

How ClearNAND™ Flash Simplifies and Enhances System Designs

Jim Cooke
Senior Technical Marketing Manager

Micron Technology, Inc.
December 27, 2010

NAND Flash Trends and Complexities

NAND Flash ECC trends have been on the rise since NAND was first introduced. Although it's not a new issue, the ECC required to support newer multilevel (MLC) and three-bit-per-cell technologies is becoming increasingly difficult for system designers to keep up with.

ECC has historically been used to improve the overall data reliability of NAND subsystems. However, as NAND cells shrink, fewer electrons are stored per floating gate. To compensate for the increasing bit error rates of these smaller geometry cells, ECC requirements have to dramatically increase to maintain the desired system reliability.

As system requirements for ECC increase, the number of gates required to implement the logic also increases, as does the system complexity. For example, 24 bits of ECC requires about 200,000 gates, while 40 bits of ECC requires about 300,000 gates. It is estimated that in the future, advanced algorithms will approach close to 1 million gates. See Figure 1.

Many high-performance Flash systems require multiple channels of NAND to reach the desired performance. In these systems, each channel typically has its own ECC logic. For example, a 10-channel SSD may have 10 channels of ECC logic implemented. If 60 bits of ECC were required for each of the 10 channels, the result would be 3 million gates just for the ECC logic.

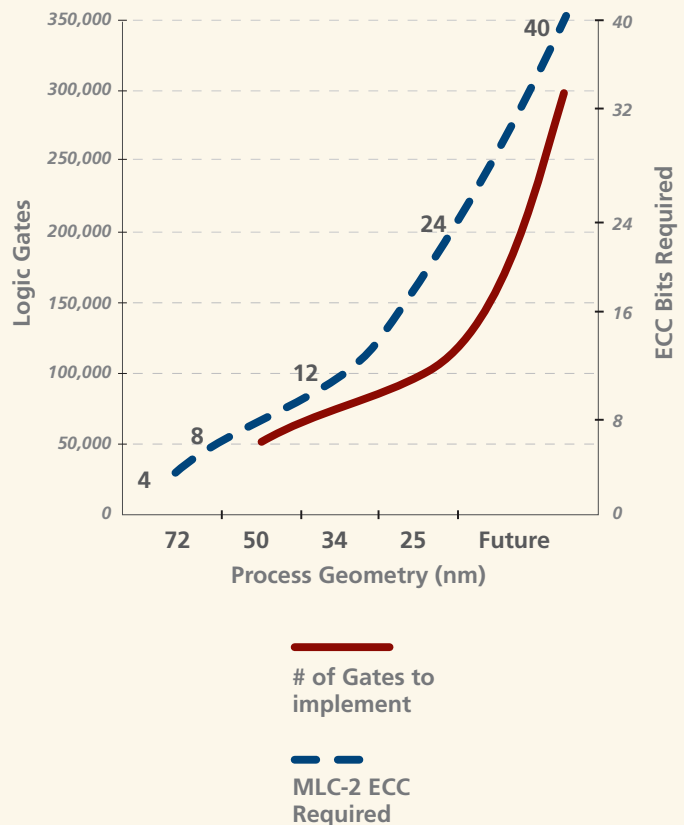


Figure 1: ECC Increases as Process Geometries Shrink



NAND Interface Choices

Fully Managed Solutions

Fully managed interfaces are typically the simplest to interface with. Examples include e-MMC™ memory or eSD, which use a traditional multimedia card or SD card interface. These multichip packages (MCPs) include a controller that implements the desired interface, as well as the ECC and block management required by the NAND.

An advantage of fully managed solutions is that they require the least amount of effort from the host's perspective. The host software uses a straightforward block-level interface; and the controller within the MCP is responsible for the ECC, block management, and wear leveling. The disadvantage is that they're single-threaded and based on a simple command, data, and acknowledge protocol, which limits performance. In addition, the processor used in these MCPs is typically a small 8-bit processor, which also impacts performance.

These implementations are typically not very deterministic, because the internal controller can be doing block management or garbage collection activities in the background. Consequently, they can be exposed to unexpected power loss. With regard to performance, sequential READ/WRITE operations provide reasonable bandwidth for large data transfers, while random READ/WRITE operations can yield lower performance for smaller data transfers.

