Memory Sandbox 2.0: A Framework for Enabling HBM2e vs HBM2 Performance and Telemetry Analysis on Xilinx FPGAs

Elias Perdomo
Barcelona Supercomputing Center
Universitat Politècnica de Catalunya
Barcelona, Catalunya, Spain
elias.perdomo@bsc.es

Joan Teruel
Barcelona Supercomputing Center
Universitat Politècnica de Catalunya
Barcelona, Catalunya, Spain
joan.teruel@bsc.es

Sajjad Ahmed
Barcelona Supercomputing Center
Barcelona, Catalunya, Spain
sajjad.ahmed@bsc.es

Behzad Salami
Barcelona Supercomputing Center
Barcelona, Catalunya, Spain
behzad.salami@bsc.es

Teresa Cervero
Barcelona Supercomputing Center
Barcelona, Catalunya, Spain
teresa.cervero@bsc.es

Xavier Martorell
Barcelona Supercomputing Center
Universitat Politècnica de Catalunya
Barcelona, Catalunya, Spain
xavier.martorell@bsc.es

Abstract

FPGA-based emulation provides a reconfigurable and efficient environment for evaluating SoC memory systems prior to full silicon deployment. Increasingly used to accelerate HPC and AI workloads, FPGAs enable hardware-level exploration of performance-critical subsystems such as memory, making them ideal platforms for prototyping accelerators in today's heterogeneous computing landscape.

Due to modern HPC and AI workloads' increasing bandwidth and energy requirements, understanding and optimizing off-chip memory behavior, especially with High Bandwidth Memory (HBM), becomes critical. In this position paper, we present the ongoing development of ***Memory Sandbox 2.0***, a modular, open-source framework for analyzing memory behavior across both **performance** (throughput, latency) and **telemetry** (temperature, power).

Memory Sandbox 2.0 enhances the original tool with increased configurability, modular extensions for emerging technologies such as **HBM2e**, flexible support for varied access patterns, and integrated telemetry monitoring to guide energy-aware system design. We aim to support design space exploration of memory-access behavior and its architectural implications across platforms. While full experimental validation is still underway, we share early insights into challenges faced while profiling HBM2 and HBM2e on cutting-edge AMD Xilinx FPGAs such as the **Versal V80** and Alveo U280. We also describe key architectural changes in recent HBM systems and how our tool models and exposes their behavior.

Memory Sandbox 2.0 helps identify architectural bottlenecks and trade-offs early in the design cycle by enabling reproducible, insightful evaluation of memory-access patterns and telemetry data. This contributes to more effective and energy-efficient FPGA-based accelerator development, paving the way for more performant and energy-efficient HPC and AI solutions.

Keywords

HBM, memory, performance, throughput, telemetry, FPGA, NoCs, emulation, tool, HPC, AI, DSE

1 Introduction

The growing computational demands of High-Performance Computing (HPC) and Artificial Intelligence (AI) workloads have exposed a critical system bottleneck: off-chip memory access. In modern accelerators, systems spend up to 80% of their execution time stalled on memory accesses, making them heavily memory-bound [11]. Moreover, memory transactions can account for as much as 60% of total energy consumption, positioning memory as both a performance bottleneck and a dominant factor in overall system efficiency [28].

Traditionally, boosting performance in critical memory-bound systems involved adding more computational capabilities and bringing more memory on-chip, especially in scenarios processing large volumes of data locally or regionally. However, such an approach no longer scales since we are reaching the boundaries of the Von Neumann, Moore's Law, and Dennard scaling trifecta. High Bandwidth Memory (HBM) has emerged as a key enabler in addressing memory performance and energy challenges in modern computing. With its wide interfaces, 3D stacking, and high internal parallelism, HBM significantly outperforms traditional DDR memory in both bandwidth and energy efficiency [22]. The HBM standard evolution, shown in Table 1, introduce further improvements in peak throughput, memory capacity, and pseudo-channel concurrency. As a result, HBM has seen widespread adoption in ASIC-based systems, many of which are now transitioning to HBM3, with HBM3e and HBM4 already on the horizon [1, 3].

In contrast, the FPGA ecosystem has only recently begun integrating HBM2e. While HBM2 has been available in AMD Alveo FPGAs targeting HPC systems such as the AMD Xilinx Alveo U280 [4] and U55C [5], HBM2e support has just become accessible in the Versal V80 advertised as their FPGA targeting AI deigns [16]. Nevertheless, real-world utilization of HBM in FPGA platforms remains far from optimal. This gap is largely due to the limited visibility and tooling available to analyze internal memory behavior, as well as platform-specific architectural constraints, including microswitch topologies, pseudo-channel interconnect structures, and the intricacies of NoC routing.

