



Laboratory of the Year 50th Annual Awards Honoring Design Excellence for New and Renovated Laboratories

Entry deadline: February 15, 2016 Presented by: **R&D & Laboratory Design**

Articles

ADVERTISEMENT



R&D 100 2016 Entries are Now Open!
The 100 Most Innovative Technologies Introduced in the Last Year

Submit your entry today!
Submission Deadline: April 15, 2016

Solid State Drives: Part 1

Thu, 06/20/2013 - 9:16am

by John J. Barbara

LISTED UNDER: [Computers and Software](#) | [Digital Forensic Hardware](#) | [Digital Forensic Software](#)

 Get today's news and top headlines for forensics professionals - Sign up now!

While browsing in the laptop section at a computer store, you overhear the following conversation between the salesperson and a customer:

"You'll love this new model laptop. It comes with a 512 GB Serial ATA - 600 Solid State Drive (SSD). You know that SSDs have been around awhile and this is top of the line. This laptop will boot your typical OS about twice as fast and all your applications will load and run up to five to ten times faster compared to a laptop equipped with a traditional hard drive (HDD). And, this laptop is virtually shock resistant and uses less power compared to a traditional laptop."

"That's sounds great. I've been looking to replace my old laptop for awhile and this appears to be the one I'm looking for. SSDs are replacing traditional HDDs in computers and they are in virtually all other types of portable electronic devices being sold today. It's the way to go. Why would anyone want to buy a laptop with a traditional HDD?"

Since you are not familiar with SSDs, you are puzzled by the conversation. "Replacing HDDs with SSDs?" "Virtually shock resistant laptops?" "Faster when loading the OS and running applications?" These are all new concepts to you. Rather than displaying your lack of knowledge to the salesperson by asking questions, you prefer to learn about SSDs yourself, all the while wondering how long they have been available and how they actually work.

Brief History of SSDs

The origin of SSDs can be traced back to the 1950s when International Business Machines (IBM) was researching methods to make their products (mostly machines with motor-assist, i.e. mechanical machines), work better and faster.¹ They realized one way to accomplish this was to make their machines programmable. However, programmable machines would subsequently require both temporary and permanent memory storage capabilities. Already familiar with punched paper cards and punched paper tape for input and output storage, they understood this was not the answer to increased performance. A major drawback to this methodology was the fact that each time an operator wanted to run a machine, its software program and initial data, programmed and stored on punched paper cards or punched paper tapes, first had to be loaded into the machine. (This methodology was used on many early digital systems into the 1960s.)

IBM researchers soon developed magnetic tapes, magnetic strip cards, and magnetic disks to replace their paper equivalents. This technological advance was based upon the physical principal that an electromagnet would cause certain earth ferrite materials to become magnetized or demagnetized. Shortly thereafter, IBM began to use magnetic tapes and magnetic disk memory to provide their machines with non-volatile memory for program input and output. Termed "Charged Capacitor Read Only Store (CCROS)," this non-volatile memory is considered to be among the first SSDs and was the predecessor of today's memory storage devices (FLASH, EEPROMS, etc.).² Shortly thereafter, SSD use began to proliferate throughout the computer industry. For example in the 1970s and 1980s they were used for storage purposes in early mainframe supercomputers such as those from Amdahl, IBM, and Cray.³ In the late 1970s, a 16 KB RAM solid-state drive was developed by Texas Memory Systems to record seismic data generated by the oil industry during oil explorations.⁴ In 1987 EMC began producing SSD storage systems for the mini-computer market which were capable of providing twenty times the performance of magnetic disk drives.⁵ Jumping ahead to the 1990s, flash based SSDs were introduced by M-Systems and other companies.⁶ Both the aerospace and military industries began to utilize SSDs in mission critical applications where storage devices needed to withstand extreme vibration and shock and endure wide temperature fluctuations.⁷ Their growth, use, performance, and reliability has expanded since then to where now they can be found in all types of electronic devices. Foremost is their use in laptops and tablets where they serve as replacements for traditional hard drives. (For those interested in a more detailed history of SSDs, see references [1 and 4] cited below.)

Electronic circuitry comprised entirely of semiconductors is referred to as "solid state." The term dates back to the electronic circuitry used in the transistor radios developed during the 1950s. At that time, the semiconductor began to replace the vacuum tube which was previously used in radio construction. Today, all electronic devices are constructed around semiconductors and microchips. Pertaining to SSDs, they have *none* of the typical components found in the traditional hard drive and use semiconductors as the primary storage media. Figures 1 and 2 illustrate these differences.

The construction and design of SSDs provide many advantages versus traditional hard drives, some of which include:

- No moving parts
- No spin-up time or noise
- Very low power consumption
- Low random access times
- Very light-weight
- Unaffected by magnetic fields
- Shock and vibration resistant
- Ability to handle extremes of temperature

There are also a number of disadvantages to SSDs, particularly as they relate to their forensic examination for potential probative evidence. These drawbacks will be discussed in future columns.

(Note: any products or manufacturers mentioned in this article are not to be considered as an endorsement of that product or manufacturer by Forensic Magazine or the author.).

References

1. "Origin of Solid State Drives." http://www.storagereview.com/ssd_reference_guide. Retrieved 04-03-2013.
2. Ibid.
3. "IBM User's Guide, Thirteenth Edition." <http://web.utk.edu/~mnewman/ibmguide03.html>. Retrieved 04-09-2013.
4. "[SSD Market History - Charting the 30 Year Rise of the Solid State Disk Market.](#)" <http://www.storagesearch.com>. Retrieved 04-08-2013.
5. "A Brief History of EMC through 1998." <http://web.archive.org/web/20000621204120/>
6. http://www.emc.com/about/emc_story/brief_history.jsp. Retrieved 04-09-2013.
7. Odagiri, Hiroyuki; Goto, Akira; Sunami, Atsushi; Nelson, Richard R. (2010). *Intellectual Property Rights, Development, and Catch Up: An International Comparative Study*. Oxford University Press. pp. 224–227.
8. Drossel, Gary (2007-02). "Solid-State Drives Meet Military Storage Security Requirements". <http://mil-embedded.com/pdfs/SiliconSysts.Feb07.pdf>. Retrieved 04-08-2013.

