Technical Note

Wear-Leveling Techniques in NAND Flash Devices

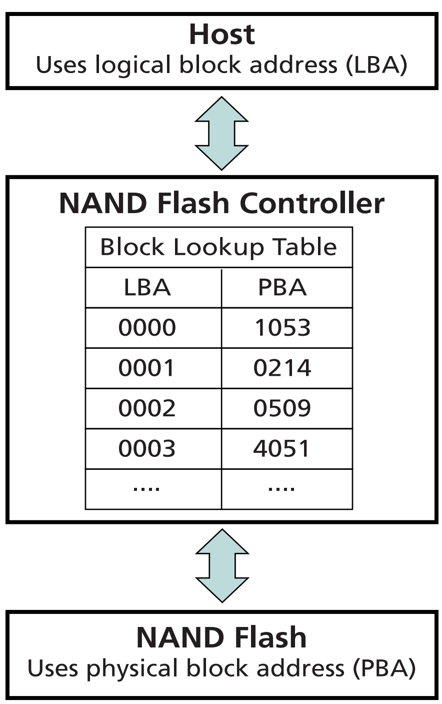
Introduction

Wear leveling is a process that helps reduce premature wear in NAND Flash devices. This technical note highlights the importance of wear leveling, explains two primary wear-leveling techniques—static and dynamic—and calls attention to other considerations involved in implementing wear leveling.

Wear Leveling Implementation

The most common implementation of wear leveling occurs in the NAND Flash controller, which manages access to the memory device and determines how the NAND Flash blocks are used. In most cases, the controller maintains a lookup table to translate the memory array physical block address (PBA) to the logical block address (LBA) used by the host system (see Figure 1). The controller's wear-leveling algorithm determines which physical block to use each time data is programmed, eliminating the relevance of the physical location of data and enabling data to be stored anywhere within the memory array.

Figure 1: NAND Flash Controller Block Address Management



Depending on the wear-leveling method used, the controller typically either writes to the available erased block with the lowest erase count (dynamic wear leveling); or it selects an available target block with the lowest overall erase count, erases the block if necessary, writes new data to the block, and ensures that blocks of static data are moved when their block erase count is below a certain threshold (static wear leveling).

The Importance of Wear Leveling

The need for wear leveling results from the finite PROGRAM/ERASE cycling capability of NAND Flash memory cells. The repeated use of a limited number of blocks can cause the device to prematurely wear out or exceed its program/erase endurance. The wear-leveling process spreads NAND Flash memory cell use over the available memory array, ideally equalizing the use of all memory cells and helping to extend device life.

Consider a case without wear leveling. In a NAND Flash device with 4,096 total blocks and 2.5% allowable bad blocks in a system that updates 3 files comprised of 50 blocks each at a rate of 1 file every 10 minutes (or 6 files per hour), where a NAND host reuses the same 200 physical blocks for these updates, the NAND Flash device will wear out in under 1 year, leaving over 95% of the memory array unused.

For example:

$$\text{Only 200 blocks are reused: } \frac{\text{10,000 cycles x 200 blocks}}{\text{50 blocks per file x 6 files per hour x 24 hours per day}} = \text{ ~278 days or <1 year}$$

In contrast, in a situation where all blocks are used equally by programming and erasing 4,096 of them at a rate of 50 blocks every 10 minutes (or 6 files per hour), even distribution of wear across all blocks dramatically increases the useful life of the device.

For example:

$$\text{All 4,096 blocks are evenly used: } \frac{\text{10,000 cycles x 4,096 blocks}}{\text{50 blocks per file x 6 files per hour x 24 hours per day}} = \text{ ~5,689 days or >15 years}$$

Wear leveling not only extends the life of a Flash device, it also enables the memory capacity to be used more efficiently. Reusing only 200 blocks results in programming and erasing about 2 million effective blocks over the life of the device, while evenly using all 4,096 blocks results in programming and erasing more than 40 million effective blocks.

Wear-Leveling Methods

Wear leveling can be implemented using several methods, each of which adds complexity, but increases the life of NAND Flash devices. Depending on the size of the memory area used for wear-leveling purposes relative to the total available memory, a lack of wear leveling can dramatically reduce the useful life of the NAND Flash device.

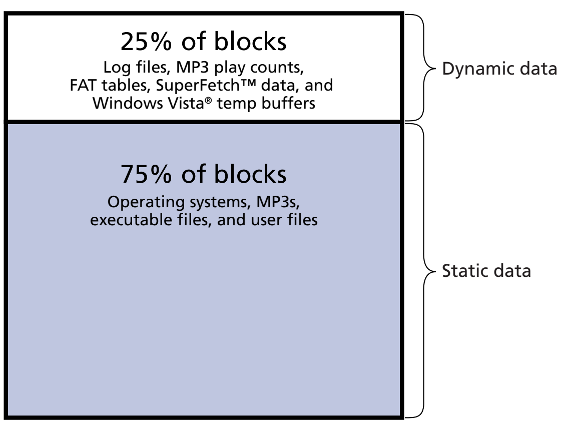
Two types of data exist in NAND Flash devices: static and dynamic. Static data is information that is rarely, if ever, updated. It may be read frequently, but it seldom changes and can theoretically reside in the same physical location for the life of the device. Dynamic data, on the other hand, is constantly changing and consequently requires frequent reprogramming.