FPGAs offer an unparalleled platform for exploring accelerator-memory interactions, combining hardware-level observability with

Table 1: HBM standards evolution

HBM Specification	HBM	HBM2	HBM2e	HBM3	HMB3e	HMB4*
JEDEC Standard	Oct 2013	Jan 2016	Aug 2018	Jan 2022	May 2023	2026*
Die Density	2Gb	8Gb	16Gb	16Gb	24Gb	24Gb/32 Gb
Max dies per stack	4Hi	4Hi/8Hi	4Hi/8Hi/12Hi	8Hi/12Hi/16Hi	8Hi/12Hi/16Hi	4Hi/8Hi/12Hi/16Hi
Channels to SoC per stack	8x 128-bit channels			16x 64-bits channels		32x 64-bits channels
Total HBM width	1024 bits (for 8-Hi stack)					2048 bits (for 8-Hi stack)
Interface to SoC	Interposer or direct stack					
Max Pin Transfer Rate	1 Gb/s	2.4 Gb/s	3.6 Gb/s	6.4 Gb/s	9.8 Gb/s	8 Gb/s
Max Capacity per stack	1GB	4GB/8GB	8GB/16GB/24GB	16GB/24GB	36GB	64GB
Max Bandwidth per stack	128 GB/s	460.8 GB/s	820 GB/s	1.2 GB/s	1.6 TB/s	2 TB/s

full reconfigurability. Their flexibility allows designers to prototype and iterate low-level memory access behaviors that would be impractical in ASICs or GPUs [8, 9]. As such, FPGAs are uniquely positioned to support early-stage performance modeling, hardware/software co-design, and comprehensive design space exploration. However, existing evaluation frameworks often narrow their scope to bandwidth or latency metrics, overlooking the thermal and power dynamics that are increasingly critical in modern design decisions. To address this gap, we present *Memory Sandbox 2.0*, a tool that contributes to:

- Studying HBM architectural features, standards, and their addition to AMD HPC and AI-oriented boards.
- Providing an enhanced open-source framework aimed at enabling reproducible, telemetry-aware exploration of memory system behavior across HBM2e and HBM2-based FPGA platforms.
- Providing an infrastructure to better understand, profile, and eventually optimize memory-bound accelerator designs, in both HPC and AI domains, by combining modular performance benchmarking with fine-grained telemetry access.

2 Memory Sandbox 2.0

We originally developed our *Memory Sandbox 1.0* to enable users to understand memory behavior more effectively, leading to better hardware resource utilization. By allowing early evaluation of accelerator-specific memory access patterns before full hardware availability, our *Memory Sandbox 1.0* reduced design time and supported early performance modeling.

As a practical application of our Memory Sandbox 1.0, we profiled HBM2 performance on Xilinx platforms and empirically confirmed that throughput and latency are significantly influenced by the microswitch architecture, as also acknowledged in AMD documentation [27]. In Figure 1, we illustrate the internal routing

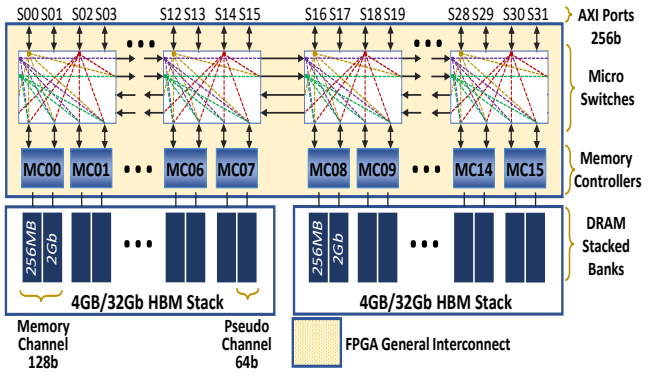

Figure 1: Alveo U280 HBM topology.

of microswitches. Full connectivity is preserved within a given microswitch instance during vertical access, resulting in minimal performance loss. However, when transactions cross into a different AXI Switch instance (i.e., lateral routing), a dead cycle is inserted, degrading throughput, especially for small write bursts.

Our *Memory Sandbox 2.0* framework is redesigned for greater flexibility and extensibility. As shown in Figure 2, the system is now built around multiple Configurable Pattern Generators (CPGs), each supporting sequential, pseudo-random, and trace-driven access modes. These generators are instantiated independently and configured via AXI-Lite, allowing users to simulate realistic or stress-test scenarios across the full range of HBM pseudo-channels.

In *Memory Sandbox 2.0*, we refined the control interface by integrating QDMA, though the CPGs remain compatible with any AXI-Lite master. To streamline experimentation, we consolidated design-time generics into just three design parameters:

- **Design Select:** Repetitve Sequential Transversal (RST), Inverted RST, Pseudo-random and Sparse Matrix Vector.
- **Memory Type:** DDR U280, HBM U280/U55C and V80.
- **Address Mapping:** 4 for DDR U280, 6 for HBM U280/U55C and 4 for HBM V80.

