SSD POWER FAIL PROTECTION WHITEPAPER

Whitepaper

Document #AN0025 – Viking SSD Power Fail Protection | Rev. E

Abstract

Viking SSD's that contain MLC, TLC or QLC NAND, use special Power Fail Management (PFM) hardware, firmware and data protection mechanisms to prevent data loss or data corruption for any in-flight data being written to NAND or any in-flight DRAM write cache data, should an unexpected power fail event occur. NOR flash or SLC NAND flash based SSD's without DRAM cache, do not need Power Loss Protection (PLP). Some examples of how an SSD Sudden Power Failure (SPF) could occur while the SSD is writing, reading, erasing, mapping updates and performing background firmware operations, would be 1) removing the computer system power cable 2) a defective power supply in the computer system, 3) removing an SSD from its socket, while the computer systems power is on (if the HotPlug feature is not enabled), 4) a low voltage condition at the SSD connector, 5) an ungraceful system shutdown without using software commands 6) flipping the power switch off.

Table of Contents

Table of Tables

Table of Figures

1 Introduction to SSD Power Loss Protection (PLP)

Viking's solid state drives are available in Enterprise and Industrial/Client versions.

The power loss protection mechanisms for Enterprise-class SSD's are vastly different then the Industrial/Client-class SSD's, so this whitepaper is divided in two sections to avoid confusing the Enterprise SSD with the Industrial/Client power loss protection schemes.

Section 2 describes how Enterprise SSD power loss protection is implemented.
Section 3 describes how Industrial SSD power loss protection is implemented.

Important Note:
Only MLC, TLC or QLC NAND flash based SSD's need power loss protection.
NOR flash or SLC NAND flash based SSD's without DRAM cache, do not need power loss protection.

2 Enterprise SSD's that use a DRAM capacitive hold-up circuit

An Enterprise SSD based on MLC, TLC or QLC NAND contains power fail protection (PFAIL) hardware and firmware that detect and manage power failures. This allows the drive to flush the volatile DRAM controller cache and harden data to NAND flash without data loss or corrupted.

2.1 Supported Enterprise-class SSD Part Numbers

The Viking part numbers for Enterprise SSD's that support the power fail protection features described in Section 2 of this document are listed below:

Table 2-1: SATA Supported Enterprise SSD's (Commercial Temperature 0-70°C)

Viking Part Number*	Description
VPFyyyxxxxTCxxx	PHISON S10 controller based SSD's
VPFyyyxxxxZCxxx	PHISON S11 controller based SSD's
VPFyyyxxxx3Cxxx	PHISON S12 controller based SSD's

Notes: "x" indicates a wild card character that provides specific PN/BOM information.
"y" indicates a wild card character that provides form factor information.
*** Contact Viking to implement this optional feature**

Table 2-2: PCIe Supported Enterprise SSD's (Commercial Temperature 0-70°C)

Viking Part Number*	Description
VPFyyyxxxxVCxxx	PHISON E7 controller based SSD's
VPFyyyxxxx4Cxxx	PHISON E8 controller based SSD's
VPFyyyxxxx5Cxxx	PHISON E12 controller based SSD's

Notes: "x" indicates a wild card character that provides specific PN/BOM information.

"y" indicates a wild card character that provides form factor information.
*** Contact Viking to implement this optional feature**

*2.2 The Integrated Hold-Up Circuit in enterprise-class SSD's**

Viking Enterprise-class SSD's, contain *optional** PFAIL Integrated Hold-Up Circuit hardware and firmware that detects and manage power failures. This allows the drive to flush the controller cache and harden data to NAND flash. The integrated hold-up circuit powers the SSD for short period of time after a power failure using a capacitor and voltage regulator. In the event of an unexpected loss of power, the hold-up circuit is used to supply power to the SSD to allow the controller time to harden data in the DRAM to the non-volatile NAND flash.

In an event of unexpected power drop, the SSD controller firmware detects the lower voltage level through GPIO (General Purpose Input/Out) Pin, and all the internal activities of SSD will be suspended immediately, including garbage collection, wear-leveling, etc. The cached user data and P2L table will be quickly flushed to a temporary assigned block for emergency data backup. On the next power up of the SSD, the drive will read out the flushed data from the "saved" block and rearrange the data to a dynamic block where it can be properly stored. The time interval for a DRAM cache save operation can vary depending on the capacity of the SSD cache and SSD configuration.

*** Contact Viking to order this optional feature**

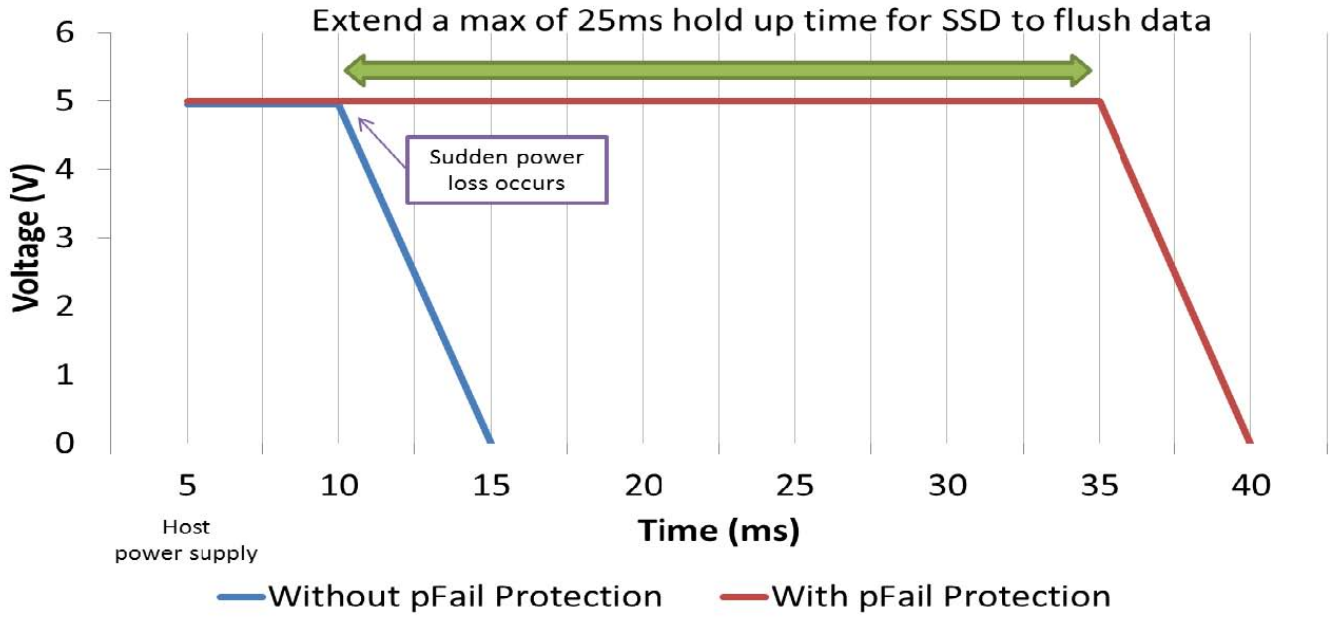

Figure 2-1: 512GB example of Controller/DRAM Capacitive hold-up time

Enterprise SSD's that need to operate in the industrial temperature cannot use capacitive hold-up circuits, which could fail at elevated temperatures causing data to become corrupted if power is removed during a write (also known as lower page corruption). A system UPS (Uninterruptible Power Supply) would be needed or the use of Industrial-class SSD's that do not use DRAM or hold-up circuits. See Section 3 below.