Implementing Dynamic Wear Leveling

Dynamic wear leveling is a method of pooling the available blocks that are free of data and selecting the block with the lowest erase count for the next write. This method is most efficient for dynamic data because only the nonstatic portion of the NAND Flash array is wear-leveled. A system that implements dynamic wear leveling enables longer NAND Flash device life than a system that does not implement wear leveling.

For instance, in a device with a 25%/75% split of dynamic data versus static data, respectively, dynamic wear leveling targets the 25% of the blocks of dynamic memory area, while the other 75% of the blocks remain idle with static data. In this case, 25% of the available blocks are used to their maximum cycle count (see Figure 2).

Figure 2: NAND Flash Data Distribution



In a 4,096-block MLC device with a 10,000-cycle count, 75% static data, and a program and erase rate of 50 blocks every 10 minutes (or 6 files per hour), dynamic wear leveling results in device wear-out after approximately 4 years, with 75% of the blocks nearly unused (see Figure 3 on page 4).

For example:

$$\text{Wear leveling only dynamic data:} \frac{10{,}000 \text{ cycles x } 1{,}024 \text{ blocks}}{50 \text{ blocks per file x } 6 \text{ files per hour x } 24 \text{ hours per day}} = \sim 1{,}422 \text{ days or } <4 \text{ years}$$

Implementing Static Wear Leveling

Static wear leveling utilizes all good blocks to evenly distribute wear, providing effective wear leveling and thereby extending the life of the device. This method tracks the cycle count of all good blocks and attempts to evenly distribute block wear throughout the entire device by selecting the available block with the least wear each time a program operation is executed. Static data is managed by maintaining all blocks within a certain erase count threshold. Blocks that contain static data with erase counts that begin to lag behind other blocks will be included in the wear-leveling block pool, with the static data being moved to blocks with higher erase counts.

Although the additional step of moving static data to free up space in low erase count blocks can slow write performance (because it requires additional controller overhead) and can consume some block life, overall, static wear leveling is the best method for maximizing the life of a NAND device.

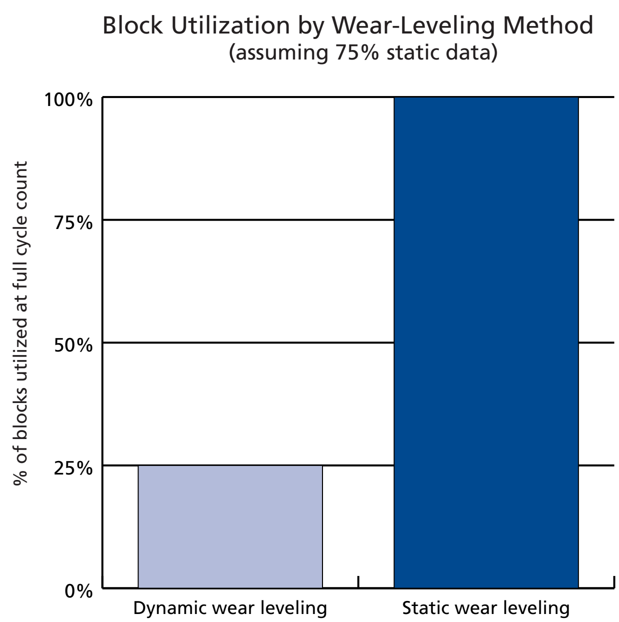
Using the same example of a 4,096-block MLC device with a 10,000-cycle count, 75% static data, and a program and erase rate of 50 blocks every 10 minutes (or 6 files per hour), static wear leveling provides the best chance of extending the device life span beyond 15 years.

For example:

$$\text{Wear leveling static and dynamic data: } \frac{10{,}000 \text{ cycles x } 4{,}096 \text{ blocks}}{50 \text{ blocks per file x } 6 \text{ files per hour x } 24 \text{ hours per day}} = \sim 5{,}689 \text{ days or } >15 \text{ years}$$

The drastic difference in block utilization between dynamic and static wear leveling is illustrated in Figure 3.

Figure 3: Static vs. Dynamic Block Utilization



Choosing Wear-Leveling Methods

Choosing a wear-leveling method involves evaluating the advantages and disadvantages of each method (see Table 1).