Legacy NAND Interface

The NAND interface has traditionally been an asynchronous interface. Although interface speeds have improved up to 50 MHz in recent years, not much else has changed on this interface.

Several years back, Micron and several other forward-thinking companies joined together to form a NAND Flash organization that was focused on simplifying the myriad of timing and command specifications offered by the industry. The Open NAND Flash Interface (ONFI) developed the first version of their specification, ONFI 1.0. While there are many advantages to the original ONFI 1.0 specification, one of the biggest is the ability for the host to electronically detect the type of Flash device that is connected, as well as other important parameters, like timing modes, page size, block size,

ECC requirements. This feature has been carried forward to all of the ONFI specifications and remains an important aspect of all ONFI standards.

Another significant accomplishment of the ONFI organization was the development of the synchronous NAND interface, also known as ONFI 2. ONFI 2.2 currently supports up to 200 mega transfers per second (200 MT/s) using a DDR, source-synchronous interface. That is, after powering up, it can be used in asynchronous mode. However, for higher performance, the host interrogates the Flash device to see if it is able to support the higher-speed synchronous interface before changing to it.

Direct NAND Solutions

Implementations that connect NAND directly to the host processor are responsible for managing the NAND. Hardware manages the ECC, while software typically performs all block management and wear-leveling operations. At first this may seem like a disadvantage. However, with today's typical embedded processors running at speeds of hundreds of megahertz and often over one gigahertz, these high-performance processors can accomplish block management much faster and can take advantage of deterministic, multithreading techniques to improve performance. In addition, with the host managing the Flash device directly, the host software can make real-time decisions that can help eliminate exposure to unexpected power failures.

As shown in Figure 2 (page 3), the ONFI 2.2 specification (200 MT/s) was designed to accommodate up to 16 standard NAND loads. A typical implementation of this would be using two 8-die NAND packages. The standard 8-die, 100-ball BGA package includes two separate NAND buses (DQ[7:0]1 and DQ[7:0]2), with each bus having four NAND Flash wired together. Each of the four die stacks are controlled with two chip enables. A typical design would wire the two data or DQ buses together, forming a single 8-bit data bus for each package. A maximum configuration would consist of two 100-ball BGA packages, each containing eight die. Each of these standard 100-ball BGA packages requires four chip enables (CE#) to select a specific NAND die. Thus, the system controller needs to supply eight chip enables to support this configuration.



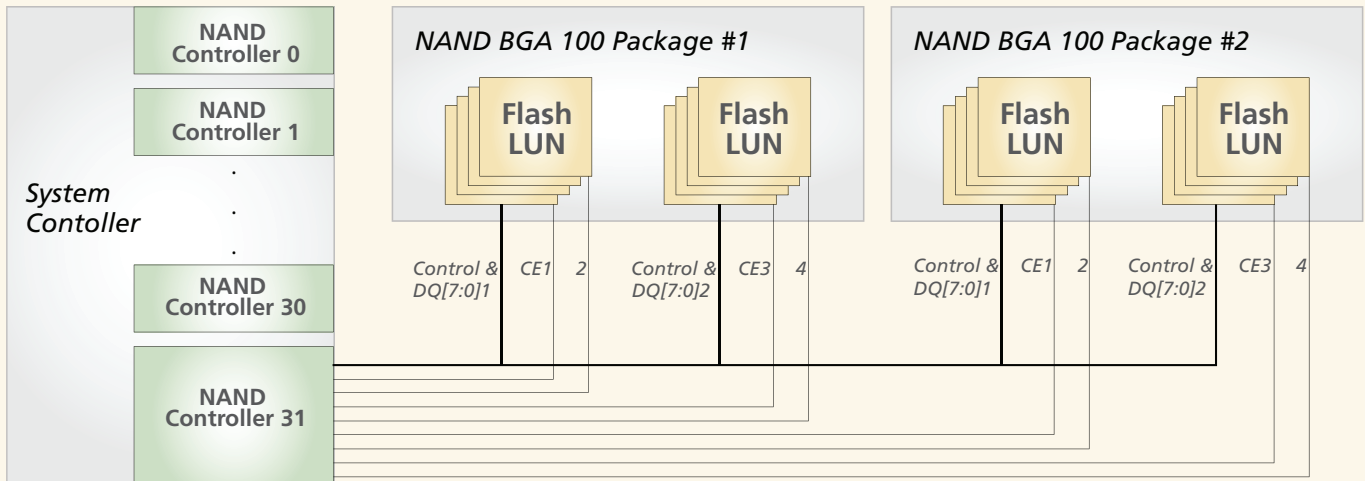


Figure 2: Typical High-Capacity Implementation Using Two 8-Die, 100-Ball BGA Packages

