Leveraging Flash Memory to Address Enterprise Class Workloads

Matt Pujol

Sr. Hardware Product Manager
Flash Products

Hitachi Data Systems

Consider Enterprise Workloads and Flash

- Usually the discussion centers around tiering, dedupe and RAID types.
  - Really just a bunch of software (typing, really ;))
- Let's focus on the hardware architecture
  - Where the heavy lifting is done

http://search.dilbert.com/comic/Overuse%20Colons

Overview

- Review current architectures
- Discuss some of the limitations
- Outline some design considerations to overcome these limitations

Flash Controller Evolution

- First designs were mostly USB keys and CF cards
  - Not necessarily enterprise class
  - Data integrity or performance was not key
- Designs focused on low cost and low power
  - Smaller microprocessor
  - Limited error recovery
- With the evolution of high speed large capacity devices came the dreaded “Write Cliff”

A Representative Architecture

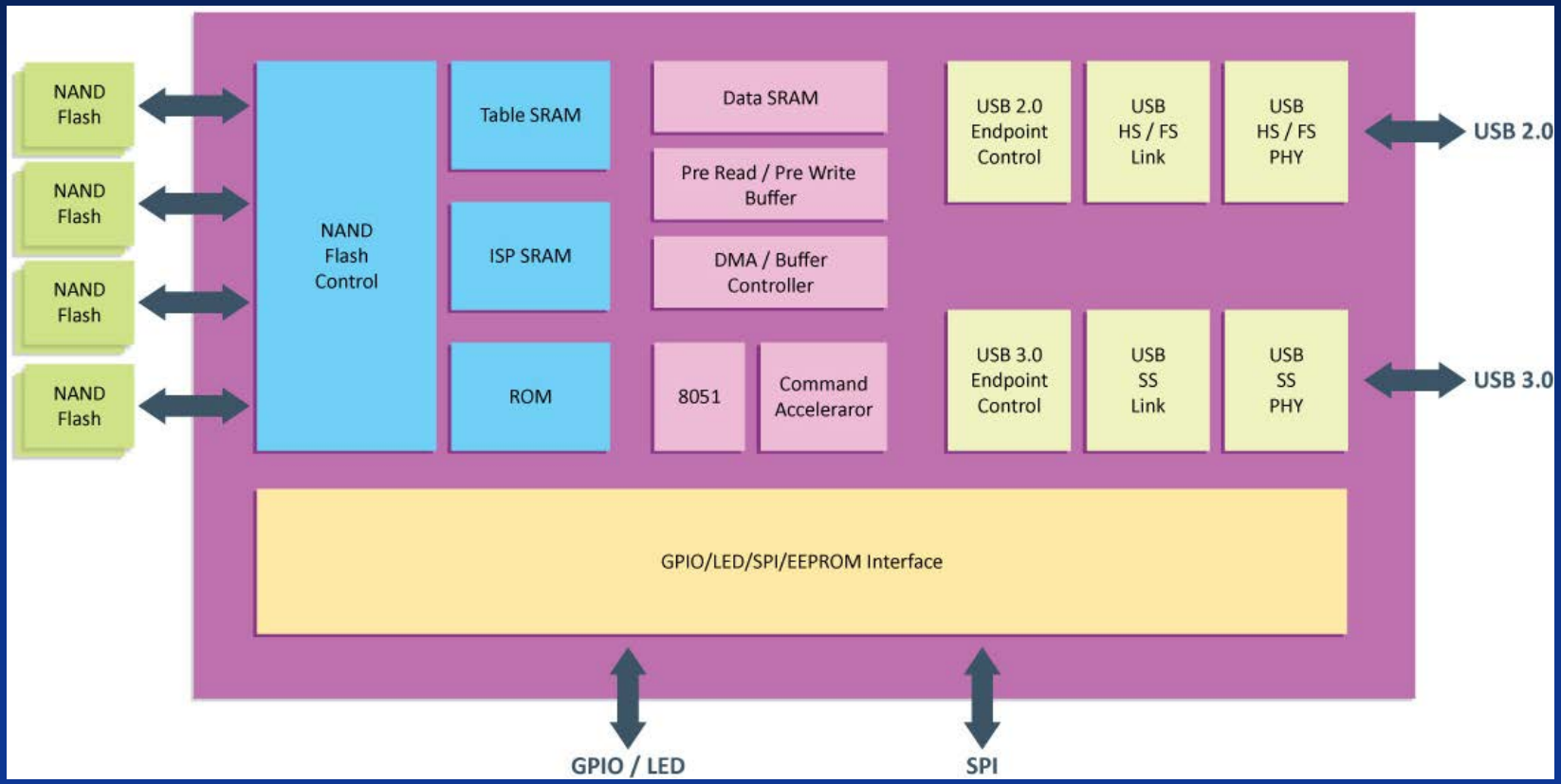


Image borrowed from http://electronicdesign.com/site-files/electronicdesign.com/files/archive/electronicdesign.com/content/content/63425/63425_fig1.jpg

What's needed for an "Enterprise Class" solution?

- High Speed Interface
  - SAS, PCIe
- Rock solid data protection