John J. Barbara owns Digital Forensics Consulting, LLC, providing consulting services for companies and laboratories seeking digital forensics accreditation. An ASCLD/LAB inspector since 1993, John has conducted inspections in several forensic disciplines including Digital Evidence. jjb@digforcon.com

TOPICS [DIGITAL FORENSIC INSIDER](#) [COMPUTER FORENSICS](#) [DIGITAL FORENSIC HARDWARE](#) [JUNE/JULY 2013](#) [DIGITAL FORENSIC INVESTIGATION](#)

 Share { 2 } [Pin it](#) [Share](#) { 23 }



0 Comments Forensic Magazine

Login

Recommend [Share](#)

Sort by Best



Start the discussion...

Be the first to comment.

[Subscribe](#) [Add Disqus to your site](#) [Privacy](#)

DISQUS



Laboratory of the Year 50th Annual Awards Honoring Design Excellence for New and Renovated Laboratories

Entry deadline: February 15, 2016 Presented by: **RD & Laboratory Design**

Articles

ADVERTISEMENT



Join DFI News for an Educational Webinar Now Available ON-DEMAND

Exclusive Webinar: 5 Habits of Highly Effective Data Collection

Sponsored by: **ZAPPROVED**

[View it Today!](#)

Solid State Drives: Part 2

Wed, 08/28/2013 - 5:49am

by John J. Barbara

LISTED UNDER: [Digital Forensic Hardware](#)|[Digital Forensic Software](#)

[Get today's news and top headlines for forensics professionals - Sign up now!](#)

SSD Architecture and Function

One commonality between a typical hard drive and an SSD is that they both store data. However, the way in which they do so is totally different. To fully comprehend how SSDs function, it is necessary to understand SSD terminology. Doing so will also provide insight into the "pitfalls" of their forensic examination.

- **Controller:** Every SSD contains an internal embedded processor that functions as a bridge between its NAND flash memory and a host (such as a computer). The SSD's firmware provides code to the Controller for execution to perform all the data requests from the host. Controllers are responsible for the SSD's performance and its features. Typical features include reading and writing, erasing, encryption, error checking and correction (ECC), bad block mapping, garbage collection, RAISE, wear-leveling, and over-provisioning. The Controller is not to be confused with the actual I/O controller interface, which is typically a SATA interface used to physically attach the SSD to the host. The Controller and its NAND non-volatile memory are the two primary components of all SSDs.
- **NAND non-volatile memory:** The most common non-volatile memory in most SSDs is NAND Flash memory. To store information, flash memory uses an array of multiple-layer NAND memory cells which are connected in series and somewhat resemble a typical NAND gate. Each cell is similar to a standard Metal Oxide Semiconductor Field Effect Transistor (MOSFET, the most common type of transistor used for switching and amplifying electronic signals in analog and digital circuits). However, these specialized transistors contain two gates instead of one; a top control gate (CG) as in other MOS transistors, and a lower floating gate (FG) which resides between the CG and the MOSFET channel. The FG is electrically separated from the CG by an insulating layer consisting of a dielectric material.

A technique called "tunnel injection" (the process of injecting electrons through an electric insulator to an electric conductor) is used to charge (write to) the FG. Because the FG is electrically separated by the dielectric material, electrons become trapped in the FG and can remain there for many years without discharging. As long as the FG maintains its charge, it will alter the threshold voltage of the cell, partially cancel the electric field from the CG, and not allow a current to flow through the transistor. This is a very stable environment which allows minuscule amounts of charge to be stored for years without the need for any additional power. At this point in time, the FG bit state is "0" (Figure 1).

For a read-out to occur, more voltage would have to be applied for the channel to conduct. This would require an intermediate voltage between the threshold voltages to be applied to the CG causing the MOSFET channel conductivity to be tested to determine if it is conducting or insulating. When a current flow is sensed, it forms a binary code which then reproduces the stored data. The removal of electrons (uncharging or erasing) from the FG allows the current to flow and its bit state now becomes "1." This is known as "tunnel release" which is the reverse of tunnel injection (Figure 2). The addition of electrons to an FG and their removal is the basis for how non-volatile flash memory works.

- **Program Erase (P/E) Cycles and Wear-leveling:** There are many peculiar facts associated with NAND flash memory, one of which is that existing data in a particular location cannot be readily overwritten. To facilitate writing, there are two types of operations that can be performed: set all bits to "1" (erase) or set all bits from "1" to "0" (program). Bits cannot set from "0" to "1," they must all be reset to "1" (erase). To write to a particular block, all bits must be set to "1" (erase) and then the bits programmed appropriately from "1" to "0." However, before information can be rewritten, it must first be erased. Why is this so? All write operations occur on a block by block basis. To change data in a given block, that block must first be copied to another block, the original block must then be erased or cleared, and the copied block must be rewritten to another different cleared block along with the updated information. To enhance performance, SSDs set aside a certain percentage of space to ensure that there are always extra blocks available for wear-leveling and other performance optimizing features. This space is referred to as over-provisioning space and is reserved for the controller and is not available to the user.

Each read operation may introduce a potential error. After a number of reads to the same blocks, there is an increase in the likelihood of error for further consecutive reads from those blocks. Likewise, rewriting always requires the use of the erase program cycle. NAND flash memory cells can only be erased (written to) a finite number of times, generally 100,000 cycles, and each erase (write) operation reduces the life of the SSD, making it vulnerable to failure. This is known as the write-endurance or Program Erase (P/E) Cycle. To mediate this effect, the controller performs a technique called wear-leveling, using certain algorithms to dynamically arrange the data such that erasures are distributed over all of the memory cells. The result is that no one block will approach its P/E cycle due to constantly changing data. Wear-leveling reduces the potential premature wear or unreliability of the SSD and essentially ensures that all blocks will fail at the same time. There are two types of wear-leveling: dynamic and static. In dynamic leveling, a least erased block from a free list is chosen to be written to. Static leveling involves the periodic moving of "static" non-free blocks with a low erasure count (such as those blocks used for applications, the OS, etc.) to a block with a high erase count. This then allows those low usage blocks to be used more frequently.

This discussion will continue in the next Digital Forensics Insider column.

John J. Barbara owns *Digital Forensics Consulting, LLC*, providing consulting services for companies and laboratories seeking digital forensics accreditation. An ASCLD/LAB inspector since 1993, John has conducted inspections in several forensic disciplines including Digital Evidence. jjb@digforcon.com

TOPICS [DIGITAL FORENSIC INSIDER](#) [COMPUTER FORENSICS](#) [DIGITAL FORENSIC HARDWARE](#) [DIGITAL FORENSIC SOFTWARE](#)
[AUGUST/SEPTEMBER 2013](#) [DIGITAL FORENSIC INVESTIGATION](#)

Share 0 Share 7



0 Comments Forensic Magazine

Login

Recommend Share

Sort by Best

Start the discussion...

