SOLID STATE STORAGE TECHNOLOGY CORPORATION

LITE-ON Storage is now Solid State Storage Technology Corporation

The Brains Behind Flash

SSD Controller ASICs

Perhaps the most important component of Solid State Drives, aside from the flash, is the controller. While there is a Microcontroller Unit (MCU) at its heart, the overall package has multiple elements that work

together as an Application-Specific Integrated Circuit (ASIC). SSDs require such a specialized solution because storage Input/Output (I/O) is latency-sensitive. Data access is a real-time application and SSDs are now capable of a million or more I/O Operations Per Second (IOPS). The controller must therefore be able to handle heavy workloads without adversely impacting the response time.

The MCU, as opposed to a microprocessor, is usually of a Reduced Instruction Set Computer (RISC) design. The implication is that a simplified architecture, in comparison to one that is a Complex Instruction Set Computer (CISC), is more efficient for specialized use. While that distinction has blurred with modern processors, SSD controllers tend to be simulated with Field-Programmable Gate Arrays (FPGA) which are akin to a programmable ASIC. Having some flexibility is important, but SSD controllers are not general-purpose; they are often developed as a drop-in, scalable solution.

The most common architecture for the MCU is made by Advanced RISC Machines (ARM), as with the Cortex-R5, although there are competitors such as the Argonaut RISC Core (ARC). These are typically 32-bit designs. One difference between the two would be in terms of customizability, but in general these architectures serve a similar goal: the RISC Central Processing Unit (CPU) is designed to facilitate I/O as efficiently as possible. However, SSD controllers are increasingly attractive for use with in-storage computing, or computational storage. They are, in any case, at the center of multiple modules that work intelligently to maintain the usually NOT-AND (NAND) Non-Volatile Media (NVM).
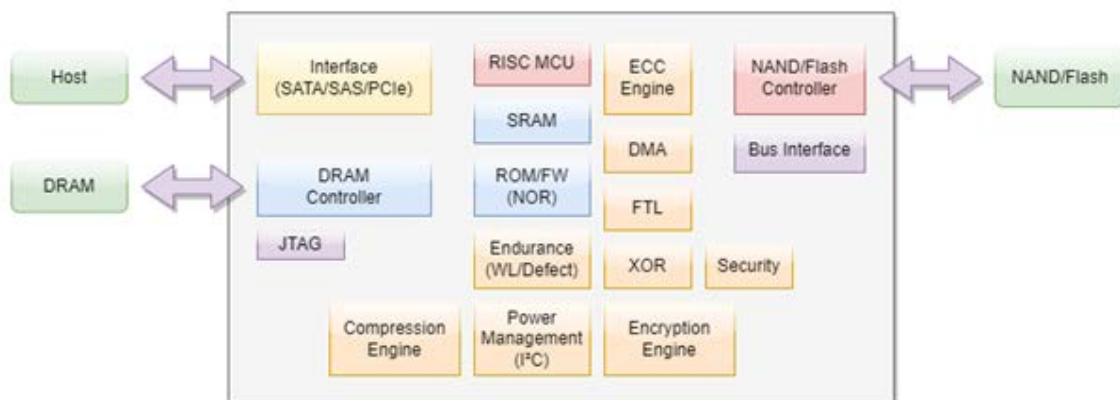
General SSD Controller ASIC

I/O Management

The principal purpose of an SSD controller is to manage the NVM storage, handling communication between the host and flash. Although host-SSD collaboration is possible through the NVM Express (NVMe™) specification, most often the storage is abstracted by the controller's management of the Flash Translation Layer (FTL). The FTL helps translate between physical and logical addresses, bridging the file system and the physical storage. This is necessary as NAND has unique characteristics: it must be erased before being rewritten, and is also structured in pages and blocks for Read/Write (RW) or erase access, respectively.

The SSD ASIC therefore contains a NAND or flash controller that communicates with the NVM through a bidirectional Data Bus (DQ), normally 8-bit wide, and a bidirectional data strobe (DQS). Modern flash operates with a Double Data Rate (DDR) bus, for example through the ONFi or Toggle standards. This means that data is sent on each edge of the clock cycle. The bus must also transfer information such as I/O commands and addresses. The flash itself is divided into channels and banks for simultaneous I/O through interleaving, with die selection coming through a Chip Enable (CE) signal.

Data moves from host to NVM via Direct Memory Access (DMA), from host to controller and then controller to flash. The data may reside in the Static Random-Access Memory (SRAM) for write combining which aids with interleaving at the superpage level. From there it enters volatile latches, also known as Page Buffers (PB), which in turn output to the NAND cells via bitlines. Each flash die contains its own latches and latch management which includes row and column decoders for addressing. This process adds some latency but is done rapidly, with the actual flash I/O access taking the largest portion of time. SSDs may also employ a compression engine, which reduces the amount of storage used on the device while also decreasing the Write Amplification Factor (WAF).

Other Modules

The controller SRAM, which tends to be limited in capacity, is also utilized to cache boot and instruction code. This is based on the controller's firmware which is stored in Read-Only Memory (ROM) which is typically made

of NOT-OR (NOR) NVM. NOR is more expensive than NAND and is therefore also limited in capacity, but has faster random access and is byte-addressable for RW. The SSD controller engages in a start-up process when the device is powered on, for example initializing the mapping tables in the FTL, using these memory elements as well as data within a fast portion of the NAND acting in a single-bit mode.

