Intel® SSD Technology Terminology Guide

An Informative Guide Explaining Key Intel and Industry SSD Technology Features & Terminology

Information Guide

June 2015

Revision 1.3
INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request. Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

This document contains information on products in the design phase of development. Do not finalize a design with this information. Revised information will be published when the product is available. Verify with your local sales office that you have the latest datasheet before finalizing a design.

Code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and other countries.

*Other names and brands may be claimed as the property of others.

Copyright © 2015, Intel Corporation. All rights reserved.
# Contents

1. Document Purpose ........................................................................................................ 9
2. PLI (Power Loss Imminent) .......................................................................................... 11
3. NCQ (Native Command Queuing) ................................................................................ 13
4. AES Encryption ........................................................................................................... 15
5. End-to-End Data Protection ......................................................................................... 17
6. XOR (Exclusive “OR”) ............................................................................................... 19
7. HET® (High Endurance Technology) ......................................................................... 21
8. OPAL* .......................................................................................................................... 23
9. Trim .............................................................................................................................. 25
10. Power Safe Write Cache ............................................................................................. 27
11. DIPM and HIPM (Device and Host Power Management) ........................................ 29
12. Write Amplification ..................................................................................................... 31
13. RAS (Reliability, Availability, & Serviceability) ......................................................... 33
14. DevSleep (Device Sleep) ............................................................................................ 35
15. NVMe (Non-Volatile Memory Express) ..................................................................... 37
16. RBER & UBER ......................................................................................................... 41
17. MTBF .......................................................................................................................... 42
18. M.2 SSD Form Factor ................................................................................................. 43
Figures

Figure 1: PLI Functional Description Block Diagram .........................12
Figure 2: AES Security Diagram.................................................16
Figure 3: “End-to-End” Data Flow Diagram ..................................17
Figure 4: ETE Appended Protection Information...............................18
Figure 5: Write with “XOR” Operation Illustration..........................20
Figure 6: HET (High Endurance Technology) Diagram ....................22
Figure 7: SSD Trim Operation Flow Diagram..................................26
Figure 8: Energy Supply Capacitors for Power Safe Write Cache on Intel® Solid State Drive 710 Series.................................28
Figure 9: SATA Power Management States ..................................30
Figure 10: Write Amplification Formula ........................................32
Figure 11: Dev Sleep Block Diagram .............................................36
Figure 12: NVMe Structure .........................................................38
Figure 13: NVMe Command Set...................................................38
Figure 14: Feature Comparison of AHCI and NVMe .........................39
Figure 15: PCIe SSD ACHI vs. NVMe Flow ....................................40
Figure 16: M.2 Edge Connector Pin out and Key Identification .............44
Figure 17: M.2 SSD Photo Illustration of Sample Lengths ................45
## Revision History

<table>
<thead>
<tr>
<th>Document Number</th>
<th>Revision Number</th>
<th>Description</th>
<th>Revision Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>327756</td>
<td>000-US</td>
<td>Initial Release</td>
<td>August 2012</td>
</tr>
<tr>
<td></td>
<td>1.2</td>
<td>Add Storage Device Sleep and NVMe Definitions</td>
<td>September 2014</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Updated glossary terms with AHCI, JBOD, NVMe</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.3</td>
<td>Updated glossary terms with RBER/UBER, MTBF, M.2 SSD Form Factor</td>
<td>June 2015</td>
</tr>
</tbody>
</table>

§
<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AHCI</td>
<td>Advanced Host Controller Interface: Developed in conjunction with and for SATA interface storage devices</td>
</tr>
<tr>
<td>BIOS</td>
<td>Basic Input / Output System</td>
</tr>
<tr>
<td>Chipset</td>
<td>A term used to define a collection of integrated components required to make a PC function.</td>
</tr>
<tr>
<td>DOS</td>
<td>Disk Operating System</td>
</tr>
<tr>
<td>HBA</td>
<td>Host Bus Adapter</td>
</tr>
<tr>
<td>HDD</td>
<td>Hard Disk Drive</td>
</tr>
<tr>
<td>I/F</td>
<td>Interface</td>
</tr>
<tr>
<td>JBOD</td>
<td>“Just a Bunch of Disks”: a storage architecture using multiple drives, while making them accessible either as independent drives, or as a combined (spanned) single logical volume with no actual RAID functionality.</td>
</tr>
<tr>
<td>LBA</td>
<td>Logical Block Address</td>
</tr>
<tr>
<td>NAND</td>
<td>Negated “AND” – A NVM Flash Memory Architecture</td>
</tr>
<tr>
<td>NVM</td>
<td>Non-Volatile Memory</td>
</tr>
<tr>
<td>NVMe</td>
<td>Non-Volatile Memory Express: PCIe based storage interface optimized for solid-state drives. Also known as NVMeHCI, for NVMe host controller interface.</td>
</tr>
<tr>
<td>OEM</td>
<td>Original Equipment Manufacturer</td>
</tr>
<tr>
<td>Term</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-----------------------------------</td>
</tr>
<tr>
<td>ONFI</td>
<td>Open NAND Flash Interface</td>
</tr>
<tr>
<td>OS</td>
<td>Operating System</td>
</tr>
<tr>
<td>RAID</td>
<td>Redundant Array of Independent Disks</td>
</tr>
<tr>
<td>ROM</td>
<td>Read Only Memory</td>
</tr>
<tr>
<td>SATA</td>
<td>Serial Advanced Technology Attachment</td>
</tr>
<tr>
<td>SSD</td>
<td>Solid-State Drive</td>
</tr>
<tr>
<td>UI</td>
<td>User Interface</td>
</tr>
</tbody>
</table>
1. Document Purpose

The purpose of this document is to provide interested readers explanations of many Intel and industry Solid-State Drive technology functions, features, and acronyms.