3 Industrial SSD's that do not use DRAM or hold-up circuits

An Industrial/Client SSD that only contains SLC NAND and no DRAM cache, does not need power loss protection.

Industrial/Client SSD's using industrial temperature DRAM as a cache as well as MLC, TLC or QLC NAND cannot use capacitive hold-up circuits, which could fail at elevated temperatures causing data to become corrupted if power is removed during a write (also known as lower page corruption).

Therefore, a non-SLC NAND based Industrial/Client SSD using is well-suited in a system that already manages power fail events, allowing for graceful SSD shutdown. Accordingly, system support should include issuing a Standby Immediate command to the SSD while maintaining power for at least 50ms.

If a non-SLC based Industrial/Client SSD is used in a system that does not manage power failures and shutdowns, there is a possibility of small chance of data corruption. Viking Industrial/Client SSD's take sophisticated hardware and firmware measures to prevent or mitigate such issues making the chance of corruption very small.

3.1 Supported Industrial-class SSD Part Numbers

The Viking part numbers for Industrial/Client SSD's that support the power fail protection features described in Section 3 of this document are listed below:

Table 3-1: SATA Supported Industrial/Client SSD's (Industrial Temp -40 to +85°C)

Viking Part Number	Description
VPFyyyxxxxTIxxx	PHISON S10 controller based SSD's
VPFyyyxxxxZIxxx	PHISON S11 controller based SSD's
VPFyyyxxxx3Ixxx	PHISON S12 controller based SSD's

Notes: "x" indicates a wild card character that provides specific PN/BOM information.
"y" indicates a wild card character that provides form factor information.

Table 3-2: PCIe Supported Industrial/Client SSD's (Industrial Temp -40 to +85°C)

Viking Part Number*	Description
VPFyyyxxxxVIxxx	PHISON E7 controller based SSD's
VPFyyyxxxx4Ixxx	PHISON E8 controller based SSD's
VPFyyyxxxx5Ixxx	PHISON E12 controller based SSD's

Notes: "x" indicates a wild card character that provides specific PN/BOM information.
"y" indicates a wild card character that provides form factor information.

3.1 Power Loss Protection using Link Table Cross-Checking

The link table stored in NAND flash, contains the Physical Block Address(PBA) to Logical Block Address (LBA) translation maps for the data blocks in the SSD. The maps are constantly updated by host read/write and TRIM commands as well as SSD housekeeping functions (garbage collection, wear-leveling and read disturb management).

Under normal and safe power shutdowns conditions, the SSD controller completes all in-flight write transactions to the NAND and properly updates the link table. However, if power to the SSD has been suddenly and abruptly terminated, any in-flight write data to non-SLC NAND could be lost; a condition known as lower page corruption. Viking SSD's protect against this type of data loss, without the use of capacitive hold-up circuits, which could fail at elevated temperatures, by using the following two data protection alternatives:

1. Assigning Link Table Status Flags to SSD data blocks to properly identifying valid data vs. invalid data on the next SSD boot-up
2. Storing/restoring write cache system metadata from NAND

3.1.1 Link Table Status Indicators

The SSD flash controller tags the data blocks referenced in the link table using three types of status flags.

Table 3-3: Link Table Block Status Indicator Flags

Link Table Block Status Indicator Flags	Description of the Block
Static (Valid)	Valid data block stored in NAND
Dynamic	Temporary "Work-in-Progress" data block
Invalid	A spare block that needs to be erased

The status flags in the link table are used by the SSD controller to tell whether the data written to flash is valid or invalid.

1. **Static Block:** Indicates a valid data block was stored/written ("harden") into the non-volatile NAND.

2. **Dynamic Block:** Indicates data was on the last block written before a power loss. (The last written pages can be detected by scanning the dynamic block.)
3. **Invalid Block**: Indicates invalid data blocks needing erasure via a host TRIM command or an SSD cleanup during garbage collection.

There are two link table cross-checking rules:

1. Dynamic Blocks are defined as "Work-in-Progress" blocks until they are stored in flash. They are then retagged as Static Blocks to show they contain valid data.
2. Upon the next SSD boot-up, the status flags of the tagged blocks in the link table are checked as follows:
    a) Static Block = Valid data block
    b) Dynamic Block = A data block needing a re-scan
    c) Invalid Block = A data block needing erasure

The SSD controller firmware has an algorithm that cross-checks the validity of the data blocks in the SSD after two types of power fail conditions:

Condition 1: A power loss during Host Write to SSD
Condition 2: A power loss during internal SSD housekeeping activities

3.1.1.1 Condition 1: A Power Loss during a Host Write Command

Block D is defined as a new data block tagged as Dynamic in the link table. As long as Block D is in dynamic status, the firmware will always start scanning from Block D and check what page data is valid in Block D at SSD power-up.

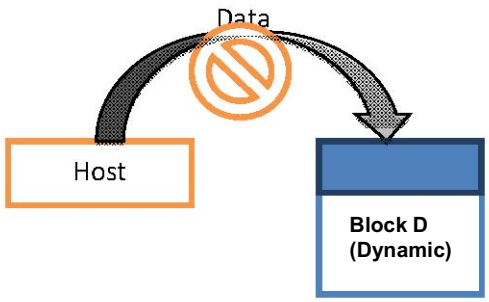

Figure 3-1: Link Table Tagging for New Data

3.1.1.2 Condition 2: A Power Loss during SSD Housekeeping

If the SSD is in a housekeeping mode when a power fail occurs, the following steps will be taken:

3.1.1.2.1 Blocks A and B are tagged as Static Blocks in the Link Table

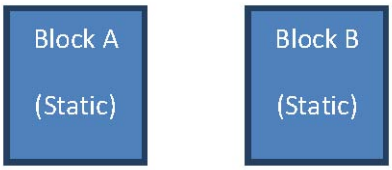

Figure 3-2: Valid Data Blocks Tagged as Static

3.1.1.2.2 Block D is a new block created by merging Blocks A and B

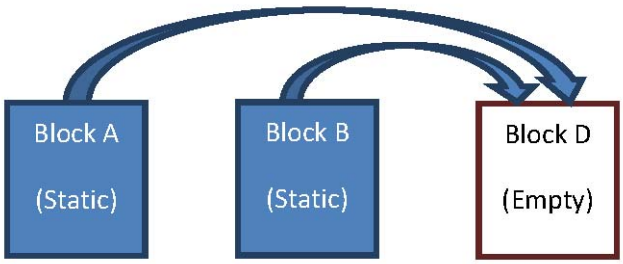

Figure 3-3: Data Block Merge

There are two possible outcomes during the data block merge:

3.1.1.2.3 Scenario I – An Invalid Data Block Merge

If a power loss happens when programming Block D and the link table has not been updated, Blocks A and B will remain tagged as Static (valid data), but Block D will be ignored as a spare block and the firmware will find a new spare block to redo garbage collection (GC).