Table 1: Static vs. Dynamic Wear-Leveling Methods

Wear-Leveling Method	Advantages	Disadvantages
Static	Maximizes device life Most robust wear-leveling method Most efficient use of memory array	Requires more controller overhead Can slow WRITE operations Higher power consumption More complicated to implement than dynamic wear leveling
Dynamic	Improves device life over no wear leveling at all Easier to implement than static wear leveling No impact on device performance	May not optimize device life

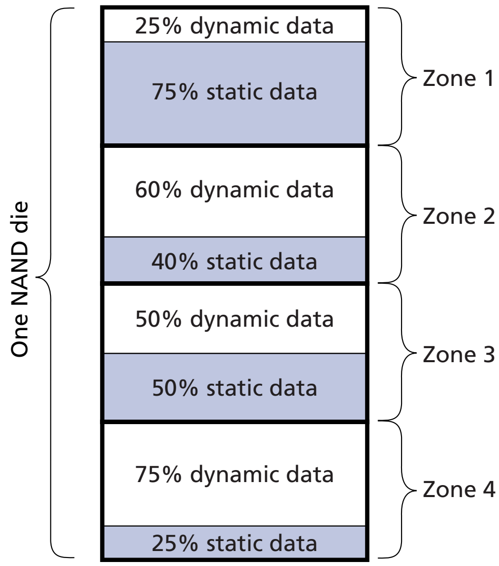
Wear-Leveling Considerations in Die Memory Arrays

Impact of Zoning

Some controllers use a zoning approach, in which wear leveling is segmented into subsets of the total die memory array. Because wear leveling is contained within each zone, if one zone wears out sooner than the remaining zones, the entire die prematurely wears out before all blocks reach their maximum cycling count. Even when static wear leveling is used, this approach can result in a shorter device life span.

If dynamic wear leveling is applied in a die where zone 1 contains 25% dynamic data and 75% static data, only 25% of the blocks in this zone will be used for wear leveling (see Figure 4 on page 6). Zone 1 will likely become worn out sooner than the remaining 3 zones, causing the die to wear out before all blocks reach their maximum cycle count.

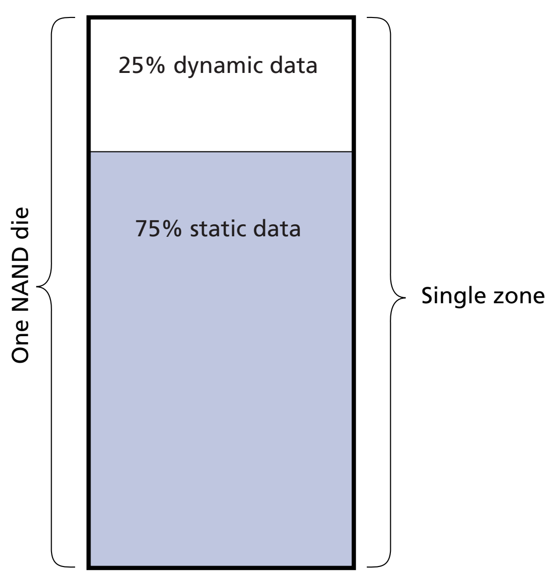
Figure 4: Data Distribution Using Multiple Zones



Even though static wear leveling helps ensure equal block use within each zone if the controller treats each zone separately, different zones might receive different levels of use. As such, one zone is likely to wear out before another, which would cause the die to wear out prematurely.

The most effective approach is to encompass all good blocks in the memory array into a single zone to help ensure equal wear leveling. Micron recommends using a single zone to maximize the life of the die.

Figure 5: Data Distribution Using a Single Zone (Preferred Method)



Wear Leveling Across Multiple Die

In devices with multiple die, wear leveling should be applied across all die to ensure equal use of blocks and avoid premature wear-out. The best implementation of wear leveling in devices with multiple die is to consider the wear-leveling pool to be all the good blocks within the device. This helps ensure that a single die does not wear out before the remaining die, which enables the device to reach maximum operating life.

For example, in a solid state drive (SSD) application with many NAND devices, if wear leveling is not completed across all devices, it is highly likely that some devices may experience greater wear and wear out more quickly. These prematurely worn blocks (or entire devices) can cause the SSD to operate as read-only before all blocks and/or devices are fully used. The proper implementation of wear leveling can ensure that SSDs continue to function to the full extent of their maximum available useful life.

Conclusion

Wear leveling can help extend the useful life of NAND Flash devices and is often necessary to ensure that the devices reach the specified endurance rating by equalizing the wear of good blocks. The use of wear-leveling techniques is imperative in NAND Flash devices, regardless of the individual device's endurance rating.

The most effective wear-leveling method is static wear leveling because it typically provides more uniform block usage than dynamic wear leveling. Although dynamic wear leveling is typically inferior to static wear leveling, this method is easier to implement and can still provide enough wear leveling to meet the needs of many applications.

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

Revision History

Rev. A 10/08

- Initial release