Subscribe Add Disqus to your site [Privacy](#)

DISQUS

Search Forensic Magazine



ADVERTISEMENT

R&D 100
54 Years of Innovation

The 100 Most Innovative Technologies Introduced in the Last Year

2016 Entries are Now Open!

Submit your entry today!

Submission Deadline: April 15, 2016

Trending

Evidence of Mass Extinction, Behind a South Jersey Hardware Store

2 comments · 6 days ago

White House Responds to 'Making a Murderer' Petition

Web2PDF

[converted by Web2PDFConvert.com](http://converted.by/Web2PDFConvert.com)

Exclusive Webinar: 5 Habits of Highly Effective Data Collection

Sponsored by: **ZAPPROVED**

View it Today!

Articles

ADVERTISEMENT

Exclusive Webinar: 5 Habits of Highly Effective Data Collection

Sponsored by: **ZAPPROVED**

View it Today!

Solid State Drives: Part 3

Tue, 10/22/2013 - 5:55pm

by John J. Barbara

LISTED UNDER: [Digital Forensic Consulting](#) [Digital Forensic Hardware](#)

Get today's news and top headlines for forensics professionals - Sign up now!

SSD Architecture and Function

NAND memory cells are comprised of MOS transistors which contain a floating gate (FG). There are two common memory cell types, Single Level Cell (SLC) and Multi Level Cell (MLC). SLCs save one bit (0 or 1) per transistor and because of this, they can be written to and read from very quickly. MLCs save two bits (00, 01, 10, or 11) per transistor, thus allowing twice as much data to be stored. However, the average read (2x) and write-speeds (3x) fall per NAND cell. The technique of "tunnel injection" is used to charge or write to an FG and once charged, the FG will maintain that state (i.e. store data) indefinitely. The FG can be uncharged or erased by removing electrons through a process called "tunnel release." Continual write/erase cycles will inherently cause degradation to the dielectric oxide insulation material in the FG leading to eventual failure of the cell. Typically, this would take about 30 KB - 1 MB Program Erase (P/E) Cycles for a SLC and 2.5 KB - 10 KB, P/Es for a MLC.

NAND memory cells are grouped into *Pages*, *Blocks*, *Planes*, *Dies*, and *TSOPs* (Thin Small Outline Packages which are the actual Integrated Circuit [IC] chips). Different NAND memory from different manufacturers will vary as to the amount of storage capacity. A typical 128 Gigabit (x8) NAND flash memory array is organized as follows:

- **Pages (x8)** consist of multiple memory cells and are the smallest structure to which data can be written. Pages are the basic programmable units of flash memory and are typically 4,314 Bytes. (See Figure 1). Of this total, only 4,096 Bytes (4KB) are usable, the remaining 218 Bytes are used for ECC and management. **[One Page (x8) = (4KB + 218 Bytes)]**
- **Blocks** are comprised of 128 Pages which provide 512KB of usable memory. An empty page in a block can be written to, but once written, that page cannot be overwritten. The entire block would have to be erased before that particular page (now erased) can be rewritten. **[(4KB + 218 Bytes) x 128 Pages = (512KB + 27KB)]**
- **Planes** are comprised of 2048 Blocks, providing a total of 8,624 Megabits of usable memory. **[(512KB + 27KB) x 2,048 Blocks = 8,624 Megabits]**
- **Dies** consist of 2 Planes, each of which provides a total of 17,248 Megabits of usable memory. **[1 Die = 8,624 Megabits x 2 Planes = 17,248 Megabits]**
- **TSOPs** (the actual IC chips that are placed on the circuit board) normally consist of two or more Dies, however they could contain as many as eight. Figure 1 shows two Dies providing a total of 34,496 Megabits: **[17,248 Megabits x 2 Dies = 34,496 Megabits]** of usable memory. For a 128 Gigabit device, the 32 Gigabit array would apply to each Chip Enable (CE#, CE2#, CE3#, and CE4#). Chip Enable input is used to select memory devices, activate memory control logic, input buffers and decoders, and sense amplifiers.
- **8 Bits = 1 Byte.** The 34,496 Megabits translates to a TSOP consisting of 16 Gigabytes (GB). **[34,496 Megabits / 8 Bits = 16 Gigabytes (GB)]**. Adding multiple TSOPs to a circuit board (usually eight or more) provides the total memory capacity of an SSD. Figure 2 illustrates a typical SSD containing eight TSOPs, with a total of 128 Gigabytes (GB) of storage capacity. There are currently SSDs available with capacities of more than 600 Gigabytes (GB).

How It All Works Together

The architecture of a typical Controller is shown in Figure 3. The Flash File System (FFS) is firmware designed specifically to enable files to be stored in flash memory with each sub-layer performing a specific function. Typical functions include wear-leveling, garbage collection (GC), bad block management (BBM), error checking and correction (ECC), over-provisioning (OP), reading and writing, erasing, and encryption.

Wear-Leveling (WL)

Blocks containing frequently updated information are subjected to many more P/Es as compared to blocks which are rarely updated. After a number of reads to the same blocks, there is an increase in the likelihood of error for further consecutive reads because MLC NAND memory can only be erased (written to) a finite number of times (~100,000 P/Es). To avoid potential failure, the number of P/Es for each page is monitored as is the maximum number of allowed P/Es for each block (i.e. its endurance). Each time the host sends update information to the same logical sector, the controller will map the sector to a different physical sector, tagging the now out of date sector as eligible for erasure. This allows all physical blocks to be evenly used. If not for wear-leveling, most SSDs would quickly fail if the logical and physical block addresses were mapped one to one.

There are two types of wear-leveling, dynamic and static. In dynamic leveling, a least erased block from a free list is chosen to be written to. Static leveling involves the periodic moving of "static" non-free blocks with a low erasure count (such as those blocks used for applications, the OS, etc.) to a block with a high erase count. This then allows those low usage blocks to be used more frequently. Dynamic leveling and static leveling require the availability of free sectors which can be filled with updated information.

Garbage Collection (GC)

When a user empties the Windows recycle bin or deletes data on an SSD, no actual erasing takes place. The Windows OS uses the *TRIM* command to notify the SSD controller that certain pages contain stale data and to mark those pages as no longer being valid. They are waiting to be reclaimed and made available for reuse. The controller then knows not to relocate this stale data, which then decreases the number of P/Es.