The document is structured to limit the explanations to one per each SSD technology item. For each item, overview and context is provided along with the “short” and “long” explanations suited to the user’s need and interest. Feel free to pull individual topic sheets out as needed.
2. PLI (Power Loss Imminent)

Overview and Context

Worried about data loss during unplanned power shutdowns or inadvertent drive removal in data centers? Data safety features in SSDs can prepare for unexpected power-loss and protect system and user data.

Definition and Explanation

Short:
PLI is a hardware and firmware feature on SSDs that provides enough stored energy for the SSD to safely store user and system data in temporary buffers to the non-volatile NAND flash storage during an unexpected loss of power. Not all SSDs have the PLI feature, but the Intel® SSD 320 Series and Intel® SSD 710 Series do. PLI is sometimes referred to as “power safe write cache” (also included in this document; see section 10).

Long:
During a “clean” shutdown, most host systems initiate a command (the STANDBY IMMEDIATE command) to an SSD to give the SSD enough time to prepare for the shutdown. This allows the SSD to save data currently in transition (in temporary buffers) to the non-volatile NAND media. However, during an unsafe power shutdown or a loss of power, the SSD abruptly loses power before the host system can initiate the ATA STANDBY IMMEDIATE command. This prevents data in the temporary buffers from being saved in the non-volatile NAND.
The Intel SSD 320 Series and Intel SSD 710 Series contain hardware and firmware-based power-loss data protection features. These SSDs includes a power-fail detection circuit, which sends a signal to the ASIC controller in the SSD indicating there is an imminent drop in power level. The SSD then relies on its on-board power-loss protection capacitors to provide enough energy for the SSD firmware to safely move user and system data from the transfer buffer and other temporary buffers to the NAND.

Figure 1: PLI Functional Description Block Diagram
3. NCQ (Native Command Queuing)

Overview and Context

Native Command Queuing (NCQ) was originally a technology designed to increase performance of SATA hard disks. It does this by allowing a hard disk to internally optimize the order in which it executes the read and write requests it received from the host. For hard drives, this can reduce the amount of unnecessary drive head movement and resulting in better performance, in particular for workloads where multiple read/write requests are outstanding at the same time. This situation most often occurs in server-type applications. Since NCQ helps hard disk drives performance by optimizing the internal order of execution, NCQ will further the performance of solid-state drive even more so.

Definition and Explanation

**Short:**

Native Command Queuing (NCQ) is a technology designed to increase performance of SATA hard disk and solid-state drives by allowing them to internally optimize the order in which it executes read and write requests received from the host. All Intel solid-state drives support the Native Command Queuing (NCQ) command set, which includes: “READ FPDMA QUEUED” and “WRITE FPDMA QUEUED”. Maximum Queue Depth is 32.

**Long:**

Native Command Queuing (NCQ) was originally a technology designed to increase performance of SATA hard disks. It allows a hard disk to internally optimize the order in which it executes read and write requests received from the host. Since NCQ helps hard disk drives performance by optimizing the internal order of execution, NCQ will further the performance of solid-state drive even more so.
For NCQ to be enabled, it must be supported and enabled in the SATA host bus adapter and in the hard drive or SSD. Additionally, the appropriate device driver must be loaded into the operating system to enable NCQ. Many newer chipsets support the Advanced Host Controller Interface (AHCI), which should allow a generic driver supplied by the operating system to control them and enable NCQ. Newer mainstream Linux kernels support AHCI natively.
4. AES Encryption

Overview and Context

Encryption of data in storage devices such as PC hard drives, USB keys, or SD removable cards is increasingly important to protect sensitive personal or proprietary company information. Encryption protects the data from being read without a decryption key if the device is lost or stolen. Advanced Encryption Standard (AES) is one of many specifications for the encryption of electronic data and it has been adopted by the U.S. government and widely used in industry.

Definition and Explanation

Short:

Advanced Encryption Standard (AES) is a specification for the encryption of electronic data. It has been adopted by the U.S. government and is now used worldwide. The algorithm described by AES is a symmetric-key algorithm, meaning the same key is used for both encrypting and decrypting the data. AES is the first publicly accessible and open cipher approved by the National Security Agency (NSA) for top secret information when used in an NSA approved cryptographic module.

Long:

The Advanced Encryption Standard (AES) is the United States Government’s Federal Information Processing Standard for symmetric encryption, defined by FIPS Publication #197. AES is a block cipher that encrypts a 128-bit block (plaintext) to a 128-bit block (ciphertext), or decrypts a 128-bit block (ciphertext) to a 128-bit block (plaintext). AES uses a key (cipher key) whose length can be 128, 192, or 256 bits. AES-128, AES-192, and AES-256 process the data block in, respectively, 10, 12, or 14 iterations of pre-defined sequences of transformations, which are also called AES rounds. The rounds are identical except for the last
one, which differs slightly from the others by skipping one of the transformations.

