



SoC Architecture to Multichannel Memory Management Using Sonics IMT

July 2008

**Phil Casini,
Vice President Marketing and Business Development
Sonics, Inc.**

Introduction

The high-definition video trend continues to drive new consumer products. These products are now moving to higher resolution, more sophisticated video compression, such as H.264, and improved image and scaling algorithms up to 120 Hz. This trend is affecting digital TVs, set-top-boxes, game consoles, and even mobile devices. There is clearly a new high quality, high definition (HQHD) segment forming which requires an exponential increase in SoC processing capability to support the more complex algorithms associated with HQHD.

Increasing processing requirements in turn means increasing the memory performance required to supply the processors the data they need in a timely manner. Memory performance requirements grow exponentially along with processing needs. For example, MPEG2 requires on the order of 2 gigabytes per second memory throughput. However, the state-of-the art today is dual-channel H.264, which requires in excess of 10 gigabytes per second to support HDNM (High-Definition Natural Motion), 120 Hz video scaling and advanced image quality.

This white paper focuses on identifying the inflection point in SoC architecture that is emerging as a result of these increased processing and memory performance requirements, and the economic value of transitioning SoC architecture from single to multiple DRAM channels, or multichannel memory management, using an innovative new Interleaved Multichannel Technology (IMT) from Sonics. IMT helps SoC developers avoid the inflection point by providing a seamless transition to multichannel memory management that operates transparently to hardware and software.

An Inflection Point is Emerging

Conventional reaction to increasing memory performance is to transition from DDR2, which is being used today to achieve HD performance, to DDR3, in order to achieve HQHD performance. Although alternative DRAM technologies exist, such as Enhanced DDR2, the history of DRAM economics indicate that DDR3 will be the predominant choice over time. **However, for the first time, an inconsistency arises between the optimal burst length required for DDR3 and the optimal data object sizes for the applications driving the higher bandwidth, which creates a new SoC architecture design challenge.**

The most popular microprocessors used for high-definition applications support 32 byte cache lines. Many video related protocols, such as H.264 macro blocks are also optimized around 32 bytes. However, inspection of DDR3 memories reveal that their optimal burst lengths are 64 bytes.

Conventional single-channel design is breaking down because the natural progression to DDR3 causes an architecture incompatibility between the optimal 64 byte burst sizes required for high DDR3 DRAM efficiency and the optimal access sizes required for processing efficiency, which are normally 32 bytes or less. DDR3 does not permit early

termination of bursts. Data object fetches smaller than the DRAM burst size waste data cycles, resulting in dramatic drops in external DRAM efficiency as unneeded words are accessed and then discarded.

Since bill-of-material costs are very tight, compensating for large losses in channel efficiency by adding more DRAMs is not cost effective.

Conventional reaction will be to make the DDR3 transition anyway and attempt to solve the efficiency loss problem other ways, such as raising the average access burst size. Existing caches can be changed and new cache levels introduced that have a 64 byte line size. But this represents new hardware. Software will also need new data structures to further optimize temporal and spatial locality around 64 byte blocks. New tiling algorithms are necessary to manage how data is stored. The development and performance testing of such algorithms are both time consuming and risky, and the area impact of local buffering they may require is equally uncertain.

Creating massive changes in hardware and software will drive project costs up, create significant risks through hardware and software development, and may still not yield the needed performance required for HQHD. SoCs now need another memory management model.

Transition to Multiple Shared DRAM Channels a Must

The inflection point is avoidable by migrating the SoC architecture from single to multiple DRAM channels. Expanding the number of DRAM channels on a device enables SoC developers to move to DDR3 and NOT forfeit channel efficiency. Splitting the traffic enables architects to double the accesses to DRAM while maintaining 32 byte burst lengths. Multichannel requires the same peak bandwidth as a single channel. Therefore, for the same number of DDR3 DRAMs used, the effective memory performance increases. Assuming DRAM economics favors DDR3, multichannel then becomes a more cost effective approach than single channel.