All additional configurations, such as burst length, number of transactions, and randomization dimensions (row, column, bank group, bank address, pseudo-channel), are now programmable via *CPG* registers through **QDMA**. This architecture allows to execute hundreds of distinct experiments from a single bitstream containing multiple *CPGs*, supporting rapid design space exploration.

Our exploration revealed that accessing different microswitches introduces noticable latency increases and throughput degradation, as shown in Figure 3. Randomization in access patterns and pseudo-channel switching amplify this performance drop, highlighting the need for controlled and observable memory interactions.

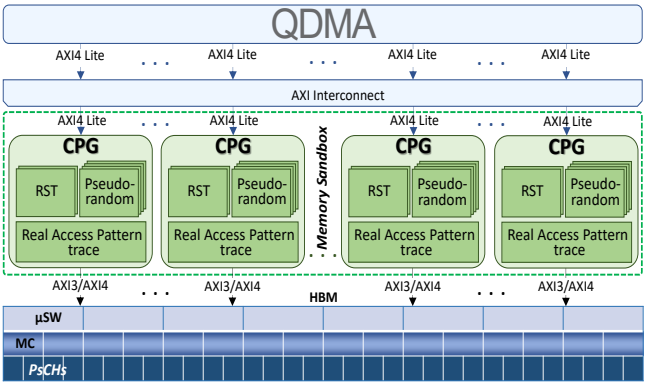

Figure 2: Memory Sandbox 2.0 Hardware architecture.

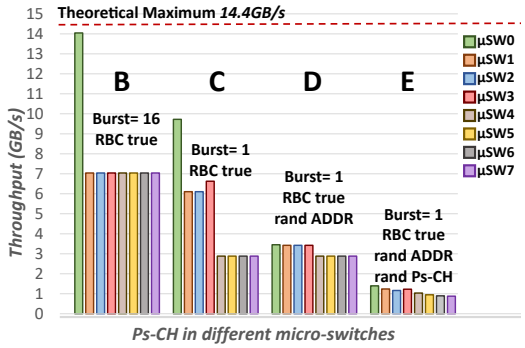

Figure 3: HBM Throughput for Different micro-switches.

To address broader design goals, *Memory Sandbox 2.0* now includes telemetry integration, enabling visibility into physical characteristics such as power consumption and temperature evolution. Beyond static reporting, our framework allows correlation of telemetry with access patterns, memory mappings, and burst parameters, providing crucial insights for optimizing throughput-per-watt, thermal headroom, and energy efficiency under constrained operation.

3 Memory Sandbox 2.0 Telemetry

Understanding how physical variables such as temperature, voltage, and current affect system behavior is critical for designing robust and efficient hardware. These telemetry variables provide essential insight into the performance and energy profile of memory systems and platforms reliability, failure prediction, and thermal-aware scaling. For HBM-based accelerators in particular, ignoring temperature can lead to degraded performance, reduced reliability, or even frequency throttling, as shown in prior work on row hammer effects and thermally-induced fault propagation in DRAM and HBM-based systems [13, 15, 18].

While Xilinx vendor IPs provide reliable access to telemetry data, they focus on accessing raw signals but do not enable systematic correlation between telemetry and memory system behavior. This is precisely the gap addressed by *Memory Sandbox 2.0*, which integrates telemetry tracking directly into a configurable framework that allows performance and thermal/power data to be collected and analyzed as a function of access patterns, memory mapping, burst lengths, and more. By bridging the division between raw sensor data and its architectural implications, we allow users to reason about energy-aware scaling, thermal headroom, and system-level efficiency early in the design cycle. Therefore, our framework not only contributes to more insightful performance tuning but also enhances platform reliability through historical tracking of telemetry trends, supporting predictive maintenance and long-term system monitoring.

3.1 Telemetry Acquisition via CMS & PMC

Telemetry is implemented in Memory Sandbox 2.0 using already successfully deployed methods called '*in-band access*' for Alveo platforms and '*out-of-band access*' for Versal platforms [24].

On Alveo platforms, the telemetry bridge is implemented through Xilinx's Card Management Solution (CMS) IP, which interfaces internally over I²C with an embedded TI-MSP432 microcontroller.

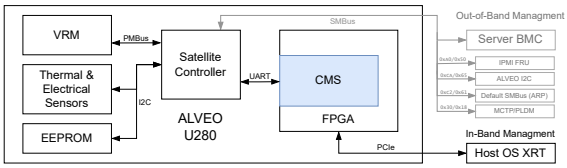

Figure 4: Alveo Ultrascale telemetry methodology.

