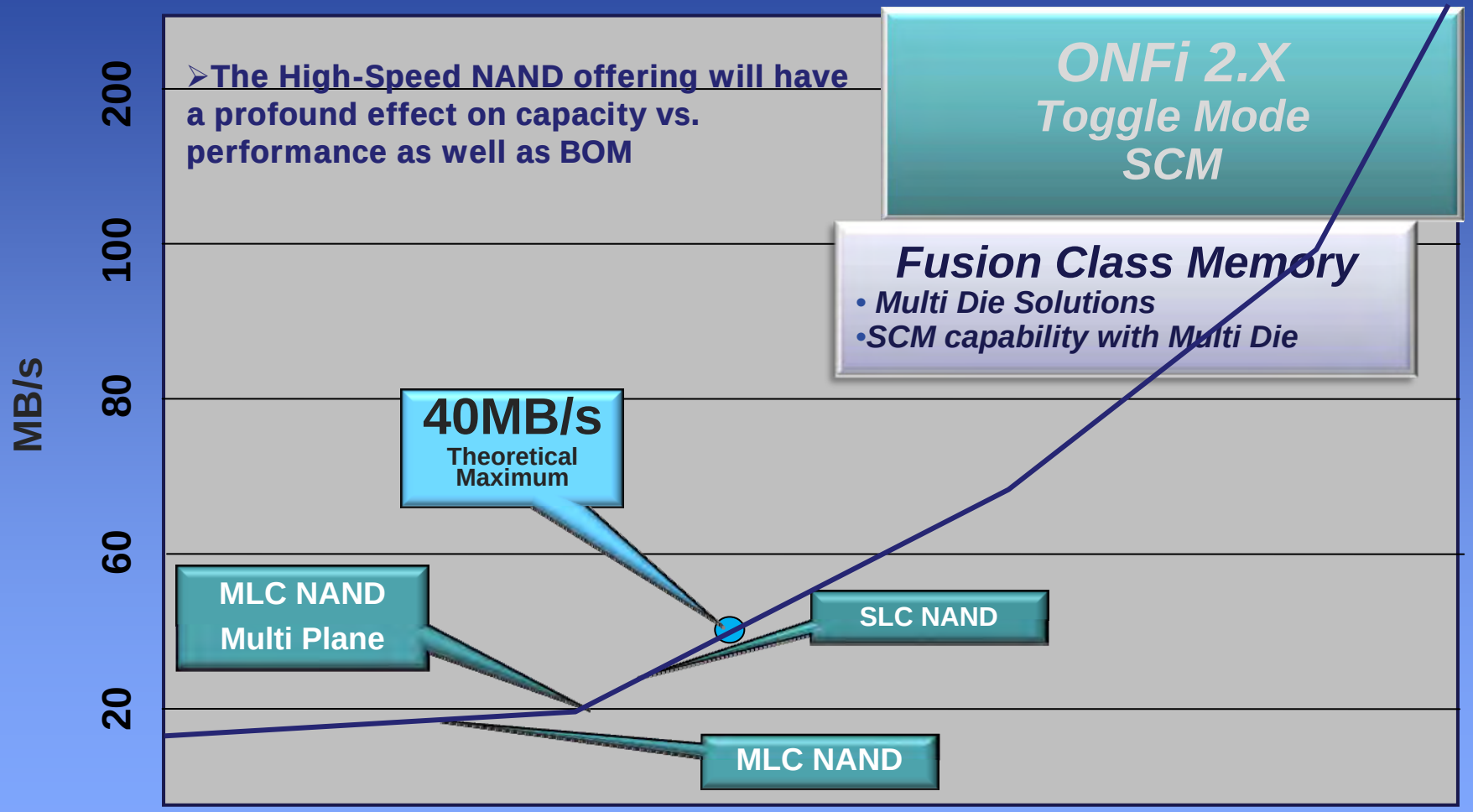
High-Speed NAND Flash

Design Considerations to Maximize Performance

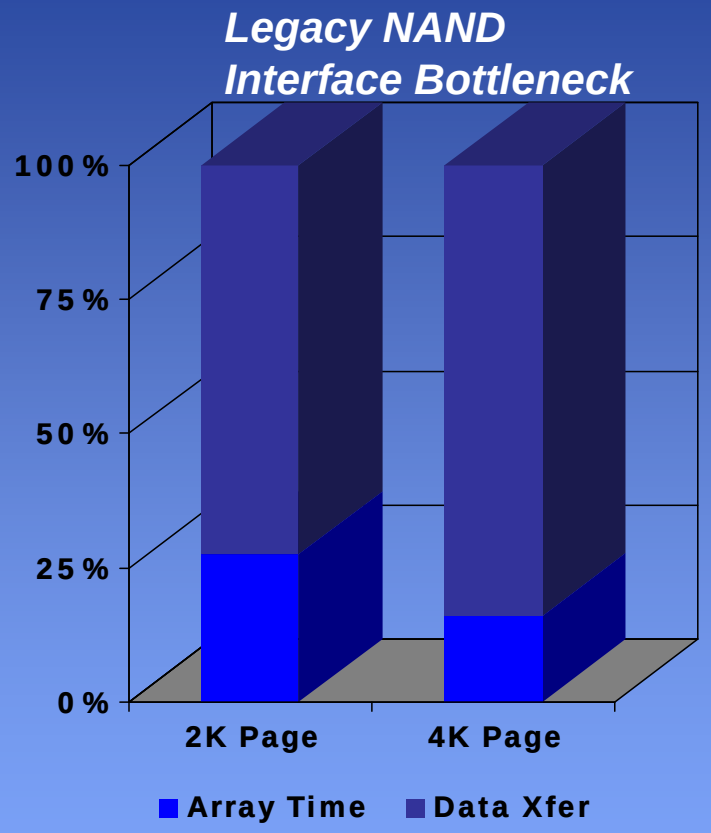
Presented by: Robert Pierce
Sr. Director, NAND Flash
Denali Software, Inc.

History of NAND Bandwidth Trend



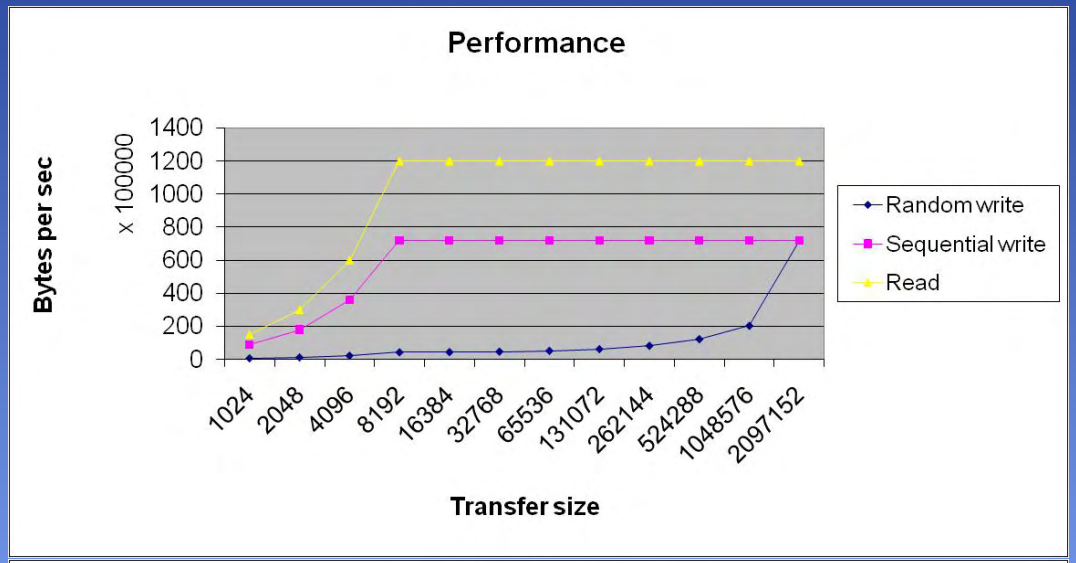
High-Speed Flash Interfaces

- The NAND Flash interface has been a bottleneck in achieving high performance for system applications
  - As page size increases to 4KB, the SLC tR time of ~20 µs is completely unbalanced with the data transfer time of ~100 µs in legacy/native NAND
- High performance applications (i.e. Cache, SSD's , etc.) have been unable to show the true capability for random operations required by today's systems and OS's
- Changes to the flash device architecture will have even more effect for these new devices
  - Page size increases
  - Multi Plane
  - Additional Spare area for Metadata
  - Enhanced commands

