Integrated Device Technology

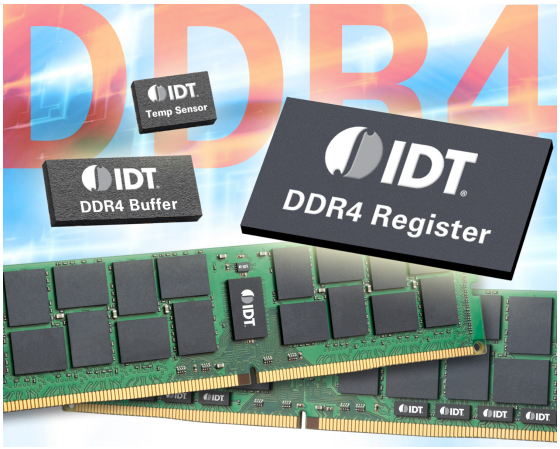
SAMSUNG

WHITE PAPER

DDR4 LRDIMMs

Unprecedented Memory Bandwidth on Samsung DDR4 LRDIMM Enabled by IDT's Register and Data Buffer

By Douglas Malech, IDT



Introduction

As Big data business analytics, real time data for social media and mobile application continue their explosive growth, the need for higher speed and memory capacity could not have been greater. Over last few years, Samsung, IDT and other companies in the memory eco-system have worked closely to continue to advance the system memory roadmap for enterprise applications. This white paper aims to highlight the advances the industry has made with the latest memory technology DDR4 and more specifically on LRDIMM (Load reduced DIMM).

DDR4 LRDIMM (load reduced memory module) technology uses a distributed buffer approach to accomplish memory bandwidth efficiencies when scaling to higher capacities and speed on the upcoming DDR4 enterprise server systems, as compared to DDR4 RDIMM (registered memory modules). LRDIMM, in general, have continued to evolve and improved their value to system users. In Figure 1 below, Gen1 DDR3 enterprise systems such as E5-2600 had sub-optimal speed on LRDIMM for all capacities. In addition, the controller performance also left a lot to be desired. E5-2600 v2 made significant progress in improving LRDIMM value to end-users and reversed the speed inversion issue that existed on E5-2600. DDR4 LRDIMM is expected to launch memory subsystem performance to a new paradigm. DDR4 LRDIMM not only appeals to the highest capacities, but also to a much wider range of applications that require highest bandwidth and/or highest capacities.

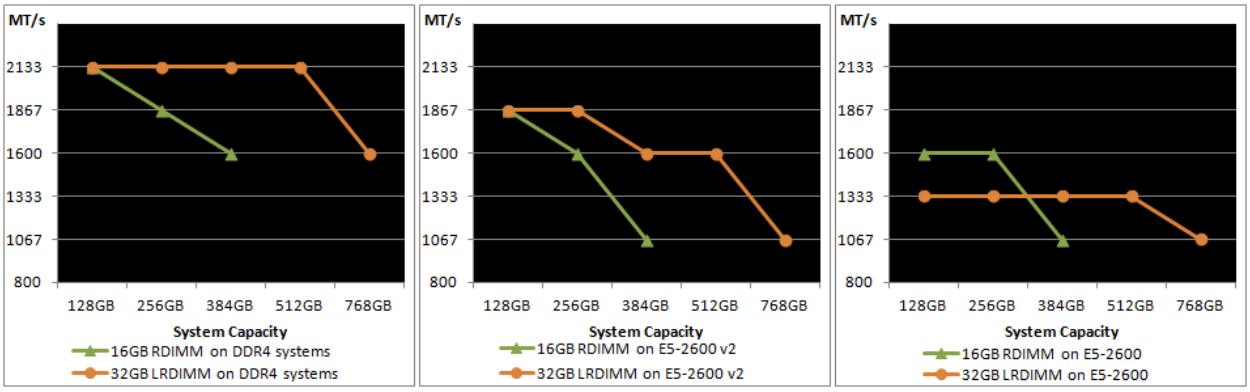


Figure 1: LRDIMM vs. RDIMM speed improvement

In addition to speed, the eco-system has collectively made huge strides in ensuring that increase in speed with LRDIMM translates into a corresponding scaling of memory bandwidth in GigaByte/second (GB/s) which is essentially what a customer pays for.