This controller collects data from voltages, currents, and thermal readings sensors distributed across the board, making them accessible via a mapped register space. Figure 4 shows the system-level interaction between CMS, the host, and the FPGA fabric. CMS is available on all Xilinx UltraScale+ and Alveo platforms, though the granularity and availability of sensors vary by board. Notably, HBM-capable devices expose more fine-grained temperature telemetry, while non-HBM platforms may lack some voltage or HBM-specific temperature sensors.

On Versal platforms, telemetry is handled by the Platform Management Controller (PMC), which manages power sequencing and exposes sensor data, including die temperature, voltage, current, and HBM stack thermals through the Adaptive Management Interface (AMI). This is accessed via AMD's *ami_tool*, enabling '*out-of-band access*' telemetry logging without requiring bitstream-level instrumentation [6].

3.2 Integration into the BSC FPGA-SHELL

We embedded both, *Memory Sandbox 2.0* and CMS telemetry system within the BSC FPGA-SHELL [20], a modular infrastructure for emulating accelerators or RISC-V processors. Once a compatible bitstream is loaded, the shell detects the hardware and initializes the telemetry environment. It periodically issues read requests to CMS and logs the collected data into a time-series database. Key parameters such as sampling interval, buffer size, and storage format can be customized by the user.

Figure 5 depicts long-term monitoring of thermal behavior across the Alveo U280 and Versal V80 platforms performing different experiments. The top subplot compares the server CPU temperature during test execution, while the bottom subplot tracks the FPGA die temperature over several days of testing. We observe that workload bursts correlate strongly with sharp rises in FPGA temperature, especially on the V80, which lacks active thermal management for HBM. These empirical platform aware experiments in our servers demonstrate the need of a more fine grained analysis of the thermal impact in the FPGA and its host of a given workload.

Figure 6 presents telemetry data from the Alveo U280 with 8 CPGs accessing memory intensively, showing a direct relationship between power consumption and die temperature during controlled memory access experiments which is more FPGA and its memory components aware. Notably, we observe temperature plateaus followed by a reset, a valuable insight for designing memory-bound workloads that operate near thermal limits.

3.3 Platform-Specific Thermal Challenges

Memory Sandbox 2.0 thus offers a reusable and extensible platform for analyzing memory performance and telemetry behavior in heterogeneous accelerator environments. Its ability to explore both

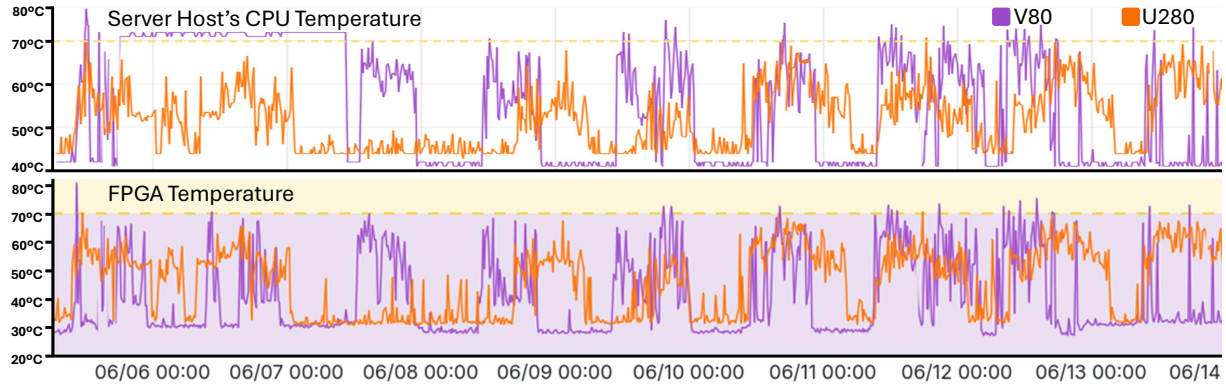

Figure 5: Temperature in Alveo U280 and Versal V80.

physical and behavioral aspects of memory systems, even in the absence of fully developed accelerators, makes it a valuable co-design and performance analysis tool for system architects and researchers alike. By integrating telemetry into our evaluation framework, *Memory Sandbox 2.0* extends its role from a performance analysis tool to a platform-aware co-design environment. The ability to correlate access patterns with physical behavior across power and thermal domains makes it a powerful aid for designing accelerators that must meet strict energy, reliability, and thermal constraints.

On the Versal V80, telemetry monitoring has revealed a critical reliability issue. As shown in Figure 5, once the temperature exceeds a threshold (≈ 70°C), the board enters an unstable state that causes a full reset. This interrupts all services, including telemetry logging and low-level control tools like *ami_tool*. These findings emphasize the importance of integrating preventive telemetry-based monitoring early in the hardware design cycle.

