IP-Based NVMe Development Platform

Mickael Guyard
Product Marketing Director (IP-Maker)
Agenda

- The need for NVMe IPs
- NVMe device platform
- NVMe host platform
- Use cases and applications
Part 1 – The need for NVMe IPs
NVMe the new universal interface

- The new universal interface for storage
  - First specification released in 2011
  - 13 board members
  - 90 companies with NVMe-based products (G2M research report)
- But not only…
NVMe applications

- Storage: PCIe SSD
- Cache: PCIe MRAM and NVRAM
- Processing accelerator
Heterogeneous architecture
FPGA in the data centers

Cloud Example: Data Center FPGA Acceleration
Up to 1/3 of Cloud Service Provider Nodes to Use FPGAs by 2020

Applications:
- Image Identification
- Security
- Big Data

Algorithms:
- Convolutional Neural Network
- Encryption
- Compression

Today

>2X performance increase through integration

Reduces total cost of ownership (TCO) by using standard server infrastructure
Increases flexibility by allowing for rapid implementation of customer IP and algorithms

Source: Intel Presentation
The need for NVMe IPs

- Massive usage of FPGA in data centers
- NVMe as a universal interface
- New architectures

=> NVMe IPs for FPGA are needed
  - Both device and host
# IP-Maker IPs

## IPM-NVMe-Device
- NVMe Controller Device
  - 1.3 specification
  - Multi Channel DMA
  - Automatic command processing
- **300ns Latency / 1.5M IOPS**

## IPM-NVMe-Host
- NVMe Controller Host
  - Automatic NVMe Command management
  - Automatic PCIe/NVMe init
- **3GB/s**
  - 1Q1RP
  - 128Q
  - Multi-RP

## IPM-UNFC
- NandFlash Controller
  - ONFI 4 Compliant
  - SLC/MLC/TLC
- **Error Correction Code**
  - Configurable BCH
    - Error number
    - Block size

## IPM-ECC
- **800MB/s Channel**
  - All IPs: AXI, Avalon or proprietary interface

---

*250MHz FPGA/ASIC clock, Gen3x8*
Part 2 – NVMe Device Platform
NVMe protocol

1- Host driver
2- NVMe command ready
3- NVMe command submission fetch
4- Data transfer
5- NVMe command executed

CPU
1-Host driver Setup
DDR
PCIe root complex
VID: Phy + Ctrl
NVMe IP
DMA
DDR
NVMe IPs

Key Features

- 1.3 NVM Express specification
- Automatic NVMe command
- Up to 65536 I/O queues
- Queue arbitration
- All mandatory commands / log management
- Legacy interrupt/MSI/MSI-X
- AXI/Avalon interface
- Up to 32 Read DMA channels + 32 write DMA channels
- Scalable data buswidth (64/128/256 bits)
- Available for PCIe Gen1/2/3

Full hardware

Hardware + Software
For more flexibility, such as vendor specific commands
HW/SW architecture

- Automatic command processing => Low latency
- Multi-channel DMA => IOPS acceleration
Validated platforms

Fidus Sidewinder – Zynq Ultrascale+

VC709 – Virtex7

Nallatech 250S+ - Kintex Ultrascale+

KCU105 – Kintex Ultrascale
Reference design
Performances

- Setup
  - Hardware: reference design
  - Standard NVMe driver
  - Use of standard benchmark tool for storage: FIO
    - Latency
    - IOPS
Latency

- QD=1, IO=4kB

12.8μs
- Measured with FIO
- FPGA clock 125MHz
- Gen3x4, OS IRQ
- 600ns from the NVMe IP

7.8μs
- Measured with FIO
- FPGA clock 125MHz
- Gen3x4, OS polling mode
- 600ns from the NVMe IP

5.3μs
- Estimated
- ASIC clock 1GHz
- Gen3x8, OS polling
- 75ns from the NVMe IP
IOPS

- Gen3x4, QD8, 4kB IO, random R/W
  - 700k IOPS
- High IOPS at low queue depth
- Scalable data path: up to Gen3x16, Gen4 x8
Part 3 – NVMe Host Platform
NVMe Host IP overview

- Memory or FIFO
- AXI / Avalon
- Data to Transfer
- NVMe Command Manager
- User Interface
- Data transfer engine
- Automatic init engine (state machine)
- Control
- PCIe Root Port
- AXI / Avalon
- NVMe SSD

IP-Maker IP
3rd party IP
NVMe commands setup by the host

Data to Transfer

Data transfer request

User Interface

Automatic init engine (state machine)