To summarize the DDR4 improvements made by various stake-holders in the eco-system to improve the usable bandwidth in GB/s

- Improved LRDIMM design for better signal integrity on the data signals.
- Lower component latency through the data buffer components
- Better intelligence and post-buffer awareness by the memory controller

How is it done?

To understand how DDR4 achieves this breakthrough in LRDIMM technology, stark contrasts can be shown with DDR3. DDR4 and DDR3 LRDIMMs both reduce the number of data loads to improve signal integrity on the memory module's data bus from a maximum of 4 data loads down to 1 data load; however, DDR4 introduces some additional features to reduce overall latency and improve signal integrity, leading to speeds comparable to DDR4 RDIMM. The following tables summarize the key points in this article:

DDR3 LRDIMM Design	Characteristic	Downside
Centralized memory buffer	Longer I/O trace lengths	Longer stubs connected to memory channel cause worse signal reflections, resulting in poorer signal integrity
		Longer traces cause increased latency
		Longer traces cause longer I/O bus turnaround time
Component latency	Delay is approximately 2.5ns	Longer propagation delay adds to latency

DDR4 LRDIMM Design	Characteristic	Upside
Distributed buffers	Shorter I/O trace lengths	Shorter stubs connected to memory channel cause less pronounced signal reflections, resulting in improved signal integrity
		Shorter traces reduce latency
		Shorter traces reduce I/O bus turnaround time
Component latency	Delay is approximately 1.2ns	50% less latency than DDR3 memory buffer,

LRDIMM Design: Centralized vs Distributed Buffers

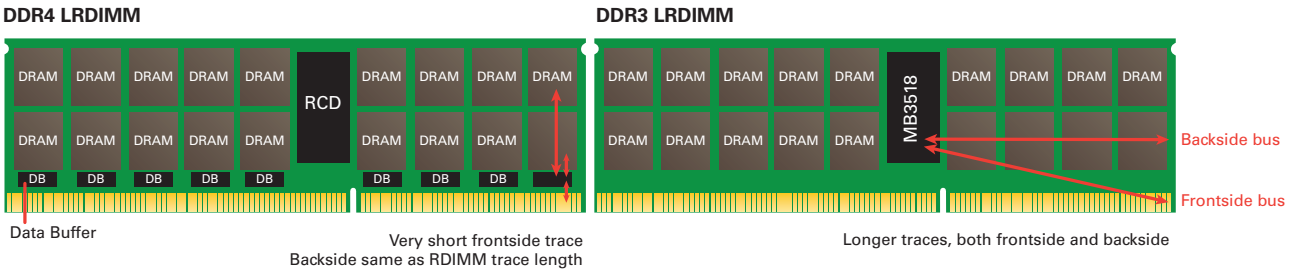


Figure 2: LRDIMM Design topology

A DDR3 LRDIMM has a memory buffer located in the center of the module, as shown in Figure 2. The memory buffer (MB3518) not only buffers and retransmits the command, address and clock signals to the DRAMs, but also buffers the DRAM I/O data bus. By buffering the I/O data, up to four DRAMs loads are reduced to one load on the memory channel. Fewer loads improves signal integrity; however, while DDR3 RDIMM I/O data bus traces go directly from the edge connector to the DRAM I/O loads[1], DDR3 LRDIMM traces are routed to the centrally located memory buffer, thereby increasing trace lengths up to six inches[2] on both the frontside and backside of the memory buffer (Figure 3). These longer trace lengths place an additional burden on the DDR3 high speed data path's signal integrity and latency by introducing very long stubs onto the memory channel bus, translating into slower maximum I/O speeds when compared with DDR3 RDIMM.