4 Extending Memory Sandbox 2.0 for Versal V80

While *Memory Sandbox 1.0* was developed for UltraScale+ platforms (e.g., Alveo U280/U55C with the mature HBM2 technology), our *Memory Sandbox 2.0* extend its support to the Versal V80. The first FPGA that introduces HBM2e, a more advanced memory technology targeting AI and data-intensive workloads. This transition allows us to analyze how architectural changes in the Versal family impact performance and telemetry, and whether HBM2e can alleviate memory system limitations previously observed in HBM2.

We hypothesize that HBM2e mitigates key bottlenecks, such as pseudo-channel contention and microswitch latency, by leveraging a significantly improved memory interconnect architecture:

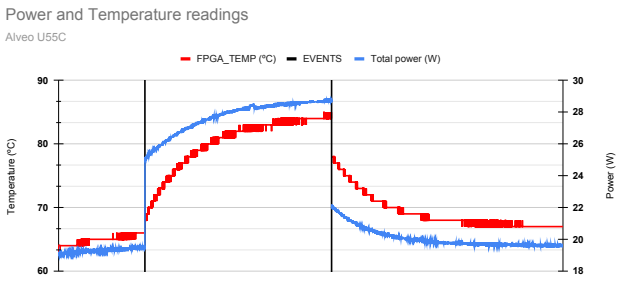

Figure 6: Temperature vs Power Consumption in Alveo U280.

- Transitioning from 8 4×4 microswitches (in HBM2) to
- 8 8×8 microswitches, independently configured for read and write operations, interconnected via both Horizontal and Vertical Networks-on-Chip (NoCs) in the Versal V80.

These enhancements expand the effective communication paths to memory. In contrast to the U280's single lateral microswitch traversal, which can incur a 50% bandwidth drop when crossing domains, Versal's dual NoC topology enables more flexible routing, effectively reducing the penalty of domain transitions. As a result:

- The nominal HBM2e bandwidth in V80 (≈820 GB/s) nearly doubles that of U280 (≈460 GB/s).
- The usable high-throughput memory region increases significantly—from 4 × 256 MB domains (U280) to 8 × 1 GB domains (V80).
- Most importantly, the microswitch locality bottleneck is alleviated, as data can be accessed through multiple NoC paths rather than being constrained to a single switch domain.

This architectural shift not only improves the raw memory performance envelope but also enables more scalable and predictable accelerator-memory interactions, precisely the scenarios *Memory Sandbox 2.0* aims to explore.

To enable consistent profiling between HBM2 and HBM2e platforms, we are adapting our *Memory Sandbox 2.0* framework to support the V80. However, this development effort has encountered several challenges:

- **Toolchain Immaturity:** The V80 is so recent that it does not yet appear in Vivado's board selector UI. Manual TCL commands are required to target the board and create a valid project. This is not an issue itself, but a clear demonstration of the immature status V80 workflow.
- **Higher level of abstraction:** The Versal family introduces new hardware layers, such as:
  - ***CIPS:*** Control Interfaces and Processing System
  - ***AXI NOC:*** A high-throughput, scalable Network-on-Chip interconnect connecting logic blocks with hard IPs (ARM CPUs, DDR, HBM).

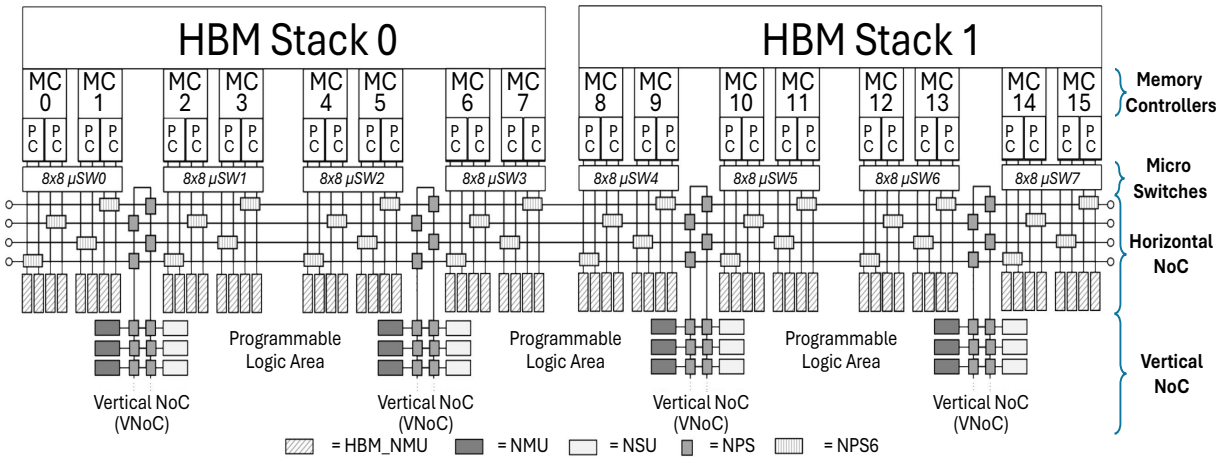

Figure 7: Versal V80 HBM topology.