Steps to Enable AES Encryption on an SSD:

1. Set ATA password (also known as the drive password) in the BIOS to enable the drive AES-128 /AES-256 encryption (The ATA password is stored on the drive as a non-reversible hash, and is used to encrypt the encryption keys on the drive)
2. Perform a secure erase on the SSD if a new password is desired or needed
3. Set your new password

![AES Security Diagram](image)

**Figure 2: AES Security Diagram**

The following Intel SSDs support 128-bit AES encryption:

Intel® SSD 320 Series, Intel® SSD 330 Series, Intel® SSD 520 Series and Intel® SSD 710 Series

References:

5. End-to-End Data Protection

Overview and Context

Data integrity is extremely important, especially in data center storage environments where there are many layers hardware devices and software that the data must traverse.

![Figure 3: “End-to-End” Data Flow Diagram](image)

Chances for user data to get corrupted as it is passed from device to device are possible and problematic. Therefore, there are mechanisms and algorithmic schemes that insure that saved and retrieved data is indeed correct without error from the beginning end to the furthest end, where the data is stored, thus the name “end to end” data protection.

Definition and Explanation

**Short:**

End-to-end Data Protection is a feature for Solid-State Drives and hard drives that extend error detection to cover the entire path from the computer system to the hard drive media and back. Data protection information is appended to the data in the computer system. It stays with the data from the computer, through connections, through RAID controllers, HBAs, and through drive electronics to the storage device. When read, the same data protection information returns with the data.
to the computer system. The protection information is used to verify the correctness.

**Long:**

End-to-End Data Protection is a feature for storage hard drives that extends error detection to cover the entire path from the computer system to the hard drive media and back. Data protection information is appended to the data in the computer system, and stays with the data from the computer, through connections, through RAID controllers, and through drive electronics to the storage device. The appended “end to end” data integrity field (DIF) is 8 Bytes that is composed of CRC, App Tag, and Ref Tag (See Figure 4).

![Figure 4: ETE Appended Protection Information](image)

**References:**

6. XOR (Exclusive “OR”)

Overview and Context

Striving to improve the reliability of SSD is always of paramount value at Intel. Improving the reliability the NAND flash components is critical to ensure that user data is preserved and performance is sustained for the life of product. Many schemes exist to improve the reliability of the NAND components; Intel has chosen XOR reliability enhancement. The “XOR (Exclusive “OR”)” reliability enhancement operation is named after the Boolean logical function (See logical gate and truth table), and allows ability to replace up to an entire defective NAND die in a component through XOR operations.

Definition and Explanation

Short:

XOR significantly improves the NAND component reliability by providing protection against die failure and extrinsic UBER (Uncorrectable Bit Error Rate) events by replacing entire or defective portions of a NAND die array utilizing spare NAND die array that can be built with reconstructed data via the XOR parity data bits. This scheme would fail in the event that there are 2 or more die failures simultaneously, or if the SSD runs out of spare XOR NAND die space. The detailed “XOR” function will be explained in the long explanation below.
Long:
XOR’s primary goal in an SSD is to REDUCE any catastrophic data loss failure and AFR (average failure rate). This is accomplished by using the Boolean XOR logical function that $A \ XOR \ B \ XOR \ B = A$ which becomes the underlying principle on rebuilding data. The XOR-ing of incoming data packets from the host generates XOR parity bits that later can be used to rebuild data if needed. This ability to rebuild or replace bad NAND bytes with spare good NAND bytes significantly enhances the SSD’s reliability while providing consistent performance. The illustration below shows the XOR operation with some example packets of data.

**Figure 5: Write with “XOR” Operation Illustration**

Since every SSD has some spare NAND area, the XOR rebuilds—if needed—uses this spare area.
7. **HET® (High Endurance Technology)**

**Overview and Context**

Improving the reliability of the core storage elements of an SSD—its NAND flash components—is critical to ensure that user data is retained along with optimum performance for the life of product. Intel has developed technologies that combine the enhancements of NAND wafer/die sorting, component testing, and software at the SSD system level that provide a very effective solution to the overall reliability of the SSD.

**Definition and Explanation**

**Short:**

HET® (High Endurance Technology) is Intel proprietary technology that combining NAND silicon enhancements and SSD NAND management techniques to extend the write endurance of MLC-NAND-based SSDs.

**Long:**

Intel developed “HET,” High Endurance Technology, which comprises enhancements at both the NAND die/wafer and component testing levels with the SSD firmware enhancements to improve the endurance and reliability of its NAND components and the SSD product. At the NAND component level, the NAND is optimized with fine-tuned read, program and erase voltages and extensively tested at temperature and voltage skews to ensure the highest-binned products.
In the SSD, firmware enhancements are implemented to minimize bit errors and to reduce write amplification. Innovative and efficient bundling of writes to minimize excessive background data manipulation and management was created to reduce write amplification. In the real world, SSDs with HET technology from Intel will provide reliable performance far beyond the expected lifetimes of standard MLC-NAND-flash-based devices.

