Proteus – a Custom Platform for Hybrid Memory Research

Kenneth Wright, Sr Director
Rambus Labs
Outline

• The Research: The Future of Memory Subsystems
• Proteus: Hardware Platform for Hybrid Memory Research
• Collaboration
Cost gap between DRAM and NAND continues to increase. Need cost-effective emerging memory to fill this gap.
The Research

- **Goal**: Investigate memory subsystem architectures attachment strategies.
- **Specifically**:
  - Multiple memory types
  - Multiple memory attachment types
  - HW and SW Management Schemes
- **Collaboration**: Create an ecosystem with academic and industry partners to develop solutions and explore the path forward for hybrid memory subsystems

Emerging memories (RRAM, MRAM, PCM and fast flash) have promise to fill the gap in the memory hierarchy.

Managing them effectively could relieve concerns with latency, bandwidth and endurance.
Hybrid Memory Research Tracks

**Performance Modeling**
- Very Promising Results to date
- Challenges: slow, limited scenarios, simulation assumptions

**Hardware Prototyping**
- Custom memory board development
- Run real world applications
  - POWER9 CPU
  - OpenCAPI interface

**Management SW and HW**
- Real time HW Policies
- SW Data placement
- SW Data movement
Proteus Hardware Platform
Proteus Hardware Platform

- Designed to accommodate present and future needs for the group developments
- It uses a Xilinx XCVU9P, with bigger devices possible
- Three types of high bandwidth links:
  - OpenCAPI: 3 SlimSAS x8 interfaces
  - Gen-Z: 2 Gen-Z x8 interfaces.
  - PCIe x8
- 4 DDR4-288 pin connectors
  - Interface compatible with NVDIMM-N, NVDIMM-P
  - Other DIMM types with emerging technologies also possible
- On board memory:
  - DDR3 memory for internal operation assistance
- Flexible clock architecture
Modular and Flexible

Processor

Applications and Application Interfaces

SW Hybrid / Heterogeneous Tier Manager

Interface Controller

OpenCAPI, Gen-Z, CCIX

Hierarchical Controller Interface

OpenCAPI, Gen-Z, CCIX

Modular and Programmable Interface

Modular and Programmable Interface

Fast Line Controller

EM / Block Controller

Hybrid Controller

HW Hybrid Tier Manager

Config and Control from SW

Processor Interface Controller

SW Interface

Master Interface

Line Interface

Block Interface

Fast Line

Emerging Memory

DDR4

ONFI, Toggle

Rambus Labs Area of Research

Memory

Processor

Modular/Programmable

Interface
Proteus Internal FPGA Architecture v0.7

- **OpenCAPI/Gen-Z/CCIX**: Provides high bandwidth interface with adaptation logic.
- **PHY DL TL**: Core components for signal transmission.
- **Proprietary logic Interfaces**: Enables custom configuration.
- **Configuration registers**: Store configuration data.
- **Adaptation logic**: Processes incoming data for efficient transmission.
- **AXI Xbar**: AXI crossbar for data exchange.
- **Hybrid Ctrl (HC)**: Controls hybrid logic for efficient data flow.
- **AXI Xbars**: AXI crossbars for data routing.
- **Policy**: Controls data flow based on predefined rules.
- **Custom configuration registers**: Tailored for specific application requirements.
- **EMM/EMMC**: Interfaces for EMMC and policy management.
- **Line controller**: Manages line-level communication.
- **Block device controller**: Controls block-level operations.
- **3rd party IP**: Integrates third-party intellectual property for enhanced functionality.
- **Rambus IP**: Provides specialized data handling capabilities.
Hybrid Management

Test/benchmark applications
“Standard” Linux OS + drivers
Memory management & mapping

Power Server platform
Power 9 scale out CPU

AFU_C0-M2
DDR4 controller IP
DDR4 PHY
(Xilinx)
PCI-e
12V, 3.3V, 3.3Vaux

Open CAPI

Physical Interfaces

SlimSAS connector

Host Data path

Cached data path

FPGA

RAMS NAND DIMM
SR/DR DRAM DDR4
DDR4
NAND over DDR connector

Host
Data path

Cached data path

NAND over DDR connector

FPGA

SR/DR DRAM DDR4

RAMS NAND DIMM

Data path

Cached data path

NAND over DDR connector

FPGA
Hybrid Memory Simulation Results – Enhanced Flash

Experiment Setup
- Large access granules
- Erase blocks
- High latency

Experiment Results
- Modeling shows cost/performance advantage with management techniques
- EM device critical attributes identified; see next slide for details