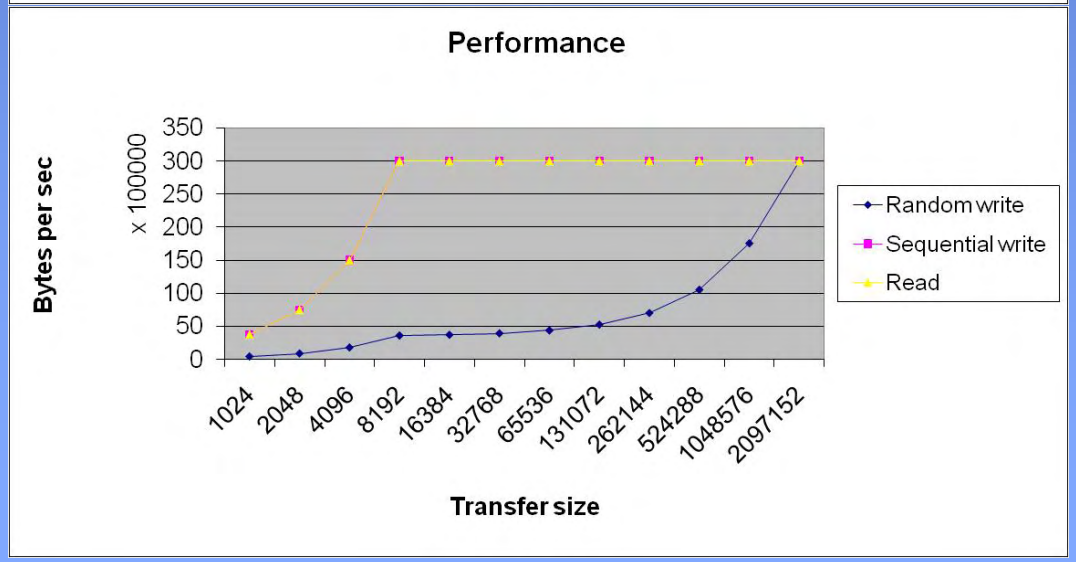


Performance Impact

- Toggle-based MLC NAND
- Eight controllers
- 4 CE
- 8K page size



- Asynchronous MLC NAND
- Eight controllers
- 4 CE
- 8K page size

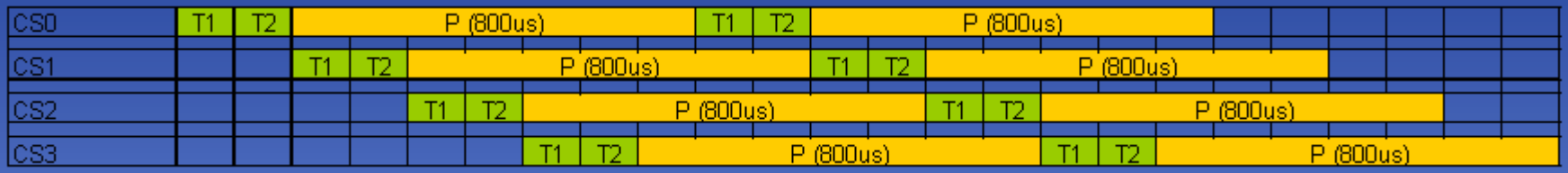


Key Aspects to Higher Interface Performance Improvements

- Increase the number of commands to the flash device
  - Maximizes the number of transactions for a device
  - Multi-plane architectures are very useful
- Interlacing, by CE or LUN
  - CE interlacing uses more pins
  - Polling mode not as useful
  - LUN (Logical Unit Addressing) very useful, with pin reduction
- Transaction size
  - 8K page size can increase Read BW

