Flash Memory Summit

System Design Challenges for Gen5 NVMe Solutions & Portable BMC Module

Presenter: Odie Killen, VP of Engineering at Viking Enterprise Solutions

Solution Level Design Challenges

- Designing at PCIe Gen5 data rates introduces significant challenges over Gen4 data rates
  - Data lanes run twice as fast, resulting in higher transmission line loss
  - Packaging and connector losses are higher
  - Signal integrity becomes more critical
  - Materials used for Gen4 become non-viable, resulting in higher product cost
  - Physical layer constraints drive a higher level of design discipline
- Without proper design rules, Gen5 designs will not reliably work.

Gen5 PCIe (Physical Layer) Challenges

- What Does This Mean
  - Current trend in the industry is higher performance, capacity and density
  - These constraints require tighter packaging and smaller solutions
  - All components should be located closer to CPUs
  - Counters the overall industry trend for higher density and performance
- Designs such as a 2U-24 (Typical Industry Solution) become more challenging to implement
- Higher drive and PCIe slot counts become harder to implement
- Overall solution design rules become more challenging

Impact on Performance

- Design constraints have a direct impact on performance if not overcome
  - SI errors result in command retries, adding latency and lowered throughput
  - SI errors can cause links to detune to Gen4 data rates (half the throughput)
  - Trace lengths can limit the drive count, reducing capacity and available back end performance
  - Trace lengths can limit the PCIe slot count, reducing available front end performance
  - Without very carefully developed design rules, Gen5 designs will have lower drive and PCIe slot counts, resulting in an overall lower system performance

Impact on Cost

- Gen5 End-to-End solutions that adhere to proper design rules will have a higher cost of acquisition than comparable Gen4 solutions
  - PCB material cost is higher to support solution
  - Overall power, packaging and cooling design will be more complex, potentially resulting in a higher product cost
  - Higher power consumption will result in higher wattage (more expensive) PSUs
    - Further made worse by EU Lot9 efficiency requirements for servers

Ways to Solve these Challenges

- Design with the absolute best case materials (increases cost)
  - Higher PCB etch cost, higher component cost, etc.
- Design with the absolute shortest traces possible (reduces density and performance)
  - Fewer drives and AICs, overall lower density/performance solution
- Implement carefully constructed design rules to find a balance between performance, density and cost

Cost Optimized Approach

- Implement a balanced solution combining Gen4 and Gen5 devices
  - Account for SW and application overhead when architecting your solution
  - Matched lane count between host ports and SSD devices
  - Leverage processing and throughput improvements of Gen5 host ports
  - Support Gen5 Host ports and design guidelines for maximum input BW
  - Support Gen4 Device ports to maximize their throughput

Do We Need a Gen5 End-to-End Solution

- Typical Gen4 SSDs operate at: 6.2 GB/sec READ, 2.6 GB/sec WRITE
- Available SSD throughput for 24 SSD system is: ~149 GB/sec READ, ~62 GB/sec WRITE
- Available Gen5 Input (4x 16-lane AICs): 252 GB/sec
  - Assumes no bottlenecks in AIC performance
- Assume 50% SW overhead (NICs and Application); expected throughput is: ~126 GB/s per single socket controller
- Assume a 50/50 WRITE/READ workload; SSD backend supports ~106 GB/sec
  - Gen4 SSDs provide enough available BW to saturate a Dual controller solution with 4 AICs per controller
- Accounting for 50% SW overhead, an additional 5 SSDs would result in a completely optimized solution (in this example)
- Workloads vary, but this example illustrates that for many workloads, a Gen4 device back end with a Gen5 host side is adequate to optimize system capabilities

