

# CAST



## NANDFLASH-CTRL

### NAND Flash Memory Controller Core

Implements a NAND Flash memory controller for ONFI 2.2 compliant, high-capacity Multi-Level Cell (MLC) and Single-Level Cell (SLC) NAND flash memories. The core supports memory devices from Hynix, Micron, Samsung, ST-Micro, Toshiba, and others, and is configurable to handle diverse application, performance, and memory requirements.

The controller core efficiently manages the read/write interactions between a master host system and NAND flash memory devices by implementing hardware blocks that off-load the processor and increase system performance. It uses a comprehensive command set for easy memory access, automatically remaps corrupted memory blocks to improve reliability (Bad Block Management), has built-in power-saving features, can perform AES 256 encryption, and can boot software directly from flash memory.

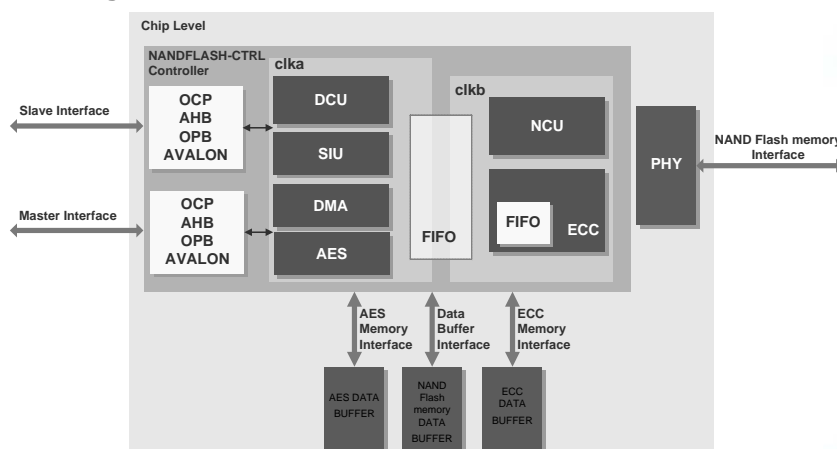
The core performs error code correction (ECC) based on a sophisticated implementation of the BCH (Bose, Ray-Chaudhuri and Hocquenghem) algorithm. The ECC strength—fixed or run-time programmable—is 2 to 32 bits (strengths up to 64 bits are available by request).

The NAND flash controller core is easy to integrate. Its deliverables include an OS-independent device driver, and a soft-PHY that can be easily targeted in any technology. A built-in, high-performance DMA engine is available. Common system buses are supported—AMBA® AHB, OCP, PLB, or Avalon—and other interfaces such as AMBA AXI are available by request.

Two versions of the core are available: a cost-efficient asynchronous mode-only version, and a high-speed, fully-featured version. Pre-synthesis and run-time configurability features make the core suitable for applications with different of bandwidth, area, and power requirements.

This sixth-generation product builds on production-proven previous versions of the controller. Developed for reuse in ASICs and FPGAs, the controller core is fully synchronous with positive-edge clocking, has no internal three-state buses, and uses a synchronous reset so scan insertion is straightforward.

### Block Diagram



### Features

#### Wide Devices Support

- ONFI 1 & 2.2 compatible
- Source synchronous (DDR) and asynchronous mode
- Toggle Mode DDR
- Clear NAND
- SLC and MLC devices
- Small and large block devices
- All popular device manufacturers

#### High Memory Bandwidth

- Up to 200MT/s
- At-speed ECC
- Advanced command sequencing & parallel operation of different devices.

#### Easy SoC Integration

- Portable device driver in C
- Technology-independent PHY
- Booting from flash, with configurable boot sequence
- Available system interfaces for:
  - AMBA™ AHB
  - OCP
  - PLB
  - Avalon
  - AXI (soon)
- Built-in optional Direct Memory Access (DMA) engine

#### Hardware Functions to Offload Host Processor

- Optional AES256 encryption
- 2 to 32 bits of BCH ECC, optionally extendable to 64 bits
- Bad Block Relocation
- Multipage transfers

#### Flexible Configuration & Run-Time Operation

- Lower cost version for asynchronous memories only is suitable for long-term storage and boot applications
- Configurable number of memory banks and devices per bank
- Adapts to a variety of system and memory types, with configurable (via special function registers):
  - timing parameters
  - 0/1/2/3/4/5 address cycles
  - ECC calculation turn on/off
  - Write/Erase Protection size
  - Interrupt enable/disable
  - Optional support for the Write/Erase Protection

## Applications

The core is suitable for controlling embedded storage (e.g. in mobile devices, network routers, and point-of-sale systems) and solid state device (SSD) mass storage for USB flash drives, digital cameras, laptops, and more.

## Functional Description

The NANDFLASH-CTRL core is partitioned into modules and comes with external elements as shown in the block diagram and described below.

### DCU — Design Control Unit

Controls all other modules based on the SFR values and current controller state. The main tasks of this module are:

- Provide enable/disable signal to the DMA and SIU units when they try to get access the FIFO module. Only one of the two units can be active at time.
- Enable/Disable the ECC module.
- Provide control signals to the NCU unit
- Execute the boot sequence
- Interrupt controller

### SIU — System Unit Interface

Opens a window in the address space where all SFRs are visible, providing access to these elements. The SIU works as glue logic between the system interface and internal controller bus, coordinating their interaction. It is responsible for generating the internal request signal, when the controller buffer must be read directly using the controller interface, and it holds transmission on the external bus if access can't be granted.