NAND memory cannot directly overwrite existing data. Data can only be written page by page and erased block by block. Once the number of free sectors falls below a set threshold, the GC module selects the blocks containing the marked, invalid sectors, copies the latest valid copy onto pages in free sectors in another block or blocks and then erases those blocks which contained the old data. It also consolidates pages by moving and rewriting them from multiple blocks to newer blocks. This now provides free storage sectors in the old blocks. The data is only erased when new data is being written to the drive (except for a brand new drive).

This discussion will continue in the next Digital Forensics Insider column.

John J. Barbara owns Digital Forensics Consulting, LLC, providing consulting services for companies and laboratories seeking digital forensics accreditation. An ASCLD/LAB inspector since 1993, John has conducted inspections in several forensic disciplines including Digital Evidence. jjb@digforcon.com

TOPICS [DIGITAL FORENSIC INSIDER](#) [COMPUTER FORENSICS](#) [DIGITAL FORENSIC HARDWARE](#) [OCTOBER/NOVEMBER 2013](#) [DIGITAL FORENSIC INVESTIGATION](#)



Share 0 Share 3



0 Comments Forensic Magazine Login

Recommend 1 Share Sort by Best

Start the discussion...

Be the first to comment.

Subscribe Add Disqus to your site [Privacy](#) DISQUS

Search Forensic Magazine

ADVERTISEMENT



Articles

ADVERTISEMENT



Solid State Drives: Part 4

Sat, 01/04/2014 - 4:28am

by John J. Barbara

Digital Forensics Consulting, LLC

LISTED UNDER: Digital Forensic Consulting | Digital Forensic Hardware

[Get today's news and top headlines for forensics professionals - Sign up now!](#)

SSD Architecture and Function

The way in which an SSD stores data is totally different from how data is stored on a traditional hard drive. To fully comprehend how an SSD functions and provide insight into their forensic examination, it is necessary to understand SSD terminology. (Note: Controllers, NAND non-volatile memory, and Program Erase Cycles (P/E) were discussed in Part 2. Pages, Blocks, Planes, Dies, TSOPs, Wear-Leveling (WL), and Garbage Collection (GC) were discussed in Part 3).

- **Write Amplification (WA)**

This is essentially a measure of how efficiently the controller operates. Normally the amount of physical information written to the SSD is greater than the logical amount being provided by the host. Although this will negatively impact performance and P/E cycles, it is necessary to optimize the performance and maximize the lifespan of the SSD. The primary causes of WA are Wear-Leveling and Garbage Collection. WA requires that there be free space available on the SSD, ~20% of its capacity, to serve as a buffer.

- **Over-Provisioning (OP)**

Generally ~20% of the space on an SSD is reserved for the controller to use in managing the Flash File System. Since this space is normally not available to the user, an OP enabled SSD will have non-traditional actual storage capacities (such as 120 Gigabytes instead of the usual 128 Gigabytes and so on). OP allows the controller to create pre-erased blocks that are ready to be used. This extra capacity can also be used to replace some of the bad memory blocks.

- **Bad Block Management (BBM)**

An SSD has billions of flash cells. Due to the sheer number, a certain percentage (~2%–10%) will be unusable, unreliable, or prone to errors. The SSD design process includes a built-in bad block budget. Until the budget is reached, the SSD will operate satisfactorily, however, once the budget of bad blocks is exceeded, the SSD generally fails. The BBM module maintains the mapping of all bad blocks and is created at the time the memory is initialized and the NAND memory modules tested by the manufacturer.

Once in use, data in normally good blocks can be corrupted in a number of ways such as interference from writes in adjacent blocks, charge leakage, variability in design tolerances of the read/write process, and so forth. The controller remaps each time it writes to a block. When it determines a block is bad, that block is marked as being unusable and the BBM module is updated to keep track. This occurs throughout the SSD's lifetime.

- **Ensuring Data Integrity**

With a traditional HDD, data is normally written to free sectors and at a later time, read from those same sectors. If the data is changed, the HDD controller will simply overwrite the original sector and assign a new CRC value. On the other hand, an SSD erases blocks of data before storing new data and then uses newly erased blocks to store data that has changed. This leaves the old "stale" data unchanged in the old block(s). Address translation and versioning techniques then become necessary to prevent the drive controller from returning "stale" data. There are several methodologies which can be used to maintain data integrity.

- **Error Correction Code (ECC)**



Solid State Drives: Part 4

These date back to the 1950s. Essentially, they are mathematical algorithms that work by checking individual data bits as they move between the host and its storage device(s). ECC is common to both traditional HDDs and flash memory devices, however, ECC serves a more critical role in the function of an SSD. The enormous number of flash cells and their architecture predisposes the fact that some of them will not function at all. Over time, many NAND memory cells will deteriorate and fail during normal operations. This can lead to random bit errors and data corruption in the data stored on the various pages. Although the chance of a particular stored bit becoming corrupted is very small, the vast number of stored bits dramatically increases the probability. It is the function of the SSD controller to detect and correct errors when they occur in the storage system. Specifically, to maintain data integrity, ECC enables the SSD controller to detect and correct errors. The data is written to pages and then read back using various ECC algorithms to ensure that it has not been changed or corrupted.

- **Cyclic Redundancy Check (CRC)**

CRC normally provides “end-to-end” protection of data by ensuring that the data written and stored is the same when it is returned and read. Data normally passes through RAM, firmware, and other components of the SSD. At any juncture, software or hardware errors could cause the drive to change the data before writing it to memory or after reading it and providing it to the user. These errors can occur without the user’s knowledge. CRC can prevent these occurrences from happening. The drive controller generates a CRC value for data as it passes through the interface and embeds this value with the file’s other metadata. The controller can use a simple CRC check to determine if the data about to be provided matches what was originally stored. If the CRC value does not match, the drive reports that an error has occurred, thereby notifying the user of potentially incorrect data. It is important to note that the CRC cannot correct the error(s).

- **Logical Block Address (LBA), Correct Address Translation, and Correct Data Version**

SSDs cannot directly overwrite data in flash memory. The controller has to read the block to be modified, modify the data to be changed in the buffer, and then write the data to an entirely new block in a new location. This leaves the old “stale” data in its original block, and potentially multiple slightly different versions of the same data in multiple blocks. The SSD’s LBA table has to be continually updated such that it can properly identify the correct address or location of the newest data. Otherwise, since the old “stale” data does not contain a record of being rewritten to a new block, the drive controller could point to an old location and return old “stale” data. Even though ECC and CRC would work properly, the user would get the old data instead of the latest correct data.

This discussion will continue in the next Digital Forensic Insider column.