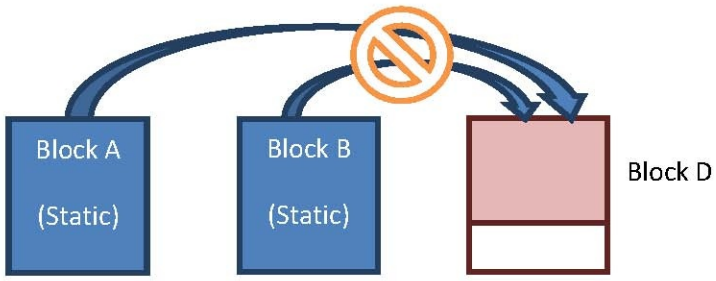

Figure 3-4: Invalid Data Block Merge

3.1.1.2.4 Scenario II – A Successful Valid Data Block Merge

If a power loss happens after finishing internal housekeeping activities on the SSD, the link table will be updated with Blocks A and B retagged as Invalid and Block D will be retagged as a Static (Valid).

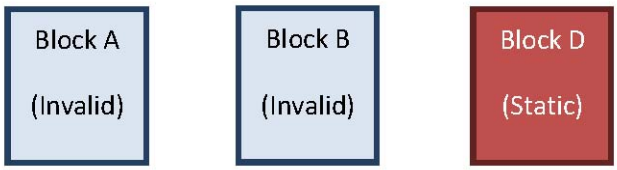

Figure 3-5: Valid Data Block Merge

3.2 Power Loss Protection using Cache Flushing Commands/Algorithms

3.2.1 SSD Flash Controller Cache

SSD cache can provide performance improvements and higher MLC flash endurance by consolidating multiple small transfers into larger groups prior to writing to NAND flash. This write amplification reduces the number of block writes or erasures that are required.

During a proper and graceful shutdown, the host computer would typically issue a STANDBY IMMEDIATE command to allow the SSD controller enough time to flush its volatile DRAM cache to non-volatile NAND Flash. However, during an unexpected power shutdown, where power has been abruptly and unexpectedly terminated, the in-flight write cache data could become corrupted, if data protection measures are not in place.

To “harden” the DRAM cache data into NAND, the host could issue an ATA command called FLUSH CACHE that would request the SSD to flush its volatile write cache into NAND. The command does not complete until the SSD controller sends an acknowledgement (ACK) back to the host indicating the cache flushing has completed. Note that the flush cache needs to be enabled by a host register setting. If the write cache is disabled by the host, maximum power fail immunity could be achieved, but SSD write performance will be reduced accordingly.

The figure below describes how the FLUSH CACHE command works in a Microsoft Windows OS environment for a file copy operation. The process is shown using sub-tasks.

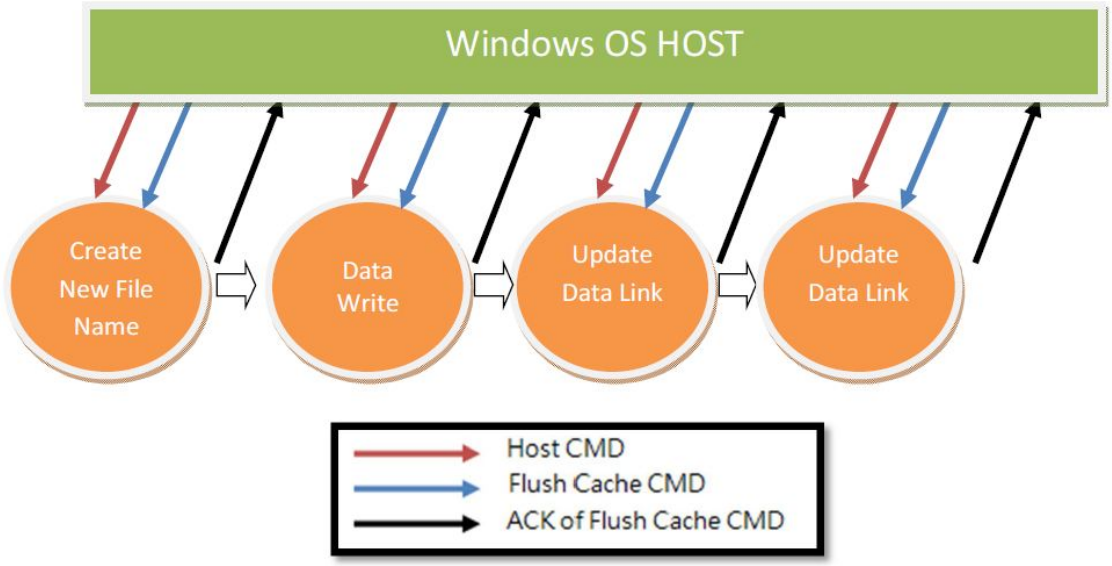

Figure 3-6: Flush Cache during a File Copy Operation

The flush command following each stage of a given task allows the operating system to re-build the system table (and file system table) upon the return from a power fail event. For Microsoft Windows, it would be CHKDSK and for Linux it would be FSCK.

3.2.2 Cache Flushing Firmware Enhancements

3.2.2.1 SmartCacheFlush

The goal of SmartCacheFlush is to prevent DRAM cache data loss by flushing the cached data in DRAM into NAND flash at appropriate timings to avoid a data loss to an unexpected power off condition. There are suggested timings for when to launch a DRAM cache data flush into NAND:

- **Timing Option 1**: When the data size in DRAM cache is larger than a page in flash.

  Since SSD Flash types are page-based programmable, do not flush DRAM cache data into NAND flash until the size of cache data is more than the capacity of one single page in flash.

- **Timing Option 2**: When the System/Host stops sending commands to the SSD.

  From a user viewpoint, cache flushing would be desirable when performed undetected in background mode when the host stops sending requests to the SSD. However if SmartCacheFlush is done too

often, it may effect the fluency of SSD keeping it in a busy state (even when host is not sending commands to the SSD). As a result, SSD will be not allowed to enter sleep mode and power consumption will be greater.

- **Timing Option 3**: Limit write Data cache to a maximum of 15% of total cache.