They provide too many configurations options and interdepencies between parameters which are not properly documented yet for the user to make an efficient use of these IPs.

- **Incomplete Constraint Files:** As of this writing, AMD/Xilinx has not released official XDC constraint files for the V80, hindering timing closure and signal integrity efforts.
- **Thermal Constraints:** Versal devices have tighter thermal limits than UltraScale+ boards. While U280s operate reliably at up to ≈55°C, the V80 is prone to thermal throttling or shutdown near ≈45°C, creating significant integration challenges in standard lab environments like ours.
- **Complex Bitstream and System Integration:** Unlike UltraScale+, bitstream deployment on Versal involves managing both the programmable logic (PL) and a semi-configurable Processing Subsystem (PS), which adds extra complexity. Proper system operation requires extra layers of vendor specific software which adds dependency and complexity.

These hurdles emphasize the development overhead of migrating to Versal platforms and motivate the need for our *Memory Sandbox 2.0* to abstract hardware architectural details or requirements while enabling reproducible performance and telemetry analysis.

5 Background & Related Work

To the best of our knowledge, our work is the first one aiming to profile HBM2e on FPGAs in terms of performance and telemetry and its correlation. Our goal is to create a comprehensive baseline for HPC developers which includes: real time physical parameters monitoring, accesses across pseudo-channels, concurrent access and a real application validation. Table2 lists the related works compared to our *Memory Sandbox 2.0*. To understand the novelty of our paper, we broadly classify the related work from the recent literature into four categories:

- Benchmarking features HBM on FPGAs.
- Telemetry on FPGAs.
- HBM2e analysis.

The closest endeavor in comparison to ours is [10], where they also perform a thorough approach to profile HBM. Nevertheless, our solution incorporates Telemetry and targets HBM2e. Moreover, being RTL-based gives more control over signals, shows concurrent access behavior, and enables real access pattern evaluation.

Other Vendor-specific tools, such as the Versal AVED[2], provide platform-level support with no pattern-aware telemetry, deep debug with no access pattern emulation. The AXI Performance Monitor provides bandwidth and latency visibility [26], but cannot correlate telemetry data and custom access pattern behavior, which is where Memory Sandbox 2.0 fills the gap.

6 Conclusions

This work introduces Memory Sandbox 2.0, a modular and extensible framework designed to analyze memory system behavior across performance and telemetry dimensions. By refactoring our original tool, we have made it capable of easily integrating new memory technologies (such as HBM2e) and supporting custom access patterns, enabling more systematic design space exploration.

Enhanced parameterization and user interface improvements allowing developers to replicate hardware-relevant memory behavior, such as sequential, pseudo-random, and sparse patterns, without requiring a full accelerator implementation, as demonstrated with our SpMV-inspired access pattern.

Moreover, moving most of the configuration parameters to runtime allows to execute multiple experiments on a same bitstream. By means, a single bitstream might contain multiple CPGs performing concurrent experiments, or one after another.

Each Configurable Pattern Generators (CPGs) can emulate diverse access schemes, allowing to study how memory-bound accelerators or CPUs behave in heterogeneous computing environments. To validate these capabilities, we are currently conducting a comparative analysis of HBM2 (Alveo U280) and HBM2e (Versal V80) in terms of throughput, latency, and telemetry (temperature, power).

Ultimately, Memory Sandbox 2.0 contributes toward enabling more efficient and performance-aware accelerator design, particularly in HPC and AI domains where memory bottlenecks and energy constraints play a critical role.

Acknowledgements

This work has been co-financed by the Barcelona Zettascale Laboratory under project reference REGAGE22e00058408992, with support from the Spanish Ministry for Digital Transformation and Public

Table 2: Comparison of our *Memory Sandbox 2.0* with the state of the art solutions

Endeavour	Open source	HLS based	RTL based	HBM+FPGA profiling								
				FPGAs	**Telemetry**	**Latency**	**Within PSCHs**	**Across PSCHs**	**Concurrent PSCHs**	**RST**	**Pseudo-random**	**Real access pattern**
Shuhai[12, 25]	□	☑	□	Alveo	□	☑	☑	□	□	☑	□	□
General HBM profilers [7, 14, 17, 19, 23]	□	☑	□	Alveo	□	☑	□	□	□	☑	□	□
MAO[10]	□	☑	□	Alveo	□	☑	☑	☑	□	☑	☑	□
Versal AVED[2]	□	□	☑	Versal	≈	☑	☑	☑	☑	□	□	□
Memory Sandbox 1.0[21]	☑	□	☑	Alveo	□	☑	☑	☑	☑	☑	☑	☑
Memory Sandbox 2.0	☑	□	☑	**Alveo Versal**	☑	☑	☑	☑	☑	☑	☑	☑

Services, within the framework of the Recovery and Resilience Facility and the European Union – NextGenerationEU; from the research projects ST4HPC (PID2023-147979NB-C21) and the ACAP project (PID2023-146511NB-I00) sup- ported by the Spanish Ministry of Science and Innovation (MICIU), the State Research Agency (AEI), and the European Regional De- velopment Fund (FEDER). Additional support was provided by the Lenovo–BSC Contract Framework (2022) and the LOCA project (CEX2021-001148-S) funded by MCIN/AEI /10.13039/501100011033.