John J. Barbara owns Digital Forensics Consulting, LLC, providing consulting services for companies and laboratories seeking digital forensics accreditation. An ASCLD/LAB inspector since 1993, John has conducted inspections in several forensic disciplines including Digital Evidence. jib@digforcon.com

TOPICS [DIGITAL FORENSIC INSIDER](#) [COMPUTER FORENSICS](#) [DIGITAL FORENSIC HARDWARE](#) [DECEMBER 2013/JANUARY 2014](#) [DIGITAL FORENSIC INVESTIGATION](#)

Share 1 | Pin it | Share 7

0 Comments Forensic Magazine

Login

Recommend Share

Sort by Best

Start the discussion...

Be the first to comment.

Subscribe Add Disqus to your site Privacy

DISQUS

Search Forensic Magazine

ADVERTISEMENT

**R&D
100**
54 Years of Innovation

2016 Entries are Now Open!

The 100 Most Innovative Technologies
Introduced in the Last Year

**Submit
your entry
today!**

Submission Deadline:
April 15, 2016

DFI News
Digital Forensic Investigator®

[Home](#)
[EXCLUSIVES](#)
[NEWS](#)
[COLUMNS](#)
[DIGITAL FORENSICS](#)
[PRODUCTS](#)
[RESOURCES](#)
[FREE SUBSCRIPTIONS](#)

[LOG IN](#)
[REGISTER](#)

Articles

ADVERTISEMENT

Laboratory of the Year 50th Annual Awards
 Entry deadline: February 15, 2016
 Presented by: **R&D & Laboratory Design**

Solid State Drives: Part 5

Wed, 04/30/2014 - 10:37am

by John J. Barbara

LISTED UNDER: [Computers and Software](#) [Digital Forensic Consulting](#) [Digital Forensic Hardware](#)

Get today's news and top headlines for forensics professionals - Sign up now!

SSD Architecture and Function

Controllers, NAND non-volatile memory, and Program Erase Cycles (P/E) were discussed in Part 2. Pages, Blocks, Planes, Dies, TSOPs, Wear-Leveling (WL), and Garbage Collection (GC) were discussed in Part 3. Write Amplification (WA), Over-Provisioning (OP), and Bad Block Management (BBM) were discussed in Part 4.

Brief Discussion of Cylinders, Heads, and Sectors

Early traditional hard drives were supported by a PC's BIOS using Cylinder, Head, and Sector (CHS) addressing. Data was written using movable recording heads which were controlled via drive control commands. Once stored, the data could then easily be read by moving the heads over a particular cylinder. However, to read or write from a specific sector, that sector had to be specified in terms of its CHS. The combined limitations of the BIOS Int 13h routines and the IDE/ATA standard restricted the capacity of early hard drives to 504 MB ($1024 \text{ Cylinders} * 63 \text{ Sectors per Track} * 16 \text{ Heads} * 512 \text{ Bytes per Sector} = 528 \text{ Million Bytes or } 504 \text{ MBs}$). To circumvent the 504 MB size limit, Extended CHS addressing was implemented. Although this added a translation step that changed the way the hard drive geometry appeared to the BIOS, CHS addressing was still used. Unfortunately this introduced another size limiting factor for hard drives, namely the 8 GB barrier [$1024 \text{ Cylinders} * 63 \text{ Sectors per Track} * 256 \text{ Heads} * 512 \text{ Bytes per Sector} = 8 \text{ GBs}$].

Logical Block Addressing, and Physical Block Addressing

Logical Block Addressing (LBA) was developed to circumvent this issue and is now the method used with conventional hard drives to translate the CHS of the drive into addresses that can be used by an enhanced system BIOS. Instead of referring to CHS, each sector is assigned a unique "section number," starting at "0" and ending at "N-1" where "N" represents the number of sectors on the disc. (As an analogy, CHS can be considered as an individual's home address which is comprised of the street number, street name, city name, and state name. LBA would be analogous to every house in every state having a unique identifying number.) LBA itself is a run time function of a system's BIOS which uses LBA for commands such as reading, writing, format tracks, and so forth. Information pertaining to the hard drive's actual true geometry is stored in the system CMOS. LBA BIOS performs a translation from the traditional MS-DOS Track, Head, and Sector to the logical block numbers used by the drive.

Although they function totally differently, from the perspective of the host OS, an SSD appears similar to a conventional hard drive with rotating discs. The Logical to Physical Sector Block Address Translation Layer manages the placement of sectors. The SSD's Controller constantly writes new data or updates previous data to the first available free block which contains the least number of writes. This is to ensure that the number of write cycles per block is minimized, thereby maximizing the drive's longevity. Blocks containing old data are marked as "not in use" by the host OS. However, the data remains in the blocks until eventually erased by the GC function. The constant movement of data between blocks and pages can result in parts of any file being stored in any physical sector. The data's location, its Physical Block Address (PBA), must be tracked. To maintain organization, the Controller uses a mapping table to remap the LBA to the PBA. The table is referred to as the Logical to Physical Block Address Translation Table, or LBA-PBA Translation Table and has to be continually updated such that it can properly identify the correct address or location of data. As long as the index is changed when the data is physically moved, the data can still be located. (This is somewhat analogous to the function of the index of a book which points to the page number or location of a specific topic.) It is important to note that the physical location of any block will inevitably not match the external Logical Block Address.

"TRIM" Command

A traditional hard drive with an NTFS file system contains a Master File Table (MFT). The MFT is essentially an index file which maps everything on the hard drive. All file, directory, and metafile data (size, date and time stamp, data location, data content, permissions, etc.) is stored in MFT entries or in space outside the MFT that is described by MFT entries. When a user deletes a file, the file's MFT entry is marked as free and available for reuse. However, the actual disk space where the file is located is not reallocated and the data is not deleted, removed, or relocated. Essentially, all the hard drive "knows" is that this space can be reused at some future time. When additional space is needed, the OS will send new data to that location, directly overwriting the old data.

This is not the case with an SSD. An SSD uses OP to improve its longevity and overall performance. However, at some point, an SSD can eventually fill up with both valid and invalid data which can reduce its OP functionality and its performance. NAND memory pages containing old or invalid data cannot be directly overwritten.