**NAND Optimizations**
- Custom NAND optimizations
- Extensive NAND testing and binning

**Firmware Enhancements**
- Optimized error avoidance techniques
- Write amplification reduction algorithms
- System level error management beyond ECC

**Figure 6: HET (High Endurance Technology) Diagram**
8. **OPAL*  

Overview and Context

Secure protection of data on storage devices such as solid state drives and hard disk drives has become very important given the broadly available user connection points such as WiFi, Broadband, LTE, etc. that dramatically increases vulnerability to malware and virus entering our devices.


Definition and Explanation

Short:

TCG’s Opal SSC (Security Subsystem Class) specification has been adopted as an international standard with the goal of creating more secure IT environments. When used in combination with supporting application software, SSDs and HDDs that support Opal SSC will enable sophisticated security solutions for a wide range of computing platforms, including notebook PCs.

Long:

When Opal is used in combination with application software supporting Opal SSC, it will enable advanced security features such as pre-boot authentication and secure partition.

Pre-boot authentication performs user authentication when starting up the computer. The Opal SSC standard will allow for the use of advanced authentication techniques that include biometric authentication or smart-card authentication, even before starting up Windows or other
operating systems that rely on traditional keyboard password entry access.

Secure partition is a technology enabling SSD or HDD storage to be partitioned into a number of secure storage regions—each with its own encryption key—giving access to only the valid owner of partition and users given access by the owner. This allows, for example, for content and its licensing data to be stored and protected in separate partitions, so that different partitions can be used depending on the nature of the data stored, enabling more secure data management.

There are many independent software vendors to implement these functions, including McAfee*, SECUDE*, Wave Systems*, and WinMagic*, that will work in conjunction with the Opal SSC. Many vendors also provide their own consoles to manage these features.

Other features that Opal enables with independent software suppliers are:

- Security Provider Support
- Interface Communication Protocol
- Cryptographic Features
- Authentication
- Table Management
- Access Control & Personalization
- Issuance
- SSC Discovery
9. Trim

Overview and Context

Erasing data is different between solid-state drives and hard drives. In hard drives, existing data can simply be overwritten. On the other hand, SSDs cannot overwrite old data in NAND components until that old data has been erased with a separate operation. Therefore, SSDs must efficiently aggregate the erase operations without interrupting any active SSD read and write operations. To facilitate these NAND erase operations, also known as “garbage collection,” the Windows 7* OS issues a TRIM command to the SSD when files or data are no longer needed by the user.

Definition and Explanation

Short:

TRIM is a command issued by the operating system to inform the solid-state drive which blocks of data are no longer in use and can be wiped or erased internally. TRIM enables the SSD to handle its garbage collection to free up space for future writing of new data at a high sustained rate.
Long:

The TRIM command is designed to enable the operating system to notify the SSD which pages of data are now invalid due to erases by the user or operating system itself. During a delete operation, the OS will not only mark the sectors as free for new data, but it will also send a TRIM command to the SSD with the associated LBAs (Logical Block Address) to be marked as no longer valid. After that point, the SSD knows not to relocate the data from those LBAs during garbage collection. This will result in fewer writes to the flash, reducing write amplification and increasing drive life. Different SSDs will act on the TRIM command somewhat differently so the final performance can vary based on the SSD models.

TRIM is not supported in older Windows generation operating systems and in RAID configurations.

**Figure 7: SSD Trim Operation Flow Diagram**
10. Power Safe Write Cache

Overview and Context

Data integrity is the utmost concern with all computing and storage devices and environments. Unexpected events such as a power loss can cause serious data integrity issues, especially for data that is “in flight”—in temporary volatile DRAM write caches— and not securely saved in the storage device. SSDs providing a means to capture and save data “in flight” add to the overall system computing and storage platform integrity; this is the benefit that a power safe write cache provides.

Definition and Explanation

Short:
In the event of a power failure, a power safe write cache will have energy-storing capacitors to ensure that there is no data loss by providing enough energy to complete all writes to the NAND flash memory.
**Long:**

In the event of a power failure, a power safe write cache will have energy storing capacitors to ensure that there is no data loss by providing enough energy to complete all writes to the NAND flash memory.

Please see Section 2 on “PLI” (Power Loss Imminent) for more details of the capacitor operation in supplying enough energy to complete write operation.

![Figure 8: Energy Supply Capacitors for Power Safe Write Cache on Intel® Solid State Drive 710 Series](image)
11. DIPM and HIPM (Device and Host Power Management)

Overview and Context

SATA allows “PHY” Power Management to be Host Initiated (HIPM) or Device Initiated (DIPM), thus providing the flexibility to optimize the SATA components like Solid-State Drives and hard drives for a wide range of usages and applications. The host will have the ability to put SATA peripherals directly into Idle, Standby, and Sleep modes, and report the current power management mode of SATA peripherals.

SATA Link Power Management requires cooperation between the host and the device. Either can request the link to enter a low-power state, but the corresponding host or device must accept or reject the link state change request. Each of these provides power savings by themselves; maximum power savings, however, are achieved when both are implemented together.

Definition and Explanation

Short:
In HIPM (Host Initiated Power Management), the Host either in hardware or software manages the power state of SATA PHY to enter a low power state. In DIPM (Device Initiated Power Management) the Device manages the power state.

