

An Overview of NAND Flash Memory Controllers

The demand for NAND Flash memory is growing at a phenomenal rate. In 2005, worldwide revenue for NAND Flash rose to \$10.8 billion, up 60% from 2004. The inclusion of NAND Flash in an increasing number of MP3 players, high-end cell phones, and digital cameras has fueled expectations that NAND will overtake NOR in a number of markets. Rapid increases in the capacity of NAND Flash, coupled with mobile consumer products that demand ever-increasing amounts of data storage, lay out a very bright future for this technology in the coming years.

What is NAND Flash?

NAND Flash is a form of non-volatile memory introduced by Toshiba and Samsung in 1989. Its benefits include high storage density, fast access times, low power requirements in operation, and excellent shock resistance. These benefits are tempered somewhat by the inherent limitations of the technology:

- Due to production yield constraints, NAND Flash ships from the factory with a number of bad blocks that cannot be used.
- NAND Flash has a serialized data interface similar to a hard disk and so cannot be used for execute in place functionality, such as running code directly from the chip.
- Over time and after multiple erase cycles the memory will tend to “wear” and become less reliable.
- The bits stored in the chip can sometimes flip - or reverse - from 0 to 1 or vice versa.

Over time, NAND Flash technology and software has evolved to the point where these limitations are invisible to the end user. Owners of iPod Nanos do not overly concern themselves with bad blocks or bit-flipping.

What does a NAND controller do?

Increased Performance

Approximately 3% of the overall flash array is reserved as a “spare area” in order to cope with flash vulnerabilities, like bit-flipping and bad blocks. Approximately 3 to 6 bytes in the spare area are reserved for error detection and correction algorithms, while the remainder of the spare area is used for remapping bad blocks. Without a NAND controller the algorithms that handle these functions would be executed by the general purpose CPU. However, NAND

controllers are available that will perform these functions in hardware. This reduces the CPU load on the device which has the effect of increasing battery life as well as increasing performance – both important factors in mobile technology. In addition, the advent of Multi-Level Cell (MLC) NAND Flash Technology requires *even more* rigorous error detection algorithms due to the increased chance of an error occurring. A NAND controller is perhaps the only reasonable way of performing this task.

NAND controllers can implement read and write caching and transfer data to and from the NAND Flash chip independently of the general purpose CPU. This increases the overall throughput and can again reduce the load on the system.

Faster Integration

The serialized data interface to NAND presents a difficult scenario for transferring data in and out of the chip. Unlike other memory technologies, like NOR or DRAM, an address must be fed in a bit at a time, at just the right time, and then read or write the actual data in a similar fashion. A NAND controller encapsulates the interface with the NAND chip and handles this communication, thereby presenting an effortless interface to the user.

The lack of standardization among NAND Flash manufacturers has been problematic throughout its brief history. The use of NAND controllers allows some flexibility in the choice of a NAND Flash chip as any given controller will support a range of NAND chips. Changes in NAND Flash, therefore, will not necessarily require a change in NAND controller or software.

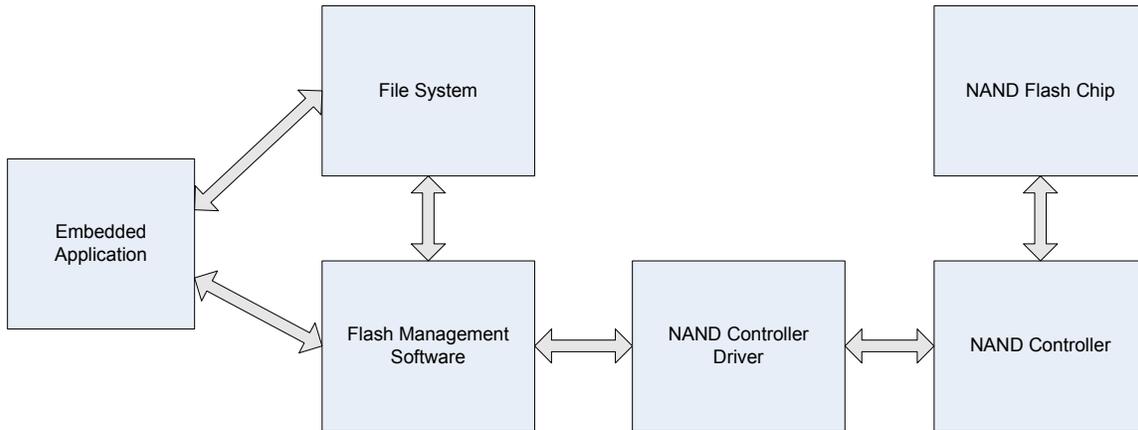
Integrated NAND Controllers

Integrated NAND controllers are growing in popularity as mobile processor vendors, such as Freescale and Texas Instruments, build NAND controllers directly into their processors. The advantages of this approach are that the design can be kept small and the costs can be minimized. The disadvantage is the inevitable delay between a new NAND Flash technology appearing and the release of an integrated processor that supports it.

A dedicated, or external, NAND controller is one in which the NAND controller is a separate chip that can be sourced independently of other parts. The advantage of choosing a dedicated NAND controller is the wide range of latest and greatest NAND Flash chips available. This is due to the relatively short time to market for a chip with this level of complexity. The disadvantage of using a dedicated NAND controller is that chip uses additional PCB space and will cost more money than the integrated approach.

Specialized NAND Controller Software

The diagram below shows an overview of the interaction between the embedded application, file system, Flash management software, NAND controller, NAND controller driver, and the NAND Flash memory chip.



NAND Controllers require specialized software to hide the unpleasant details of the NAND controller and NAND Flash itself and presents the user with a clean and highly usable interface to the memory.

At a higher level, the NAND controller driver is often utilized by an intelligent block device driver, such as FlashFX Pro from Datalight, or a flash file system. Some important functions of this software are:

- Reducing the integration time for developers of embedded applications.
- Handling bad blocks present on the NAND Flash chip.
- Providing small block emulation, which is required due to the relatively large “erase blocks” present on NAND Flash.
- Mitigating the effects of wear on the NAND flash through the application of wear leveling algorithms that spread writes and erases over a wider area.
- Providing a file system interface that allows you to interact with the Flash chip at the file level.

Summary

The strong desire for more storage and better performance in consumer devices has elevated the pressure on NAND Flash and introduced new technologies like NAND Controllers to device engineers. In order for new flash technologies to be adopted, it is critical that the enabling software also continues to evolve.