<table>
<thead>
<tr>
<th>Workload</th>
<th>DRAM</th>
<th>No Mgmt.</th>
<th>Rambus Techniques</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Caching</td>
<td>1</td>
<td>48.95</td>
<td>1.40</td>
</tr>
<tr>
<td>Data Serving</td>
<td>1</td>
<td>63.28</td>
<td>1.45</td>
</tr>
<tr>
<td>Graph Analytics</td>
<td>1</td>
<td>101.69</td>
<td>1.51</td>
</tr>
<tr>
<td>In-memory Analytics</td>
<td>1</td>
<td>79.06</td>
<td>1.73</td>
</tr>
<tr>
<td>Media Streaming</td>
<td>1</td>
<td>3.67</td>
<td>1.13</td>
</tr>
<tr>
<td>Web Search</td>
<td>1</td>
<td>17.25</td>
<td>1.30</td>
</tr>
</tbody>
</table>

EM latency
3us read/100us write

Relative execution time
Note: Lower values = better performance

Reference

Poor bandwidth, latency and endurance
Rambus management policies enable improved performance
NVDIMM-P Demonstration Configuration

Test/benchmark applications
“Standard” Linux OS + drivers
Memory management & mapping

Power Server platform
Power 9 scale out CPU
AFU_C0-M2
DDR4 controller IP
DDR4 PHY
(Xilinx)
PCI-e
12V, 3.3V, 3.3Vaux
Open CAPI
Physical Interfaces
SlimSAS connector
FPGA

“PCI-e” config space
8x PHY
DLX
TLX
VC1 (cmd)
VC1 (cfg)
ADD Mapping
DCP1
DCP0
AFU_C0-M2
ADD/CMD decoder

Data FIFO's
DDR4 controller/DDR4-P controller
DDR4 PHY
DDR4 Electrical interfaces

NVDIMM-P – DRAM behind
NVDIMM-P – DRAM behind
SR/DR DRAM DDR4
SR/DR DRAM DDR4

NVIDIA

OpenPOWER®

Data • Faster • Safer
## Research Opportunities for Rambus and Partners

<table>
<thead>
<tr>
<th>Rambus</th>
<th>Processor Leaders</th>
<th>Memory Leaders</th>
<th>System Leaders</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Hardware research platform access</td>
<td>• Programming models</td>
<td>• Analysis of EM types</td>
<td>• Functional testing of Protocols (OpenCAPI)</td>
</tr>
<tr>
<td>• Benchmarking</td>
<td>• Resource sharing / partitioning / provisioning</td>
<td>• Demo EM</td>
<td>• Designs that can be modified to be a product</td>
</tr>
<tr>
<td>• Management policies/algorithms</td>
<td>• Interface comparisons</td>
<td>• Functional testing of NVDIMM-N/P</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Real world application testing</td>
<td></td>
</tr>
</tbody>
</table>

Rambus Labs and IBM are looking for collaborators interested in using the Hardware Platform to prove the benefit of DRAM and Emerging Memory.
Thank You to Our Collaborators

• IBM
• Wistron
• Toshiba
• Sony

• Xilinx
• EverSpin
• Anacode
• Inteliprop
Backup
Terms and Nomenclature

- **Memory subsystem**
  - The subsystem that can be accessed from the processor by use of loads and stores.
  - It does not require a driver in the non-exception case.
  - It may attach to the processor in a variety of ways not just DDR

- **Hybrid Memory Subsystems**
  - A memory subsystem that contains more than one storage media that has different attributes such as latency, bandwidth, energy, endurance etc.
  - The subsystem is managed by hardware in such a way that user code does not directly deal with the differences in the storage media.

- **Heterogeneous Memory Subsystem**
  - A memory subsystem that contains more than one storage media that has different attributes such as latency, bandwidth, energy, endurance etc.
  - The subsystem is managed by software in such a way that the user/OS code can make decision about placement and migration of data.

- **Emerging Memory**
  - Any storage media that has been proposed for used in the Memory subsystem but is not shipping in volume yet

- **Line-type EM -- an emerging memory with the following attitudes**
  - Low (< 3us) access latency
  - Small (64B – 128B) access granularity
  - Hardware media management enabled by, Intrinsically low BER, deterministic latency, and excellent endurance
  - Candidate Technologies: STT-MRAM, NRAM, PCM/3D Xpoint, RRAM

- **Block-type (asynchronous) EM -- an emerging memory with the following attitudes**
  - High (> 3us) read latency
  - Large (512B – 4KB) access granularity (transport latency $O$(access latency))
  - Firmware or dedicated controller management due to, non-deterministic latency, block level ECC, wear leveling or other logical/physical translations
  - Candidate Technologies: High Speed Flash, RRAM, PCM/3D Xpoint