Embedded DRAM Architectural Trade-Offs

Norbert Wehn

University of Kaiserslautern
Institute of Microelectronic Systems
Erwin-Schrödinger-Strasse
D-67663 Kaiserslautern, Germany
e-mail: wehn@e-technik.uni-kl.de

Søren Hein

Siemens AG
Semiconductor Group
Balanstraße 73
D-81617 München, Germany
e-mail: soeren.hein@hl.siemens.de

Abstract

In this paper we discuss system-related aspects in embedded DRAM/logic designs. We focus on large embedded memories which have to be implemented as DRAMs.

1 Motivation

The term "system-on-silicon" has been used to denote the integration of random logic, processor cores, SRAMs, ROMs, and analog components on the same die [1, 2]. But up to recently, one major component had been missing: high-density memories such as DRAMs. Today's integration densities are beginning to allow the integration of significant amounts of DRAM memory for applications such as data buffering, picture storage, and program/data storage. In quarter-micron technology, chips with up to 128 Mbit of DRAM and 500 kgates of logic, or 64 Mbit of DRAM and 1 Mgates of logic are feasible. This new possibility enlarges the system design space tremendeously since system architects are no more restricted to the use of standard commodity DRAMs [3].

The main characteristics of embedded DRAM[1] (eDRAM) can be summarized as follows [1, 4, 5]; note that '–' denotes either a challenge or an inherent disadvantage:

On-chip memory interface:

+ Replacing off-chip drivers with smaller on-chip drivers can reduce power consumption significantly, as large board wire capacitive loads are avoided. For instance, consider a system which needs a 4Gbyte/s bandwidth and a bus width of 256 bits. A memory system built with discrete SDRAMs (16-bit interface at 100 MHz) would require about ten times the power of an eDRAM with an internal 256-bit interface.

– Although the power consumption per system decreases, the power consumption per chip may increase. Therefore junction temperature may increase and DRAM retention time may decrease. However, it should be noted that DRAMs are low-power devices.

– Some sort of minimal external interface is still needed in order to test the embedded memory. The hybrid chip is neither a memory nor a logic chip; should it be tested on a memory or logic tester, or on both?

+ Embedded DRAMs can achieve much higher *fill frequencies* [6] than discrete SDRAMs.[2] This is because the on-chip interface can be up to 512 bits wide, whereas discrete SDRAMs are limited to 16-64 bits. Continuing the above example, it is possible to make a 4-Mbit eDRAM with a 256-bit interface. In contrast, it would take 16 discrete 4-Mbit chips (organized as 256K x 16) to achieve the same width, so the granularity of such a discrete system is 64 Mbit. But the application may only call for, say, 8 Mbit of memory.

+ As interface wire lengths can be optimized for the application in eDRAMs, lower propagation times and thus higher speeds are possible. In addition, noise immunity is enhanced.

System integration:

+ Higher system integration saves board space, packages, and pins, and yields better form factors.

+ Pad-limited design may be transformed into non-pad-limited ones by choosing an embedded solution.

[1] We speak of "embedded DRAM" or "embedded logic" depending on whether the master process is a logic or a memory process. Note that some authors use the terms in exactly the opposite way.

[2] The fill frequency of a memory is defined as the bandwidth in Mbit/s divided by the memory size in Mbit. In other words, the fill frequency is the number of times per second a given memory can be completely filled with new data.

– More expensive packages may be needed. Also, DRAMs and logic require different power supplies; currently the DRAM power supply (2.5V) is less than the logic power supply (3.3V), but this situation will reverse in the future due to the back-biasing problem in DRAMs.

– The eDRAM process adds another technology for which libraries must be developed and characterized, macros must be ported, and design flows must be tuned.

– DRAM transistors are optimized for low leakage currents, yielding low transistor performance, whereas logic transistors are optimized for high saturation currents, yielding high leakage currents. If a compromise is not acceptable, expensive extra manufacturing steps must be added.

– DRAM processes have fewer layers of metal than do logic processes. Layers can be added at the expense of process cost.