Long:
AHCI Link Power Management is a technique where the SATA AHCI controller puts the SATA link to the internal HDD and/or SSD into a very low power mode when there is no IO (input/output) activity for an extended period. The controller automatically puts the link back into active power state when there is real work to be done. This is done to save power consumption by the HDD and/or SSD.
Host-initiated power management can be implemented either in the host hardware or the host software. In the first case, the host controller requests a link power management transition immediately after all outstanding commands to the drive have been completed. This allows the link to enter a low-power state immediately upon completion of the commands to the disk. Since the host has the best knowledge of what commands have been posted, or will be posted to the device, the host is able to make an immediate link power state change without invoking a time-out period.

**Figure 9: SATA Power Management States**

Device-initiated power management is implemented by the drive. The drive knows best how long a specific command might take to complete, and is best equipped to request a link power management state change while processing the command. The host controller can automatically put the link into either Slumber or Partial after the command completes, typically, this will be Partial. However, after some extended period of idleness, the link will transition from Partial to Slumber. This can be done either by the host software or the device. Since the host is best equipped to manage the PHY between commands and the best device within a command, the best power management is obtained when the host and device cooperate.

**References:**

1. Intel Technology Journal, Volume 9, Issue 1, 2005
2. SATA Article Brief: SATA Power Management: “It’s Good to Be Green”; April 8, 2009
12. Write Amplification

Overview and Context

SSDs that use NAND Flash Memory as storage will have some data write amplification due to the nature of NAND Flash memory—that it must be erased before new data can be written—which requires extra NAND operations to move existing data possibly more than once. These extra NAND operations produce a multiplying effect that increases the number of writes required, producing an “amplification” effect; thus the term, “write amplification.” The write amplification factor constantly changes over the life of the SSD. Write amplification has many implications to the read/write performance and the reliability of the SSD. Depending on the SSD’s intelligence in managing its data, write amplification can cause extra wear and extra read/write/erase cycles on the NAND components reducing the life of the NAND Flash component. Additionally, the extra erase and write operations could cause an IOP latency outlier if these operations were done at inopportune times. Because write amplification is very impactful to the life of a SSD, SSD controller companies, such as LSI-SandForce*, have developed compression algorithms where the amount of data written to the NAND Flash is less for every host write. Therefore, with write compression techniques, an SSD can achieve a write amplification that is less than 1.0, which enhances the reliability of the NAND components because less data is written. While there are merits to compression techniques, it is not a panacea to all the challenges of developing great Solid-State Drives.

Definition and Explanation

Short:

Due to the nature of NAND Flash Memories in that it must be erased before new data can be written may require extra NAND operations to move existing data more than once. This extra movement of data may involve erases and writes to accommodate the single host write
request. These extra NAND write operations create a multiplying effect producing an “amplification” effect; thus the term, “write amplification”. Write amplification is typically measured by the ratio of writes coming from the host system and the number of actual writes required of the flash memory.

\[
\text{Write Amplification} = \frac{\text{Data Written to the Flash Memory}}{\text{Data Written by the Host System}}
\]

**Figure 10: Write Amplification Formula**

**Long:**

The following illustration shows the extra operations that NAND flash must do to accommodate one host write request:

1 Host Requests to Write
3 Pages of NAND Data:

**Resultant NAND Component Operations:**

1) 3 copy or write operations to move pages “A”, “B” and “C” from Block 2 to Block 1, 3, and 4 respectively
2) 1 erase operation of Block 2
3) Page Program Operations of Page 1, 2, and 3 into Block 2
4) Grand Total: 7 Program and 1 Erase Operation
13. RAS (Reliability, Availability, & Serviceability)

Overview and Context

RAS is a term created by IBM* to describe the robustness of their mainframe computers. In the past when mainframe computers were the hub of all computing servicing many remote terminal users, “uptime” was key feature or metric to the value of the mainframe computer. All users of such computers dreaded to hear that the computer (mainframe) was “down”. IBM wanted to differentiate their mainframes from others in that they had RAS features to ensure their computers would be “up” more than “down”. So, computers designed with higher levels of RAS have a host of features that help them stay available for long periods of time without failure—with some computer vendors offering uptimes on the order of years! While RAS was a term created for hardware, it is also being applied to software.

Definition and Explanation

Short:

RAS (Reliability, Availability, Serviceability) was a term created by IBM to differentiate their mainframe computers that offered higher levels of reliability features to detect and avoid crashing faults, availability even with a fault occurrence, and serviceability of repairs is fast and easy to ensure higher levels of “uptime”. Now this term once applied to mainframes is also being used for servers and data centers.

Long:

Computers designed with higher levels of RAS have a host of features that help them be Reliable, Available, and Serviceable.

Reliability means features that help avoid and detect faults. A reliable system does not silently continue and deliver results that include uncorrected corrupted data. Instead, it detects and corrects the corruption when possible.
**Availability** is the amount of time a device is actually operating as the percentage of total time it should be operating. Availability features allow the system to stay operational even when faults do occur.

**Serviceability** is the simplicity and speed with which a system can be repaired or maintained, and includes various methods of easily diagnosing the system when problems arise.

