# Rambus Smart Data Acceleration

***Back to the Future***

## Memory and Data Access: The Final Frontier

As an industry, if real progress is to be made towards the level of computing that the future mandates, then the way computing problems are attacked must change. The von Neumann execution model has and will continue to serve us well, but additional techniques must be brought to bear.

The next logical focus area is data—how it is accessed, and how it is transformed into real information—that leads to newer solutions. No longer can all of memory simply continue to be an element that holds the program commands and data during execution. Memory must become an active part of the solution, rather than a necessary evil.

The first step in what is likely to be a protracted journey is to provide a vehicle that allows it to be exploited independently of the CPU. Rambus is doing just that through the creation of their Smart Data Acceleration (SDA) Research Program.

## So what's this all about?

The von Neumann model, first described in 1945 by physicist John von Neumann, is an enduring principle upon which virtually all modern computer systems have been built. While its meaning in modern terminology has evolved in many aspects, it has come to mean the following:

- **Von Neumann Model:** Program execution from memory with a singular linear address space and access method (address, data, and control) containing both program information and data used by computational elements.

Intel, in particular, has advanced the x86 architecture by being among the industry leaders who implemented symmetric multiprocessing (SMP)—in their case, through the use of multiple engines ("sockets") and further by extension, through the inclusion of multiple execution units ("cores") in each socket. Their implementation of SMP allowed for some departure from the central von Neumann definition. It includes a semiprivate memory for each socket with the ability to concatenate these memories together into the single linear address space. The connection method, while an extremely high speed interconnect, does introduce non-uniform memory access characteristics (NUMA). In simple terms, this means that obtaining information from the semiprivate ("near") memory is faster than an equivalent access to other ("far") memory.

Modern operating systems must detect and plan for these NUMA characteristics if they are to achieve maximum performance. Extensive caching techniques have also been employed in this design and reduce both the impact of NUMA and of simultaneous access to memory by multiple execution cores. By also including "write back data

caching" (not only caching data reads but also caching data writes—with extensive hardware based techniques that guarantee data coherency), program execution is allowed to advance without waiting for a response from the slower main memory.

Even with all of these remarkable methods, the performance impact of the singular, shared data resource is inevitable.

At this point, a couple of other core principles that system architects continually ponder should be mentioned.

- **Data at rest should remain at rest.** If anything occurs that disturbs data and does not directly contribute to the solution—such as inspecting, smelling, tasting, or moving data—it should be stopped! It is a waste of valuable time and resource. Historically, moving data has been a very impactful offender.
- **Work should be performed as close as possible (in time) to the location of the data.** Simply speaking, dispatch the work to the resource that can get to the data fastest. There is a temptation for a variety of reasons which are generally good to move or make copies of data. But unless the operation is destructive, we should avoid moving or copying it. Unfortunately, modern OS and application complexity sometimes trigger these events that may be invisible to both the programmer and the solution developer. (There are also many other NUMA implications here.)

These core principles are driven by the harsh realities of modern memory hierarchies. Accessing data on local disks or in remote nodes can take one thousand times (or more) longer compared to accessing data in local DRAM.

We should also note that CPU manufacturers have done a good job in advancing the DRAM / CPU ecosystem with incredible speed and difficult signal integrity. No one would advocate changing that, but there also need to be other opportunities to take advantage of the advancement of DRAM in overall system performance.

**Providing solutions to the class of problems identified in this section, along with the ability to offload much of the data transformation, is what the Rambus SDA Research Program is all about.**

## The Rambus SDA Research Program

Almost all modern solution providers agree that current and emerging computing and storage resources are out-of-balance with the need. Datacenters are under stress due to the demands for real-time access to large amounts of information, driven by Big Data and new applications. The Rambus Smart Data Acceleration (SDA) Research Program focuses on architectures that offload computing, making it close (in time) to these very large data sets and at multiple points in the memory and storage hierarchy.

The SDA Research Program is designed to tackle the major issues facing datacenters in the Big Data era. The program has been exploring new architectures for servers and

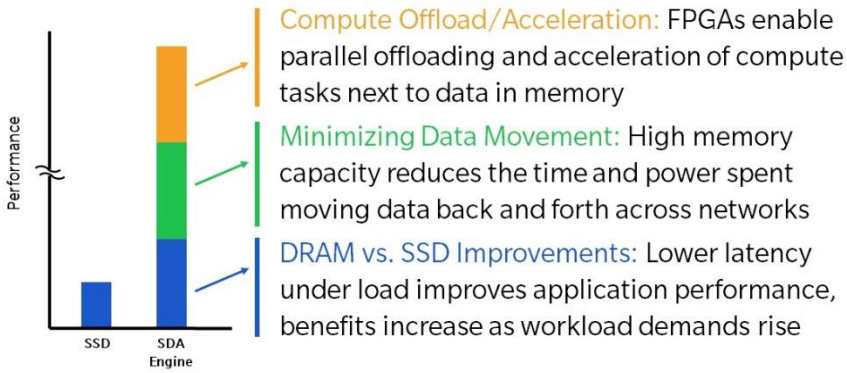
datacenters that bring computing closer to data, and it is targeting significant improvements in performance and power efficiency.