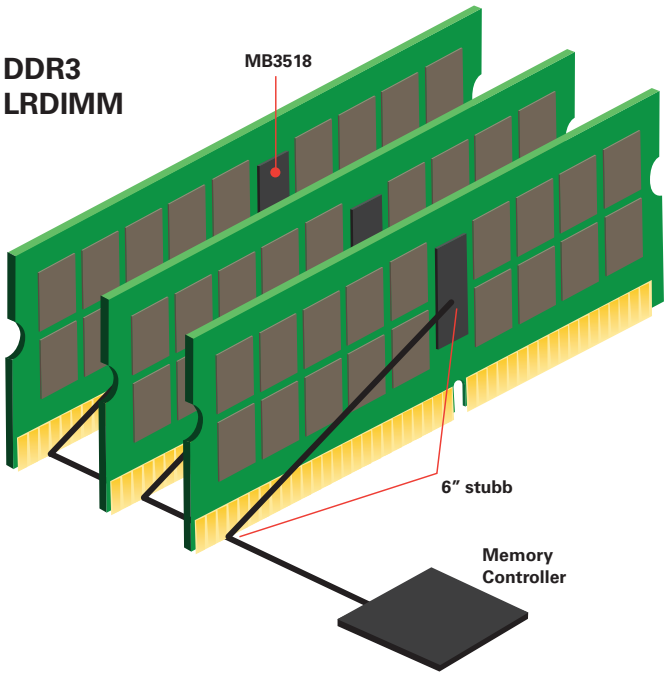


Figure 3: Trace-lengths on DDR3 LRDIMM topology

[1] Series resistors are placed close to the edge connector in both LRDIMM and RDIMM cases, so are not discussed in this comparative study

[2] Six inches of trace adds about 0.5ns of additional latency

As shown in Figure 4 below, DDR4 LRDIMM technology gets rid of the longer trace lengths introduced by DDR3 memory buffers by breaking out the data buffer function and distributing nine data buffer devices between the edge connector and the nine columns of DDR4 DRAMs. The six inch traces found on DDR3 LRDIMMs are gone and are replaced with 2mm frontside bus traces and backside traces equivalent in length to a DDR4 RDIMM. In addition to trace lengths comparable to DDR4 RDIMM, the propagation delay through the DDR4 data buffers is about 33 percent faster than through the DDR3 memory buffer[3], further reducing latency. DDR4 LRDIMM's shorter trace length and quicker buffering scheme make the overall latency and bus read/write turnaround time comparable to DDR4 RDIMM.

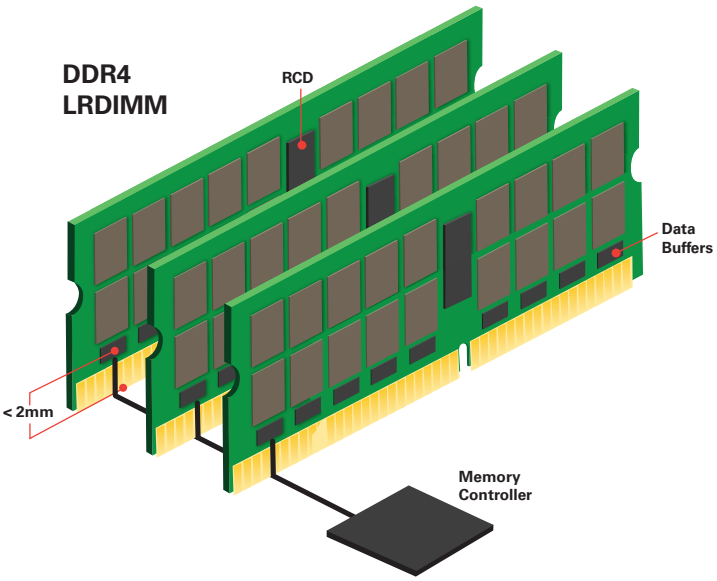


Figure 4: Trace-lengths of DDR4 LRDIMM topology

As shown in Figure 5, these better transmission line characteristics allow faster DDR4 data rates than DDR3 by contributing to a more pristine data eye opening.