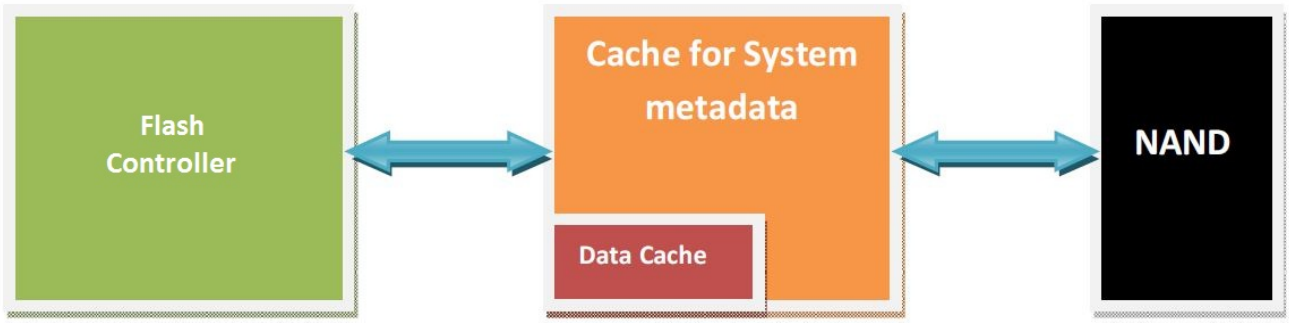

Figure 3-7: Data Cache vs. Metadata Cache

The data cache limit reduces the time for a Cache Flush if it occurs during a power fail event. Viking SSD's use a write-through cache where data is written simultaneously to the DRAM cache and NAND flash, but since it takes longer to write to flash, the controller will consolidate its write cache data into larger groups for higher write amplification prior to writing it to the NAND.

3.2.2.2 Metadata Self-Recovery from a Power Fail Event

Metadata is always stored in NAND and updated in DRAM cache, so some updated metadata will be lost during a power failure. The 85% cache reserve for system metadata can be restored on the next SSD power-up sequence during the page scanning phase. The flash controller rebuilds the metadata blocks using spares taken from the appropriate NAND page using the following process:

1. When new data is written to NAND, firmware always updates the mapping table.
2. The master mapping table is stored in NAND and is used to update/refresh DRAM cache metadata
3. Each new page data is transmitted and tagged along with a few spare bytes for any LBA information, ECC etc.
4. The spare data is then used to rebuild the mapping table in the cache on the next SSD power-up after the power failure, as shown in the figure below:

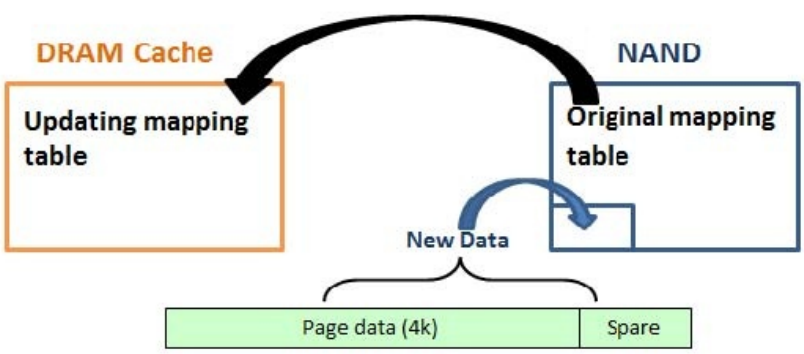

Figure 3-8: Rebuilding Metadata Mapping Table

Contact Viking for other SmartCacheFlush algorithms or firmware options that take into consideration different user scenarios.

3.2.2.3 GuaranteedFlush™

The Viking Industrial SSD uses a firmware algorithm called GuaranteedFlush™ to provide two data protection enhancements for power fail events:

1. The flash controller will ACK the host only when the data is fully committed and stored in NAND, unlike other SSDs implementations, where the ACK is sent to the host when the write is completed to cache but without waiting for the write completion to NAND.
2. Once the data is committed to NAND, the following page writes will not impact the previous committed data. This is made possible by intelligently managing the pair-page of the MLC flash.

Note: GuaranteedFlush™ is a registered trademark of the Phison Electronics Corporation

3.2.2.3.1 NAND knowledge needed to understand how GuaranteedFlush™ works

NAND Flash is a non-volatile memory which composed of millions of floating-gate transistors to capture electrons within the gate. These floating-gate transistors can be identified as many memory cells. Millions of memory cells are connected in array. Each array is consisted of blocks and each block contains numbers of pages. The figure below gives a brief illustration of NAND array from the viewpoint of schematic level.

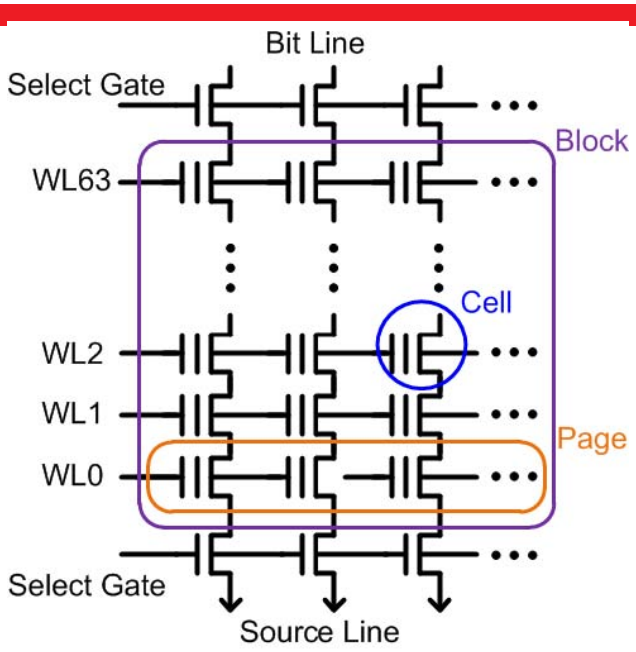

Figure 3-9: NAND Composition in Transistor Level

Each Word-Line (WL) can be regarded as a page which is the basic unit of Read / Program operations in NAND. However, the number of pages for each WL is different in Single-level Cell (SLC) / Multi-level Cell (MLC) / Triple-level Cell (TLC) type NAND.