RAS features are available for most computer/server components, including:

**Processor**: Processor instruction error detection and instruction retry, including alternative processor recovery

**Memory**: Parity or ECC protection of memory components as well as memory bus

**I/O**: Cyclic redundancy check checksums for data transmission/retry and data storage

**Storage**: RAID configurations for Solid-state or magnetic disk storage; Journaling file systems for file repair after crashes; Checksums on both data and metadata, and background scrubbing

**Power/cooling**: Duplication of components to avoid failures (for example power-supplies). Systems are over-designed for the specified operating ranges of clock frequency, temperature, voltage, vibration. Temperature sensors are included to throttle operating frequency if temperatures are exceeded. Surge protector, uninterruptible power supply, and auxiliary power are also provided.

**System**: Hot swapping of components capability is provided. Predictive failure analysis is done to predict which intermittent correctable errors will lead eventually to hard non-correctable errors.

**References**:

1. Wikipedia.com :
   http://en.wikipedia.org/wiki/Reliability,_Availability_and_Serviceability
3. Whatis.com: (http://whatis.techtarget.com/definition/Reliability-Availability-and-Serviceability-RAS)
14. DevSleep (Device Sleep)

Overview and Context

In an effort to further reduce the power consumption of a solid-state drive at the platform level, the SATA-IO standards body added a new feature to the Serial ATA (SATA) specification. This new feature is designed to further reduce storage device battery consumption in portable computer products such as Intel® 4th Generation Core™ based Ultrabooks™ products. This new feature is DevSleep, or Device Sleep, which enables solid-state drives (SSDs) to enter a deeper lower power mode.

Definition and Explanation

**Short:**

DevSleep or Device Sleep is a feature in some SATA 3.0 solid-state storage devices that allows them to go into a lower power state, "device sleep", mode when sent the appropriate signal from the host controller chipset. This side-band active high signal enables the PHY to be powered down while enabling the drive to recover in 20ms. The modulated dev sleep signal instructs the SSD to enter the dev sleep mode allowing, one or two orders of magnitude less power than a traditional idle, about 5 mW and some as low as 2.5 mW.

**Long:**

With DevSleep enabled, a host has a middle ground between today’s interface power management states of Slumber, ~0.05 Watt and “off”, 0 Watt. It can now go into a low latency power mode where both the host and device PHY can be completely powered off, as well as possibly other sub-systems, but still maintain an exit latency much closer to Slumber than to a full shutdown. The DevSleep specification does not state what power levels a device will reach while in the DevSleep state, but SSDs are targeting 5mW or less.
The dev sleep block diagram illustrates how the side-band devslp signal from the host modulates the SSD in and out of the devslp modes and relative exit or recovery latency times from devslp.

![Dev Sleep Block Diagram](image)

**Figure 11: Dev Sleep Block Diagram**

**DevSleep operates as follows:**

- The host may assert the DEVSLP signal from any state, provided that:
  - Device supports the Device Sleep feature (per ATA IDENTIFY DEVICE command)
  - The Device Sleep feature is enabled by host (per ATA SET FEATURES command)
  - There are no commands outstanding
- On DEVSLP Assertion
  - Host must assert DEVSLP for >= 10ms, or as specified in Identify Device Data Log;
  - Host and device may power down PHY and other systems (e.g., PLL’s, clocks, media);
  - Neither host nor device shall initiate PHY communications while DEVSLP asserted
  - All PHY communications ignored by host and device while DEVSLP asserted
- On DEVSLP Negation
  - Device must detect OOB in <= 20ms, or as specified in Identify Device Data log
  - Host and device can use COMWAKE or COMRESET/COMINIT for renegotiation

**References:**

1. Intel and SanDisk Corporation White Paper: December 2011
15. NVMe (Non-Volatile Memory Express)

Overview and Context

The Advanced Host Controller Interface (AHCI) is a technical standard that specifies the operation of Serial ATA (SATA) storage devices. The specification describes a system memory structure for computer hardware for detecting, configuring, programming, and exchanging data between host system memory and attached storage devices. However, AHCI was developed and optimized for hard disk drives in its days of development, but with the onset of solid-state drives, a new standard, NVMe, was developed to fully utilize the capabilities of solid-state drives and the PCIe interface protocol. Additionally, with the SATA interface ending at 3.0 (6 Gb/s) provided further impetus for the transition to an established faster scalable interface, PCIe, from which NVMe is based.

Definition and Explanation

Short:

NVM Express, NVMe, or Non-Volatile Memory Host Controller Interface Specification (NVMHCI), is a high performance specification for accessing solid-state drives (SSDs) attached through the PCI Express (PCIe) bus for Data Center and Client systems. This open industry standard consisting of 90+ companies driven by 13 promoter group was incorporated in January, 2014. The interface specification provides for reduced latencies and faster performance with support for security and end-to-end data protection.