	DDR2	DDR3	DDR3
Channels	1	1	2
Data Width (B)	4	4	2
Effective BW	100%	84%	100%

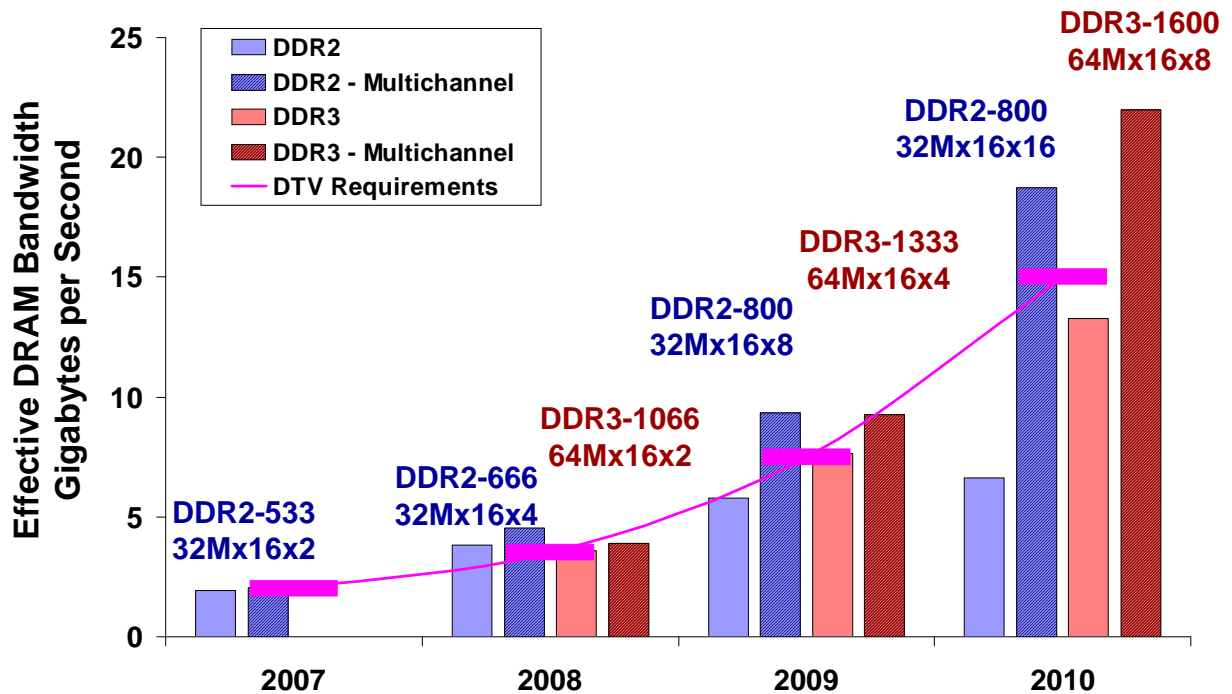
Source: Customer (HDTV) System Dataflow
Constant Frequency/Ideal Load Balancing

As an example, the table above represents a comparison of possible approaches to maintaining high memory efficiency including multichannel. The DDR2 table entry is used for reference as it is generally assumed that DRAM economics will favor DDR3 over DDR2. The second table, highlighting DDR3 conversion using a single channel, indicates a 16 percent loss of effective bandwidth by employing a conventional approach. This was measured by Sonics in an experiment whereby actual HDTV traffic was used to simulate a single DDR3 channel. The second part of the experiment was to split the same traffic into two channels, which is seen in table three. As anticipated, the channel bandwidth for this case was fully utilized.

What the table indicates is that multichannel does achieve memory efficiencies using DDR3 that are equal to memory channel efficiencies using a single channel DDR2 solution. Memory performance can now increase when transitioning from DDR2 to DDR3.

