The Challenges of Scaling Nonvolatile Memory in Embedded Systems

How Micron e•MMC Embedded Memory Simplifies High-Capacity Storage

Daniela Ruggeri  
NAND Product Marketing, Embedded Segment Group  
Micron Technology, Inc.  
July 31, 2013

Overview

This paper discusses how Micron’s e•MMC embedded memory provides necessary NAND Flash functions in an easy-to-use BGA package, saving significant resources that would otherwise be allocated to hardware and software development.

Traditional NAND Flash Solutions

Electronic product designers in all market segments—not just mobile and consumer—need high-capacity media that can enable their devices to store more content, multimedia files, and simple data. NAND Flash memory meets these demands with single-level cell (SLC) and multilevel cell (MLC) technologies.

SLC NAND Flash for High-End Systems

SLC NAND Flash continues to be a good choice for designs using low- to mid-range densities, giving engineers more flexibility to fine-tune their system and maximize SLC’s best-in-class performance and reliability. SLC is ideal for high-end systems where cost reduction is not the priority, and investments can be made in additional system resources, like fast microprocessors, RAM buffer space, and optimized software that can minimize internal latency and enable parallel operations, pipelines, and multi-threads with full control over time-consuming operations (e.g., garbage collection and wear leveling).

The introduction of NAND Flash was significant for engineers managing error correction code (ECC). For many years, software algorithms met 1-bit/512-byte ECC requirements and were used in spite of other technological advances. When moving from 1-bit/512-byte to 4-bit/512-byte ECC, the transition to 34nm lithography presented the biggest challenge because software algorithms were no longer appropriate and many processors were not yet equipped to handle more than 1-bit ECC.

Current SLC NAND Flash devices require only 1-bit ECC for smaller-density, legacy solutions and 8-bit/512-byte ECC for the higher-density, newer solutions. Current-generation embedded processors include sophisticated ECC capabilities with direct support for 2xnM SLC NAND Flash memory and future devices.

As SLC geometries shrink, ECC requirements are expected to increase to 24 bits/1024 bytes. e-MMC can be a good solution for managing this complexity.

Note: Micron’s Product Longevity Program (PLP) supports current 8-bit/512-byte products to ensure availability for an extended period of time regardless of upcoming shrinks and ECC requirement changes.

MLC NAND Flash for Cost-Sensitive Systems

Current MLC NAND Flash devices require 24–40 bits per 1024 bytes of ECC, which results in tight technology coupling with host controllers. MLC requires a tradeoff between endurance and performance for cost control. Additionally, MLC requires an efficient interface between the controller and the NAND Flash device. Designers can smooth implementation by selecting an interface that is supported by several controllers.
Traditional NAND Flash Design Challenges

Rapidly changing requirements present obvious design challenges for embedded system designers and processor manufacturers. To keep up with new hardware ECC requirements and architectural changes to improve NAND performance, such as page size increases and dual-plane array options, additional resources must be allocated for hardware and software development, which can present significant challenges.

ONFI Standard

To minimize the impact of competing and sometimes incompatible NAND Flash architectures, Micron and several controller manufacturers have joined together to create the Open NAND Flash Interface (ONFI) standard. The primary goal of the ONFI standard is to increase compatibility and make the embedded system designer’s job easier. Nevertheless, it is still important for embedded processor vendors to be aware of current NAND Flash development.
A Simpler Approach

While controllers that provide direct NAND Flash support will typically yield the lowest bill-of-materials (BOM) cost, a managed NAND solution like e•MMC embedded memory reduces the complexity of NAND Flash operations. Because of the high-level, abstracted interface, the controller and software can treat NAND Flash devices like simple, block-oriented file systems; and the NAND device can handle NAND Flash management tasks, such as error correction, bad block management, and wear leveling. A managed NAND solution can also reduce development time and resource allocation and can insulate designers from concerns about high-density NAND Flash technology changes.