SSDs also often rely on Dynamic Random-Access Memory (DRAM) to enhance the performance of the address Lookup Table (LUT). DRAM has to be continually refreshed, unlike SRAM, but is cheaper and more plentiful. SSDs tend to have up to one kilobyte of DRAM for every megabyte of storage, a ratio that assumes 32-bit addressing for 4-kilobyte logical pages. The SSD controller requires a separate DRAM controller to manage this memory. The DRAM will cache the most-accessed and most-recently-accessed addresses in order to facilitate future I/O.

Other modules include the Error Correcting Code (ECC) engine, Exclusive-OR (XOR) logic, and the encryption/decryption engine. Error correction is critical for End-to-End (E2E) data path protection including the SRAM, and for metadata as well as data. XOR is used during the encryption process but also to scramble data to probabilistically enable an even distribution of bits. The encryption engine includes key storage as part of the security module, whether for a Self-Encrypting Drive (SED) via the Trusted Computing Group's (TCG's) Opal specifications or for internal encryption of the flash dies to defeat physical, forensic data recovery attempts. Typically, this is using a 256-bit key with the Advanced Encryption Standard (AES or AES-256).

The SSD controller also has a way to handle power management and to communicate with other devices and modules. This is done through the Inter-Integrated Circuit (I2C or I2C) protocol or the System Management Bus (SMBus), which has similarities to a Serial Peripheral Interface (SPI) and Universal Asynchronous Receiver-Transmitters (UARTs). The ASIC communicates with external components through various buses for the NAND, DRAM, and host. The host interface often uses Peripheral Component Interconnect Express (PCIe®) with four or more lanes, but previously Serial AT Attachment (SATA) was common. Additionally, the controller can be accessed through the Joint Test Action Group (JTAG) standard, for example to aid in design and debugging. There may also be General-Purpose Input/Output (GPIO) pins for connection.

Endurance and Maintenance

NAND has a limited lifespan as the Raw Bit Error Rate (RBER) will climb as the flash cells are worn. The SSD ASIC has multiple ways to manage and improve flash endurance. Part of this is defect management, as flash dies arrive with bad blocks from the factory but also develop more as time goes on. Spare blocks must then be rotated in for replacement. The controller also engages in wear-leveling to ensure the flash is worn equally, reducing the chances of early block retirement.

The controller relies on ECC to improve NAND lifetime through Low-Density Parity-Check (LDPC) code for both hard and soft encoding. Each flash page also has extra cells outside of the addressable area used for error correction and spare bits. This spare space can be distributed over multiple planes and dies into what resembles a Redundant Array of Independent Disks (RAID), enabling repair through parity data. This is effective if the LDPC check, managed by the ECC engine, is insufficient to fix all of the bit errors.

The SSD controller must also work to avoid errors through the reduction of disturb – for example, program and read disturb – via the use of feedback and historical tables. As flash is non-uniform due to production peculiarities, this includes things such as voltage bias depending on the block range. The controller additionally controls maintenance and Garbage Collection (GC) to both reduce write amplification and improve endurance. Making sure there are free blocks available for writing is important to this process, which requires knowledge of page status as through TRIM.

Conclusion

Modern SSD controller ASICs are composed of a RISC MCU and dedicated modules designed to facilitate storage performance while mitigating errors and flash's innate drawbacks. The package could be understood as a singular solution, externally linked to the host, flash, and DRAM. This includes a bidirectional bus operating in a toggle mode. Internal monitoring and management, as well as external probing, follow communication protocols meant to be lightweight and secure.

Errors are handled through various data protection schemes, including ECC, disturb mitigation, wear-leveling, RAID parity, and defect management. While some modules are optional, in general security is a crucial part of

the design meant to protect the data. Perhaps more importantly, the SSD controller must manage the FTL for efficient data storage. Aside from endurance, performance, particularly latency, is paramount, given that consistency is an all-important metric in enterprise. Data integrity must also be assured, end-to-end, within the device.

With all components acting together, the SSD controller ASIC can be specialized for I/O while additionally delegating tasks to further-specialized modules. There is a necessary level of abstraction which enables the host to command I/O without having to worry about the details. The controller, for its part, can intelligently control the flow of data to maximum performance and efficiency. In this way it becomes possible to use relatively capacious NVM effectively, even at scale. As such, the importance of the controller and its workings cannot be understated.

Our SSD Solutions

PCIe^{TM} - Our ED1 Series is a powerful, high performance SSD made for edge storage applications. It comes in M.2 and U.2 form factors.

SATA - Our ER2 SSD Series delivers affordability and performance with superior random read/write speeds of up to 90,000/45,000 IOPS. It comes in M.2 and 2.5" form factors.

Please contact our Solid State Storage Technology Corp. expert for more information.

*Specifications and features are subject to change without prior notice. Images are samples only, not actual products.

Request Full Specs Sheets

ABOUT US

A subsidiary of KIOXIA Corporation, **Solid State Storage Technology Corporation** is a global leader in the design, development, and manufacturing of digital storage solutions. We offer a comprehensive lineup of high-performance customizable SSDs for the Enterprise, Industrial, and Business Client markets. With various form factors and interfaces, our SSD solutions help businesses simplify their storage infrastructures accelerating variable workloads, improving efficiency, and reducing total cost of ownership.

Learn more at www.ssstc.com