ClearNAND™ Solutions

Figure 3 (page 4), shows two system implementations: a traditional system where the processor is interfacing directly with NAND and a system using ClearNAND Flash. Both implementations use the same ONFI hardware interface and similar 100-ball BGA ballouts. The ClearNAND example includes a thin controller package with the NAND die in an MCP. The ClearNAND controller implements the ECC required by the NAND inside the MCP package. Utilizing the same ONFI asynchronous or synchronous interface enables designers to migrate easily from standard NAND to ClearNAND Flash.

Micron offers two versions of ClearNAND Flash: Standard and Enhanced. Standard ClearNAND Flash, suggested mainly for consumer devices, implements the required ECC and provides a traditional asynchronous ONFI bus for easy migration.

Enhanced ClearNAND Flash manages ECC, in addition to offering several performance-enabling features that are of most value to enterprise applications. It also supports both the asynchronous and synchronous versions of the ONFI 2.2 interface and is available in densities up to 64GB.

By abstracting the ECC, both versions of ClearNAND Flash will be able to handle the additional ECC that future versions of NAND will require. This will eliminate the need for designers to continually redesign their circuitry to keep up with manufacturers' latest NAND ECC requirements.

Enhanced ClearNAND Flash

Figure 4 (page 4), shows the Enhanced ClearNAND architecture. It supports a single ONFI 2.2 interface and up to 200 MT/s command, address, and data bus. The V_{DDI} decoupling capacitor is common in e-MMCTM products and other devices that include a controller. It is required to decouple the internal voltage regulator. For backward compatibility with standard NAND devices, the V_{DDI} connection is located on an unused pin. The ClearNAND controller supports two internal Flash buses, one for even logical unit number or LUNs and one for odd LUNs. These two independent Flash buses can operate at up to 200 MT/s. In addition, each bus has its own ECC engine and can manage simultaneous READ or WRITE operations on the two buses. It is envisioned that future versions of the controller will support the ONFI 3 specification, which is targeting up to 400 MT/s.