Long:

NVM Express, NVMe, or Non-Volatile Memory Host Controller Interface Specification (NVMHCI) is a high performance highly optimized specification that utilizes the full capabilities of solid-state drives. It takes full advantage of all the PCIe Gen 2 and Gen 3 bus performance and protocol benefits in delivery low latency direct attach CPU storage performance.
**NVMe Structure:**

![NVMe Structure Diagram]

**NVMe Simple Optimized Command Set:**

Only 10 admin commands and 3 I/O commands are required for NVMe:

<table>
<thead>
<tr>
<th>Admin Commands</th>
<th>NVM Admin Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) Create I/O Submission Queue</td>
<td>Format NVM (optional)</td>
</tr>
<tr>
<td>2) Delete I/O Submission Queue</td>
<td>Security Send (optional)</td>
</tr>
<tr>
<td>3) Create I/O Completion Queue</td>
<td>Security Receive (optional)</td>
</tr>
<tr>
<td>4) Delete I/O Completion Queue</td>
<td></td>
</tr>
<tr>
<td>5) Get Log Page</td>
<td></td>
</tr>
<tr>
<td>6) Identify</td>
<td></td>
</tr>
<tr>
<td>7) Abort</td>
<td></td>
</tr>
<tr>
<td>8) Set Features</td>
<td></td>
</tr>
<tr>
<td>9) Get Features</td>
<td></td>
</tr>
<tr>
<td>10) Asynchronous Event Request</td>
<td></td>
</tr>
<tr>
<td><strong>Firmware Activate (optional)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Firmware Image Download (optional)</strong></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NVM I/O Commands</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1) Read</td>
<td></td>
</tr>
<tr>
<td>2) Write</td>
<td></td>
</tr>
<tr>
<td>3) Flush</td>
<td></td>
</tr>
<tr>
<td><strong>Write Uncorrectable (optional)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Compare (optional)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Dataset Management (optional)</strong></td>
<td></td>
</tr>
<tr>
<td>Write Zeros (optional)</td>
<td></td>
</tr>
<tr>
<td>Reservation Register (optional)</td>
<td></td>
</tr>
<tr>
<td>Reservation Report (optional)</td>
<td></td>
</tr>
<tr>
<td>Reservation Acquire (optional)</td>
<td></td>
</tr>
<tr>
<td>Reservation Release (optional)</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 12: NVMe Structure**

**Figure 13: NVMe Command**
**NVMe Controllers and Drivers:**

With PCIe being a ubiquitous interface to so many devices, it is important to know and understand the differences with PCIe storage devices with its ACHI or NVMe supported. There will be performance differences between the ACHI and NVMe interfaces resulting in drive performance differences. The table and the flow diagram below highlights the interface feature and the operational flow differences.

<table>
<thead>
<tr>
<th>Feature</th>
<th>AHCI</th>
<th>NVMe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum queue depth</td>
<td>1 command queue; 32 commands per queue</td>
<td>65536 queues; 65536 commands per queue</td>
</tr>
<tr>
<td>Un-cacheable register accesses (2000 cycles each)</td>
<td>6 per non-queued command; 9 per queued command</td>
<td>2 per command</td>
</tr>
<tr>
<td>MSI-X (Message Signaled interrupts) and Interrupt steering</td>
<td>single interrupt; no steering</td>
<td>2048 MSI-X interrupts</td>
</tr>
<tr>
<td>Parallelism and multiple threads</td>
<td>requires synchronization lock to issue a command</td>
<td>no locking</td>
</tr>
<tr>
<td>Efficiency for 4 KB commands</td>
<td>command parameters require two serialized host DRAM fetches</td>
<td>gets command parameters in one 64 Bytes fetch</td>
</tr>
</tbody>
</table>

**Figure 14: Feature Comparison of AHCI and NVMe**
As with many storage devices, NVMe drivers are very important and many times a key performance differentiator in how well the driver is designed and optimized for a particular supplier’s NVMe supported solid-state drive. Driver supported for the following OS:

- Linux, Windows*, UEFI, FreeBSD
- Solaris (In Development)

**Hardware Connectors Supporting NVMe:**

Since NVMe is an interface protocol, there are no real hardware requirements or limitation, so all PCIe supported hardware will work with the NVMe interface. Example hardware connector interface supported are:

- 2.5-inch Form Factor
  - 15mm Z-height
  - 8639-compatible connector
- AIC Form Factor
  - Half-height, Half-length
  - Single slot x4 connector

**NVMe Development Core Philosophy:**

- Simplicity and Efficiency
- Architected for performance
- Scalable from Client to Enterprise
- Standardized, consistent feature set
- Supports the current and next generation of NVM

**References:**

4. SNIA NVMe Overview Presentation (Chander Chadha)
16. RBER & UBER

Overview and Context

Just like hard disk drives, NAND flash based solid-state storage devices are not intrinsically error-free but rely on error correction coding (ECC) to correct its raw bit errors. These raw bit errors are quantified in terms of RBER, raw bit error rate, and UBER, uncorrectable bit error rate.

Definition and Explanation

Short:

The bits of NAND data that contain incorrect data before applying ECC (error correction coding) is called the raw bit error rate (RBER). Some of these raw bad bits can be corrected by applying ECC and read out correctly. The resulting code word error rate per bit of data after ECC has been applied is called the uncorrectable bit error rate (UBER). These values are quoted typically as one sector in \(10^{13}\) to \(10^{16}\) bits read, smaller fraction is better.

Long:

There are several NAND component mechanisms that can lead to the creation of these raw bit errors that include program disturb, quantum level noise effects, erratic tunneling, SILC (Stress Induced Leakage Current) related data retention, read disturb, and detrapping-induced retention. The data error rate caused by the above mechanisms must be managed by the SSD and will determine the reliability of the SSD. One way that the SSD manages the error rate is through ECC methods.