References

[1] 2023. JEDEC® and Industry Leaders Collaborate to Release: High Bandwidth Memory (HBM3) DRAM | JEDEC. https://www.jedec.org/standards-documents/docs/jesd238b01

[2] 2024. *Designing with the Versal Adaptive SoC: Hardware Debug.* Technical Report. Accessed: 2025-06-15.

[3] 2024. JEDEC® and Industry Leaders Collaborate to Release JESD270-4 HBM4 Standard: Advancing Bandwidth, Efficiency, and Capacity for AI and HPC | JEDEC. https://www.jedec.org/news/pressreleases/jedec%C2%AE-and-industry-leaders-collaborate-release-jesd270-4-hbm4-standard-advancing

[4] AMD Xilinx Inc. 2022. *Alveo U280 Data Center Accelerator Card Data Sheet.* Technical Report DS963. Xilinx Inc. https://www.xilinx.com/content/dam/xilinx/support/documents/data_sheets/ds963-u280.pdf

[5] AMD Xilinx Inc. 2022. *Alveo U55C Data Center Accelerator Cards Data Sheet.* Technical Report DS978. Xilinx Inc. https://www.xilinx.com/content/dam/xilinx/support/documents/data_sheets/ds978-u55c.pdf

[6] AMD/Xilinx. 2024. Versal Adaptive SoC System Monitor Architecture Manual (AM006)). https://docs.amd.com/r/en-US/am006-versal-sysmon. Accessed: 2025-06-15.

[7] Young-Kyu Choi, Yuze Chi, Weikang Qiao, Nikola Samardzic, and Jason Cong. 2021. HBM Connect: High-Performance HLS Interconnect for FPGA HBM. In *The 2021 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays (FPGA '21)*. Association for Computing Machinery, New York, NY, USA, 116–126. doi:10.1145/3431920.3439301 event-place: Virtual Event, USA, doi: 10.1145/3431920.3439301.

[8] Eric S. Chung, Peter A. Milder, James C. Hoe, and Ken Mai. 2010. Single-Chip Heterogeneous Computing: Does the Future Include Custom Logic, FPGAs, and GPGPUs?. In *2010 43rd Annual IEEE/ACM International Symposium on Microarchitecture*. 225–236. doi:10.1109/MICRO.2010.36

[9] Jason Cong, Zhenman Fang, Michael Lo, Hanrui Wang, Jingxian Xu, and Shaochong Zhang. 2018. Understanding Performance Differences of FPGAs and GPUs. In *2018 IEEE 26th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM)*. 93–96. doi:10.1109/FCCM.2018.00023 ISSN: 2576-2621.

[10] Philipp Holzinger, Daniel Reiser, Tobias Hahn, and Marc Reichenbach. 2021. Fast HBM Access with FPGAs: Analysis, Architectures, and Applications. In *2021 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*. 152–159. doi:10.1109/IPDPSW52791.2021.00030

[11] Sungpack Hong, Sang Kyun Kim, Tayo Oguntebi, and Kunle Olukotun. 2011. Accelerating CUDA graph algorithms at maximum warp. *SIGPLAN Not.* 46, 8 (Feb. 2011), 267–276. doi:10.1145/2038037.1941590

[12] Hongjing Huang, Zeke Wang, Jie´ Zhang, Zhenhao He, Chao Wu, Jun Xiao, and Gustavo Alonso. 2022. Shuhai: A Tool for Benchmarking High Bandwidth Memory on FPGAs. *IEEE Trans. Comput.* 71, 5 (May 2022), 1133–1144. doi:10.1109/TC.2021.3075765 doi: 10.1109/TC.2021.3075765.

[13] Yoongu Kim, Ross Daly, Jeremie Kim, Chris Fallin, Ji Hye Lee, Donghyuk Lee, Chris Wilkerson, Konrad Lai, and Onur Mutlu. 2014. Flipping bits in memory without accessing them: An experimental study of DRAM disturbance errors. *ACM SIGARCH Computer Architecture News* 42, 3 (2014), 361–372.

[14] Seyed Saber Nabavi Larimi, Behzad Salami, Osman S Unsal, Adrián Cristal Kestelman, Hamid Sarbazi-Azad, and Onur Mutlu. 2021. Understanding power consumption and reliability of high-bandwidth memory with voltage underscaling. In *2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 517–522.