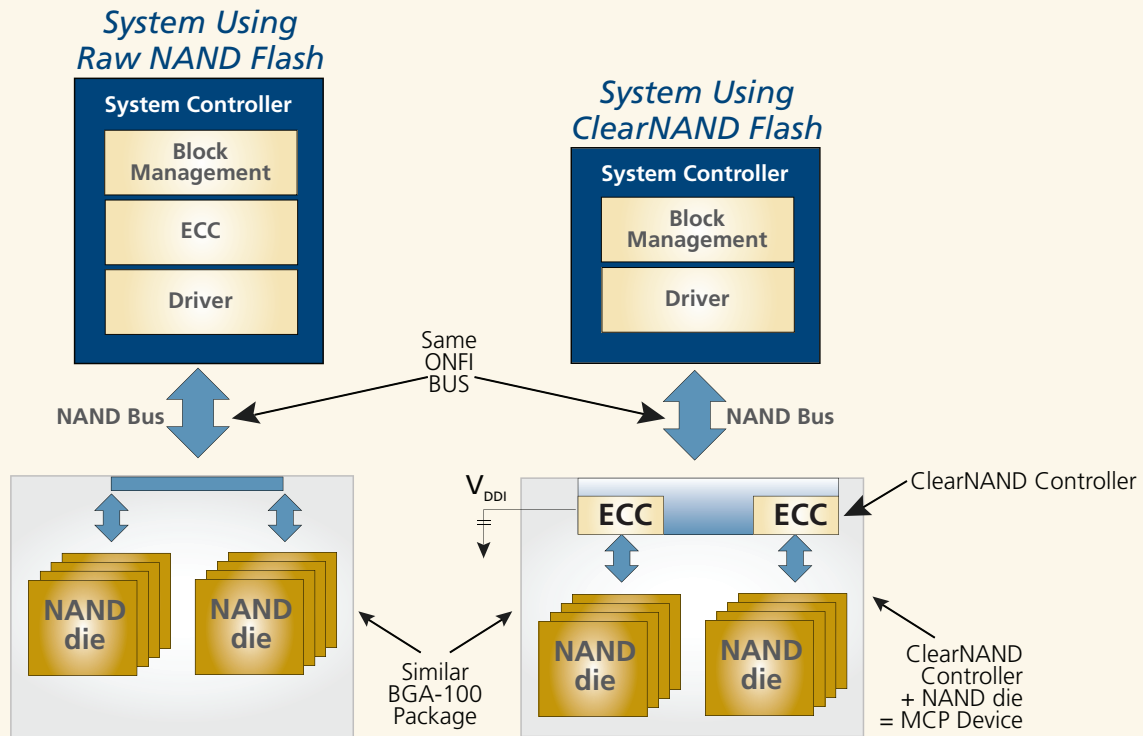


Figure 3: Standard NAND vs. ClearNAND

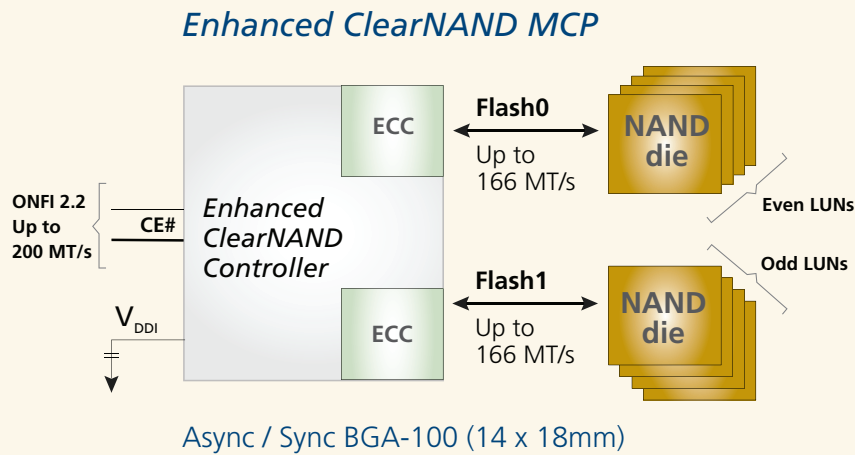


Figure 4: Enhanced ClearNAND Architecture

<i>Feature</i>	<i>Benefit</i>
<i>ONFI-compatible controller and NAND device in a standard package (14 x 18mm; 100-ball BGA)</i>	<ol style="list-style-type: none"> 1. Footprint compatible with raw NAND 2. Manages and abstracts ECC 3. Improves endurance (future versions) 4. Provides platform for future enhancements
<i>Single 200 MT/s front side bus</i>	<ol style="list-style-type: none"> 1. Reduces loading on bus 2. Allows more NAND per channel 3. Improves signal integrity
<i>Dual 166 MT/s internal NAND buses</i>	<ol style="list-style-type: none"> 1. Enables parallel operations 2. Local copyback offloads wear-leveling operations

Table 1: Enhanced ClearNAND Features and Benefits

Many of these basic features are covered in the following discussion of the four advanced functions offered by Enhanced ClearNAND: volume addressing, electronic data mirroring, interrupt and internal copyback.

Volume Addressing

Volume addressing allows a single chip select or chip enable (CE#) to address up to 16 ClearNAND volumes. Each ClearNAND controller can support up to eight die packaged in an MCP. The ClearNAND controller provides a buffer for the system controller access.

The Enhanced ClearNAND design, shown in Figure 5 (page 6), offers an eightfold improvement in density while maintaining or improving signal integrity and reducing the active number of chip enables that are required. This is because a single ClearNAND controller represents only one load to the system controller, but supports up to eight NAND die in the MCP package.

There are two aspects to the volume addressing concept. The first is establishing the volume address for each of the ClearNAND packages. The volume address is appointed only once at initialization and is maintained until the power is cycled. The second aspect is the volume select command itself. This is a new command that is followed by a single byte (actually only 4 bits) volume address. Once the intended volume is selected, it remains selected until another volume is selected. The chip enable pin savings can be significant. For example,

with a standard NAND implementation, a single channel requires eight chip enables to control two 8-die packages. The 32-channel example would require a total of 256 chip enable pins whereas the Enhanced ClearNAND volume addressing feature can address the same amount of NAND using only 32 chip enable pins. What's more, these same 32 chip enable pins can be used to address eight times the density.

