SSD & FTL 从底向上 (P1)

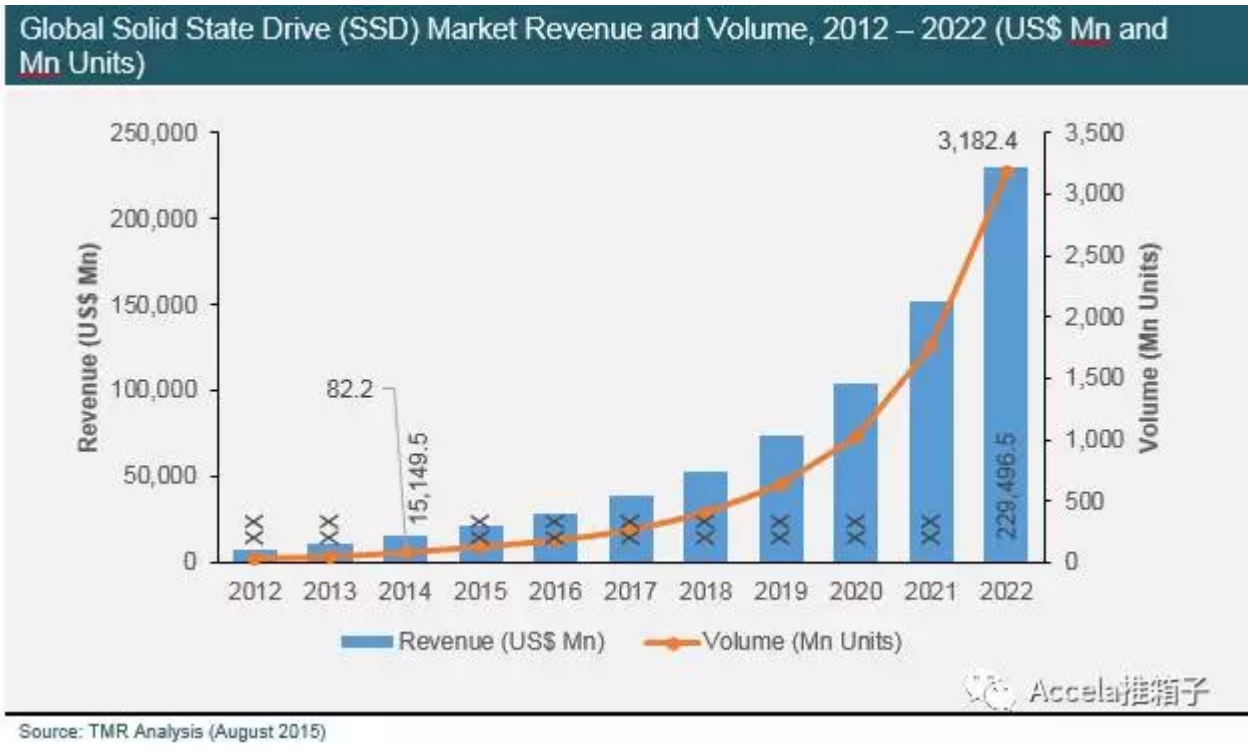
Original Accela Zhao Accela推箱子 2017-12-25

SSD近年来发展迅速，与之而来催生出PCIe SSD、NVMe SSD、NVDIMM、PersistentMemory等新型存储类别，还有Intel3D XPoint Optane SSD这款跨越性产品。存储介质速度的跃进，带来了弥补其它硬件速度的需求，例如FPGA之于CPU Offloading，RDMA高速网络，绕过PCIe总线等做法。

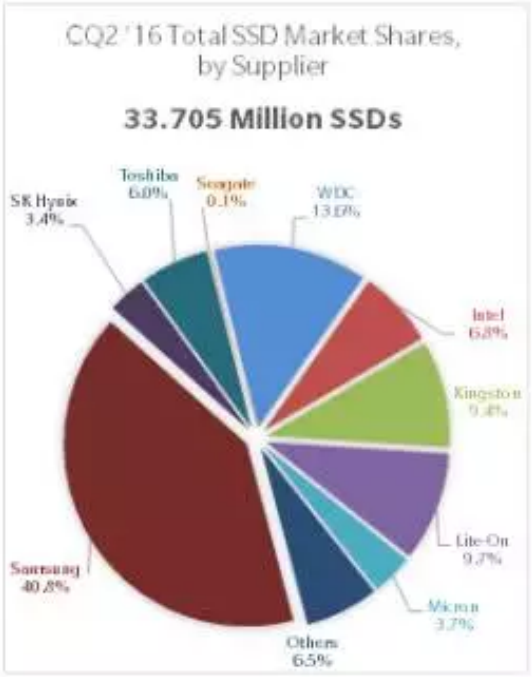
而SSD的核心技术之一，就是其Flash Translation Layer（FTL）。SSD存储媒介与磁盘不同，需要有FTL将块设备操作映射成为符合NANDFlash介质特性的操作。FTL管理了接口转换，块地址映射、磨损平衡、垃圾回收、写放大等诸多功能；技术丰富，引人遐想。

