Cactus® Technologies

Wear Leveling – Static, Dynamic and Global

White paper CTWP013

Cactus Technologies Limited
Suite C, 15/F, Capital Trade Center
62 Tsun Yip Street, Kwun Tong
Kowloon, Hong Kong
Tel: +852-2797-2277
Email: sales@cactus-tech.com

Cactus USA
3112 Windsor Road
Suite A356
Austin, Texas 78703
Tel: +512-775-0746
Email: americas@cactus-tech.com

Cactus-Tech.com

01 Introduction

NAND flash memory has a finite number of program/erase cycles due to several defect causing mechanisms inherent in the program/erase operations. These mechanisms are:

- electrons trapped in the thin oxide layer that insulates the floating gate
- oxide breakdown due to hot carrier injection

Today's 4Xnm SLC NAND flash devices typically are guaranteed for 70,000 program/erase cycles. As the number of program/erase cycles approaches this limit, the reliability of the cells starts to decrease and will eventually become unusable, requiring the entire block to be replaced by spare blocks.

02 Wear Leveling

As the number of available spare blocks in a flash storage device is limited, special flash management techniques are used to overcome and manage the flash wear out phenomenon. One such technique is wear leveling.

Wear leveling tries to even out the distribution of program/erase operations on all available blocks in the flash drive. This is done by writing all new or updated data to a free block and then erasing the block containing old data and making it available in the free block pool. The wear leveling operation is done in the background and completely transparent to the host system.

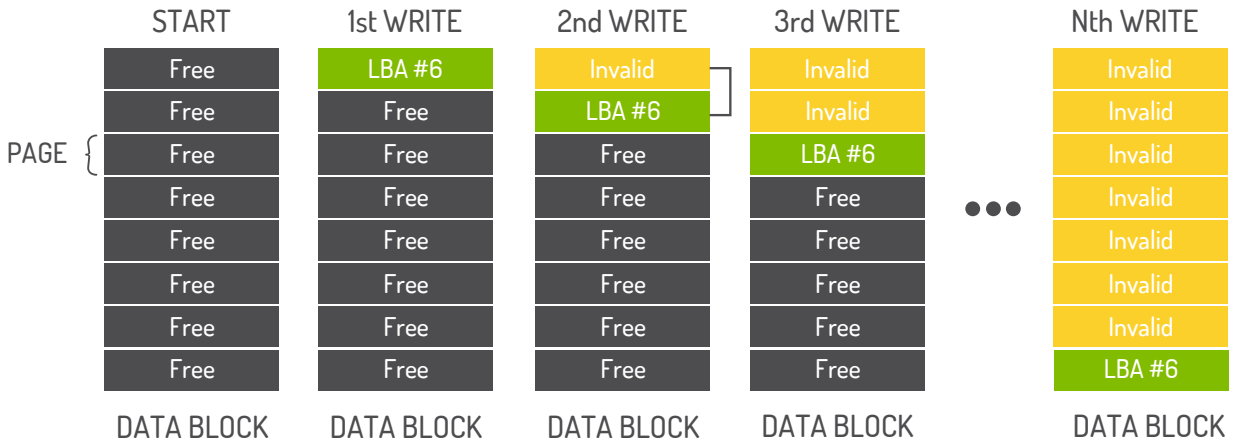
Two methods of wear leveling are used – static and dynamic.