Electronic Data Mirroring

Enhanced ClearNAND supports electronic data mirroring, which allows the data bus signal order to be electronically remapped to one of two configurations. This is particularly useful for high-density designs with ClearNAND devices mounted on both the top and bottom of the PCB. The ClearNAND package is able to electronically detect whether it is mounted on the top or bottom of the PCB. This is accomplished using a specific initialization or reset sequence. For example, it's common practice to issue a reset or FFh command to the Flash device on power-up. To accomplish the electronic DQ mirroring, the host must follow this FFh command with the traditional READ STATUS (70h) command. The top die detects this command sequence as FFh-70h; the bottom die recognizes this same sequence as FFh-0Eh and can establish that it is the bottom package and reorder its data bus to align directly under the top die. This not only improves PCB routing but also improves signal integrity.



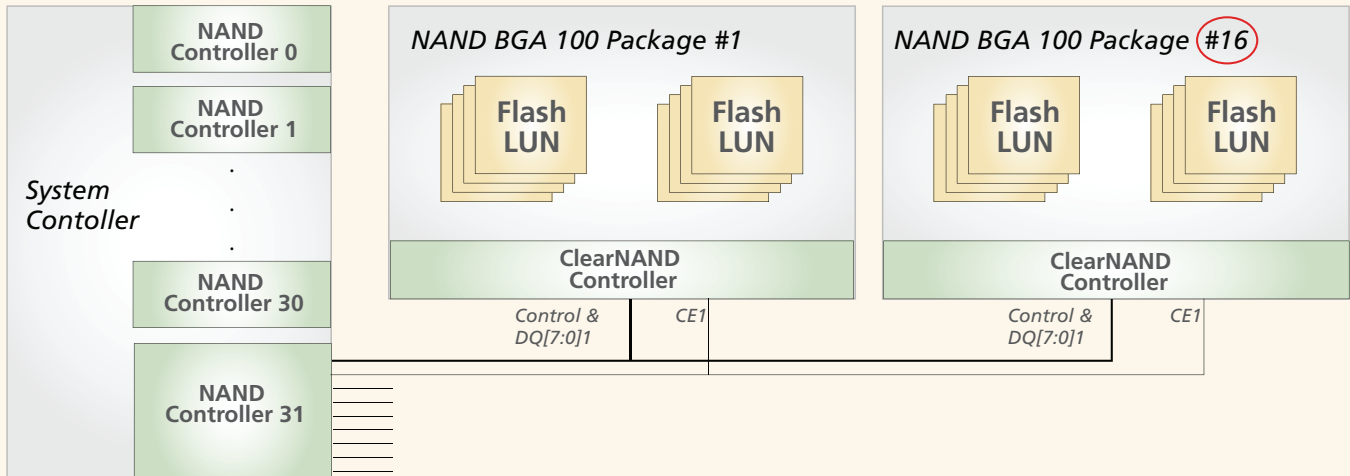


Figure 5: Typical High-Capacity Implementation Using up to Eight Die in 100-Ball BGA ClearNAND Packages

Ready/Busy# Redefined to Interrupt

Figure 6A shows the READY/BUSY# operation of standard NAND. The READY/BUSY# function is an open-drain signal. The standard 8-die, 100-ball BGA package includes four ready/busy (R/B#) pins. Like chip enable, each of these pins is connected to two individual die. Referring to Figure 6A, R/B# 1 would be connected to Die 1 and Die 2. Because signal connections are at a premium, designers will either not use the R/B# signals, or if they do, they will wire-OR all of the R/B# signals from one package (shown at the bottom of Figure 6A). This is not ideal, because any single die that is busy pulls the entire R/B# signal LOW, preventing die that become ready from being detected. If the implementation does not use the R/B# signals, it has the option of using firmware to poll the status of each of the die in the package, which is time-consuming and wastes power. Neither case is ideal.

Enhanced ClearNAND Flash redefines the existing R/B# pin to be an interrupt pin. As shown in Figure 6, the interrupt# signal, which is still open-drain, provides a real-time interrupt when the ClearNAND volume or die becomes ready. Designers can use this signal to provide real-time status to the system controller. In larger configurations supporting multiple ClearNAND packages on a single bus, the interrupt# signals can be wired

together. When the system controller detects an interrupt, it can simply interrogate each of the ClearNAND packages or volumes to learn which volume has posted the new status. The INTERRUPT# operation can save signals on the system controller while improving the ability for the system controller to respond to status updates.