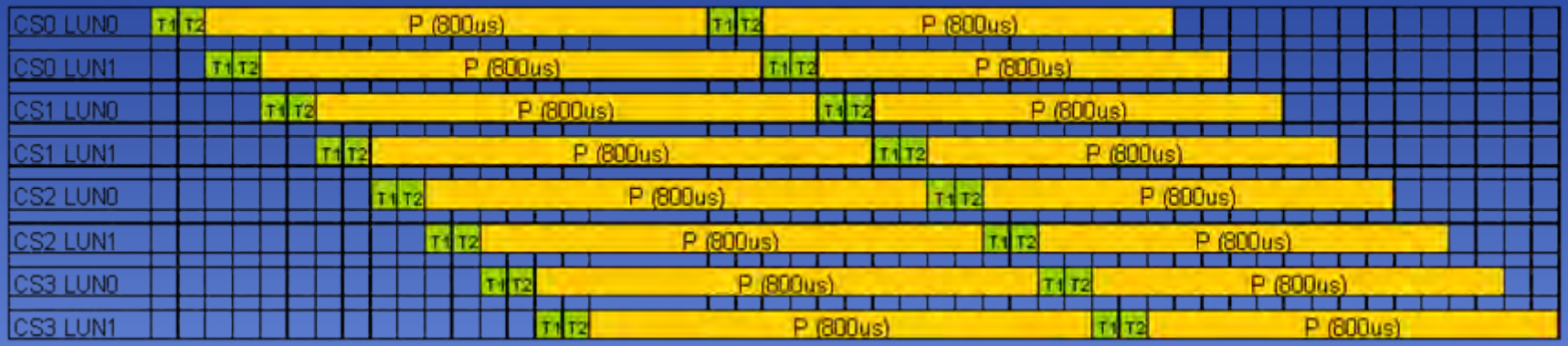
denali

Parallelism using Chip Enables



- 4CS Interleave
- 4KB Page size (transfer time is ~30us - 4096*7.5ns)
- Program time ~800us typical
- Dual plane support (T1 is transfer for plane 1 and T2 is transfer for plane 2)
- 860 us Program Cycle = One program time + two transfer times (800+30+30)
- 32 Kbytes data written in one program cycle
- 37 MBps Theoretical max throughput per program cycle
- 20% controller and flash software overhead
- 30 MBps estimated throughput

Example of LUN Interleaving



- 4CS with LUN Interleave
- Two LUNs (0 & 1) per CS
- 4KB Page size (transfer time is ~30us - 4096*7.5ns)
- Program time ~800us typical
- Dual plane support (T1 is transfer for plane 1 and T2 is transfer for plane 2)

- 860 us Program Cycle = One program time + two transfer times (800+30+30)
- 64 Kbytes data written in one program cycle
- 74 MBps Theoretical max throughput per program cycle
- 20% controller and flash software overhead
- 60 MBps estimated throughput
- Achieved twice the throughput with LUN interleaving

Timing Considerations - Interleaving

- The earlier examples were without any interface overhead

In Reality

- There is idle time required when:
  - we switch between devices, dies during an die-interleaving operation
  - we switch between chip enables during interleaving
- System integrator needs to look at a combination of array performance timing as well as the inter-command idle time to arrive at target achievable performance

High-Speed Controller Key Features

Key Toggle Features

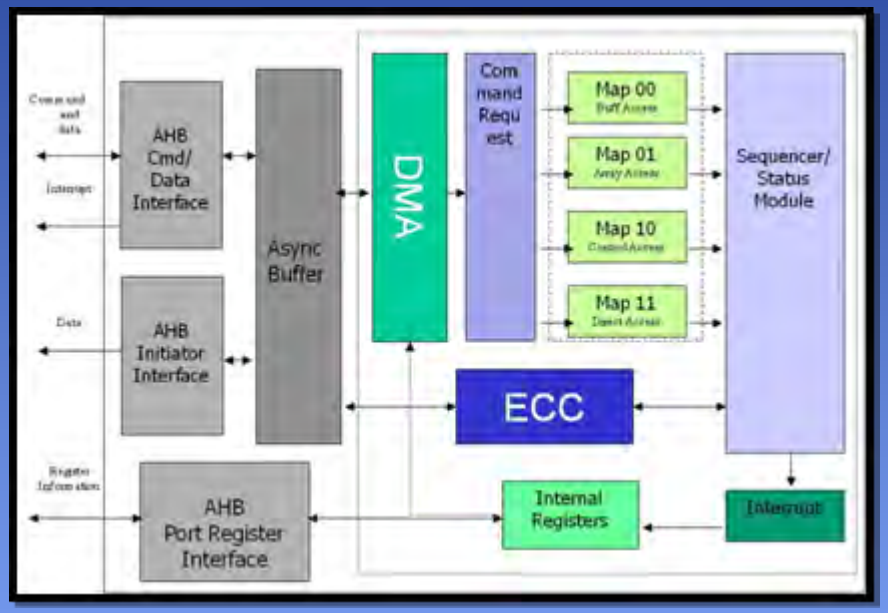
- 63 and 83 MHz operation
- Multi Plane support
- Multiple I/O voltage
- I/O strength support
- Cache Read/write commands
- Programmable/Erase lockout during power transitions