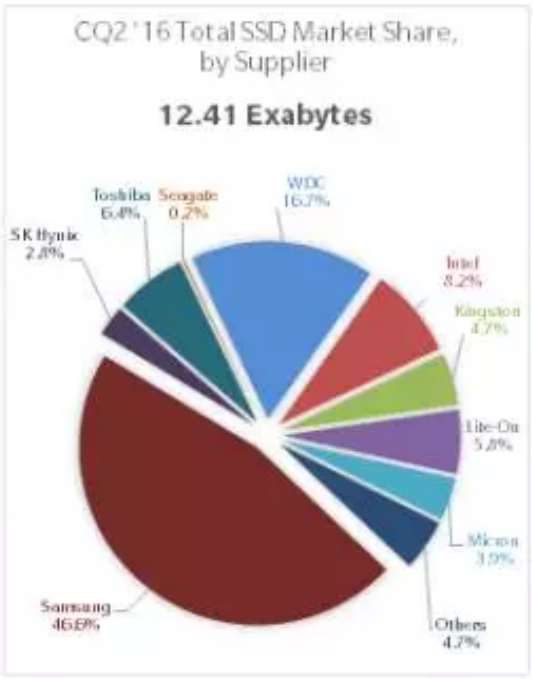
SSD市场增长

为什么SSD备受关注？近年来，SSD市场增长率维持双位数，几近年年100%。硬件厂商、存储厂商激烈竞争，许多用户将服务和数据迁移至全闪存设备，或者使用磁盘/SSD混合存储。虽然SSD价格较磁盘更加昂贵，但长期趋势逐年下降；而磁盘的发展已经十分成熟而再难速进。



[Source: TMR Analysis (August 2015) – SSD Market Revenue andVolume] (http://www.legitreviews.com/market-research-shows-ssds-sales-are-going-to-greatly-increase_172791)





TRENDFOCUS
The Data Storage Industry's Most Trusted Market Intelligence

[Source: TrendFocus – Market Trends Q2 2016 SSD Shipments] (http://www.anandtech.com/show/10706/market-trends-q2-2016-ssd-shipments-up-41-percent-yoy)

迅速增长的市场意味着能吸引大量资金、就业、和创新空间，许多Startup涌现，也是研究技术和发朋友圈文章的好去处。

SSD芯片架构

从底向上完善了解新技术，往往更能帮助写出好的存储产品和服务；存储媒介的特性决定了上层服务需要什么样的技术创新。

逐层分解一个Flash芯片的组成部分

- 一个Flash芯片或Flash设备，包含了多个DIE（存储颗粒）。DIE封装了完整独立的存储功能
- 一个DIE包含若干个Plane。Plane之间可独立执行读写操作，这是Flash能够多通道并行读写的基础
- 一个Plane包含上千个Block。Block，或被常称作块，是Flash芯片擦除操作的基本位；Flash按页写按块擦除的特性即源于此处
- 一个Block包含上百个Page。Page，被称作页，一般是4KB/8KB+ 几百字节的隐藏空间；隐藏空间用于FTL自身管理，可用于存储关于Page的元信息。Page是Flash读写的基本单元。
- Page由Cell存储比特，Cell是比特存储的基本单元。Flash媒介经常对比的SLC、MLC、TLC等等指的就是Cell的不同类型；类型不同，一大特点是最大P/E次数（编程/擦除次数，即Cell的寿命）不同。MLC在1,500到10,000次左右；SLC能到100,000次。

相关资料：

[AlanWu 博客：镁光 256Gb NAND Flash 芯片介绍](http://blog.51cto.com/alanwu/1668609)