  - Robust ECC
- Performance MPU architecture

3 Key Blocks of a Flash Controller

- Host interface
  - SAS/SATA common for traditional storage interconnect
  - PCIe
  - Is Flash going to nudge us into a Storage Class Memory architecture?
- MPU
  - USB sticks used 8051s and other 8 bit mpus
  - Enterprise really needs something with some horsepower and the ability to execute multiple threads to manage the Write Cliff

3 Key Blocks of a Flash Controller

- Flash Interface Logic
  - Physical layer interface with flash memory chips
  - Needs a highly parallel architecture
    - Again, stay out of the way of host IOs when performing garbage collection
  - A nice place to integrate an ECC?

Data Flow Considerations

- In the USB key days, the MPU handled all the data moving
- High performance designs call for a hardware/firmware balance
- The key is to streamline the host data flow

Tradeoffs

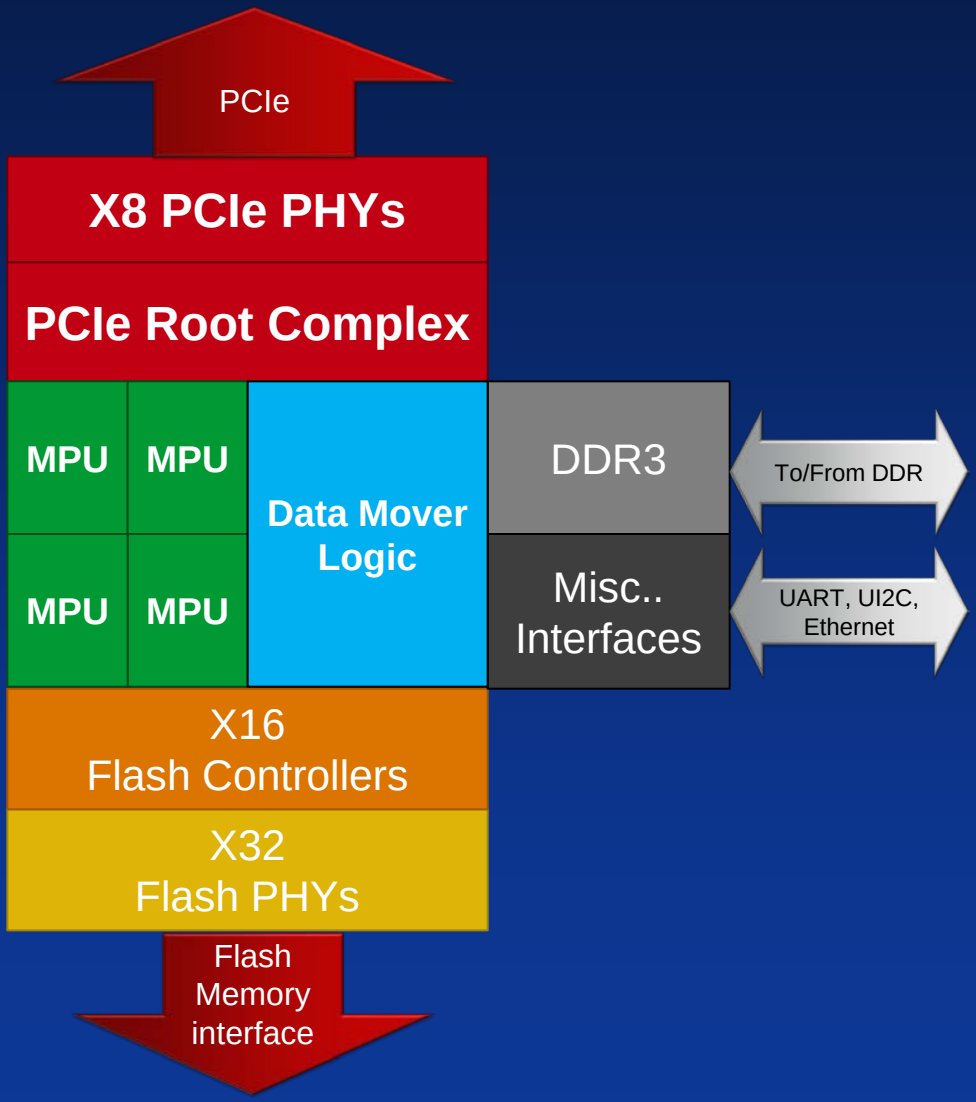
- Cost
  - CPU
  - SRAM size
- Host interface flexibility
  - SAS/SATA/PCIe
    - Integrate or use an interface chip?
  - NVMe implementation challenges
    - PCIe buffers and the DMA interface architecture aren't well defined by the spec
- Semiconductor manufacturing issues
  - Die size, Power and Package