The SDA Research Program is looking to attain performance improvements by:

1. Leveraging the performance benefits of DRAM
2. Minimizing data movement
3. Enabling compute offload and acceleration through the use of FPGAs

As part of the SDA Research Program, Rambus has also created a platform to enable architectural exploration that includes software, firmware, FPGAs, and large amounts of memory. This platform can be used to test new methods to optimize and accelerate data analytics for extremely large data sets as shown in Figure 1 below.

**Figure 1: Rambus SDA Focus Areas**

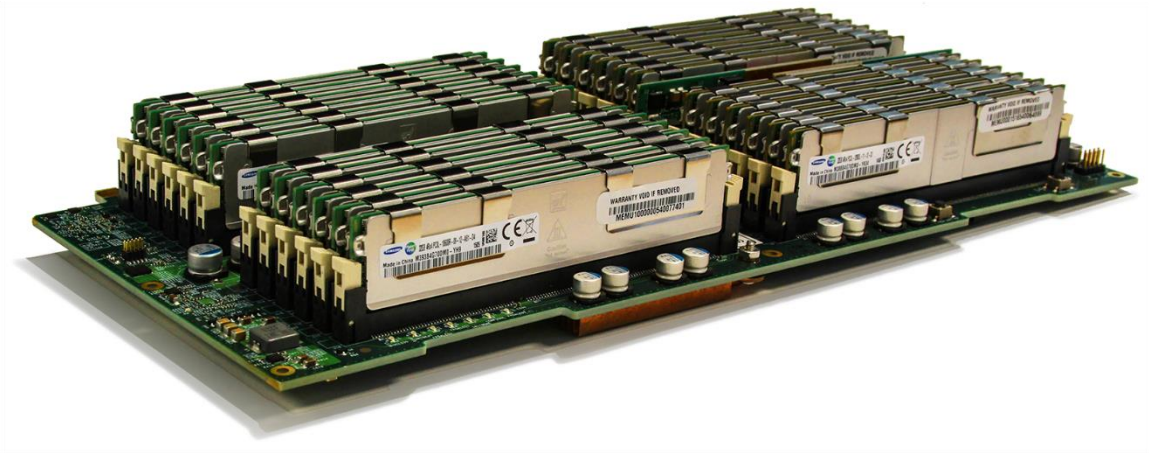



*Source: Rambus*

## The Rambus SDA Research Platform

The platform itself is a combination of hardware, software, firmware, drivers, and bit files that can be adjusted in different ways to allow architectural exploration. At its heart is the SDA engine (Figure 2), which includes an FPGA coupled with a high capacity of memory.

**Figure 2: Rambus SDA Engine**



*Source: Rambus*

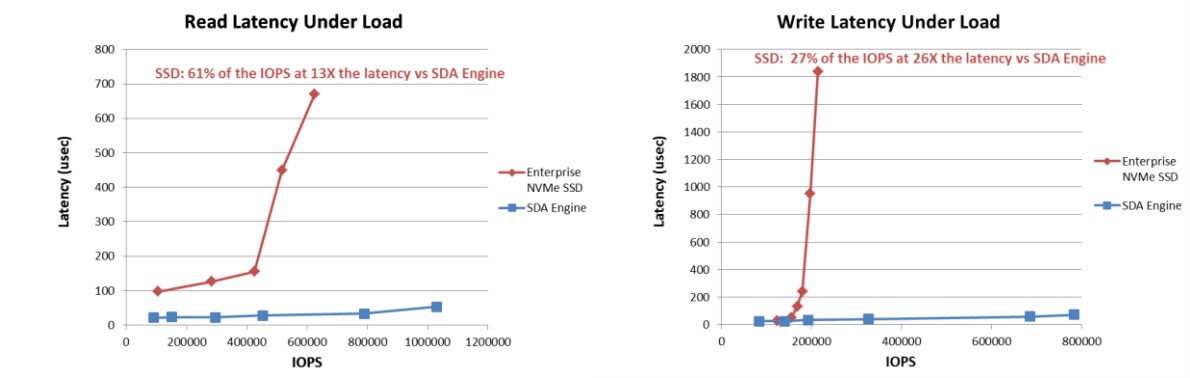
The SDA engine connects 24 DIMMs to an FPGA, providing high memory densities connected to a flexible computing resource. This form factor enables a higher density of DIMM sockets than typical servers in use today. And because these are standard DIMM sockets, other DIMM-based technologies can be supported by the architecture, such as NAND and emerging storage class memories.