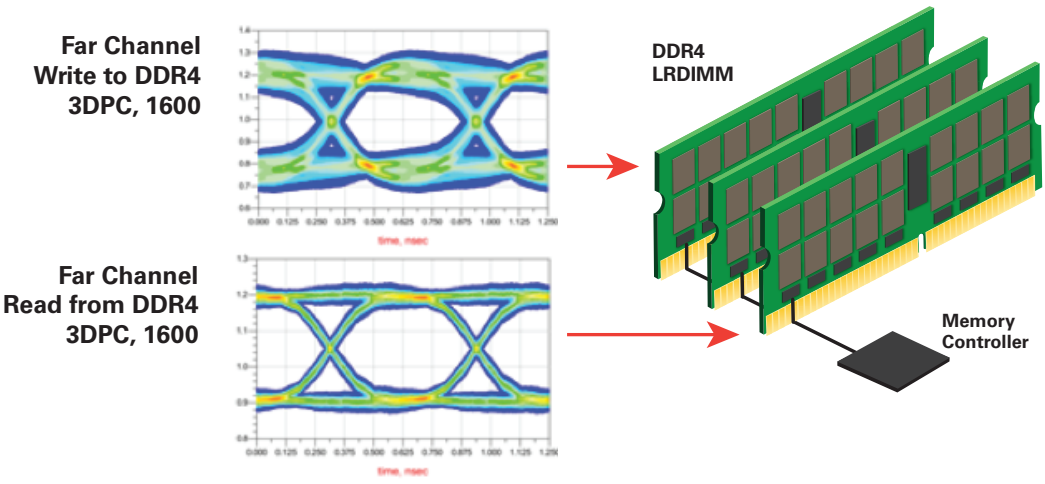


Figure 5: Pristine signal eye opening due to better transmission line characteristics due to shorter data stub lengths

[3] 1.67ns+tCK/4 compared to 2.5ns+tCK/4

Optimizing Component Latency

In addition, the centralized DDR3 memory buffer also adds 2.5ns of delay through the buffer and has an additional six inch signal path between the backside of the memory buffer and the DRAM I/O, introducing even more latency to DDR3 LRDIMM when compared to an RDIMM approach. More latency through the central memory buffer for DDR3 resulted in lowering of the effective bandwidth (Figure 6).