The assumption is that the two channels are well-balanced in terms of traffic throughput and the completion times to run the application were about equivalent as to not cause other problems within the overall application flow. The paper will address these assumptions later

The next step is to overlay the multichannel performance projections with the needs for HQHD. The graph below performs this mapping for HDTV assuming the same number of DDR3 devices are used. The multichannel approach provides clear advantage over single channel approaches, and is the only approach that will meet HQHD needs over time.



Now that multichannel has been established as the best approach the next question is how to implement multichannel.

Implementing Multichannel

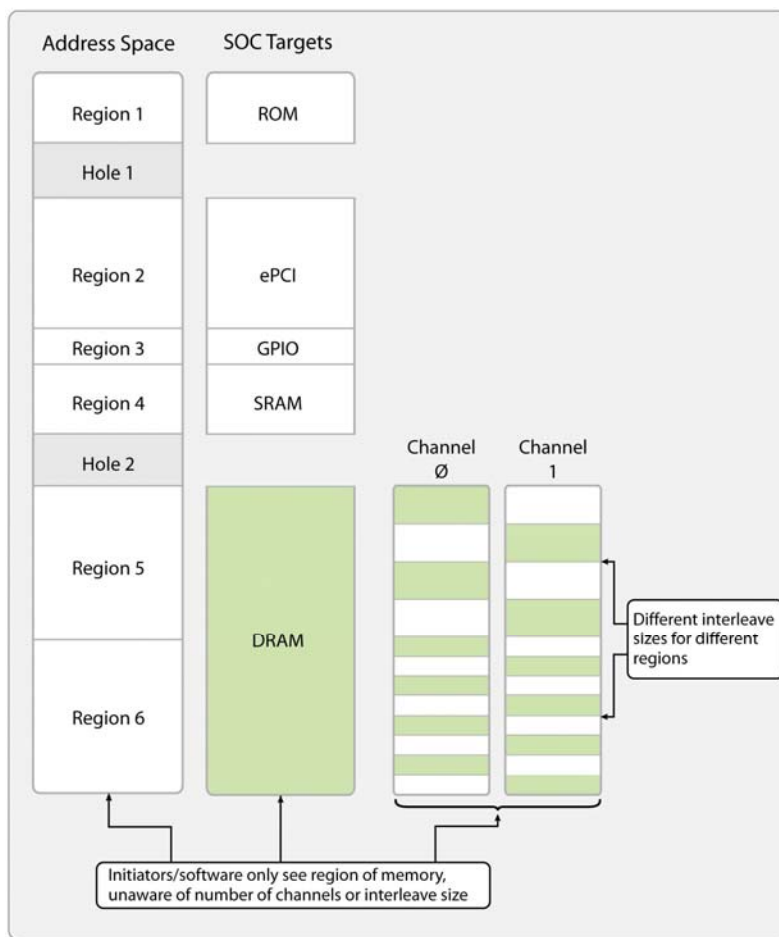
Most multichannel implementations today are simple two-channel schemes that rely on balancing traffic in the software or with some rudimentary hardware assistance to gain efficiencies. These are point solutions that are tightly coupled to the particular configuration and will need significant re-architecting for derivative products, or to add more channels. In addition, there is little predictability or scalability with such approaches. The verification burden to implement multichannel management in hardware introduces high project risks and potentially long design cycles.

However, interleaving the physical channels within the application's address space enables automatic statistical and algorithmic spreading to be employed that avoid "hot spots" in hardware AND enables the multichannel management to operate transparently to hardware and software.

Introducing IMT

The figure below depicts how Sonics IMT uses an innovative interleaving scheme to physically map multiple memory channels into the regions of the application's address map. This approach enables traffic within an address region to be split, transparently to hardware and software, for up to 8 DRAM channels, and then managed automatically for load balancing and low overall latency. IMT also manages the merging of the traffic and all ordering of the data such that no re-ordering buffers are required.

IMT offers the flexibility to balance the number of channels required today and the scaling of channels for generations to come. IMT is flexible and when coupled with a Sonics SMART Interconnect solution provides high performance.



By mapping all channels into existing regions, the resulting physical address map is identical, and no software or hardware changes are required. However, the ability to alter channels enables further performance optimizations if desired, taking advantage of more of the interleaving capabilities for certain applications.