[15] Srijeeta Maity, Anirban Ghose, Soumyajit Dey, and Swarnendu Biswas. 2021. Thermal-aware Adaptive Platform Management for Heterogeneous Embedded Systems. *ACM Trans. Embed. Comput. Syst.* 20, 5s, Article 97, 28 pages. doi:10.1145/3477028

[16] Ehab Mohsen. 2024. Unleashing Computational Power with the New AMD Alveo V80. https://www.amd.com/en/blogs/2024/unleashing-computational-power-with-the-new-amd-al.html. Accessed: 2025-06-15.

[17] Hector Gerardo Muñoz Hernandez, Veronia Iskandar, Lukas Steiner, Philipp Holzinger, Matthias Jung, Diana Göhringer, Michael Hübner, Norbert Wehn, and Marc Reichenbach. 2024. A Novel System Simulation Framework for HBM2 FPGA Platforms. In *International Conference on Embedded Computer Systems: Architectures, Modeling, and Simulation*. Springer, 72–84.

[18] Ataberk Olgun, Majd Osseiran, A Giray Yağlıkçı, Yahya Can Tuğrul, Haocong Luo, Steve Rhyner, Behzad Salami, Juan Gomez Luna, and Onur Mutlu. 2024. Read disturbance in high bandwidth memory: A detailed experimental study on hbm2 dram chips. In *2024 54th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. IEEE, 75–89.

[19] Ataberk Olgun, Majd Osseiran, A. Giray Yağlıkçı, Yahya Can Tuğrul, Haocong Luo, Steve Rhyner, Behzad Salami, Juan Gomez Luna, and Onur Mutlu. 2024. Read Disturbance in High Bandwidth Memory: A Detailed Experimental Study on HBM2 DRAM Chips. In *2024 54th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. 75–89. doi:10.1109/DSN58291.2024.00022

[20] Elias Perdomo, Alexander Kropotov, Francelly Katherine Cano Ladino, Syed Zafar, Teresa Cervero, Xavier Martorell Bofill, and Behzad Salami. 2024. Makinote: An FPGA-Based HW/SW Platform for Pre-Silicon Emulation of RISC-V Designs. In *Proceedings of the 16th Workshop on Rapid Simulation and Performance Evaluation for Design* (Munich, Germany) *(RAPIDO '24)*. Association for Computing Machinery, New York, NY, USA, 29–34. doi:10.1145/3642921.3642928

[21] Elias Perdomo, Xavier Martorell, Teresa Cervero, and Behzad Salami. 2024. Memory Sandbox: A Versatile Tool for Analyzing and Optimizing HBM Performance in FPGA. In *2024 IEEE 36th International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD)*. 206–217. doi:10.1109/SBAC-PAD63648.2024.00026

[22] Samsung Electronics. 2022. High Bandwidth Memory (HBM). https://www.samsung.com/semiconductor/dram/hbm/. Accessed: 2025-06-15.

[23] Runbin Shi, Kaan Kara, Christoph Hagleitner, Dionysios Diamantopoulos, Dimitris Syrivelis, and Gustavo Alonso. 2022. Exploiting HBM on FPGAs for data processing. *ACM Transactions on Reconfigurable Technology and Systems* 15, 4 (Dec. 2022), 1–27. doi:10.1145/3491238 Publisher: ACM New York, NY, US, doi: 10.1145/3491238.

[24] Joan Teruel, Elias Perdomo, David Castells, Xavier Martorell, and Behzad Salami. 2025. Developing telemetry tools for Alveo Accelerator cards.. In *2025 40th Conference on Design of Circuits and Integrated Systems (DCIS) (Posters Session)*. IEEE, Santander, Spain.

[25] Zeke Wang, Hongjing Huang, Jie Zhang, and Gustavo Alonso. 2020. Shuhai: Benchmarking High Bandwidth Memory On FPGAS. In *2020 IEEE 28th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM)*. 111–119. doi:10.1109/FCCM48280.2020.00024 doi: 10.1109/FCCM48280.2020.00024.

[26] Xilinx, Inc. [n. d.]. AXI Performance Monitor (axi_perf_mon) IP Core. https://xilinx-wiki.atlassian.net/wiki/display/A/Axi+Performance+Monitor+standalone. Last updated May 29, 2025; Accessed: 2025-06-15.

[27] Xilinx Inc. 2021. *AXI High Bandwidth Memory Controller v1.0 LogiCORE IP Product Guide.* Technical Report PG276 (v1.0). Xilinx Inc. https://www.xilinx.com/support/documentation/ip_documentation/hbm/v1_0/pg276-axi-hbm.pdf

[28] Lehan Zhang. 2024. Mitigating measures of memory access bottlenecks in high performance computing. *Applied and Computational Engineering* 73 (2024).