Internal Copyback

The last but perhaps most noteworthy feature of Enhanced ClearNAND Flash is the internal COPYBACK function, also known as INTERNAL DATA MOVE. This function can provide a significant advantage in SSD systems when it comes to wear-leveling or garbage collection operations; that is, the process of collecting fragments of data scattered throughout the various pages and blocks of the NAND and coalescing them in a more streamlined block or sequence of blocks. It is similar to the old hard disk defragmentation utility.



Referring back to Figure 2, when using standard NAND, moving data fragments from one block to another typically requires the following sequence of operations:

1. The system controller issues a READ command and source address to access the source page of data.
2. The system controller inputs data from the NAND device while calculating and making any necessary ECC corrections. Any updating of data or metadata is usually accomplished after this step.
3. The system controller calculates and appends the new ECC information before issuing a new PROGRAM command, destination address, and data sequence, which will store the data in the new NAND block.

In this sequential operation, the bus is busy while moving the source and destination data from and to the Flash device. The timing of these operations can be significant. An ONFI 2.2 synchronous bus operating at 200 MT/s would require about 41µs to move the data, assuming an 8K page. Because the data has to be moved from and to the Flash device, we double this time to 82µs, which doesn't include the ECC overhead. While this sequence is being carried out, the ONFI Flash bus is busy and cannot be used for other operations.

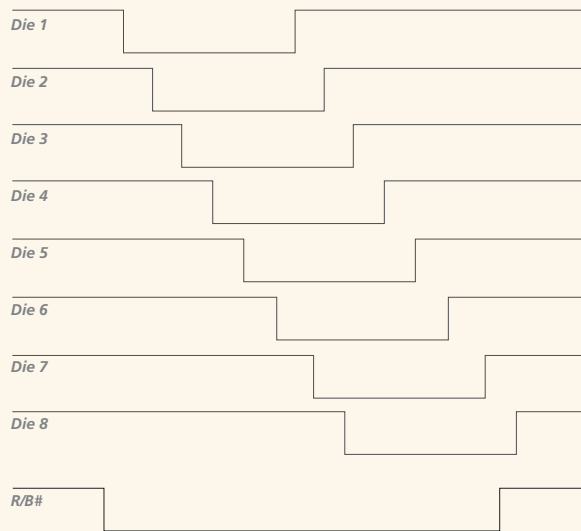


Figure 6A: READY/BUSY# Operation

Enhanced ClearNAND Flash is different in that it supports internal ECC. Using this built-in ECC enables the COPYBACK operation to be performed internal to the Enhanced ClearNAND package, assuming the source and destination of the data are within the ClearNAND package. The system controller is still responsible for issuing the commands and addresses as well as any modified data or metadata. The data movement is handled by the ClearNAND controller and does not tie up the external ONFI bus. If the system controller is able to keep its wear-leveling and garbage collection operations within a single ClearNAND package, it can have significant performance advantages.

Figure 7 (page 8), shows an example using Enhanced ClearNAND on two ONFI channels labeled Channel 0 and Channel 1. On both SSD channels, we can see that four of the INTERNAL DATA MOVE operations are occurring simultaneously without the external ONFI bus being used for the data movement. This frees the system controller and ONFI bus to move data from one ClearNAND package to another, if necessary. Depending on the architecture, some percentage of these operations may need to go between ClearNAND packages or even between ONFI buses. Taking advantage of the INTERNAL DATA MOVE operation can provide a significant performance improvement for garbage collection and wear-leveling operations.