Rather, they must first be erased at the block level using the Garbage Collection function. Unlike the traditional hard drive, an SSD does need to "know" what data is old or invalid so it can be moved and eventually deleted. The TRIM command (an innovation in storage architecture) is used by the OS to identify which addresses no longer hold valid data and which are available for clearing and re-use. The SSD then takes those addresses and updates the LBA-PBA Translation Table marking the addresses as invalid. During GC, the SSD does not move that invalid data. The net effect is a reduction in the number of write cycles and an increase of the SSD's longevity. This also provides additional space for OP. The contents of the blocks are not actually erased by the TRIM command, but rather it adds them to a queue of pending blocks which are eventually cleared by the GC function.

This discussion will continue in the next Digital Forensic Insider column.

John J. Barbara owns Digital Forensics Consulting, LLC, providing consulting services for companies and laboratories seeking digital forensics accreditation. An ASCLD/LAB inspector since 1993, John has conducted inspections in several forensic disciplines including Digital Evidence. jib@digforcon.com

TOPICS [DIGITAL FORENSIC INSIDER](#) [COMPUTER FORENSICS](#) [DIGITAL FORENSIC CONSULTING](#) [DIGITAL FORENSIC HARDWARE](#) [FEBRUARY/MARCH 2014](#)
[DIGITAL FORENSIC INVESTIGATION](#)





0 Comments Forensic Magazine

Login

Recommend Share


Sort by Best

 Start the discussion...

Be the first to comment.

Subscribe Add Disqus to your site Privacy

DISQUS

Search Forensic Magazine 

ADVERTISEMENT

Join DFI News for an Educational Webinar

Exclusive Webinar: 5 Habits of Highly Effective Data Collection

Now Available ON-DEMAND



Sponsored by: **ZAPPROVED**

[View It Today!](#)

Trending

Evidence of Mass Extinction, Behind a South Jersey Hardware Store

2 comments · 6 days ago

White House Responds to 'Making a Murderer' Petition

4 comments · 1 week ago

Robotics to the Rescue: Automated Sample Processing

1 comment · 4 days ago

Forensic Expert Finds Trace Blood on Revolutionary War Bullets

Exclusive Webinar: 5 Habits of Highly Effective Data Collection

Sponsored by:
ZAPPROVED™
[View it Today!](#)

Articles

ADVERTISEMENT

Join DFI News for an Educational Webinar

Now Available ON-DEMAND

Exclusive Webinar: 5 Habits of Highly Effective Data Collection

Sponsored by:
ZAPPROVED™
[View it Today!](#)

Solid State Drives: Part 6

Wed, 06/25/2014 - 8:29am

by John J. Barbara

Digital Forensics Consulting, LLC

LISTED UNDER: [Digital Forensic Hardware](#) | [Digital Forensic Software](#)

[Get today's news and top headlines for forensics professionals - Sign up now!](#)

SSD Architecture and Function

Previously Controllers, NAND non-volatile memory, and Program Erase Cycles (P/E) were discussed in Part 2. Pages, Blocks, Planes, Dies, TSOPs, Wear-Leveling (WL), and Garbage Collection (GC) were discussed in Part 3. Write Amplification (WA), Over-Provisioning (OP), and Bad Block Management (BBM) were discussed in Part 4. Cylinders, Heads, and Sectors, Logical and Physical Block Addressing, and the "TRIM" Command were discussed Part 5.

Solid State Drive Forensic Issues

When compared to a typical hard drive, SSDs are totally different in design and functionality which leads to some difficult issues to deal with pertaining to their forensic analysis. The SSD's use of flash memory for data storage rather than rotating magnetic discs is the cause of the forensic issues. Some important differences between SSDs and hard drives include the following:

- Flash memory in an SSD is typically divided into 4KB pages, not divided into the traditional 512 byte block size(s) as commonly found on hard drives.
- A flash memory page of NAND-based transistors must be totally erased before it can be reused (rewritten).
- The SSD controller spreads data over a number of pages to avoid excessive wear and failing pages. Each page can only be erased and rewritten a finite number of times (~10,000 – 100,000). Contrast that to a hard drive sector which can be erased and rewritten almost an unlimited number of times.
- The SSD controller handles significantly more tasks and functionality, such as wear leveling, over-provisioning, and garbage collection, which have no counterparts in hard drive architecture or function.
- Data stored on an SSD is subject to "self-corrosion" whereby it can be permanently erased or altered in the absence of commands from a user. When a user deletes a file on a hard drive, the file's MFT entry is marked as free and available for reuse. The actual disk space is not reallocated and the data is not deleted, removed, or relocated. To permanently erase the data requires direct user intervention.



To put everything into perspective regarding SSD forensics, consider the following case scenario:

One evening after work, a disgruntled employee successfully hacked into his company's Human Resources Department database and downloaded the confidential files of several co-workers. Since the company had previously disabled all USB ports on the computers, he was forced to print the information. After printing, he deleted the downloaded files and performed a quick format of the SSD itself. The next day, members of the company's internal IT security discovered that several confidential personnel files were illegally accessed and downloaded to a company computer. They were able to trace the security breach to the disgruntled employee's computer and its SSD was seized and turned over to the company's forensics division. Following "best forensic practice," the forensic examiner attached the SSD to a write blocker and proceeded to generate both its cryptographic hash value and its forensic image. When the imaging process was complete, he re-hashed the image to verify its integrity, only to determine that the two hash values were different. Proceeding with the examination using his forensic software, the examiner was not able to recover any deleted files from the SSD regarding confidential employee information.

Hash Value Discrepancy

The fact that there were two different hash values generated for the image of the SSD is problematic. One of the "golden rules" of digital forensics is to always preserve the evidence in its original condition (or to explain how and why it was altered). The use of write blockers and hash values has served as the methodology to provide proof that original digital evidence has not been altered. Write blockers prevent user-induced modifications to the data stored on a drive, thereby preserving the data in its original condition. Once calculated, a hash value serves as a "digital fingerprint" of the digital evidence. If that digital evidence is re-imaged at a later date, it should generate the same hash value. Matching hash values thus provides the verification that there were no changes to the original digital evidence. However, a very

important and critical fact to remember is that a hash value is unknown until it is calculated. Obtaining different hash values of the same hard drive is not an unheard of occurrence. After its hash value has been determined, changing even one bit of data in a file anywhere on a hard drive will result in a different hash value. Likewise, a different hash value will be obtained if a sector or cluster becomes unreadable on one of the hard drive's platters after it was originally hashed. In the above scenario there were two different hash values generated for the same image. The examiner needs to be able to explain this discrepancy and its impact upon the forensic analysis.