[AlanWu 博客：如何挖掘 NAND Flash 的 IO 性能](http://blog.51cto.com/alanwu/1544227)

[Solid State Drive Architecture](http://meseec.ce.rit.edu/551-projects/fall2010/1-4.pdf)

MLC vs eMLC vs SLC

对比这些Cell的不同类型，它们的特点各不相同

- SLC：更加昂贵，速度快，更高的P/E次数；一些高端企业级SSD存储倾向选用此种介质
- MLC：更多由个人消费产品选用。比SLC便宜2到4倍，P/E次数少10倍，写速率更慢，可靠性更低
- eMLC：“e”代表企业级MLC。本质上是使用更复杂更强大的控制器（包括FTL），来管理磨损平衡、纠错等
- TLC：主要由三星控制

另外，Google在2016的“FlashReliability in Production”论文中提出，在硬盘预期寿命内，没有证据显示高端SLC硬盘比MLC硬盘有更高的可靠性。此外，Google的分布式存储软件层，副本和EC编码，也能极大程度上屏蔽硬件层的可靠性差异。

[Flash Reliability in Production: The Expected and theUnexpected (Google)](https://www.usenix.org/node/194415)

相关资料：

[MLC vs. eMLC vs. SLC vs. TLC](http://www.tomsitpro.com/articles/flash-data-center-advantages,2-744-2.html)

性能参数

存储硬件的速率和延迟数字对直观理解存储系统的性能设计很有帮助。本质上，存储系统的性能设计最终是将硬件性能完全挖掘，齐平硬件性能。而存储系统软件层架构的种种设计和随时代的变迁，本质上是CPU、内存、网络、硬盘等随时代发展，当代硬件性能水平对比的变化所致；当一方的性能跟不上另一方的高速发展时，往往涌现出众多创新方案来弥补。

[Latency Numbers Every Programmer Should Know](https://people.eecs.berkeley.edu/~rcs/research/interactive_latency.html)

2016年的大体延迟参数。不过硬件发展很快，每隔几年，参数都会大不一样

- Cache访问：~1ns
- 内存访问：~100ns
- SSD访问：~10us
- 磁盘访问：~10ms

从中也可以看出，SSD有作为更快的磁盘，或者较慢的内存这样不同的方向。前者可以进行存储分层（Tiering），冷热数据在磁盘和SSD之间迁移；后者有字节可寻址PersistentMemory，或者将DRAM内存断电时用电池刷回后备Flash来实现的NVDIMM。

SSD接口类型

传统磁盘的接口一般是SATA和SAS。相比SATA多见于个人消费品市场，SAS兼容SATA，更为需要高性能、可接受高价格的服务器所青睐。随着SSD硬盘的发展，逐渐出现更多其它的接口方案。

- SATA/SAS：最初SSD硬盘为了易于市场接受，采用兼容传统磁盘的设备接口SATA/SAS；SATA/SAS本是为磁盘设计的接口，并不能最佳发挥SSD性能；另一方面，也因而需要FTL将磁盘操作映射成Flash操作。

- PCIe：后来FusionIO首创，一众公司逐步推进，出现了将SSD直接连接到PCIe总线的PCIeSSD；跳过了SATA/SAS总线，与CPU更近，速度更快；PCIeSSD可以达到极高的IOPS和吞吐量，远超磁盘。PCIe中，FTL功能可以由服务器，称为host-basedPCIe SSD；但占用服务器自身资源；同样，FTL功能也可继续由SSD硬盘硬件实现。

- NVMe：PCIe接口并不是最符合SSD需要的，因而又催生了NVMe接口。SSD仍然连接在PCIe总线上，但连接协议换成了NVMe。NVMe针对Flash特性作了诸多优化，如更少的消息数量，更多更大容量的队列支持等；NVMe让服务器能够最大化地利用Flash内部的并行能力。

其它还有一些接口协议，其中NVMe over Fabric新兴正在制定中；NVDIMM将Flash进一步连接到内存总线上；而LightNVM与Open-ChannelSSD应用较窄，一般是大型互联网公司定制自身SSD软硬件。

- NVMe over Fabric：类比过去通过SCSI协议连接大量磁盘组成SAN，NVMeover Fabric希望将NVMe SSD硬盘组成存储网络。将存储集中起来从计算服务器中分离出去，和将存储与计算进行co-located部署，两种趋势总是交相发展。网络传输上，NVMeover Fabric可以使用RDMA或者Fibre Channel。