The bits of NAND data that contain incorrect data before applying ECC (error correction coding) is called the raw bit error rate (RBER). Some of these raw bad bits can be corrected by applying ECC and read out correctly. The resulting code word error rate per bit of data after ECC has been applied is called the uncorrectable bit error rate (UBER). A code word is a fixed group of memory bits that the ECC engine and algorithm analyzes for error correction. These values are quoted typically as one sector in \(10^{13}\) to \(10^{16}\) bits read, lower is better for both, meaning, one sector in \(10^{16}\) is a smaller fraction and thus, better.
17. MTBF

Overview and Context

MTBF, Mean Time Between Failures, is a very common metric used to predict the elapsed time between inherent failures of a system during operation. These failures are assumed to be immediately repairable, unlike MTTF, Mean Time To Failure, where the failed system is not repairable.

Definition and Explanation

Short:

MTBF is defined as the predicted elapsed time between inherent failures of a system during operation.

MTBF refers to the failure rate of a drive over its expected lifetime. This doesn’t mean a 1.2 million hour MTBF drive will last 1.2 million hours, and a 1.5m MTBF drive will last 1.5 million hours, which equates to 136 to 171 years.

Long:

The general definition of MTBF is defined as the predicted elapsed time between inherent failures of a system during operation. As you can see, MTBF refers to the failure rate of a drive over its expected lifetime. This doesn’t mean a 1.2 million hour MTBF drive will last 1.2 million hours, and a 1.5m MTBF drive will last 1.5 million hours, which equates to 136 to 171 years, which is a long time.

Intel qualifies their SSDs using a workload (JEDEC JESD219A) that represents 20 GB of writes per day for 5 years. With this workload we have estimated a MTBF of 1.2 million hours. So what does this SSD MTBF mean for me? This generally indicates about 3 failures a year across 1000 drives that run 8 hours a day. In other words, you’d have a 0.3% chance of having a write operation failure within the drive’s warranty.

References:

18. M.2 SSD Form Factor

Overview and Context

With the ever shrinking computing platforms to be thinner, lighter, and ever so more portable, SSD’s have uniquely been able to keep up with the platform scaling factors. M.2 is such a smaller case less form factor SSD able to fit in thinner laptops, Ultrabooks*, tablets, and like devices.

Definition and Explanation

M.2 (formerly known as NGFF, Next Generation Form Factor) is a small form factor card and connector that supports applications such as Wi-Fi, WWAN, USB, PCIe & SATA, as defined in the PCI-SIG M.2 Specification. M.2 form factor has a fixed width of 22 mm, but is available in multiple lengths, 30, 42, 60, 80, and 100 mm and comes with various key ID’s that identify the product’s interface and functionality.

The M.2 specification provides four PCI Express lanes and one SATA 3.0 port, exposed through the same connector, allowing use of both PCI Express x2 and x4 and SATA storage devices in form of M.2 cards.

Short:

M.2 is a case less small form factor card and connector that supports applications such as Wi-Fi, WWAN, USB, PCIe & SATA protocols and functionality. M.2 comes in various lengths with various keys to identify its use and functionality.

Long:

M.2 (formerly known as NGFF, Next Generation Form Factor) is a small form factor card and connector that supports applications such as Wi-Fi, WWAN, USB, PCIe & SATA, as defined in the PCI-SIG M.2 Specification. M.2 form factor has a fixed width of 22 mm, but is available in multiple lengths, 30, 42, 60, 80, and 100 mm and comes with various key ID’s that identify the product’s interface and functionality. M.2 cards come in various single sided and double sided options to fit in thinner form factors.
Figure 16: M.2 Edge Connector Pinout and Key Identification

<table>
<thead>
<tr>
<th>Key ID</th>
<th>Notched pins</th>
<th>Provided interfaces</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>8–15</td>
<td>PCIe x2, USB 2.0, I(^2)C and DP x4</td>
</tr>
<tr>
<td>B</td>
<td>12–19</td>
<td>PCIe x2, SATA, USB 2.0 and 3.0, Audio, PCM, IUM, SSIC and I(^2)C</td>
</tr>
<tr>
<td>C</td>
<td>16–23</td>
<td>Reserved for future use</td>
</tr>
<tr>
<td>D</td>
<td>20–27</td>
<td>Reserved for future use</td>
</tr>
<tr>
<td>E</td>
<td>24–31</td>
<td>PCIe x2, USB 2.0, I(^2)C, SDIO, UART and PCM</td>
</tr>
<tr>
<td>F</td>
<td>28–35</td>
<td>Future Memory Interface (FMI)</td>
</tr>
<tr>
<td>G</td>
<td>39–46</td>
<td>Generic (not used in M.2 specification)</td>
</tr>
<tr>
<td>H</td>
<td>43–50</td>
<td>Reserved for future use</td>
</tr>
<tr>
<td>J</td>
<td>47–54</td>
<td>Reserved for future use</td>
</tr>
<tr>
<td>K</td>
<td>51–58</td>
<td>Reserved for future use</td>
</tr>
<tr>
<td>L</td>
<td>55–62</td>
<td>Reserved for future use</td>
</tr>
<tr>
<td>M</td>
<td>59–66</td>
<td>PCIe x4 and SATA</td>
</tr>
</tbody>
</table>
Figure 17: M.2 SSD Photo Illustration of Sample Lengths

References:

2. https://www.sata-io.org/sata-m2-card

*Other names and brands may be claimed as property of others