Wear-leveling Impact

The architecture and function of the SSD controller itself is the cause of the different hash values. The SSD flash file system firmware is designed specifically to enable files to be stored in flash memory with each sub-layer performing a specific function such as wear-leveling and garbage collection. The fact that the examiner attached the SSD to a write blocker has no effect on the SSD controller's ability to perform its designed functions. The SSD firmware is totally independent from any commands that are issued by the computer to which it is attached via a write blocker. Wear-leveling functionality is designed to increase the life of an SSD by avoiding constantly storing a charge (i.e. data) in the same group of transistors. Thus, unlike a typical hard drive which tends to store blocks of data for a file close to each other, the SSD randomly spreads the data (load) across all unused transistors. To the computers OS, this is transparent and the SSD appears similar to a conventional hard drive with rotating discs. The SSD's logical to physical sector block address translation layer manages the placement of sectors and the controller provides an abstracted list of hard drive sectors to the computer OS. A problem arises in that the OS cannot retrieve any file without going through the SSD controller which has a map of the location(s) where the file is actually stored. If for some reason, the SSD controller becomes corrupted, attempting to physically replace it with a similar one will result in the contents of the drive being viewed as a scrambled conglomeration of data. Also, data integrity would be very questionable and the resulting forensic image would not bear any resemblance to the original.

John J. Barbara owns *Digital Forensics Consulting, LLC*, providing consulting services for companies and laboratories seeking digital forensics accreditation. An ASCLD/LAB inspector since 1993, John has conducted inspections in several forensic disciplines including Digital Evidence. jib@digforcon.com

TOPICS [DIGITAL FORENSIC INSIDER](#) [COMPUTER FORENSICS](#) [DIGITAL FORENSIC HARDWARE](#) [DIGITAL FORENSIC INVESTIGATION](#) [DIGITAL FORENSIC SOFTWARE](#)
[APRIL/MAY 2014](#)

Share 0 [Print](#) Share 3



0 Comments Forensic Magazine

Login

Recommend Share

Sort by Best

Start the discussion...

Be the first to comment.

Subscribe Add Disqus to your site Privacy

DISQUS

Search Forensic Magazine

ADVERTISEMENT

SHIMADZU
Next-generation
FTIR Spectrophotometer
The fast, sensitive,
easy-to-use
platform for
forensics
applications

[learn more, click here](#)

Exclusive Webinar: 5 Habits of Highly Effective Data Collection

Sponsored by:
ZAPPROVED
[View it Today!](#)

Articles

ADVERTISEMENT

Join DFI News for an Educational Webinar

Now Available ON-DEMAND

Exclusive Webinar: 5 Habits of Highly Effective Data Collection

Sponsored by:
ZAPPROVED
[View it Today!](#)

Solid State Drives: Part 7

Wed, 08/27/2014 - 8:12am

by John J. Barbara

Digital Forensics Consulting, LLC

LISTED UNDER: [Digital Forensic Hardware](#) | [Digital Forensic Software](#)

[Get today's news and top headlines for forensics professionals - Sign up now!](#)

Solid State Drive Forensic Issues

An SSD's design and functionality causes a number of forensic issues that an examiner has to consider before beginning a forensic examination. Some important differences between SSDs and HDDs, Hash Value Discrepancies, and the impact of Wear-Leveling were discussed previously in Part 6.

Effect of Garbage Collection: SSD technology requires that a page or a block be erased before a write operation can occur. Inherently, this will slow down the write process. To circumvent this issue, the SSD Controller initiates its GC functionality to identify pages or blocks which are not in use and erases them as soon as possible. A problem arises in that neither the drive nor the Controller knows which blocks are occupied by OS system structures or user files or which blocks are no longer in use (stale). Via the "TRIM" command, the OS identifies which addresses no longer hold valid data and which are available for clearing. The Controller then initiates GC as a background process to erase the contents of the blocks within a few minutes of being marked for deletion, thereby preparing them for future re-use. Blocks of data processed in this manner are physically erased and any information they may have contained cannot be recovered. This is referred to as "self corrosion." Currently, it is not possible to prevent this from occurring using normal forensic tools or forensic methodology. (A potential solution to prevent "self-corrosion" would be to physically detach the Controller from the SSD. However this would require a level of technical expertise far beyond the capabilities of virtually all examiners. It would also require the use of some very sophisticated software and hardware which may or may not be available).

GC does not require user intervention. It is initiated by the SSD firmware and not via any commands issued by the computer. This is problematic as GC typically starts and completes within a few minutes of the SSD being powered on. The process will automatically occur regardless of whether an examiner attempts to obtain a "live" forensic image or uses a software or hardware-based write-blocker to obtain a "post mortem" forensic image. Potentially, probative data from slack or unallocated space will be erased while the SSD is being viewed "live" or is being forensically imaged! The fundamental premise of forensic imaging is to create an exact image of a computer's hard drive. This allows the examiner to demonstrate that no changes have occurred and that the imaged drive represents the exact information that was on the original hard drive. Regarding SSDs, this premise is violated and raises a serious issue concerning the validation or verification of the integrity of the data. After forensically imaging an SSD, repeating the imaging will generally result in a different hash value.

Quick Format Issue: If data were stored on a typical HDD and a quick format command were issued, all that would occur is that the file system would be reset but the data would remain persistent until actually overwritten at a later time. An examiner using typical forensic software should be able to recover most, if not all of the data "post mortem" from the drive. However, if the data is stored on an SSD and a Quick Format command is initiated by a user, the GC process occurs and examining the SSD to obtain probative data now becomes virtually impossible. By the time an examiner attempts to preview the SSD "live" with a triage tool, GC will almost certainly have permanently erased the blocks and the drive will not provide any probative information. The time interval from an initial format command to complete erasure can be a little as two to five minutes. The same situation will occur if the computer is powered down and the SSD examined "post mortem."

Problems with SSD Encryption: Encryption is generally considered an excellent security mechanism to safeguard sensitive data from being compromised. For many years, software tools have been available to encrypt data on hard drives. However, until recently, encrypting a hard drive using a software solution required considerable processing power and usually had a detrimental effect upon overall drive performance. SSDs are available which implement real-time AES encryption and decryption within the Controller hardware with little to no effect on performance. Normally the disk encryption key is encrypted with Advanced Technology Attachment (ATA) passwords and is only accessible after successful user authentication. To decrypt the encryption key, the ATA password is necessary. Since



SSDs have been in use since the 1950s. When compared to a conventional HDD, they are totally different in architecture and functionality. These differences offer many advantages such as no moving parts, having low random access times, and being shock and vibration resistant. However, their architecture and functionality creates some difficult issues to deal with pertaining to their forensic analysis.

encryption and decryption occurs within the Controller, this essentially protects the keys from being compromised since they are not present in the CPU or in RAM where they could be obtained using forensic analysis tools. Encryption is always on and there is no mechanism for a user to remove or disable protection short of removing the Controller itself. With an encrypted SSD, it is highly unlikely that a forensic examination will yield any meaningful data. Rather than performing a quick format as previously discussed, if the AES key is destroyed, no data on the SSD can be read and it can be considered securely erased.