Key ONFi 2.1 Features

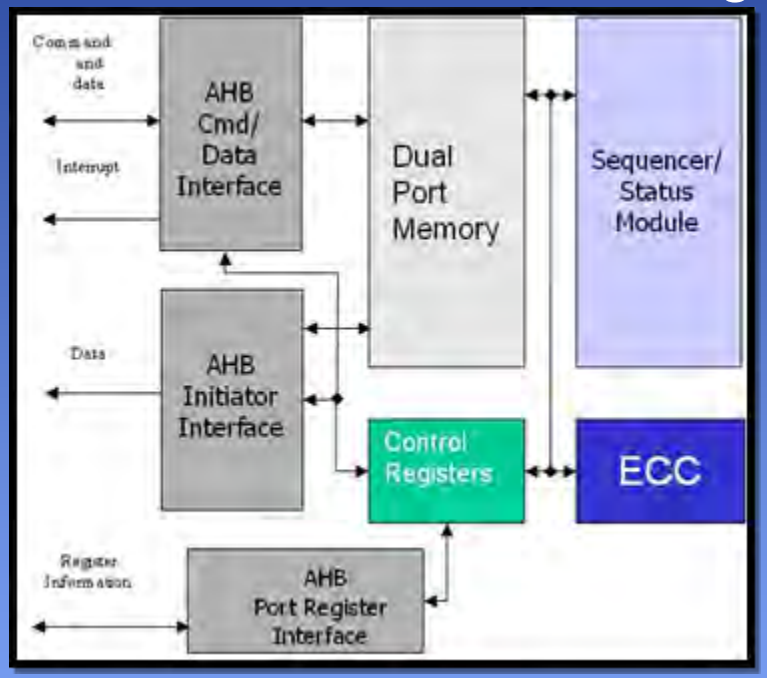
- Discovery and Initialization
- LUN addressing
- Interlaced and non-interlaced addressing
- Source synchronous operation
- Staggered power up
- I/O strength support
- ONFi 1 modes 0,1,2,3,4,5
- ONFi 2 mode support 1,2,3,4,5