Comparison of Raw NAND and e•MMC Embedded Memory

As shown in the figure below, raw NAND requires a host controller that supports a direct NAND Flash interface, and block management and wear leveling are handled by software running on the controller. e•MMC memory, on the other hand, combines a NAND controller and high-capacity NAND Flash in a single BGA package. Because the NAND Flash is managed by its own controller, the necessary software support can be provided by a simple, low-level driver. Managing several NAND Flash devices with a single controller is a cost-effective approach for any density using a fully managed solution that optimizes NAND algorithms and ECC.

Figure 3: Comparing Raw NAND and e•MMC Embedded Memory
**Benefits of e-MMC Memory**

e-MMC memory devices use the industry-standard Multi-MediaCard (MMC) interface. MMCs are used in several mobile and portable consumer devices; e-MMC memory is the chip version of the MMC. The MMC interface is becoming more common in embedded platforms for a variety of host controllers.

In addition to eliminating NAND Flash dependencies such as SLC/MLC or varying page sizes, e-MMC embedded memory offloads block management and wear-leveling tasks from the operating software to the controller. Depending on the NAND Flash support provided by the software, this can save valuable execution resources, time, and code storage space and could eliminate the need for a higher-performance controller or additional hardware/software design resources. The NAND controller inside the e-MMC chip is optimized to take advantage of specific NAND Flash performance features, including program caching, read caching, and the synchronous ONFI interface, which can help the e-MMC device provide significant performance improvements over other implementations.

A less-obvious benefit of e-MMC memory is the ongoing support for various NAND Flash densities and lithography. Because the interface to the host controller does not change, the underlying NAND Flash technology inside the BGA can change without impacting the application. This approach extends the longevity of higher-density solutions and enables support for multiple densities with a single printed circuit board (PCB).

**e-MMC Special Features**

**Partitioning:** An enhanced e-MMC attribute enables areas to be configured as SLC NAND areas via partitioning to ensure higher performance and reliability. By default, the NAND user data area in the e-MMC memory is configured as MLC. Up to the entire MLC capacity can be converted to SLC, but the converted capacity will be divided in half. See Micron Technical Note, TN-52-07: e-MMC Partitioning, for more information.

---

**Figure 4 Terms Defined**

- **Boot area partitions:** Boots the device (enhanced by default)
- **Replay-protected memory-block partition (RPMB):** Accessed through trusted mechanism and hidden security key for secret/crucial data (enhanced by default)
- **General partitions:** User-defined areas
- **User data area:** Stores data

*Note: This device supports enhanced features to distinguish certain areas from the default storage media.*

---

**Figure 4: e-MMC Memory Partitioning**
**Booting**: It is possible to boot directly from the e-MMC device, as described in Micron Technical Note, TN-FC-06: Booting from Embedded MMC – JEDEC 4.41.

**Partition Protection**: When a power loss occurs during a WRITE operation, e-MMC memory uses partition protection to protect static (previously written) data on MLC partitions. Partition protection ensures that the target address never contains undefined data; old data remains unchanged until new data has been successfully programmed, which also guarantees that data in an adjacent-paired MLC page is protected.

**JEDEC 4.51 Key Features**: e-MMC memory complies with the JEDEC 4.51 standard; its key features are summarized in the table below.