Summary

SSDs have been in use since the 1950s. When compared to a conventional HDD, they are totally different in architecture and functionality. These differences offer many advantages such as no moving parts, having low random access times, and being shock and vibration resistant. However, their architecture and functionality creates some difficult issues to deal with pertaining to their forensic analysis. Processes such as wear leveling and garbage collection lead to “self-corrosion” of data and irrespective of whether or not the drive is imaged “live” or write-blocked and examined “post mortem,” there is no expectation of being able to recover deleted data. Very problematic is that the SSD can clearly self-modify its data after being imaged, leading to Hash Value discrepancies and data corruption. Lastly, an encrypted SSD will most likely not be able to provide any meaningful data. With all these and other inherent forensic issues, probably the best approach for an examiner is to treat an SSD as any other piece of volatile evidence. Examiners would need to have a full understanding of SSD architecture and functionality and then rely on extensive documentation of their forensic methodology and procedures to (hopefully) successfully obtain and maintain the evidentiary value of any probative information gathered.

John J. Barbara owns *Digital Forensics Consulting, LLC*, providing consulting services for companies and laboratories seeking digital forensics accreditation. An ASCLD/LAB inspector since 1993, John has conducted inspections in several forensic disciplines including Digital Evidence. jjb@digforcon.com

TOPICS [DIGITAL FORENSIC INSIDER](#) [COMPUTER FORENSICS](#) [DIGITAL FORENSIC HARDWARE](#) [DIGITAL FORENSIC INVESTIGATION](#) [DIGITAL FORENSIC SOFTWARE](#)
[JUNE/JULY 2014](#)





0 Comments Forensic Magazine

Login

Recommend Share

Sort by Best

 Start the discussion...

Be the first to comment.

Subscribe Add Disqus to your site Privacy

DISQUS

Search Forensic Magazine



ADVERTISEMENT

Join DFI News for an Educational Webinar

Exclusive Webinar: 5 Habits of Highly Effective Data Collection

Now Available ON-DEMAND



Sponsored by: **ZAPPROVED**

View It Today!

Trending

Evidence of Mass Extinction, Behind a South Jersey Hardware Store

2 comments · 6 days ago

White House Responds to 'Making a Murderer' Petition

4 comments · 1 week ago

Robotics to the Rescue: Automated Sample Processing

1 comment · 4 days ago

Forensic Expert Finds Trace Blood on Revolutionary War Bullets

2 comments · 1 week ago

Making a Murderer: Forensic Expert Reflects on Testimony at Steven Avery Trial

2 comments · 1 week ago

Exclusives



Police ID Torso Stuffed in Suitcase and Dumped in Canal

January 20, 2016 12:02 pm | by Sean Allocca, Editor



Monday's Briefing: 01/18/16

January 19, 2016 12:51 pm | by Sean Allocca, Editor



New DNA Implicates Acquitted Man in Brutal Death of 77-Year-Old British Woman

January 19, 2016 12:39 pm | by Seth Augenstein, Digital Reporter



Discovery Channel True-Crime Show Follows Hunt Through 'Killing Fields'

January 15, 2016 4:45 pm | by Seth Augenstein, Digital Reporter

[View More Exclusive Content »](#)

ADVERTISEMENT

SHIMADZU
 Next-generation
 FTIR Spectrophotometer
 The fast, sensitive,
 easy-to-use
 platform for
 forensics
 applications
 learn more, click here

Tips



Impression Evidence: Admissibility and Best Practices

July 22, 2015 4:08 am | by Dick Warrington



Tips for Picking the Perfect Powder

July 8, 2015 2:01 pm | by Hilary Romig



Your Perfect Crime Scene Kit

June 10, 2015 10:30 am | by Dick Warrington



Casting: Beyond the Basics

May 6, 2015 8:32 am | by Dick Warrington

[View More Tips »](#)



December 2015

December 22, 2015 2:57 pm

ADVERTISEMENT

50th Annual Awards

**Laboratory
of the
Year**

Honoring Design
Excellence for
New and Renovated
Laboratories

Presented by:

**R&D & Laboratory
Design**

Entry deadline: February 15, 2016

ADVERTISEMENT

ADVERTISEMENT

**R&D
100**
54 Years of Innovation

The 100 Most
Innovative Technologies
Introduced in the Last Year

**2016 Entries
are Now Open!**

Submit your entry today!

Submission Deadline: April 15, 2016



In the Public Eye: Finding Fingerprints on Firearms is Actually Very Rare

September 2, 2015 11:03 am | by Forensic Magazine

New Products

WriteBlocker

December 15, 2015 5:23 pm | by CRU, Inc.

Environmental Chambers

December 15, 2015 2:46 pm | by Terra Universal Inc.

VSE Balance Enclosure

December 10, 2015 2:45 pm | by HEVCO Corporation

Desiccator Safes

December 10, 2015 12:10 pm | by Terra Universal Inc.

[View More Product Releases »](#)

ADVERTISEMENT

Join DFI News for an Educational Webinar

**Exclusive Webinar:
5 Habits of Highly Effective Data Collection**



Now Available ON-DEMAND

Sponsored by:
ZAPPROVED™

View It Today!

Featured Companies

 **Retsch, Inc.**



Hamilton Company



FactualDiagrams.com

NuAire, Inc.



FORENSIC MAGAZINE

- About Us
- Advertising Info
- Author Guidelines
- Contact Us
- Subscriptions
- Privacy Policy
- Product Release
- Submission Form
- Supplier Directory FAQ
- Terms & Conditions

RESOURCES

- Articles
- Digital Editions
- Events Calendar
- News
- Sitemap
- Tips
- Videos
- Webinars

TOPICS

- Crime Lab
- Crime Scene
- DNA
- Forensic Science
- Legal Process

CONNECT WITH US

- Facebook
- Twitter
- LinkedIn
- YouTube
- RSS

OUR PARTNER SITES



Advantage Business Media © Copyright 2016 Advantage Business Media

Sign up for our newsletter

`{{ item.Header }}`
`{{ item.Text }}`

Subscribe Now

Thank you for your subscription!

Follow

Share