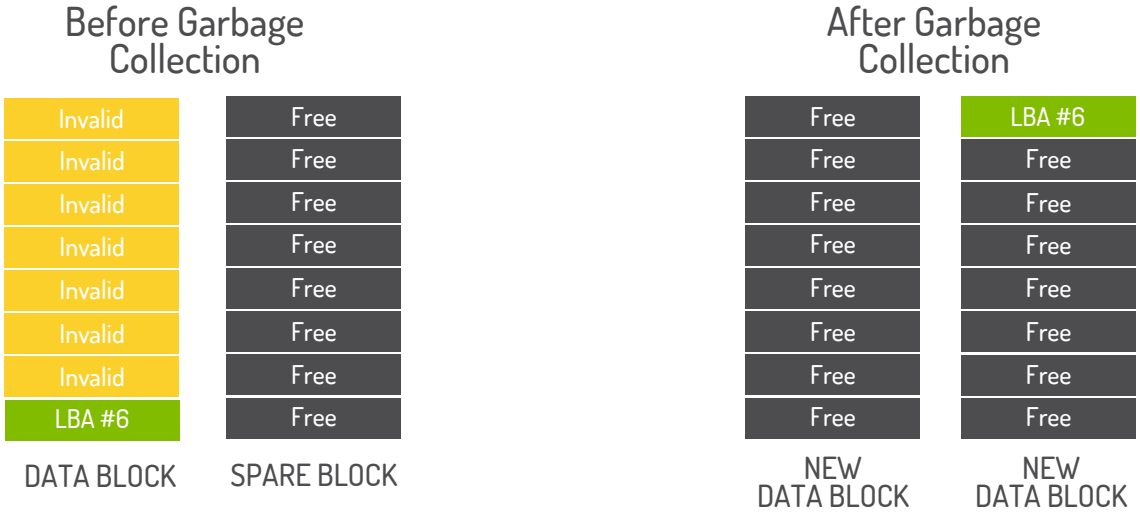
03 Dynamic Wear Leveling

Dynamic wear leveling works on data blocks that are being written to dynamically. As mentioned earlier, all new data are written to free data blocks, i.e. blocks that do not contain user data. The flash drive controller selects the new free data block based on the number of program/erase cycles that the block already has. After the new data is written, the controller then updates its internal logical to physical mapping table to point to the new physical block location. The data block containing the old data is marked invalid and is then erased and made available as a free block during garbage collection process. Dynamic wear leveling addresses the issue of repeated writes to the same blocks by redirecting new writes to different physical blocks, thus avoiding premature wear out of the actively used blocks.

The following diagram illustrates the concept of dynamic wear leveling:

The important point to note here is that if a data block is not being written to, it will not be wear leveled by this.