At the system level, one or more SDA engines can be combined into a flexible platform that can appear to the rest of the system as different types of devices, including an ultra-fast block storage device, a Key/Value store, or a caching device, among others. This flexibility allows different optimizations to be studied for different workloads.

Early work with the Rambus SDA Research Platform has looked at its suitability as an ultra-fast block storage device. At first glance, in this usage model, the platform is reminiscent of the mass memory systems that first emerged in the 1980s (hence the *Back to the Future* reference in this brief's title). In mass memory systems, large DRAM arrays were configured to emulate early hard drives. They proved to be incredibly fast compared to disks of the day and were a real performance kick.

As one would expect, the SDA engine's DRAM-based block storage proves to be nothing short of "snappy" (Figure 3). When configured as an ultra-fast block storage device, initial tests show that higher IOPS rates can be achieved under certain workloads, and with much better latency under load, compared to state-of-the-art NVMe SSDs.

**Figure 3: Comparative Read & Write Latency**




*Source: Rambus*

To date, only a limited number of benchmarks have been run, but results are in line with expectations. Notably, in 4KB random access tests, the SDA engine can deliver up to 1M IOPS with latency under load in the 10 µs to 60 µs range for both reads and writes, with additional headroom to achieve higher IOPS rates.

Of course this is a lot of trouble and work to prove the obvious fact that large DRAM arrays are faster than flash. Nonetheless, it is an aspect that deserves some discussion. Further, it should be noted that there is nothing inherent to the SDA architecture and design that restricts it to this emulation or delivery model. It is a simple matter of convenience and method to speed research.

## "Piece de Resistance"

The overarching purpose of the SDA research lies in the collection of FPGAs at the heart of the design, where each FPGA has parallel and independent access to the DIMMs. The SDA engines are also "semantically-aware", meaning structures in memory can be described to the SDA and bounded. These structures then can be operated on or subjected to transformations in a fashion that maximizes parallelism and efficiency without the need to move data. This in effect creates a unique data store that can offload significant data operations from the main processor and into the SDA engine, which essentially operates as a large data-parallel coprocessor.

Every system architect, when faced with something as mundane as matrix manipulation (for example, transpose a matrix), has longed for the ability to say to memory, "Please swap the rows and columns of this matrix and let me know when you are done. Oh, and don't move the data around while you're doing it." SDA research is heading in that direction.

Because of its nature and the early state of research, more details about the instruction set may be attained directly from Rambus. It is worth noting however, that the current, simple, extensible command set is targeted at accelerating and offloading the transformation of common data structures like those used in Big Data analytics applications.

## Dream a Little Dream with Me

Imagine, if you would, a day in the not-too-distant future where critical resources are disaggregated, and appliances are a key tool at the disposal of solutions providers. A shared resource (similar to the Smart Data Acceleration platform) could be made available over a network and serve as a key offload agent. This type of semantically-aware data transformer holds great promise. Such a resource could be applied to modern applications and address the challenges highlighted in the hierarchy off-load gap shown in Figure 1.

We believe the Rambus Smart Data Acceleration Research Program is addressing an important need and we look forward to hearing more about this program as it develops. For more information, visit http://www.rambus.com/emerging-solutions/smart-data-acceleration/.

## Important Information About This Brief

**Inquiries**
Please contact us here if you would like to discuss this report, and Moor Insights & Strategy will promptly respond.

**Citations**
This brief can be cited by accredited press and analysts, but it must be cited in-context, displaying author's name, author's title, and "Moor Insights & Strategy". Non-press and non-analysts must receive prior written permission by Moor Insights & Strategy for any citations.

**Licensing**


**Disclosures**
This paper was commissioned by Rambus. Moor Insights & Strategy provides research, analysis, advising, and consulting to many high-tech companies mentioned in this paper. No employees at the firm hold any equity positions with any companies cited in this document.

**DISCLAIMER**
The information presented in this document is for informational purposes only and may contain technical inaccuracies, omissions, and typographical errors. Moor Insights & Strategy disclaims all warranties as to the accuracy, completeness, or adequacy of such information and shall have no liability for errors, omissions, or inadequacies in such information. This document consists of the opinions of Moor Insights & Strategy and should not be construed as statements of fact. The opinions expressed herein are subject to change without notice.

Moor Insights & Strategy provides forecasts and forward-looking statements as directional indicators and not as precise predictions of future events. While our forecasts and forward-looking statements represent our current judgment on what the future holds, they are subject to risks and uncertainties that could cause actual results to differ materially. You are cautioned not to place undue reliance on these forecasts and forward-looking statements, which reflect our opinions only as of the date of publication for this document. Please keep in mind that we are not obligating ourselves to revise or publicly release the results of any revision to these forecasts and forward-looking statements in light of new information or future events.