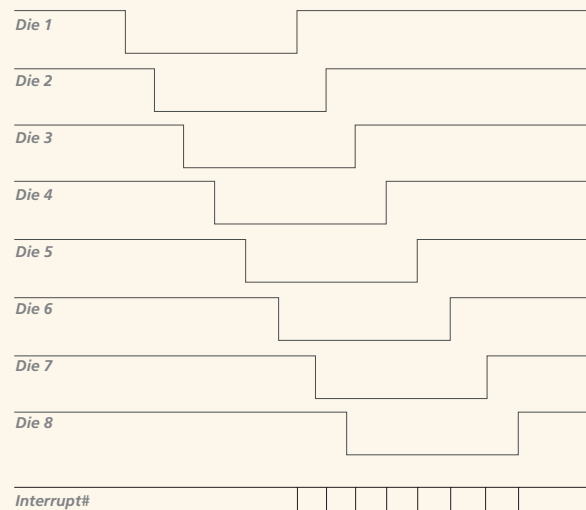


Figure 6B: INTERRUPT# Operation



Conclusion

Micron's Enhanced ClearNAND Flash provides additional performance and features while eliminating the impact of NAND's ever-increasing ECC requirements. Because Enhanced ClearNAND supports a ballout similar to the standard 100-ball BGA NAND devices, it's possible to design your product to support both. An example would be to include enough ECC in your system controller to support SLC NAND Flash directly and select Enhanced ClearNAND Flash for your multilevel cell needs, where ECC can present more of a challenge.

The volume addressing feature of Enhanced ClearNAND enables higher densities to be addressed using fewer chips, saving potentially hundreds of pins in high-capacity implementations. Electronic data mirroring simplifies PCB design and routing while improving the signal integrity of the ONFI bus. The intelligent interrupt capability provides for real-time status updates to the system controller and minimizes the polling for firmware. The dual internal NAND Flash buses provide improved COPYBACK operations, which in turn improve performance.

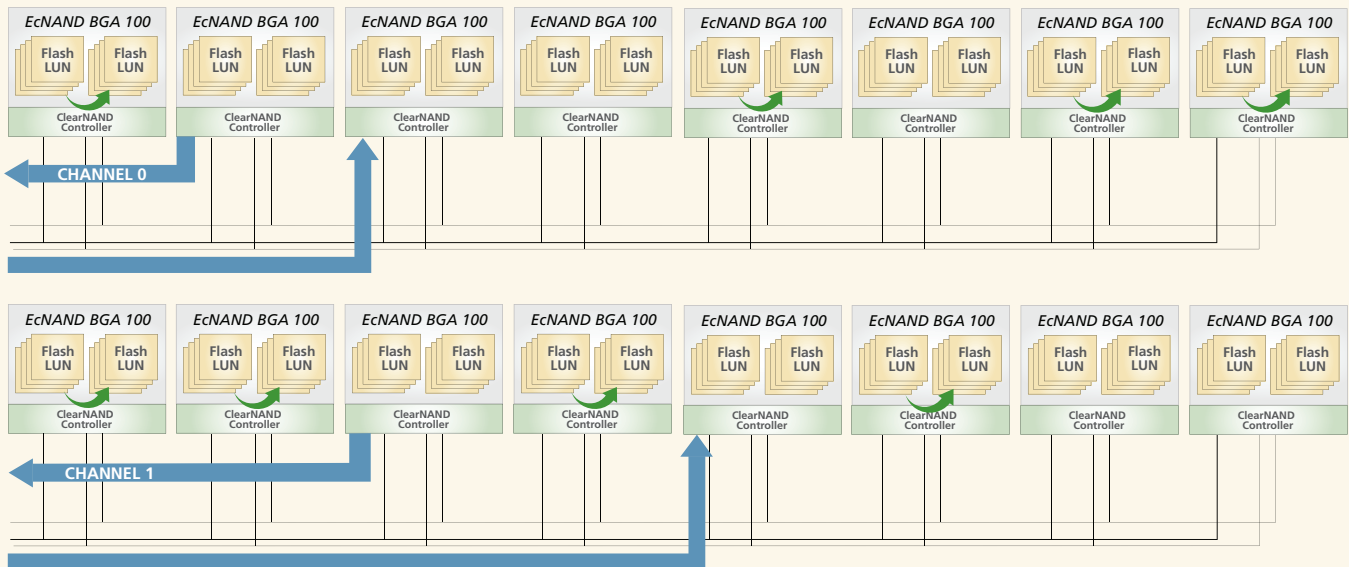


Figure 7: Copyback Using Enhanced ClearNAND Flash

micron.com

Products are warranted only to meet Micron's production data sheet specifications. Products and specifications are subject to change without notice.

Micron, the Micron logo, and ClearNAND™ are trademarks of Micron Technology, Inc. All other trademarks are the property of their respective owners. ©2010 Micron Technology, Inc. All rights reserved. 12/27/10 EN.L