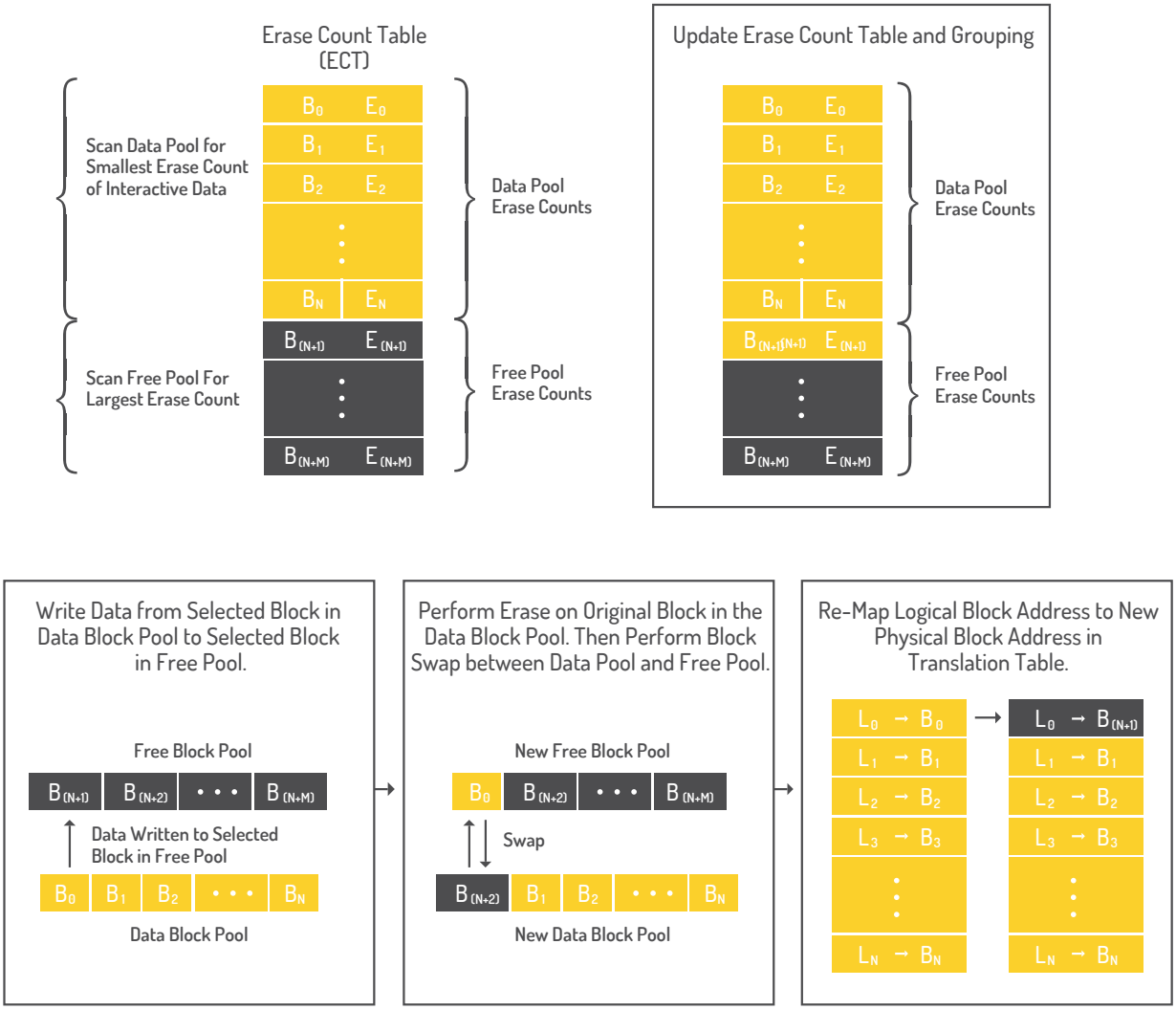




04 Static Wear Leveling

In contrast to dynamic wear leveling, static wear leveling wear levels all data blocks, including those that are not being written to. This is done in the background, completely transparent to the host system. Different vendors have different mechanisms for triggering a static wear level operating. For example, one such trigger could be the difference in program/erase counts between blocks in the static data pool and blocks in the free data pool. When this threshold is triggered, the block in the static data pool with the lowest program/erase count is swapped with the block in the free data pool with the highest program/erase count.

The following diagram illustrates a conceptual implementation of what was just described:



05 Why Static Wear Leveling

As we discussed in the previous paragraphs, dynamic wear leveling alone cannot guarantee that all data blocks are wear-leveled evenly. There are situations where a flash drive may contain large number of blocks where data is written and remain unchanged for long periods of time, e.g. operating system files. Thus, while the active data blocks are being wear-leveled dynamically, the static blocks are never touched and remains inactive in the wear level process.

The net effect of doing dynamic wear leveling only is that the program/erase cycles that are available for sharing in the static data blocks are never made available for use to the dynamic data blocks. This effectively reduces the lifespan of the flash drive. This reduction in lifespan could be significant for those cases where the static data blocks is a significant percentage of the drive capacity.

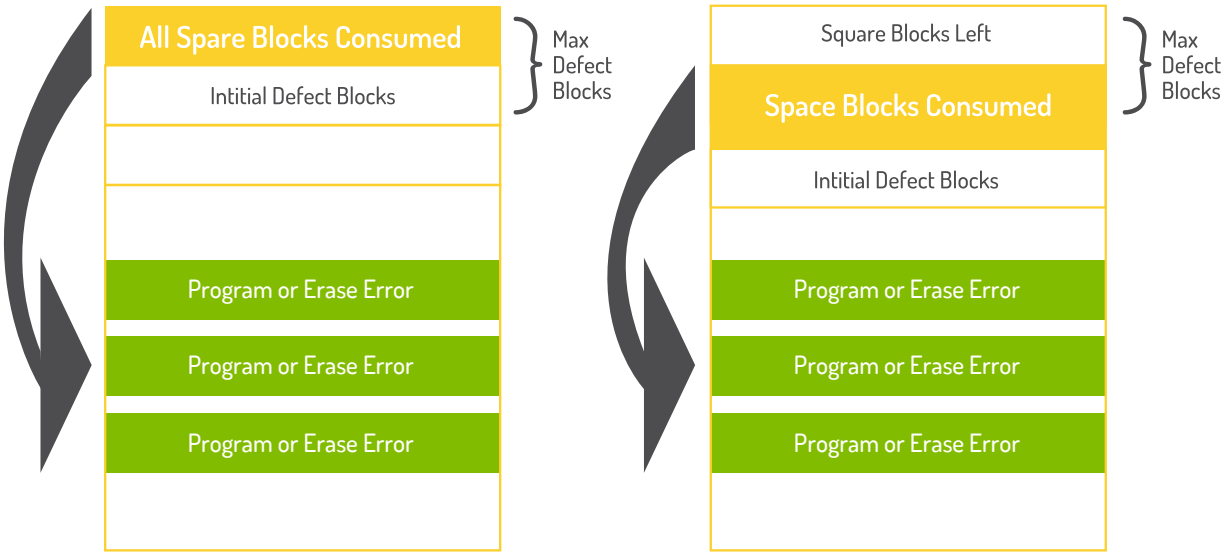
Static wear leveling solves the uneven wear-level problem by including the static data blocks in the program/erase pool. While it is true that moving the static blocks around will increase the amount of program/erase operations on the rest of the data blocks and may cause some of these blocks to fail sooner, the gain in additional product lifespan more than compensates for the small increase in blocks being retired sooner.

