以看到 U-boot 烧写成功。

-----结束



4.3 烧写 ulmage

在内存中运行起来之后在串口终端中输入：

```
mw.b <ddr_addr> 0xff 0x400000 /* 对内存初始化*/ tftp  
<ddr_addr> ulmage_hi35xx /*ulmage 下载到内存*/  
mmc write 0 <ddr_addr> 0x800 0x2000 /*从内存写入 EMMC*/
```

4.4 烧写文件系统

在内存中运行起来之后在串口终端中输入：

```
mw.b <ddr_addr> 0xff 0x2000000 /* 对内存初始化*/  
tftp <ddr_addr> rootfs_hi35xx_32M.ext4 /*文件系统下载到内存*/  
mmc write.ext4sp 0 <ddr_addr> 0x2800 0x10000 /*从内存写入 EMMC*/
```



说明

在烧写 EXT4 文件系统时，从内存写入 EMMC 必须使用 `mmc write.ext4sp`，而不能使用 `mmc write`。

4.5 eMMC 启动参数设置

步骤 1. 设置 bootargs：

```
setenv bootargs 'mem=64M console=ttyAMA0,115200 root=/dev/mmcblk0p3  
rootfstype=ext4 rw rootwait  
blkdevparts=mmcblk0:1M(boot),4M(kernel),32M(rootfs)';saveenv
```

步骤 2. 设置 bootcmd:

```
setenv bootcmd 'mmc read 0 <ddr_addr> 0x800 0x2000;bootm  
<ddr_addr>';saveenv
```

----结束



5 烧写镜像到 UFS

5.1 地址空间说明

本文以 96M UFS 文件系统为例。以下的操作均基于图示的地址空间分配，您也可以根据实际情况进行调整。

1M	9M	96M
u-boot	kernel	rootfs

5.2 烧写 u-boot

在内存中运行起来之后在串口终端中输入：

```
mw.b <ddr_addr> 0xff 0x100000 /* 对内存初始化*/  
tftp <ddr_addr> u-boot-hi35xx.bin /*U-boot 下载到内存*/  
ufs write 0 <ddr_addr> 0x0 0x100 /*从内存写入 UFS*/
```



注意

- UFS 烧写命令中的地址和长度是以 block 为单位的，每个 block 的大小是 4096Byte，所以烧写 1MB 的 u-boot 时，0x0-0x100 表示从地址 0 开始写，长度为 1MB。
- ufs write 0 中 0 表示片选，在 UFS 启动中，默认 0 片选为 UFS。

5.3 烧写 ulmage

在内存中运行起来之后在串口终端中输入：

```
mw.b <ddr_addr> 0xff 0x900000 /* 对内存初始化*/
```




```
tftp <ddr_addr> ulmage_hi35xx /*ulmage 下载到内存*/  
ufs write 0 <ddr_addr> 0x100 0x900 /*从内存写入 UFS*/
```

5.4 烧写文件系统

在内存中运行起来之后在串口终端中输入：

```
mw.b <ddr_addr> 0xff 0x6000000 /* 对内存初始化*/  
tftp <ddr_addr> rootfs_hi35xx_96M.ext4 /*文件系统下载到内存*/  
ufs write.ext4sp 0 <ddr_addr> 0xa00 0x6000 /*从内存写入 UFS*/
```



说明

在烧写 EXT4 文件系统时，从内存写入 UFS 必须使用 ufs write.ext4sp，而不能使用 ufs write。

5.5 UFS 启动参数设置

步骤 1. 设置 bootargs：

```
setenv bootargs 'mem=128M console=ttyAMA0,115200 root=/dev/sdd3  
rootfstype=ext4 rootwait blkdevparts=sdd:1M(boot),9M(kernel),96M(rootfs)';  
saveenv
```

步骤 2. 设置 bootcmd:

```
setenv bootcmd 'ufs read 0x0 <ddr_addr> 0x100 0x900;bootm  
<ddr_addr>;saveenv
```

----结束