Flash Controller HW Architectures

HW Accelerated Controller

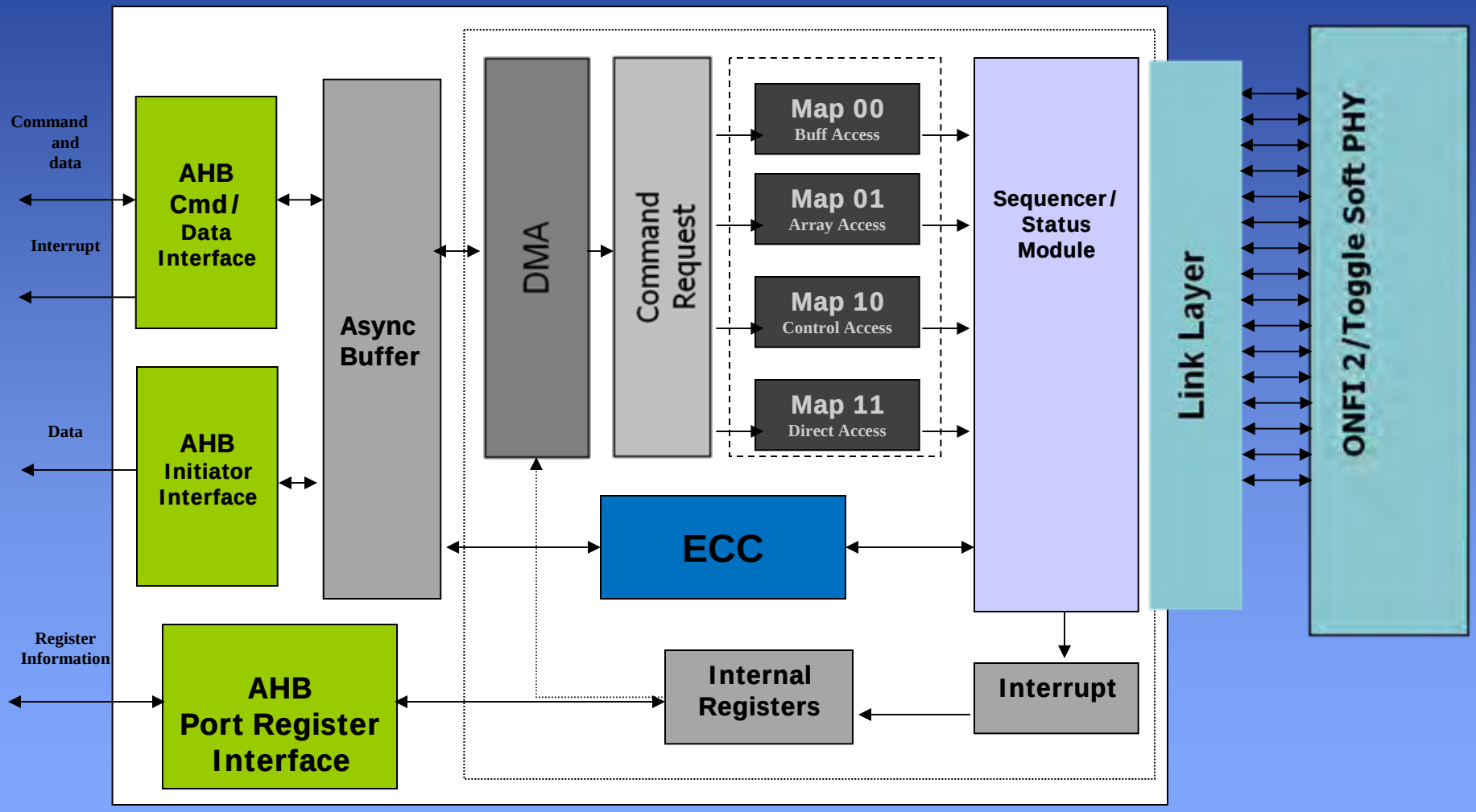


Software Driven Flash Timing



- Key Differences
  - Flash command execution
  - Interrupts
  - Processor overhead

NAND Controller w PHY Support



High-Speed NAND Challenges

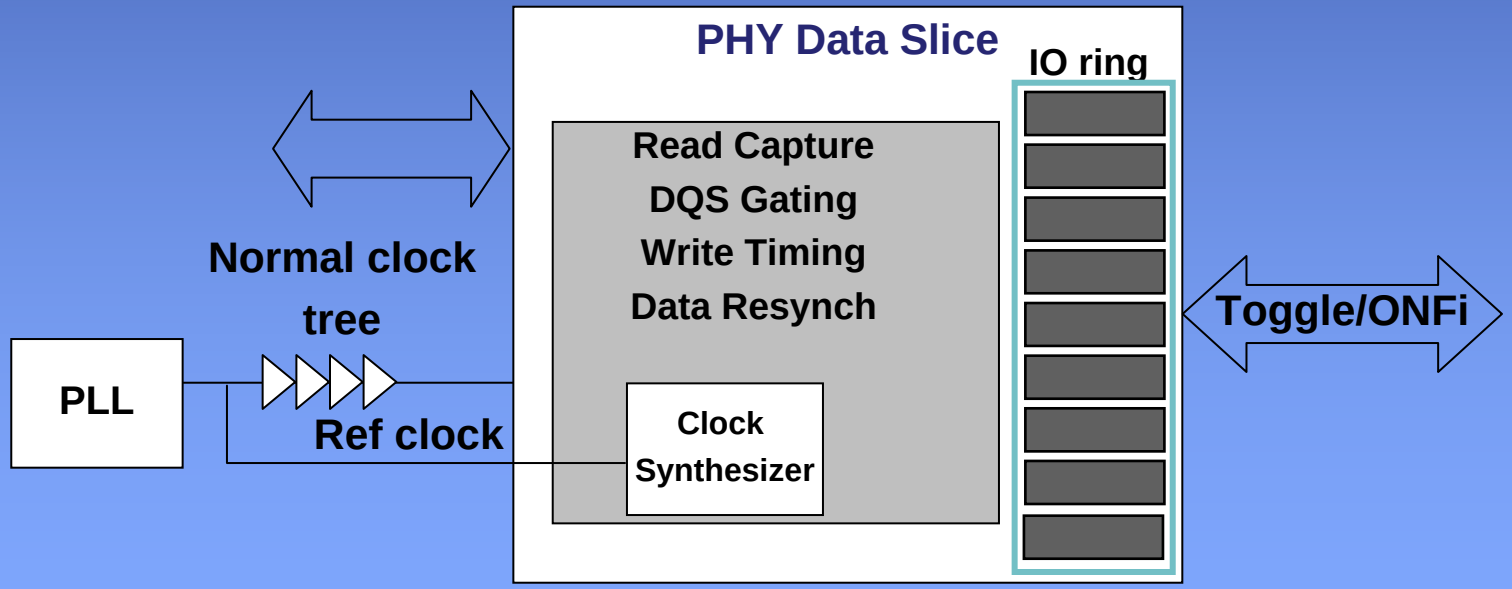
- New NAND devices (e.g.Toggle NAND, ONFi 2.X) offer tremendous performance improvements over past solutions
- Using old controller and firmware solutions will be unable to utilize this performance capability
- Physical interface requires a more defined solution, not only for timing but for legacy support
  - Multi voltage I/O's
  - Programmable drive strength
- Latency in the controller will increase buffer overhead
- Multi page size and ECC options need to be present in all HS applications