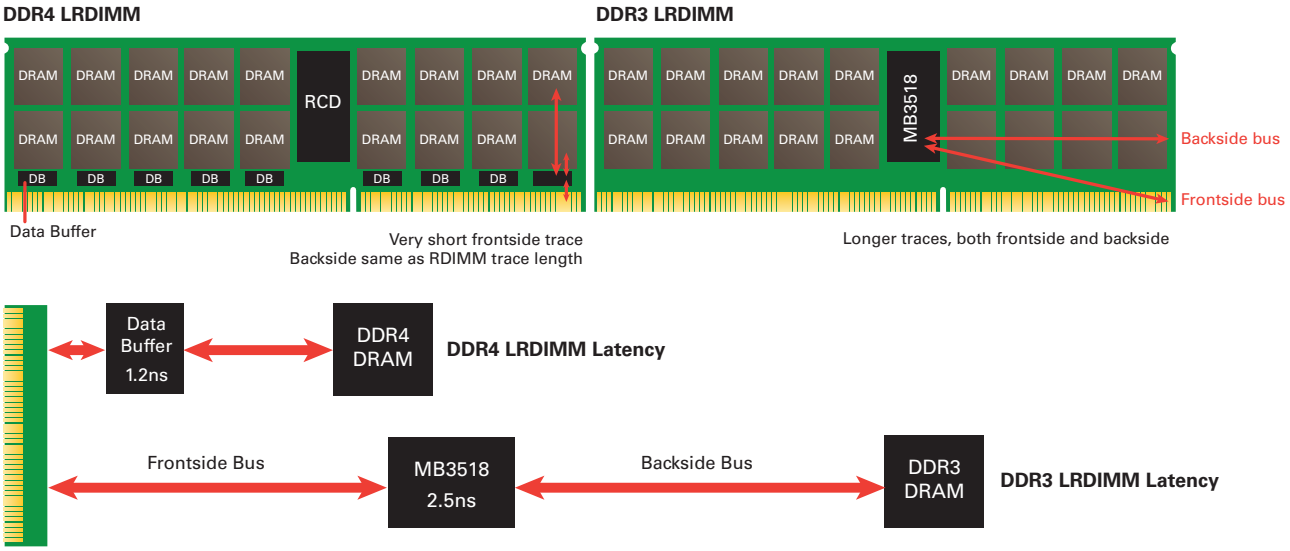


Figure 6: Component latency comparison

LRDIMM rank aware controllers

Over the last few years since the introduction of initial DDR3 LRDIMMs, memory controllers have continued to add intelligence in order to improve the utilization of the LRDIMM deeper memory capabilities. More recently, improvements in a feature called "rank multiplication" have contributed to reducing latency and improving bandwidth. Rank multiplication is shown in Figure 7. It was designed to get around the issue of having only two select bits to choose from 1 of 5 choices; choices 1-4 are to select 1 of 4 DRAMs (on a quad rank "4R" DIMM) to retrieve a piece of data and choice 5 is to not access any of the 4 DRAMs. Choice 5 may be used, for example, when the data is being retrieved from the 2nd or 3rd LRDIMM in the memory channel.

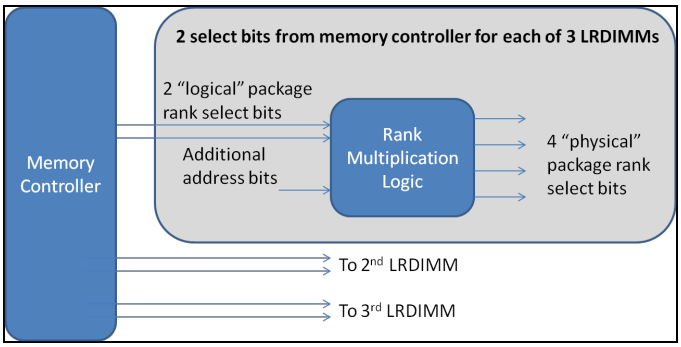


Figure 7: Rank Multiplication for 3 DIMMs per memory channel

As an example shown in Figure 8 for Gen1 DDR3 server platforms[4], the host controller was largely "rank unaware" when operating in a rank multiplication mode, required for 24 memory slot systems when fully populated with 32GB 4R LRDIMMs. This meant that such controllers were not able to send back-back read transactions to the same logical rank. This resulted in up to 25% penalty on the data bandwidth upon reads. When compared to an RDIMM solution on fully populated 24 slot systems operating at the same speed, LRDIMM provided only 70% of the memory bandwidth.

In Gen2 DDR3 server platforms[5], controllers became aware of the physical ranks behind the data buffer. They were able to request data from the memory far more efficiently. They also overcame some of the other limitations to improve the speed. So, Gen2 DDR3 server platforms achieved a speed improvement over Gen1 DDR3 server platforms, but more importantly closed the bandwidth gap with a corresponding RDIMM solution. The only remaining penalty on the DDR3 platforms was due to the component and trace length latencies mentioned above.

On DDR4, the distributed buffer architecture reduces the latency through each of the smaller data-buffers. This helps minimize the latency penalty through the buffer. Further, it allows for the memory controller to be able to hide the much lower latency in its micro-architecture.