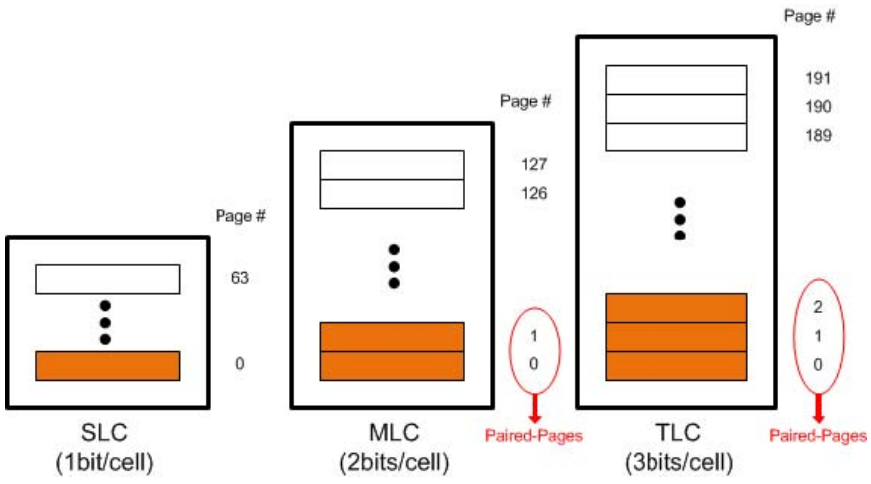

Figure 3-10: Paired-Pages

There would be two pages (Lower / Upper Page) sharing the same WL in MLC NAND and three pages (Lower / Middle / Upper Page) sharing the same WL in TLC NAND. These pages sharing the same WL are considered as paired-pages. Regarding NAND programming operation, there are two golden rules needing to be followed because of the physical characteristics of NAND flash:

- **Programming operation needs to follow the order specified based on NAND flash characteristics**
- **If the programming operation on Word-Line (WL) is not completed, the data integrity of this WL cannot be guaranteed**

For MLC or TLC NAND flash, data programming in single page will not be 1-step operation. That is, MLC/TLC pages need to be programmed 2 or 3 times to accomplish data programming purpose. This phenomenon is actually caused by the physical characteristics of flash memory.

Programming operation on MLC and TLC flash is more complicated than SLC. If we take TLC flash as an example, we need to program the paired-pages on the same WL three times to complete the whole operation. The figure below simply illustrates the complete program operation of TLC flash. Only when the 3rd programming operation on WL_x is done, the data stored at WL_x is able to be identified as reliable data.

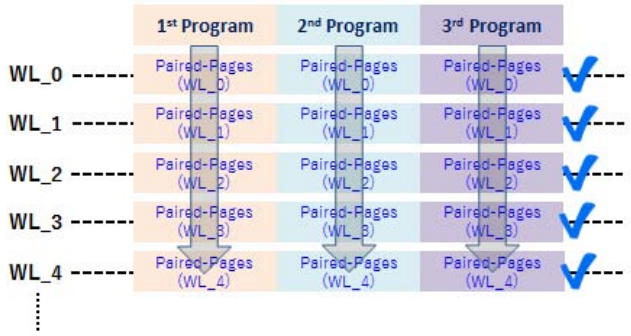

Figure 3-11: Programming Operation of TLC Flash

3.2.2.3.2 The Reasoning for GuaranteedFlush™

Unfortunately, this complexity of programming operations injects the risk of data corruption or Integrity with user data in Paired-Pages. Since the integrity of data can be guaranteed only after the entire programming sequence has been done (i.e. 1st program + 2nd program + 3rd program), any unexpected events happening prior to the completion of programming operation will cause the distortion or loss of data. The figure below is an illustration of what will happen if any unexpected event interrupts data programming operations. In this case, power loss happens before the completion of programming operation for WL_2, only the data

stored in WL_0/WL_1 can be guaranteed. In other words, other data will be distorted or lost because of the unexpected power loss.

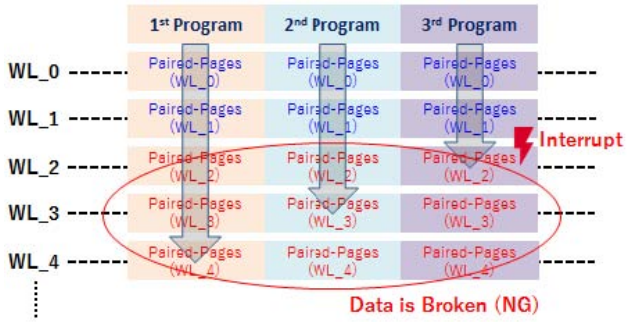

Figure 3-12: Power Loss during Data Programming

GuaranteedFlush™ is designed for preventing SSD from data corruption which caused by any unexpected interruption during programming operations. GuaranteedFlush™ would not be performed for every single data programming command from operating system. Instead, GuaranteedFlush™ feature will be automatically triggered in background once the SSD receives a FLUSH_CACHE ATA command sent from host side. Consequently, the integrity of all the data programmed into the device prior to FLUSH_CACHE ATA command can be guaranteed by this technology. The figure below is an illustration of the user data range GuaranteedFlush™ is able to cover.

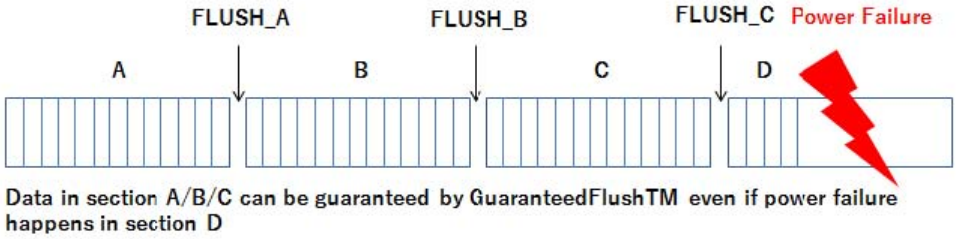

Figure 3-13: Timing of Triggering GuaranteedFlush™

GuaranteedFlush™ is actually implemented by a multi-layer algorithm. Viking SSD's protect user data from sudden power loss.

- **Concept 1**: Dummy Data Compensation

  The first idea is speeding up the programming operations. We know that the programming sequences of different NAND flash types may be different. No matter how the programming sequence varies, the duration of data programming on single WL will decide the risk of data loss due to any unexpected interruption. If this duration becomes longer, the possibility of interruption happening will be higher as well. In order to improve this phenomenon, we will insert necessary dummy data into specific WLs to make sure the WL storing user data is allowed to complete the entire programming operation (i.e. 1st

program + 2nd program + 3rd program) like the figure below shows. By inserting dummy data into the following WLs, user data programming operations must be completed. In this way, the integrity of user data can be guaranteed even if any unexpected event interrupts upcoming programming sequence afterwards.