denali

PHY Overview

PHY Architectural Overview

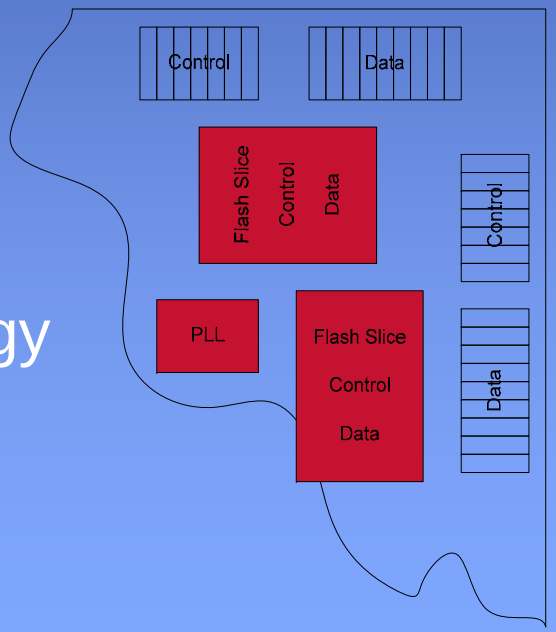
- Separate PLL
  - Use for multiple slices
- Soft PHY slice
  - Highly reusable
  - Flexible layout
- Test Logic for at-speed test
- No DLL reduces power and gate count. 4X clock at IO frequency
- Clock reference
  - Minimally buffered PLL input to slice for source synchronous domain
  - Normal clock tree for DFI, flop-to-flop timing



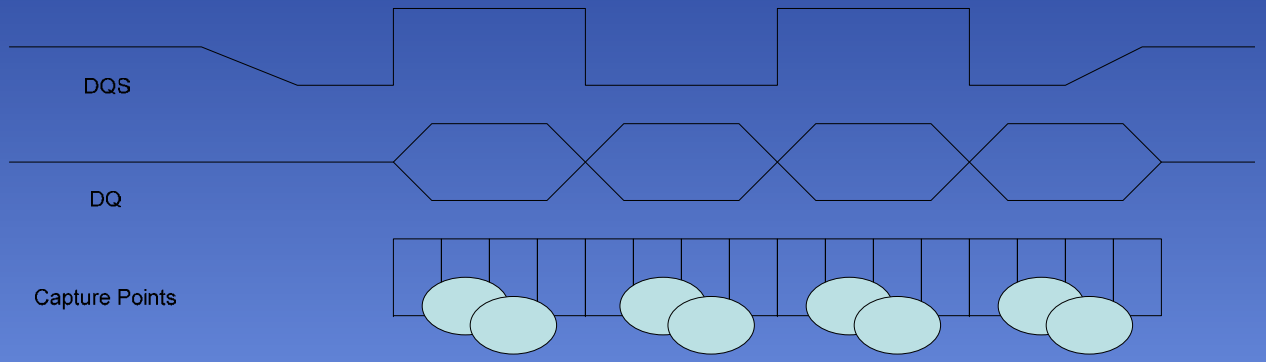
Available for SOC now, FPGA Support soon.