LRDIMM Server Platform	LRDIMM rank awareness	Latency relative to RDIMM
Gen1 DDR3		**More latency**
Gen2 DDR3	√	**More latency**
Gen1 DDR4	√	Comparable latency

Summary

Eco-system improvements on DDR4 LRDIMM have continued the trend of significantly improving the memory bandwidth along with the channel speed. This is poised to increase the usefulness of LRDIMMs in comparison to RDIMMs across a wider array of applications, whether capacity intensive, or bandwidth intensive, or both. Figure 8 below summarizes the results of the improvements on successive enterprise server platform generations from actual experiments performed at IDT's validation lab. 3DPC at 1866 can potentially be realized using LRDIMMs while only 3DPC at 1600 can be realized using RDIMMs. Owning to these improvements, we expect some server manufacturers who have always configured their server platforms for speed to also consider 16GB LRDIMM as another reduced cost alternative to the higher capacity 32GB LRDIMM option. In essence, DDR4 LRDIMM is not just for capacity. It is capacity as well as bandwidth.

[4]Gen 1 DDR3, circa 2010

[5]Gen 2 DDR3, circa 2012

IDT results running server benchmarks confirm that the smaller Gen 1 DDR4 LRDIMM buffer latency translates into bandwidth comparable to Gen1 DDR4 RDIMM

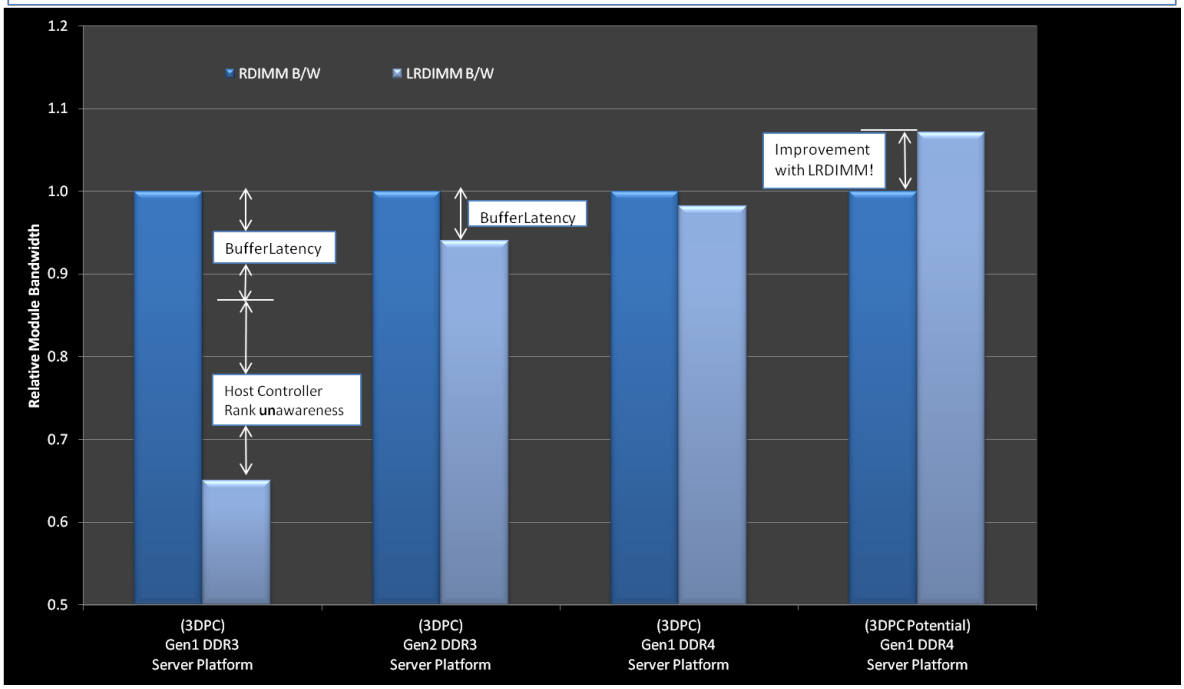


Figure 8: Bandwidth improvement normalized with RDIMM at the same speed with generations for memory controllers