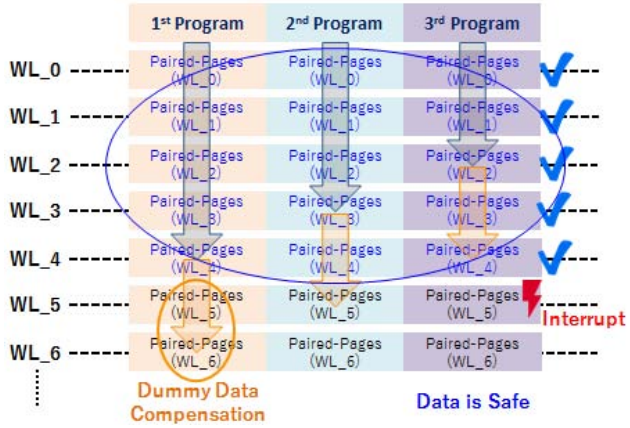

Figure 3-14: Dummy Data Compensation

- **Concept 2**: Real-time Data Backup

  The second idea to protect SSD from data corruption caused by paired-pages effect is directly backing data up to the pages without paired-paged issues in the first place. Those pages which have been configured as SLC mode are able to meet this requirement. Apparently, it is a radical way to keep user data far from the issue we discussed since no paired-pages need to be considered. However, the capacity of SLC mode will be one-third unavoidably. The implementation of Real-time Data Backup becomes another issue.

3.2.2.3.3 GuaranteedFlush™ assures data integrity in SSD prior to FLUSH_CACHE ATA command

All the data programmed into the SSD prior to FLUSH_CACHE ATA command can be guaranteed by the proprietary GuaranteedFlush™ technology. The multi-level algorithm in the SSD firmware makes sure of data correctness while any power loss happens unexpectedly.

3.3 Host-initiated and Industrial SSD-initialized power loss protection

The host-initiated and industrial SSD-initialized power loss protection mechanisms which were previously discussed in this section 3 for the Viking SSD's that operate at elevated temperatures, provide a reliable power fail protection alternative to the lower temperature enterprise commercial-grade capacitive hold-up circuits.

Host initiated **STANDBYIM** or **FLUSHCACHE** SATA ATA commands that flush SATA SSD write-through-cache, combined with link table cross-checking during SSD boot-ups, mitigates the data loss risk from an unexpected power fail events.

4 PFP for Non-SSD Flash Devices (USB)

There is a class of Viking flash storage devices that are characterized by external plug-ins or internally embedded hard-wired board-mounted devices. Examples are Universal Serial Bus (USB), both internally embedded USB (eUSB) and externally mounted thumb-drives, Discrete Flash Chips (DFC), flash BGA drives, etc..

4.1 Supported USB Part Numbers

The Viking part numbers for Non-SSD Flash Devices (USB) which support power fail management (PFM) features described in Section 4 of this document are listed below:

Table 4-1: USB Supported Device See Notes

Viking Part Number	Description
VPFyyyxxxxAxxxx	HYPERSTONE U8 USB2 controller based devices
VPFyyyxxxxQxxxx	HYPERSTONE U9 USB3 controller based devices

Notes: "x" indicates a wild card character that provides specific PN/BOM information.
"y" indicates a wild card character that provides form factor information. (ie USB2, USB3, DUC3, FEP1, FEP2)
hyMap® is a registered trademark of the HYPERSTONE Corporation

4.2 Features That Improve NAND Flash Data Integrity

4.2.1 Wear Leveling (WL)

To maintain the data on the flash memory, there are different types of wear leveling procedures carried out. The goal is to ensure that all blocks in a flash approach their erase cycle budget at the same time. Wear leveling evens out the distribution of program/erase cycles on all available blocks in the flash drive. 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 again, wear leveling ensures that the natural wear is leveled out evenly across the device. Wear leveling can be interrupted by host commands and since a SPF can happen during a WL operation, The USB controller has algorithms to ensure the WL resumes where it left off before the SPF.

4.2.1.1 Dynamic Wear Leveling

Dynamic wear leveling addresses the issue of repeated writes to the same blocks by redirecting new writes to different physical blocks, in turn avoiding premature wear out of the actively used blocks. It is important to note, dynamic wear leveling only functions on blocks being written to.

4.2.1.2 Static Wear Leveling

Static wear leveling addresses all data blocks, regardless as to whether they have been written to or not. This is all done in the background and is completely transparent to the host system.

4.2.1.3 Global Wear Leveling

In global wear leveling, all spare blocks in all flash chips in the drive are managed together in a single pool. Since different flash vendors each have a different defect block count, it is inevitable that, in a multi-chip product, one of the flash components will use up all its spare blocks before the other flash chips. Global wear leveling handles this by managing all flash chips together. The USB controller implements static wear leveling to guarantee that both written and unwritten data blocks approach their erase cycle budget (wear-out) at the same time.

4.2.2 Garbage Collection (GC)

Data blocks that are marked as invalid by the wear leveling process are erased and made available as a free block through the Garbage Collection (GC) process. The controller GC algorithm automatically selects most suited blocks to be consolidated, which is a process that runs continuously in the background, optimally interleaved and arbitrated with other tasks. GC, like WL can be interrupted by host commands and fall victim to SPF. It is, however, important to note that, like wear leveling, GC algorithms ensure that garbage collection resumes unaffected directly after the system re-boots. Garbage collection ensures that blocks that are marked as invalid by wear leveling or the TRIM ATA command are erased and made available in the free block pool. The implementation of GC algorithm insures that it won't be affected by SPF.

4.2.3 Read Disturb Management (RDM)

Reading data in any given physical location on a NAND flash involves voltage currents being transferred across the pages that make up the block. This ultimately means that the physical data qualities of the neighboring sectors, pages and blocks are affected as the voltage passes through them to read the desired data. To manage this, RDM counts the read operations of physical blocks within the flash. When the read count reaches the threshold (which can be enabled, disabled and configured during pre-formatting), the data is mapped to a new physical location on the flash. Usually, these refreshes are interleaved with other

maintenance operations. However, it may happen that when reading data, a physical write is needed. This can have significant impacts on latency and worst-case the read performance. SPFs can occur during refresh operations. The USB controller has, however, been tested and implemented so that the refresh process does not harm data in transition and resumes efficiently. Read disturb management refreshes blocks when the read count reaches its threshold. SPF can occur during a refresh; however, it does not harm data and resumes after power-up.

4.2.4 Read Retry

Should an uncorrectable ECC error occur, the read command is repeated (according to flash specifications). The flash controller supports different reference voltage scenarios, which are implemented through the flash feature. Only after no successful access is the error reported to the host, which then works to help restore weak programmed data or pages after the SPF.

4.3 USB Power Loss Protection

There are unique mapping processes and buffering technology in flash storage devices that are based on the type of controller; hence PFM is unique to these devices.

4.3.1 Log2Phy Mapping Table Protection

Data in NAND flash is usually organized in:

- Physical blocks (the erase unit)
- Physical pages (the program unit)

There is a logical to physical mapping of this flash for the host computer. A logical block is an addressing unit used by the File System usually referring to a 512Byte sector. File systems write, read and erase logical block addresses (LBAs). The flash controller and the flash Translation Layer (FTL) are mapping those LBAs and managing the activities on the physical level of the flash. A mapping table is maintained by the FTL that is vital to the system stability. For performance reasons, LBAs are often distributed over different physical blocks. A LBA or sector is an access unit of host and file systems. A physical page can store data of several sectors, and a block consists of a plurality of pages.