Flash Memory Summit

Portable BMC Module for Unified Flash System Mgt

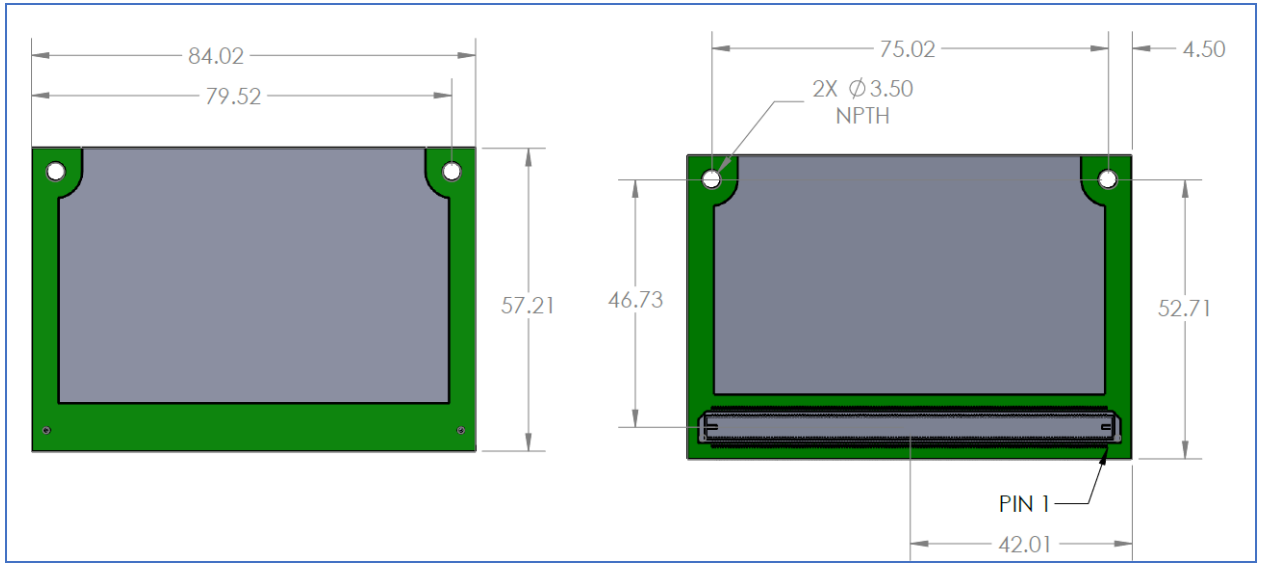
Presenter: Odie Killen, VP of Engineering, Viking Enterprise Solutions

What is a Portable BMC Module

- BMC chip integrated on a pluggable module
- Provides all BMC functionality for a CPU based solution
- Based on commodity connectors and form factors where possible
- Based on commodity BMC chipsets
- Pinout definition to support current and next generation BMC ASICs
- Designed for HW to be platform agnostic and common to many solutions
- Supports OpenBMC and custom BMC deployments

What is a Pluggable BMC Module (2)

- The mechanical outline of the pluggable BMC module is shown below
- Mates to base PCB via a SAMTEC ASP-231012-01 mezzanine connector



How Does This Differ From OCP Module

- 2019, OCP released a pluggable BMC module supporting Gen4 CPUs
  - Optimized for the AST2500 ASIC and features
  - HW pinout and definition based on needs for Gen4 solutions
  - Pinout does not support full functionality of AST2600 class ASICs
- Gen5 CPUs require AST2600 ASIC, so updates are required
  - AST2600 is more feature rich than previous AST2500
  - AST2500 is not adequate for new Gen5 CPU solutions
- Additional features drives a slightly larger form factor from previous release
- VES will submit this concept to the OCP community – target EOY ‘23

HW Design Impact of Pluggable BMC

- HW design must account for this module from the start
- Requires a connector and signal interface per specification
- Vertical height of module is higher than chip down solution
- Typical BMC signals must be routed to module connector
- Power, Packaging and Cooling of the module must be planned

SW Impact to BMC Code

- Base SW can become platform agnostic
- Increases code leverage
- Requires a platform specific abstraction layer to be developed
- Abstraction layer accounts for differences between platforms, allowing code to be deployed quickly across multiple solutions
- Abstraction layer must be mapped per platform
- Can deploy a platform agnostic OpenBMC stack

Why Implement a Pluggable BMC

- Reduction in overall design cycles
  - Once BMC module is designed, it can be re-used on multiple platforms
  - On-board circuit can be copied between various designs, reducing design time
- SW leverage between platforms is significant
  - Platform specific BMC SW development is very expensive
  - Significant portion of BMC code base becomes platform agnostic
  - Overall time to customize code per platform decreases
  - Facilitates greater leverage and re-use for host level management tools

Why Implement a Pluggable BMC (2)

- Decreases your overall SW spend while increasing your overall capabilities and efficiencies
  - Gives you more for less
- Common interface for multiple platforms and interfaces
- Common Redfish stack to be leveraged over multiple platforms
  - For example, SAS and NVMe can use the same Redfish stack
- Management tools become platform and interface agnostic
- Enables a single tool to manage your entire Enterprise

Thank You