### Table 1: Summary of JEDEC 4.51 Key Features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
<th>Benefits</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Discard</strong></td>
<td>Enables the host to identify data that is no longer required and does not need to be saved on the NAND device&lt;sup&gt;1&lt;/sup&gt;</td>
<td>• Optimizes garbage collection</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Improves system performance</td>
</tr>
<tr>
<td><strong>Sanitize</strong></td>
<td>Removes data no longer physically required by the e-MMC device</td>
<td>• Improves data security</td>
</tr>
<tr>
<td><strong>Power-Off Notification</strong></td>
<td>Enables the host to notify the e-MMC device before powering the device off; the device can then work to improve system status by creating routines to better arrange data and metadata before power-off</td>
<td>• Reduces e-MMC initialization time during the next power-up</td>
</tr>
<tr>
<td><strong>HS200 Interface</strong></td>
<td>Provides a standard, high-performance MMC interface to support up to a 52 MB/s or 200 MB/s bandwidth using an 8-bit data bus and a single data rate (SDR) up to 200 MHz CLK&lt;sup&gt;2&lt;/sup&gt;</td>
<td>• Improves e-MMC bandwidth</td>
</tr>
<tr>
<td><strong>Send Tuning Block</strong></td>
<td>Enables the host to read a predefined tuning block from the e-MMC device</td>
<td>• Finds the optimal sampling point of HS200 data input signals</td>
</tr>
<tr>
<td><strong>Updated Boot Protection</strong></td>
<td>Provides write protection independent of two boot partitions</td>
<td>• Improves flexibility</td>
</tr>
<tr>
<td><strong>Real-Time Clock</strong></td>
<td>Provides real-time clock information to the e-MMC device</td>
<td>• Executes maintenance operations (e.g., REFRESH)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Helps device handle data retention issues</td>
</tr>
<tr>
<td><strong>Packed Commands</strong></td>
<td>Sends groups of READ/WRITE commands in one transfer on the bus&lt;sup&gt;3&lt;/sup&gt;</td>
<td>• Reduces overhead</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Improves performance</td>
</tr>
<tr>
<td><strong>Data Tag</strong></td>
<td>Enables the e-MMC device to receive information about specific data types from the host; the host must manage logical block addressing (LBA) and the size of the tagged data</td>
<td>• Enables more reliable and robust storage of sensible data</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Improves read/write performance of tagged data</td>
</tr>
</tbody>
</table>

(Continued on page 6)
The Challenges of Scaling Nonvolatile Memory in Embedded Systems

Table 1: Summary of JEDEC 4.51 Key Features (Continued)

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
<th>Benefits</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cache</strong></td>
<td>Provides temporary storage space in the e-MMC device, and enables internal NAND management</td>
<td>• Reduces busy time</td>
</tr>
<tr>
<td><strong>Context ID</strong></td>
<td>Identifies related data transactions and defines data characteristics; manages commands with a logical association</td>
<td>• Reduces overhead</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Improves performance</td>
</tr>
</tbody>
</table>

**Notes:**
1. ERASE or DATA-COMPACTING operations are not executed.
2. The 200 MB/s bandwidth was introduced with the new HS200 JEDEC 4.51 standard interface.
3. The packed commands feature can be used by a host that manages a queue of commands to the e-MMC device.
4. The cache feature does not guarantee data integrity if a power loss occurs, which makes the feature ideal for non-sensible data.

**e-MMC Embedded Memory and Open Source Software**

Linux supports e-MMC memory and allows integration within its subsystems, which is an advantage for e-MMC diffusion. Initially, patches were available to support advanced features like Linux enablement introduced in JEDEC standard 4.41, but now these features are available by default. See Micron Technical Note, TN-52-05 e-MMC Linux Enablement, for more information.

e-MMC adoption in Android systems is driven by Google smartphone development and is becoming common in a broad range of embedded applications, from in-car navigators, smart TVs, and digital cameras, to washing machines and refrigerators.

**Conclusion**

For designs requiring low- or medium-density NAND Flash, SLC NAND Flash continues to be a good choice.

For higher-density designs, several host controllers already support the necessary ECC for today’s SLC NAND Flash devices. The challenge for next-generation embedded processor designers is how to meet the increasing ECC requirements of MLC NAND Flash devices.

The ONFI standard should help minimize the differences among NAND Flash devices from different vendors. However, hardware development to support even these ONFI-compatible devices will still reside with processor manufacturers. The software development required to support all direct-access NAND Flash devices will keep third-party software vendors, developers, and system integration groups busy.

Processor manufacturers that provide ongoing support for direct NAND Flash will typically provide the lowest overall pricing. For manufacturers that elect not to support future NAND requirements, or whose roadmaps do not align directly with those of NAND Flash suppliers, e-MMC embedded memory offers an attractive solution.