USB controllers are usually DRAM-less. External DRAM may significantly increase performance; however, there is a range of problems that can arise through its implementation. To ensure the mapping is saved efficiently on power down, gold cap capacitors are necessary, which have limited charge cycles. Different capacitors are used for different capacities. Secondly, DRAM involves operations being carried out w/o power. This

is risky especially when recovery and maintenance procedures are undertaken upon power-up. These two aspects considerably add to the BOM complexity and a higher failure rate.

For this reason, mapping updates are stored in a flash mapping table which is a critical activity that must be protected. Without Power Fail Management (PFM), a sudden power fail during an update of the Log2Phy table would lead to corruption of mapping data and in consequence to a defective drive. Keeping the logical-to-physical mapping consistent is vital to protect a drive against the consequences of sudden power fails.

4.3.2 Power Fail Detection

The internal voltage sensor on the controller, monitors the external supply voltage. Depending on the product, this can be 5V, 3.3V or 1.8V and enabled through flash feature configuration bits, 2 dedicated pins that are available to monitor the power supply. If the power falls below a certain threshold, the firmware might finish the currently running command (depending on flash type) and will immediately assert the flash write protect trigger. If the power continues to fall until the reset detector triggers, the controller shuts down and is prepared to do a standard power-up initialization after supply returns. If, however, supply voltage returns before the reset detector is triggered, the firmware starts the standard power-up initialization process and assumes that the host will re-initialize the card. Either way, the USB controller is designed to not rely on any final write operation. The USB controller is designed to reliably detect power-downs in order to stop and protect all operations immediately.

4.3.3 Power On Reset (POR)

Upon reset from the internal Power-on-Reset commands, the controller controls different internal signals used to appropriately schedule the different tasks upon power-up or to get out of sleep mode. POR Reset is de-asserted with some latency after both conditions apply:

- VCC rises above VCC minimum threshold
- Internal voltage rises above the internal core voltage minimum threshold.

POR Reset is asserted when VCC falls below the POR Reset voltage threshold. Those thresholds can be set by the end customer. The time between the sensing of an SPF and the reset from POR must be enough to enable the FW for a proper shut-down procedure. The USB controller manages the scheduling of the different vital tasks upon power-up.

4.3.4 Oscillating Power Supply

Voltages oscillate by nature, and this can often be mistaken for a SPF if the voltage drops below the chosen configuration. If internal voltage sense is enabled, an indication of power failure can be pushed according to the chosen configuration. If supply voltage returns within this time, the trigger is ignored. The USB controller has hardware circuitry to regulate and take care of bouncing on power-ups. It filters the power supply oscillation, which arises through high power on reset voltage thresholds. While the hardware circuitry manages bouncing on power-up, the firmware manages it on power down. Configurations can be implemented individually and should differ depending on the use case and application. For example, reliable mode and performance mode manage data differently during a SPF and ultimately trade off features to manage the data efficiently.

4.3.5 Normal Shut Down

When shutting down, if voltage sense is triggered, firmware might finish currently running command (depending on the flash), and then asserts the flash write protect signal. The detection level is at a certain threshold. If supply voltage continues to fall until reset detector triggers, the controller will go through standard power-up initialization when supply returns. If supply voltage returns before the reset detector triggers, the firmware assumes that the host will re-initialize the card, and in this case, controller will go through standard power-up initialization process. The USB controller is designed not to rely on any final write operation.

4.3.6 Flash Write Flow with hyMap®

Flash write flow with hyMap® firmware allows for a higher write granularity through its sub-page based FTL. Through this technology, pages are written consecutively to the flash (from different LBAs). When writing a physical page, information regarding the LBA origin is kept in the overhead area.

The USB controller doesn't rely on DRAM for mapping information. To improve performance, the mapping information is written in the IRAM (IP Cache). Information is flushed when the IP-Cache is full into log-book on the flash memory. Information included in the log-book, as well as the log2phy table, are maintained like any flash memory data by Garbage Collection (GC). (Higher granularity of the mapping system, results in a smaller impact from SPF.)