- NVDIMM：将Flash接入内存总线，当作内存来使用；与NVM的一大区别是字节可寻址（Byte-addressable），即内存是字节寻址的，而硬盘是块设备。最早的是NVDIMM-N，仍使用DRAM作内存，但断电时由超级电容（Supercapacitor）将数据刷回后备Flash；NVDIMM-F以Flash替代DRAM内存，内存直接读写在Flash上，性能较-N差，但容量大；NVDIMM-P混合DRAM和Flash于内存，相比上述两者居中。

- LightNVM：它是Linux内核对Open-ChannelSSD的支持。有能力高度定制SSD软硬件的用户可能希望去掉FTL，由自己管理Flash的读写擦除和垃圾回收等底层功能；LightNVM这些接口暴露给用户使用。后文Open-ChannelSSD一节会讲到更多。

相关资料：

[Difference between SATA, PCIe and NVMe](http://www.userbenchmark.com/Faq/What-s-the-difference-between-SATA-PCIe-and-NVMe/105)

[NVMe over Fabrics](http://www.nvmexpress.org/wp-content/uploads/NVMe_Over_Fabrics.pdf)

[NVDIMM-N 、NVDIMM-F 以及 NVDIMM-P](http://blog.csdn.net/wma664620/article/details/54618556)

[NOVA: A Log-structured File System for HybridVolatile/Non-volatile Main Memories](https://www.usenix.org/conference/fast16/technical-sessions/presentation/xu)

[LightNVM Open-Channel SSD](http://lightnvm.io/)

FTL – FlashTranslation Layer

由于Flash存储介质的特性与磁盘完全不同，最大的特点是无法直接覆写，必须以较大的数据单元，即块（Block），为单位进行擦除。因而需要FTL来将磁盘块设备的操作接口映射为对底层Flash介质的操作。FTL负责了如下几方面的功能：

- 接口适配：将SCSI/SATA/PCIe/NVMe硬盘接口协议映射为对Flash的操作
- 坏块管理：SSD出厂是有未知坏块的；初始化时，FTL检测并记录坏块；SSD使用过程中也会出现新的坏块。FTL通过地址映射功能对对应用屏蔽坏块的存在。
- 逻辑块映射：Flash实际读写使用物理地址，而应用读写的地址是逻辑地址；FTL从中管理映射。逻辑地址和物理地址的分离，是实现垃圾回收的基础。
- 垃圾回收：Flash无法直接覆写，而是需要先以块（Block）为最小单元进行擦除操作；读写单元却是小得多的页（Page）。因而，Write-in-place（让写操作就发生在应用指定的位置）无法实现；新的写需要被安放在一个临时位置，之后可能被挪动。这些就带来了垃圾回收的需要。

- 磨损平衡（Wear-leveling）：由上文可以看到，单个Flash存储单元（Cell）所能被擦除的最大次数（P/Ecycle）实际很小。必须妥善管理写和擦除在所有Flash存储单元之间的分配，避免一些存储单元被过早磨损耗尽。这就需要FTL来管理磨损平衡。

- 写放大（Write amplification）：相比应用的原始输入，SSD实际写入的数据体积可能更大，这加剧了Flash存储单元磨损，也影响性能。造成写放大的原因主要是垃圾回收和磨损平衡在后台重写数据；另一方面，由于擦除只能以块为单元进行，被波及的有用数据也需要挪动。值得一提的是，LSM-Tree类的Append-only式存储方式和数据结构，因为同一份数据在反复合并（Compact/Merge）中被写了多次，天生具有写放大的问题。

- 此外，FTL还可管理读写数据的纠错码、即时压缩、或者特殊功能如减少些放大的编码等（如WOMCodes）；这些功能可能会需要利用Page内的隐藏空间。在论文中可以见到更多细节。

FTL的复杂又高效的功能是如何实现的，引人遐思；后文将深入挖掘。

相关资料：

[Understanding the FTL](https://flashdba.com/2014/09/17/understanding-flash-the-flash-translation-layer/)

[AlanWu 博客：神秘的 Flash Translation Layer)(http://blog.51cto.com/alanwu/1427101)

(未完待续……)

健康中国