Data transfer engine

Memory or FIFO

NVMe Command Manager

Data transfer

Init with a state machine

PCIe rootport/endpoint settings

NVMe device/host configuration

NCMe SSD

PCIe Root Port
Different configurations

Single port, up to 128 queues

API

API

FPGA

Host NVM Express

PCIe RP

128 queues

NVMe SSD

N* root ports, 1 queue

API

PCIe RP

FPGA

Host NVM Express

PCIe RP

NVMe SSD

*depending on FPGA interfaces
Multiroot

Key Features
- NVM Express Compliant
- Automatic NVMe Command management
- Automatic PCIe/NVMe init
- Multi rootport support
- Single I/O queue
- Single Namespace
- Vendor specific commands
- Up to PCIe Gen 3x8
Key Features

- NVM Express Compliant
- Automatic NVMe Command management
- Automatic PCIe/NVMe init
- 128 I/O queues
- Vendor specific commands
- Single Namespace
- Up to PCIe Gen 3x8
Open Channel support

- Fully submission command control
  All vendor specific commands are possibly.
  The complete control is possible.
- Full completion control
Validated platforms

Fidus Sidewinder – Zynq Ultrascale+

Nallatech 250S+ - Kintex Ultrascale+

KCU105 – Kintex Ultrascale
Reference design

- Host IP configuration
  - 1 root port
  - 1 queue version
- Embedded test bench

Ultrascale Xilinx FPGA
Samsung NVMe 960EVO
Gen3x 4 PCIe interface
## Performance

- **Write**: 2.2 GB/s
- **Read**: 3.2GB/s

### Technical Specifications

<table>
<thead>
<tr>
<th>Samsung SSD 960 EVO</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Usage Application</strong></td>
</tr>
<tr>
<td><strong>Interface</strong></td>
</tr>
<tr>
<td><strong>Capacity</strong></td>
</tr>
<tr>
<td><strong>Controller</strong></td>
</tr>
<tr>
<td><strong>NAND Flash Memory</strong></td>
</tr>
<tr>
<td><strong>DRAM Cache Memory</strong></td>
</tr>
<tr>
<td><strong>Dimension</strong></td>
</tr>
<tr>
<td><strong>Form Factor</strong></td>
</tr>
<tr>
<td><strong>Performance</strong> (Up to)</td>
</tr>
<tr>
<td><strong>Sequential Read</strong></td>
</tr>
<tr>
<td><strong>Sequential Write</strong></td>
</tr>
<tr>
<td><strong>QD 1, Ran. Read</strong></td>
</tr>
<tr>
<td><strong>Thread 1, Ran. Write</strong></td>
</tr>
</tbody>
</table>
Part 4 – Applications
Use cases and applications

- PCIe Flash
- NVRAM
- Emerging NVM
- Smart SSD
- HBA NVME2NVMe
NVRAM Reference design

- NVMe device ref design close to a end-product
  - Detected as a NVMe device by the driver
- Just need to add « non-volatile » feature
PCIe NVRAM

- Using NVDIMM-N like technology
- Or using directly a NVDIMM-N
PCIe NVRAM

- Specification
  - Up to 32GB
  - 1.5MIOPS on Gen3x8
  - 10us latency
Flash Controller IP

Key Features

- ONFI 3/4 Compliant
- SLC / MLC / TLC
- 2 Channels
- Supported modes
  - Async, DDR, DDR2, DDR3
  - AXI/Avalon interface
- Configurable ECC
  - BCH
  - LDPC

16/32/64bit user interface

8/16bit ONFI interface
Evolution

• MRAM support
NVMe to NVMe HBA

- For NVMe SSD aggregation:
  - Better performance and reliability
NVMe to NVMe HBA

• 2.5 x86 cores full time at 3GHz required to sustain 750kIOPS on each SSD
• 10 cores total!

\[ \text{HBA: 10 x86 cores?} \]

• \( \Rightarrow \) Need of hardware accelerator engines
NVMe to NVMe HBA

- Let’s use both NVMe device and host IPs
NVMe to NVMe

Gen3x 16 or Gen4x4

Namespace management software

DDR3 Controller

4 x Gen3x4

PCIe Interface

NVMe Device

NVMe Host

NVMe Host

NVMe Host

NVMe Host

PCIe Interface
Namespace management

- Many configurations
  - Basic capacity aggregation: one namespace across the 4 SSDs
  - Asymmetric: one namespace on one SSD and 10 namespaces on the 3 other SSDs.
  - Multi namespaces with different characteristic (encryption, compression…) seen only as one storage SSD.
  - Raid 1 storage totally transparent for the host software.
Evolution

- Path to computational storage
  - Advanced computing accelerators can be added such as key-value store, search engine and deep learning
Thanks

Contact
mickael.guyard@ip-maker.com

Visit IP-Maker booth #710
NVMe live demo!