Design of the Future

- Choose the right MPU
  - 32 bit embedded MPUs are commonplace
- Streamline data flow
  - Optimize for minimal firmware intervention
- Ensure background functions are outside of the host data path
  - Multiple MPU architecture to handle concurrent workloads

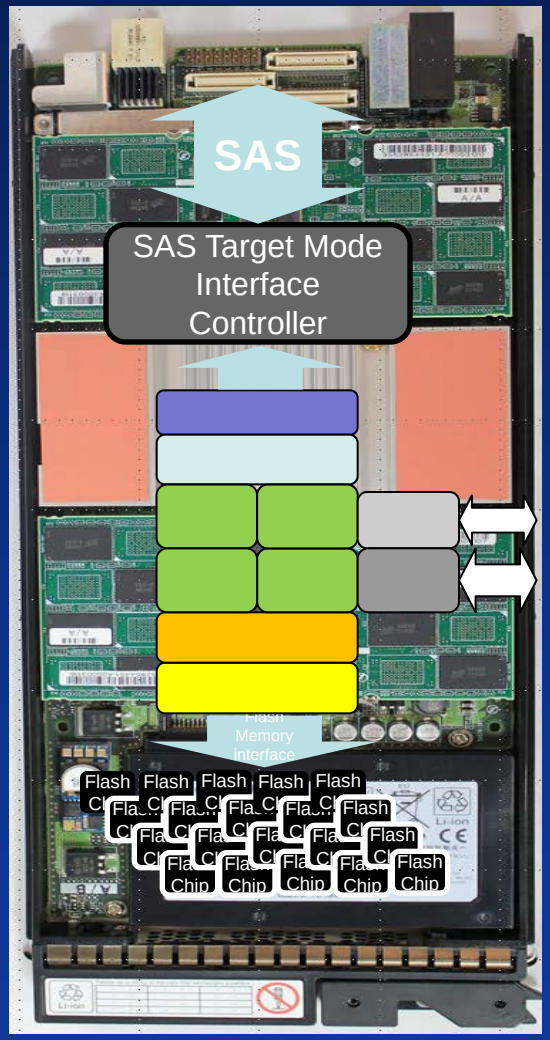
The Hitachi Accelerated Flash Controller

- 8 lanes of PCIe
- PCIe root complex
- 4X ARM9 MPU Core
- Integrated DDR-3 interface
- Patented Integrated flash controller logic supports 32 paths to the flash array
- 128 flash memory chips



The Hitachi Flash Module Drive

- Sas Host Interface
- HAFS Flash Controller ASIC
- MLC Flash Array



Questions? Discussion?

Thank You!

- http://www.hds.com/assets/pdf/hitachi-datasheet-unified-storage-vm-all-flash.pdf