IMT also enables channels to be added or removed in a programmable fashion and at boot time. This enables an SoC architect to create software profiles for a number of memory requirements such that a single SoC could serve multiple markets and price points, with functionality and external DRAM performance varying and controlled by the software.

Coupling IMT with a SMART Interconnect Solution

IMT is designed within the SMART Interconnect solution to enable a seamless transition for SoC architectures with high performance and low area and power implications.

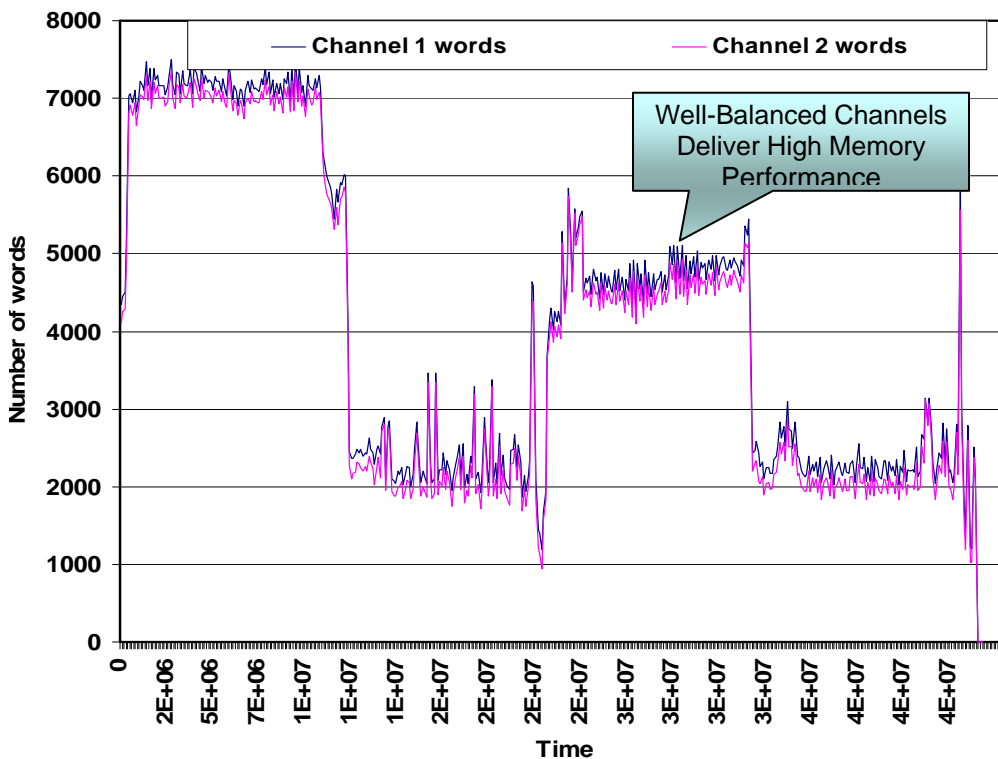
Multichannel interleaving requires intimate coordination with the traffic management of the interconnect, in order to remain optimal for every design. A key concern is where and how the traffic is split. Simple designs that are use-only channels might manage the load balancing in software. This will not scale when more than two channels are required.

IMT works well with the common infrastructure of SMART interconnect solutions to deliver a coordinated multichannel implementation. Splitting is performed close to the initiators to avoid latencies. IMT leverages the information already present for the network processing inherent to the SMART interconnect solutions architecture to help manage and load balance the traffic. The results of embedding IMT are high channel utilization with low latencies, yielding a predictable and scalable solution that occupies minimal overhead and consumes minimal power.

Having IMT embedded in SMART Interconnect solutions also allow SoC developers to take advantage of the advanced configuration and modeling automation when using SonicsStudio Development Environment. Once a configuration is determined, a push of the button outputs either a SystemC or RTL model that maps the multichannel choices into the existing address map as well as the entire data flow architecture for the SoC, fully optimized for performance, low power and minimal area. .