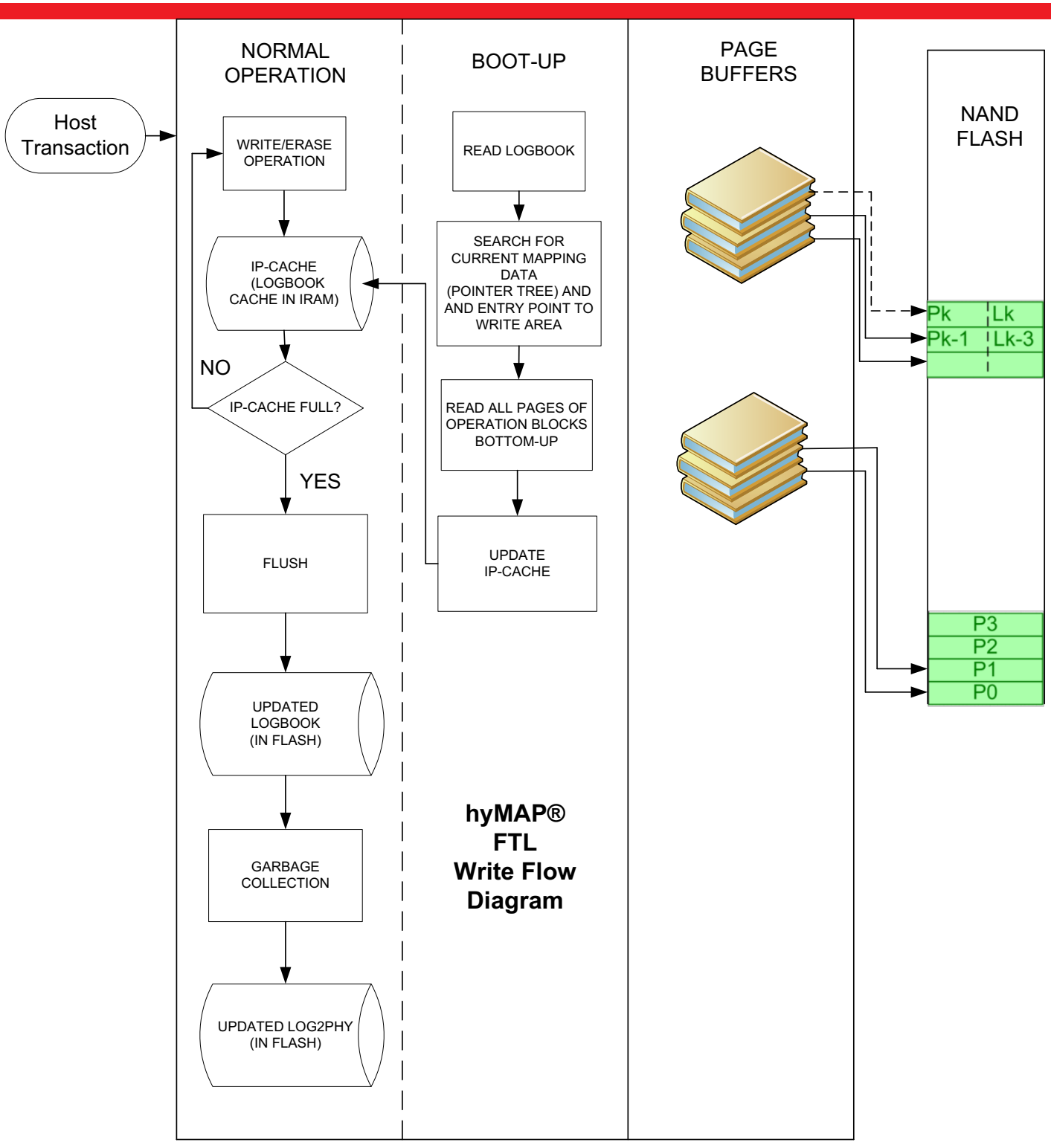

Figure 4-1: hyMAP® FTL Write Flow Diagram

4.3.7 Power Fail Consequences

Power fail consequences of a SPF can differ heavily depending on a range of aspects including the configurations and use case and application.

Figure 4-2: Consequences of a Sudden Power Failure (SPF)

The power supply of the host (depending on system or standard) can drop from 5.0V or 3.3V or 1.8V. The internal voltage detector recognizes power failures as soon as the voltage drops defined thresholds. This default configuration can be changed for individual applications. Upon noticing this voltage drop, the flash controller jumps into Write Protect (WP) by pulling the write-protect pin to zero. Finally, the flash is set to WP to minimize data corruption. Depending on the situation, recovery measures might be needed at the next power up.

4.3.7.1 Consequences on Management Data

Similarly to the firmware code, management data is invaluable to maintaining the flash device and is protected from the consequences of a SPF. If the management data were compromised, the entire flash would fail.

4.3.7.2 Consequences on User Data

User data is more susceptible to SPF and depending on the configurations and the application, the consequences can differ dramatically. Generally, it's most likely that you can lose what you most recently wrote or what you are writing during the SPF. The consequences with the controller are generally not critical due to the modes available, configuration options and complex algorithms in the background. The host interface can also redo the last transaction to ensure that all data is transferred efficiently to the flash. User data is written and managed the same way as described in the FTL write diagram on the previous page. It ensures that minimum information is lost when a SPF happens. The USB controller is designed not to rely on any final write operation.

4.3.8 Firmware Protection & Features

The USB controller guarantees that the firmware code cannot be corrupted or reached in the event of a SPF. If firmware code were compromised, the entire flash device would fail.

Firmware files are stored on the flash during pre-formatting. To act as a backup and cope with the unexpected possibility of corrupt data, the firmware is in fact written twice onto the flash during this process. During runtime, the firmware required executable part is loaded into the IRAM. Only special functions are loaded into IRAM when needed (overlay) FW files could become corrupted (e.g. due to Read Disturb problems). In the event of a non-correctable failure, the alternative FW code is used to ensure stable usage. Even when the system starts, both FW blocks are checked. If one block shows errors, it is repaired using the other backup FW block. The blocks containing FW code are also included in the near miss ECC handling. Management data is included in the standard WL and RDM processes and work as additional PFP. Redundant firmware is stored on the flash whilst the main FW is loaded onto the internal RAM for faster processing.

4.3.9 Safe Flash Handling

It is common that pages which are programmed as the power disappears (during an SPF) become weak and unstable. This ultimately generates pages that are either entirely corrupted or only readable once. Upon reading for the second time, these unstable pages are often bombarded with so many errors that not even the ECC can correct them. Examples are:

- Unsafe programming is undeniably problematic
- For a management block. This might cause loss of management information and thus damage the drive permanently
- For a user block, this might cause an error when the application wants to read the data afterwards

Safe flash handling is activated by default in the USB controller to prevent the loss of both management and user data. However, during pre-formatting, the parameters of the safe flash handling can be configured. To achieve this, management data and user blocks in the latest logbook are refreshed on every power-on sequence. Safe flash handling is activated by default in the USB controller and can be configured depending on the use-case.

4.4 Other PFP Features That Make A Difference

SPF is managed by the USB controller through a culmination of complex algorithms, flexible configurations, and features, which ensure secure data transactions and reliable usage.
Below are some additional USB controller features that ensure reliable usage:

- Hybrid of transactional mapping and a journaling log-book
- Redundant un-corruptible firmware
- Configurable “Early Acknowledge” and write caching
- Managing Program Errors
- Near-Miss ECC
- Dynamic Data-Refresh
- Reliable mode vs. Performance mode

Handling Power Fail situations efficiently and without data loss requires a very robust flash-management architecture. The USB controller has stringent hardware and hyMap® firmware to address sudden power failure (SPF) reliability issues. Many features including wear leveling, ECC, redundancy features, logbook or write confirmation have to work together reliably to adequately handle such critical situations. The log2phy mapping, as well as logbook management, are essential features for reliable data handling during PF. Temporary storage of redundant data without wasting precious memory controlled by smart FW features keeps data from getting lost in cases of sudden PF. Depending on individual application

requirements, certain trade-offs in terms of performance or cost can be supported or configured during pre-formatting.

4.5 hySMART Utility

Hyperstone Corporation provides a proprietary hySMART utility for their U8 and U9 controllers to monitor health and SPF behavior. By counting power-up cycles, it in turn analyses ECC and uncorrectable ECC specifically associated with SPF during power up procedures.

About Viking Technology

Viking Technology develops and delivers innovative high-technology products that optimize the value and performance of our customers' applications. Founded in 1989, Viking Technology has been providing Original Equipment Manufacturers (OEMs) with industry leading designs, engineering, product support and customer service for over 25 years. For more information, visit http://www.vikingtechnology.com.

Revision History

2/10/15		Initial release
5/12/15		Add PN's
9/12/17		Revise logo, address and color scheme. Add S12 PN info and holdup circuit info
3/14/19		Divide document in to two sections; one for enterprise SSD's and one for industrial SSD's. Add more info for enterprise SSD's. Add more detailed info for SmartCacheFlush and GuaranteedFlush™
4/1/19		Add info for USB PFP

Global Locations		
US Headquarters	India Office	Singapore Office
2950 Red Hill Ave. Costa Mesa, CA 92626 Main: +1 714 913 2200 Fax: +1 714 913 2202	A 3, Phase II, MEPZ-Special Economic Zone NH 45, Tambaram, Chennai-600045 India	No 2 Chai Chee Drive Singapore, 109840

For all of our global locations, visit our website under global locations. For sales information, email us at sales@vikingtechnology.com

viking TECHNOLOGY

DRAM MEMORY & FLASH STORAGE
NVDIMM, SSD, DRAM, MCP & CUSTOM
for Embedded, Industrial, Defense & Aerospace