PSS: A prototype storage subsystem based on PCM

IBM Zurich Research Lab: Ioannis Koltsidas, Peter Mueller, Roman Pletka, Thomas Weigold, Evangelos Eleftheriou

University of Patras: Maria Varsamou, Athina Ntalla, Elina Bougioukou, Aspasia Palli, Theodoros Antonakopoulos

***Abstract* – Phase-Change Memory (PCM) is emerging as one of the most promising Non-Volatile Memory technologies with cost and performance characteristics in between those of DRAM and Flash. In this presentation we describe the design and implementation of PSS, a PCM-based Storage Subsystem which is connected to the host over the PCI-e bus. We will present the hardware architecture of the card, as well as certain aspects of the firmware and the host driver. A detailed experimental evaluation of the card demonstrates the feasibility of PCM-based storage devices exhibiting performance, endurance and reliability characteristics suitable for enterprise storage.**

1. Introduction

Emerging workloads necessitate systems with highly scalable storage, not only in terms of capacity, but also in terms of performance and energy efficiency. Memory technologies like DRAM experience difficulties scaling beyond 25nm and are unfit for large-scale storage systems because of power consumption, cost per GB and volatility. Regarding traditional HDDs, although their capacity is increasing, their normalized I/O throughput (IOPS/GB) is decreasing continuously. This widening performance gap is mostly being filled in by Flash memory in different form factors. Looking towards the future, however, Flash comes with its own scalability problems: aggressively scaling Multi-Level Cell Flash results not only in low performance but also in poor endurance, which introduces additional complexity in controllers as well as performance variability. On the other hand, Phase-Change Memory is one of the most mature persistent memory technologies, promising excellent and predictable performance in terms of throughput and latency, and high endurance and scalability.

Phase change memories are built on a silicon base consisting of the driver logic (access device) to control the memory cells and an array addressing scheme, similar to other memory technologies. The active material in PCM is a chalcogenidic alloy, which is placed between two electrically conducting electrodes in each cell of the memory array. Phase-change materials exhibit two meta-stable states. The drivers create a high or medium current to program a cell as '0' (amorphous phase) or '1' (crystalline phase), respectively. A low current is then applied to read out the cells. The phase change memories used in our prototype storage system have typical programming times of 70 ns and 120 ns for '0's and '1's, respectively. The cells can be reprogrammed at least 10^6 times and with forward error correction (FEC) applied, this number increases to 10^9 cycles while guaranteeing an uncorrectable bit error rate better than 10^{-16}.

Although PCM can serve as an extended main memory to DRAM, in this presentation we will focus on a PCM-based system that is used as persistent storage, the main reason being that a PCM-based extended main memory would require significant hardware changes to the memory controller and software changes in the operating system. On the other hand, since PCM is already attractive as a replacement for Flash and/or HDDs, a PCM-based storage device would allow us to immediately reap the benefits of PCM by attaching it to the I/O bus. One of the main concerns regarding the use of PCM for high-performance applications is its relatively high write latency. The purpose of this work has been to architect a PCM-based device and implement a fully functional, high-performance PCI-e card that takes advantage of the characteristics of PCM and mitigates its limitations. The design we present targets storage workloads that are dominated by small (e.g., 4 kB) random I/O operations and aims to achieve low, predictable latency with reads and writes. Most importantly, we pursued an architecture that is simple and lightweight compared with the complexity of Flash controllers but will nevertheless allow us to scale in terms of capacity and performance with future PCM chips. Compared to Onyx [1], a PSS configuration with 4 cards and total user capacity of 8GB achieves 57% better random read IOPS at 4KB with 8% less read latency and similar number of random write IOPS at 4KB with 66% less write latency. These improvements are mainly attributed to architectural differences of our PCM controller.

In this presentation we demonstrate how we achieved these goals within the scope of our research prototype. We first give an overview of the current state of affairs of PCM and of technology trends for the future, and motivate the use of PCM in storage systems. We then present the hardware and software architecture of PSS including the key design decisions we made to meet our goals. Finally, we describe the most interesting aspects of our implementation and provide results of an experimental evaluation of the memory itself, as well as of the PCI-e card at a system level. Our results include measurements of throughput and latency under different workloads, including detailed latency profiling and breakdowns. We will also present results regarding the endurance of the PCM chips as well as the dependence of performance on data patterns.

2. Architecture and Design

The PSS card uses commercially available PCM memories of 90nm technology. These chips expose a synchronous interface with a small number of data pins and a number of control signals. Internally they use a data buffer of 64 bytes during write, whereas read is performed with very small latency, comparable with the transfer time of a single byte. Data access, for either read or write, is performed in blocks that range from a single byte up to 64 bytes. The maximum clock frequency is 66 MHz and the data program time for 64 bytes has a typical value of 120 μsecs, resulting in a data

rate of 0.52 MBps for write, whereas the read data rate is 16 MBps. To store a 512 B sector, we use multiple PCM chips that share data and control pins. For ECC and metadata, we use a spare PCM chip per sector, resulting in 89% storage efficiency and improved BER performance, because of the two-level error-coding scheme used.