### DMA

The scatter-gather first-party DMA controller speeds up the data transfers between a device on the system bus and the NAND flash memory, and decreases the system bus burden. It can read/write data between the internal BUFFER and a device on the system bus. A DMA transfer can be released by writing to the SFR or by the FSM (during automatic Page read/write); the transfer parameters are set by the SFR.

### FIFO

A 32-bit width asynchronous FIFO module that facilitates transferring data between the input module and NCU when the command sequence is executed.

### NCU — NAND Control Unit

Responsible for the generation of the NAND flash device access sequences. The unit uses the control signals provided by the DCU. The NCU uses a proprietary interface to the PHY Interface unit.

### ECC — Error Correction Code Unit

This is both an error correction code calculator and a correction unit. A correction word is calculated for each 256B or 512B (optionally per 1024 or 2048B) sub-page of the NAND flash memory page. During the read operation, the unit can automatically correct bad bits without any interaction with the external system. It has a status register, the bits of which signal errors occurring during a read, and then inform if errors were corrected.

The ECC module has an integrated FIFO that is used to transfer the calculated words to the NCU modules during the encode process, and to store the calculated partial syndromes during the decode process.

### AES — Advanced Encryption Standard Unit

Implements the encryption algorithm and can work with all standard-specified key lengths (128, 192 and 256 bits) and in ECB (Electronic Codebook) mode. It processes an 128-bit vector in 11/13/15 clock cycles.

### PHY

This module provides the DDR data interface for new High Speed devices.

### NAND Flash Controller Software Driver

An included, low-level software driver that represents a first abstraction layer of the core to relieve the higher-level application layer from hardware management.

Written in "C," it supports common memory functions as well as the custom memory features of Micron, Samsung and ST devices. A DMA support module configures the DMA and transmits data. The driver does hardware mapping of memory blocks and is configurable to reduce resource requirements.

## Support

The core as delivered is warranted against defects for 90 days from purchase. Thirty days of phone and email technical support are included, starting with the first interaction. Additional maintenance and support options are available.

## Verification

Extensive verification has been performed, using code coverage tests, simulation with multiple tools, and implementation and testing in an FPGA demonstration system.

Verification has been performed with many different devices from Micron, Samsung, STM and Toshiba.

## Implementation Results

NANDFlashCtrl reference designs have been evaluated in a variety of technologies. The following are sample Xilinx results for a typical configuration (4 devices per bank, support for high-speed devices, BCH 8, DMA, Soft PHY and memories) and optimized for speed.

Xilinx Device	Slices	Memory	I/Os <sup>1</sup>	Performance clka/clkb (MHz)	ISE
Spartan-3E 3S1200E-5	5,173	3 BRAM	266	65/45	12.4
Spartan-6 6SLX75T-2	1,812	3 BRAM	266	90/90	12.4
Virtex-5 5VLX85-3	2,467	3 BRAM	266	125/100	12.4
Virtex-6 6VLX75T-3	1,988	3 BRAM	266	190/120	12.4

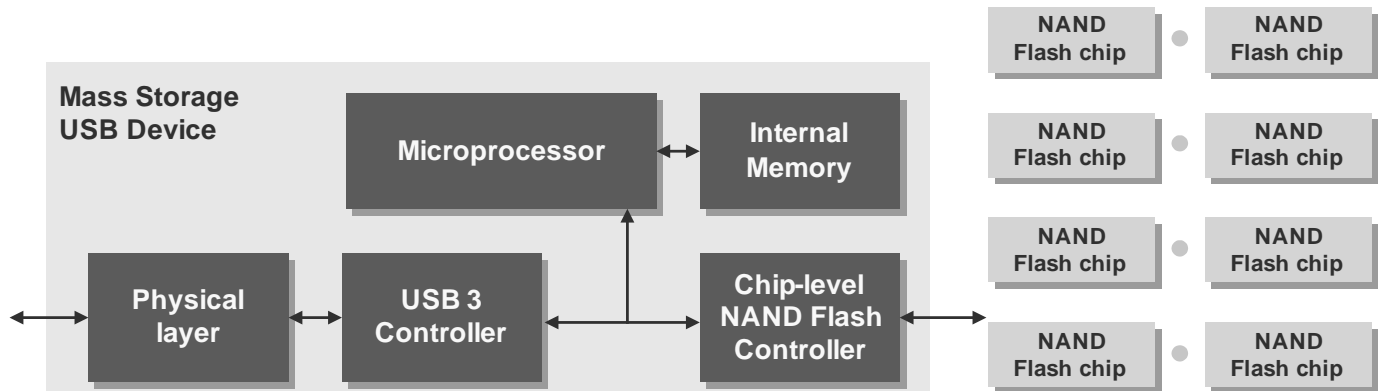
<sup>1</sup> Assuming all I/Os are routed off-chip

## Deliverables

The core includes everything required for successful implementation:

- VHDL or Verilog source code for the core
- An example design that uses the core and illustrates how to build and connect memory and tri-state buffers
- Sophisticated HDL Test Bench that instantiates the example design and related elements
- Interface wrappers when ordered
- Simulation script, vectors, expected results, and comparison utility
- Synthesis scripts
- Comprehensive user documentation, including detailed specifications and a system integration guide

## Example Application



For this USB thumb drive, the core is implemented with an internal buffer and tri-state buffers. Data goes through a USB controller, and a microcontroller manages data flow, sets parameters of data transmission for NANDFLASH-CTRL, controls the USB, and handles all aspects connected with a File System implemented in this storage device.