06 Global Wear Leveling

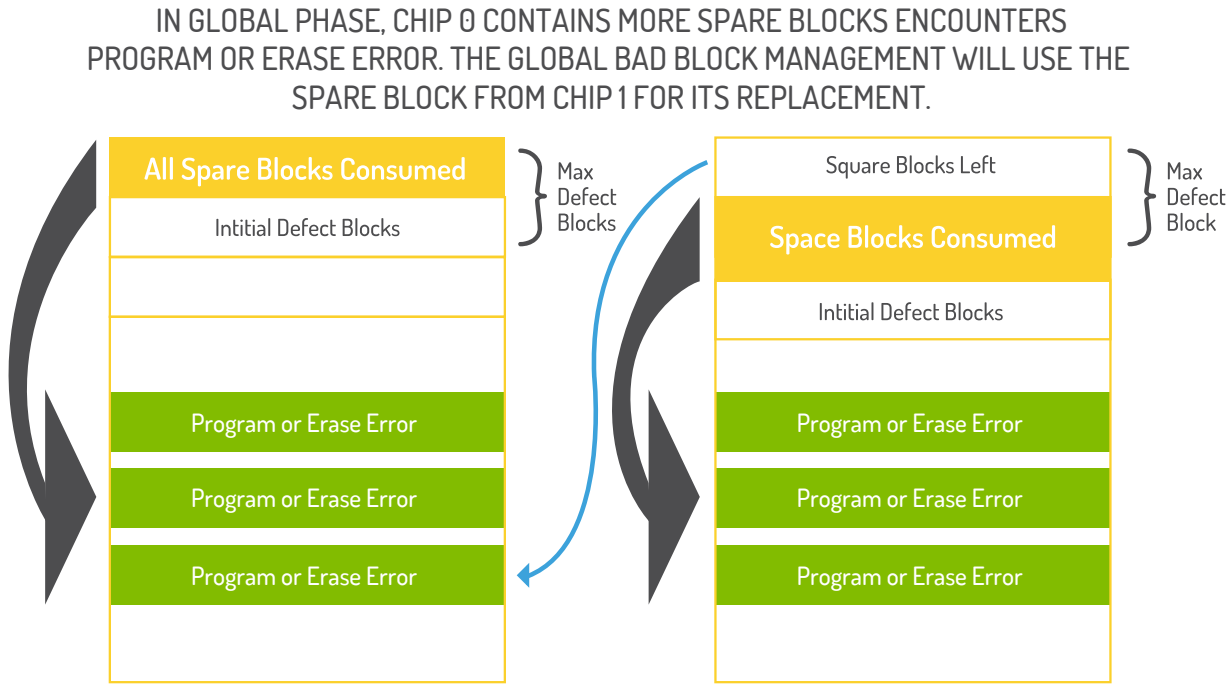
In a typical implementation of wear leveling, dynamic and static wear leveling is applied at the local level, meaning that in a multi-chip product, each chip's spare block pool is managed as a standalone resource. As flash chips have different defect block counts, it is inevitable that in a multi-chip product, one of the flash chips will use up all its spare blocks before the other chips. When this happens, the product will become write-protected to prevent any further writes from potentially causing data corruption errors. This happens even though there may be plenty of spare blocks available in the other flash chips in the product.

To remedy this situation, global wear leveling is introduced. In global wear leveling, all spare blocks in all flash chips in the product are managed together in a single pool. The following diagram shows a simple example of a two chip product in a local wear leveling setup:

IN DRIVE PHASE, DRIVE BECOMES WRITE-PROTECTED BECAUSE SPARE BLOCKS CONSUMPTION OF CHIP 0 EVEN THOUGH CHIP 1 STILL CONTAINS A FEW SPARE BLOCKS



And the following diagram shows the same product in a global wear leveling setup:



In summary, the use of dynamic and static wear leveling at the local chip level is most commonly used in order for the flash product to achieve even wear throughout the chip and to provide the best endurance and product lifespan. However, as flash technology marches towards smaller and smaller geometries, the endurance count is decreasing, thus causing the flash blocks to wear out sooner and thus using up the internal spare blocks quicker. To ensure a continued long product life, global wear leveling techniques are now used to leverage spare blocks that are available in other flash chips on the product. This technique prolongs product life at the expense of losing some performance. Cactus Technologies products utilize an intelligent algorithm that strikes the best balance of achieving long product life while maintaining good performance for as long as possible.

07 Support Information

If you would like any additional information regarding data contained in this white paper feel free to contact a Cactus representative:

Asia/Pac RIM info@cactus-tech.com

Americas americas@cactus-tech.com

EMEA info@cactus-tech.com