Through some advanced automation then, SoC architects can rapidly and cost effectively transition their architectures to multichannel.

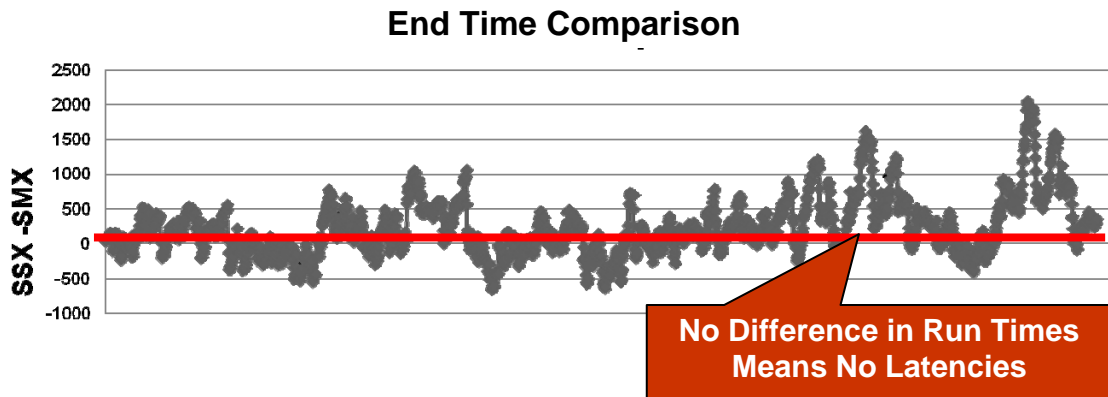
What are the Load Balancing Efficiency and Latency Impacts using IMT?



The graph above is a plot of actual production video traffic that was previously used for a single DRAM channel SoC. Sonics ran the identical traffic through IMT to produce the

results shown. The graph shows that the two IMT channels were extremely balanced.. What this picture indicates is that when utilizing IMT a relatively negligible loss in efficiency and achieved superior load balancing.

The second area of concern for multichannel approaches is making sure any overhead associated with implementing multiple channels does not impede significantly on the performance of the application. The chart below represents the delta in completion times when executing (worst case) video traffic using single and multichannel approaches. Note that this is a very conservative view, as the experiment chose to highlight the worst 200K cycles of the traffic. Depending how the traffic automatically distributes across the two channels, sometimes the access is completed more quickly, and sometimes more slowly (the gray curve). The graph clearly presents a strong picture that the difference in the completion times for the simulations is very low on average (the red curve), and remains so over time. This means IMT can be implemented into data flows with negligible impact on the overall performance of the application.



Adding Flexibility to the IMT-based SoC architecture

An important aspect of IMT is its flexible and programmable channel mapping. It includes the ability for multichannel groups to be partitioned into multiple multichannel groups; or to be partially populated; or to be fully populated with partial data path support. IMT has this built in programmability support to preserve SoC architecture while enabling product derivatives to take advantage of the technology with little engineering re-work. This translates into further amortization of the acquisition costs for IMT across an entire family of products, enabling more savings and lower risks from project to project.

IMT a Must for High-Performance SoC Architectures

The hard driving memory bandwidth requirements and the corresponding inflection point that emerges as a result of desires to move to DDR3 all point to IMT being a defacto standard “must have” technology for HQHD SoCs. Multichannel is the most practical and cost-effective method for increasing the memory performance and IMT is the most advanced multichannel technology available. IMT also represents a scalable and cost-

effective methodology to seamlessly transform from single channel to multiple channels, while maintaining architecture consistency and software transparency.

While this white paper focused on multichannel implementations that are two channels. It is quite likely that applications in the near future will require four or even eight channels. The load balancing and latency problems highlighted in this white paper are dramatically more complex as the numbers of channels grow. IMT has been designed for scalability enabling its software transparent scheme to be leveraged across up to eight channels.