Technical Note

Bad Block Management in NAND Flash Memory

Introduction

This document explains how to recognize factory-generated bad blocks to manage bad blocks that develop during the lifetime of NAND Flash memory. It covers all Micron® NAND Flash memory devices. Refer to the relative data sheets for the full list of root part numbers and for further information on the devices.

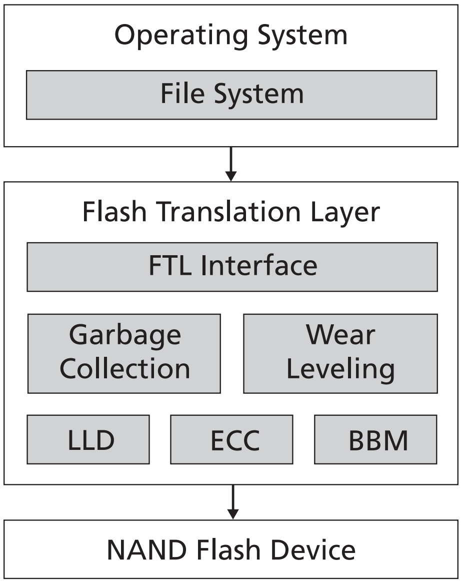
About Bad Blocks

Bad blocks are blocks that contain one or more invalid bits whose reliability is not guaranteed. Bad blocks may be present when the device is shipped, or may develop during the lifetime of the device.

Devices with bad blocks have the same quality level and the same AC and DC characteristics as devices where all the blocks are valid. A bad block does not affect the performance of valid blocks because it is isolated from the bit line and common source line by a select transistor.

Bad block management, block replacement, and the error correction code (ECC) software are necessary to manage the error bits in NAND Flash devices.

Figure 1: Software Tool Chain



Recognizing Bad Blocks

NAND Flash devices are supplied with all the locations inside valid blocks erased (FFh). The bad block information is written prior to shipping.

For single-level cell (SLC) small page (528-byte/256-word page) devices, any block where the sixth byte (x8 device)/first word (x16 device), in the spare area of the first page does not contain FFh is a bad block.

For SLC large page (2112-byte/1056-word page) devices, any block where the first and sixth bytes (x8 device)/first word (x16 device) in the spare area of the first page does not contain FFh is a bad block.

For SLC very large page (4224-byte page) devices, any block where the first and sixth bytes in the spare area of the first page does not contain FFh is a bad block.

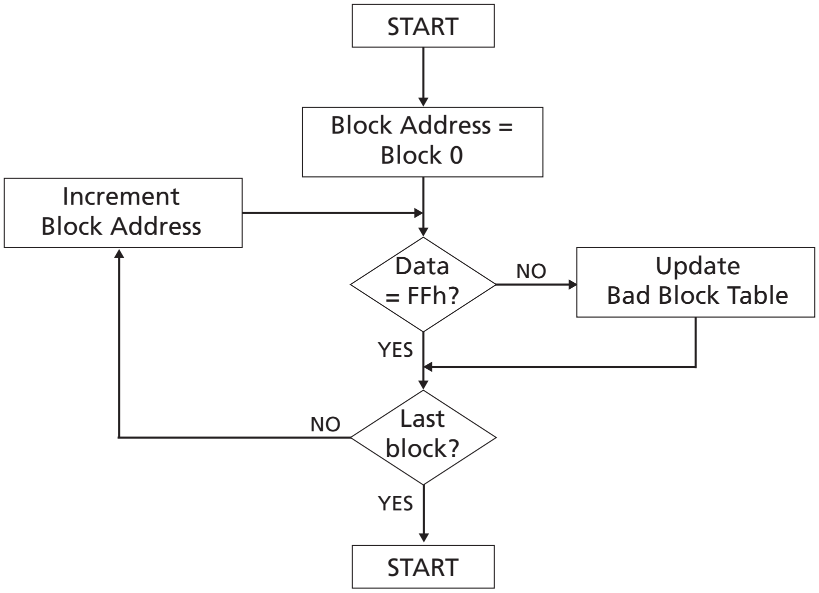
For multilevel cell (MLC) devices, any block where the first byte in the spare area of the last page does not contain FFh is a bad block.