Soft PHY Solution

- Works with ONFi2 and Toggle as well as legacy flash
- Base design has been verified by DDR DRAM controller
- Process technology agnostic
- Scalable to many multiple channels
- Multiple drive strength support for new H.S device
- No DLL, simplified clocking methodology
  - No 3rd party core IP
  - I/O's need to be supplied

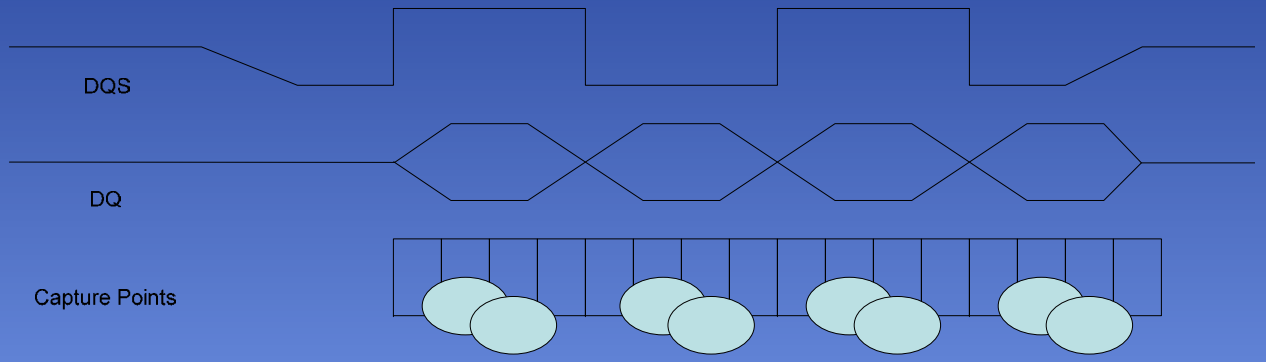


8 Phase Read Capture DRAM



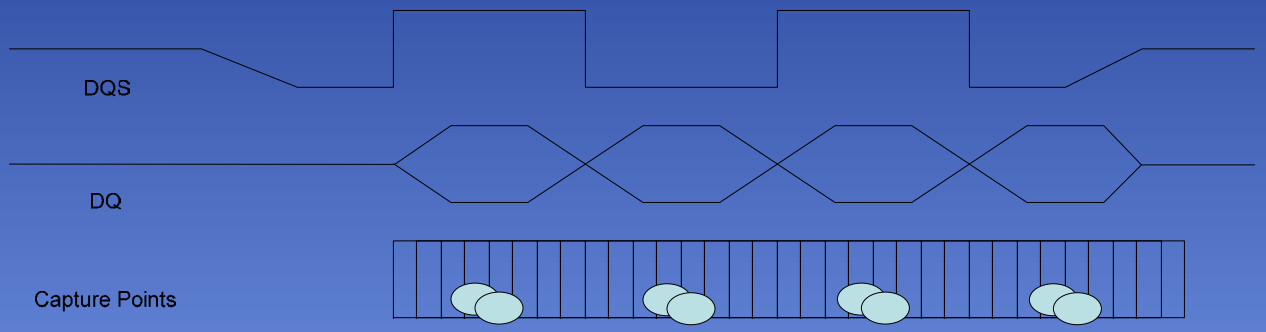
- DQS to DQ valid = tDQSS<.10clk
- DQS to DQ invalid = tDH>.38clk
- DQS capture at .125clk, .25clk and .375clk
- Three valid capture points are available when we need only two for reliable capture because of pattern matching

8 Phase Read Capture Flash



- DQS to DQ valid = tDQSS<.092clk
- DQS to DQ invalid = tQH>.322clk
- DQS capture at .125clk, .25clk and .375clk
- No read capture points; Reason is tDQSS is larger than .125clk with I/O uncertainty normally used for flash (500ps), the second read capture point is valid, the third capture point is never valid
- This could be used if I/O uncertainty was less than .033clk at 100MHz or 330ps

16 Phase Read Capture Flash



- DQS to DQ valid = tDQSS<.092clk
- DQS to DQ invalid = tQH>.322clk
- DQS capture at .125clk, .1875clk, .25clk and .3125clk
- Four read capture points: the first and last may not be reliable due to I/O uncertainty, but the two middle capture points will always work with pattern matching

Conclusion

- To maximize performance, new architectures and solutions are necessary to achieve the performance that the new High-Speed Flash devices offer
- High overhead software solutions will have difficulty achieving desired performance levels
- Trends in the Page size as well as ECC sector size will have an interesting effect for SSD and high capacity flash array applications
- It is possible to support both Legacy and High-Speed solution with one device
- The increase in commands and addresses will put more burden on the processor and the Host interface

denali