参考：<http://blog.csdn.net/jxhui23/article/details/8113217>

<http://www.cnblogs.com/mylinux/p/4180577.html>（从启动代码的角度）

一：NOR FLASH和NOR FLASH的区别

Flash编程原理都是只能将1写为0，而不能将0写成1.所以在Flash编程之前，必须将对应的块擦除，而擦除的过程就是将所有位都写为1的过程，块内的所有字节变为0xFF.因此可以说，编程是将相应位写0的过程，而擦除是将相应位写1的过程，两者的执行过程完全相反.

如何判断nand启动和nor启动：[1]

CPU从0x00000000位置开始运行程序。

1、nand启动：

如果将S3C2440配置成从NANDFLASH启动（将开发板的启动开关拔到nand端，此时OM0管脚拉低）S3C2440的Nand控制器会自动把Nandflash中的前4K代码数据搬到内部SRAM中(地址为0x40000000),同时还把这块SRAM地址映射到了0x00000000地址。CPU从0x00000000位置开始运行程序。

2、如果将S3C2440配置成从Norflash启动（将开发的启动开关拔到nor端，此时OM0管脚拉高），0x00000000就是norflash实际的起始地址，norflash中的程序就从这里开始运行，不涉及到数据拷贝和地址映射

3、总结：

nand启动时,地址0x00000000为内部SRAM映射的地址；

nor启动时，地址0x00000000为norflash的实际起始地址。

向Norflash中写数据需要特定的命令时序，而向内存中写数据可以直接向内存地址赋值。

二：FLASH LOCK和UNLOCK的应用场景

对flash进行写操作时要关闭所有中断

1>什么是FLASH LOCK和FLASH UNLOCK

为了更好地保护用户的程序以及用户的配置信息，添加了闪存锁功能。通过该闪存锁，用户可以更好地保护自己的程序不被第三方读出，从而保证自己的产品不会被第三方复制。

2>nand flash中的坏块

出现的情况：

a.出厂时的坏块

b.操作过程中由擦除失败造成的

c.擦除过程中写入操作失败引起的

d.出现超出ECC校验算法纠正能力的错误时，也认为出现了坏块

坏块管理机制：

坏块管理机制通过分配一个好块来存储当前坏块的数据并进行地址重映射，寻找新块的方法有下面两种：

a.skip block method:出现坏块，跳过，寻找下一个最近的好块，将数据存储在该好块上。这时该好块对应的坏块的信息也得存储在nandflash中

b.reserved bolock method:遇到坏块，不跳转，通过修改逻辑地址与物理地址的映射关系，将逻辑地址映射到一个已知的好块，达到替换的目的。

此时坏块管理软件将nandflash分为可寻址块区和保留区(一般保留区为nandflash的2%)。

c.动态坏块管理设计思想：

对nandflash进行操作时，出现坏块，采用动态坏块算法，标记坏块，将坏块放进坏块映射表的坏块区，从系统预留的坏块区中选择一个好块替代坏块进行操作；同时更新逻辑物理地址映射表。

1. 将nandflash存储地址映射空间划分为：

1.数据区：用于存储数据，系统应用程序，均为好块

2.交换块：所有已用块的写入操作均改为对交换块的写入，交换块同时起着更新映射表时，防止掉电发生，保存原始映射表，在更新nandflash中的映射表完成后，再将旧表擦除的作用，可以有效应对掉电状态，不至于掉电时发生信息丢失的情况，交换个数一般为一两个，或更多。

3.替换块：替换坏块

4.坏块区：用于存放初次上电扫描发现的坏块及使用过程中出现的坏块