The PSS PCI-e card uses eight PCM channels for storing data, and each PCM channel uses two banks of PCM chips to support pipelining. The PCM channel has been optimized for fast sector access. Each PCM channel uses one controller for each PCM bank, along with a zero-latency bus arbiter. An internal buffer per controller is used for storing and retrieving multiple sectors without the intervention of the card's processors, so that multiple commands can be applied in parallel. During read, the system supports two modes of operation. In the "Pass-through" mode, the PCM-stored data are provided to the host along with the retrieved metadata, but without correcting any medium-related random errors, and only a CRC-like check is performed. In the "Validate-before-Transmit" mode, the retrieved data are stored temporarily in a local buffer and error detection and correction are performed. In this case, only the user data are provided to the host. During write, the system also supports two modes of operation. When the "Early Completion" mode is used, the write command is acknowledged once the user data have been transferred to the internal buffer of the PCM chips, whereas in the "Late Completion" mode the write is acknowledged once the user data have been stored in the PCM cells. All modes of operation are user selectable.

The PSS PCI-e card is based on the Zynq-7045 FPGA chip and uses 4 PCI-e Gen.2 lanes for connection to the host and two DMA engines for fast access of the host memory. Moreover, it implements page write caching and read prefetching on two ARM Cortex-A9 cores. Write caching is used to absorb the latency introduced during bursts of write commands. The same DRAM-based cache is used for fast responses of previously stored pages, or for read prefetching when possible.

The operating system accesses the PCI-e card using a block device driver. The device driver uses a shared area in the host memory to exchange commands and information with the card's processors. This shared memory is organized as a variable-length set of descriptors, which are served in a First-Come-First-Served order. To minimize the overhead introduced at this interface, the number of commands exchanged per processing cycle is optimized by taking into account the number of pending user commands, the status at the PCM channels (pending read or write commands) and the number of pending commands at the internal write cache. One DMA engine is used for updating and retrieving information from the shared memory, whereas the other DMA engine is used for page transfers between the host memory and the PCI-e card's internal memories, either the internal DRAM or the memory associated with each PCM channel. The mapping of logical sectors/pages to PCM resources (chips and channels) is performed in a way that maximizes sequential write performance and allows to perform start-gap like wear-leveling.

3. Experimental Evaluation

We have conducted a detailed experimental study of the PCM chips as well as of the PSS PCI-e card. For the PCM chips, we present measurements about latency and endurance, and provide experimental evidence that suggests that the write latency of the chips depends on the data patterns. At the system level, we have measured the throughput and latency of the card under various workloads, including random and sequential, read-only, write-only, and mixed read and random workloads. We have captured a detailed latency profile of the device during steady-state operation and compare it with the respective latency profiles of MLC- and TLC-Flash-based devices. Our results indicate the following:

a) The endurance of the PCM chips allows several million writes cycles before the first error. With FEC, this number extends to tens of millions.

b) The card exhibits excellent read and write latency and throughput, given the interface and performance characteristics of the PCM chips we used, validating that the proposed architecture can realize the potential of PCM in terms of performance. We achieved a steady-state average latency of 35 μsec for random 4kB reads and 61 μsec for random 4kB writes, achieving a maximum sustained throughput of about 75 kIOPS and 15 kIOPS, respectively.

c) The PSS PCI-e card has predictable and stable latency: for several hours of sustained random writes, 99.9% of the requests completed within 240 μsec, and the highest latency observed was 2 msec. Conversely, for an MLC-based enterprise-class Flash PCI-e card we put to the same test, the latency for the 99.9th percentile was 3 msec (i.e., 12x higher) and the highest observed latency was 14 msec (i.e., 7x higher). Moreover, a TLC-based Flash SSD we tested showed a 99.9th percentile latency of 66 msec (i.e., 275x higher) and a highest observed latency of 122 msec (i.e., 61x higher).

4. Conclusions

The presentation focuses on a PCI-e attached storage subsystem based on Phase Change Memory. We present the hardware and software architecture of the card, and describe the key design decisions we made that allowed us to achieve our goals. Our experimental evaluation of the memory and of the card shows how PCM can achieve predictable low latency at a fraction of the complexity associated with Flash, and demonstrates the suitability and applicability of PCM to enterprise storage when adopting a device architecture such as the one presented.

References

[1] A. Akel, A. M . Caulfield, T. Mollov, R. K. Gupta, S. Swanson, "Onyx: A Prototype Phase Change Memory Storage Array", in Proc. of the 3rd USENIX conference on Hot topics in Storage and File Systems, 2011