The bad block information must be read before any erase is attempted because the bad block information is erasable and cannot be recovered once erased. It is highly recommended to not erase the original bad block information. To allow the system to recognize the bad blocks based on the original information, it is recommended to implement the bad block management algorithm shown in Figure 1.

The bad block table is created by reading all the spare areas in the NAND Flash memory. The bad block recognition methods that build the bad block table without using the original bad block information provided in the spare areas of the memory are not equally effective. The invalid blocks are detected at the factory during the testing process, which involves severe environmental conditions and PROGRAM/ERASE cycles as well as proprietary test modes. The failures that affect invalid blocks may not all be recognized if methods different from those implemented in the factory are used.

Once created, the bad block table is saved to a good block so that on rebooting the NAND Flash memory the bad block table is loaded into RAM. The blocks contained in the bad block table are not addressable. So, if the flash translation layer (FTL) addresses one of the bad blocks, the bad block management software redirects it to a good block.

Figure 2: Bad block Management Flow Chart



Block Replacement

During the lifetime of the NAND device additional bad blocks may develop. NAND devices have a status register that indicates whether an operation is successful. Additional bad blocks are identified when attempts to PROGRAM or ERASE give errors in the status register.

As the failure of a PAGE PROGRAM operation does not affect the data in other pages in the same block, the block can be replaced by reprogramming the current data and copying the rest of the replaced block to an available valid block.

Blocks can be marked as bad and new blocks allocated using two general methods:

- Skip block
- Reserve block

Skip Block Method

In the skip block method the algorithm creates the bad block table and when the target address corresponds to a bad block address, the data is stored in the next good block, skipping the bad block.

When a bad block is generated during the lifetime of the NAND Flash device, its data is also stored in the next good block. In this case, the information that indicates which good block corresponds to each developed bad block must also be stored in the NAND Flash device.

Reserve Block Method

In the reserve block method, the bad block table is created in the same manner as described in Figure 2. In this method bad blocks are not skipped but replaced by good blocks by redirecting the FTL to a known free good block. For that purpose, the bad block management software creates two areas in the NAND Flash: The user addressable block area and the reserved block area as shown in Figure 3.

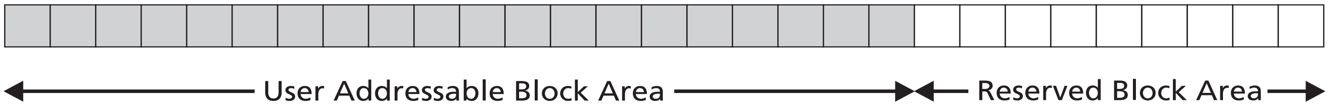
The FTL can use the **user addressable block area** to store data whereas the **reserved block area** is only used for bad block replacement and to save the bad block table, which also keeps track of the remapped developed bad blocks.

To define these two areas, determine the start address and the size of the reserved area. The size may either be given by the user or imposed by the bad block management software (for Micron NAND Flash devices, the maximum number of blocks that may become bad during the device's lifetime is 2% of the total, and as a result the same number of blocks are commonly reserved).

Each time the FTL writes a logical sector, it calculates the physical address of the block to which it will write. Then, before the FTL starts writing, the bad block management software checks whether the block is bad or not. If it is bad, it returns the address of the good block to which the sector is remapped. If the block becomes bad during the NAND Flash lifetime, the bad block management software remaps the bad block and copies the data it contains to the block that will replace it.

Bad block management is completely transparent to the FTL. For the FTL it is as if the data are written to the same address.

Figure 3: Reserved Block Method



References

- Garbage Collection in NAND Flash Memory technical note
- Wear Leveling in NAND Flash Memory technical note
- Error Correction Code in Single-Level Cell NAND Flash Memory technical note

Conclusion

Micron recommends the implementation of bad block management to detect factory produced bad blocks and manage any bad blocks that may develop over the lifetime of a NAND Flash device. It is also recommended that garbage collection and wear leveling algorithms be implemented. It is mandatory to implement ECC algorithms.

To help the integration of NAND memory devices into applications, Micron provides a full range of software solutions, such as:

- File system
- Sector manager
- Drivers
- Code management.

For more details, contact the your Micron representative or visit micron.com.

8000 S. Federal Way, P.O. Box 6, Boise, ID 83707-0006, Tel: 208-368-3900
www.micron.com/productsupport Customer Comment Line: 800-932-4992