– DRAM fabs are optimized for long runs of identical products, for high capacity utilization and for high yield. Logic fabs, while sharing these goals, are slanted towards lower batch sizes and faster turnaround times.

Memory size:

+ Memory sizes can be customized.

– On the other hand, the system designer must know the exact memory requirement at the time of design. Later extensions are not possible, as there is no external memory interface.

– From the customer's point of view, the memory component goes from a commodity to a highly specialized part which may command premium pricing. As memory processes are quite different, second-sourcing problems abound.

In this paper we will discuss the architectural trade-offs which arise in eDRAM systems, touching upon technological and economic issues as appropriate.

2 Applications of Embedded DRAM

Due to the complexity of the advantages, disadvantages and challenges of eDRAM described above, it is not possible to give a simple formula for the advisability of eDRAM in a specific project. However, some rules of thumb can be given:

- The product volume and product lifetime are usually high.
- Either the memory content is high enough to justify the higher DRAM process costs, or eDRAM is required for bandwidth or other reasons.
- Other things being equal, eDRAM will find its way first into portable applications.

The market for eDRAM has been estimated at 100 – 200m in 1997, rising to 4 – 8bn in 2001. Embedded DRAM has already conquered a large part of the market for 3D graphics accelerator chips for laptops; in this segment, the advantages of lower power consumption and higher performance cannot be ignored. Embedded DRAM is also slated to conquer a large part of the desktop PC and games market for graphics chips in the next few years. Memory sizes of 32-64 Mbit are likely to be required, mainly for frame storage.

The three other main markets for eDRAM are likely to be controllers for hard-disk drives, controllers for printers, and network switches. The first two of these markets are driven mainly by system cost; the products contain embedded processors, and the memory is used for storage of programs as well as data. Memory requirements are more modest than for graphics controllers, both in terms of size and bandwidth. Examples of embedded processors with embedded DRAM are given in [7, 8, 9].

Network switching is the high-end market for eDRAM: memory sizes of up to 128 Mbit and interface widths up to 512 are required for reading and writing data packets out of large buffers. As switches are not consumer products, the number of chips sold is lower than in the other markets, but higher prices partially compensate for this.

Several other markets are possible for eDRAM, including mobile phones, personal digital assistants (PDAs), *etc.* However, it is unlikely that eDRAM will capture the PC market for main memory, as the need for flexibility and an upgrade path is too strong.

3 The Memory Design Space

An eDRAM designer faces a design space which contains a number of dimensions not found in standard ASICs, some of which we will subsequently review.

The designer has to choose from a wide variety of memory cell technologies which differ in the number of transistors and in performance.

Also, both a DRAM technology and a logic technology can serve as a starting point for embedding DRAM. Choosing a DRAM technology as the base technology will result in high memory densities but suboptimal logic performance. On the other hand, starting from a logic technology will result in poor memory densities, but fast logic. To some extent, one can therefore trade logic speed against logic area. Finally, it is also possible to develop a process

that gives the best of both worlds, most likely at higher expense.

Furthermore the designer can trade logic area for memory area in a way heretofore impossible.

Large memories can be organized in very different ways. Free parameters are number of memory banks, which allow the opening of different pages at the same time, the length of a single page, the wordwidth and the interface organization.

Since eDRAM allows to integrate SRAMs and DRAMs, decisions on the on/off-chip DRAM- and SRAM/DRAM-partitioning have to be made.

Especially the following problems have to be solved at the system level:

- Optimizing the memory allocation.
- Optimizing the mapping of the data into memory such that the sustainable memory bandwidth approaches the peak bandwidth.
- Optimizing the access scheme to minimize the latency for the memory clients and thus minimize the necessary FIFO depth.

The goals are to some extent independent of whether or not the memory is embedded. However, the number of free parameters available to the system designer is much larger in an embedded solution, and the possibility of approaching the optimal solution correspondingly greater. On the other hand, the complexity is also increased. It is therefore incumbent upon eDRAM suppliers to make the trade-offs transparent and to quantize the design space into a set of understandable if slightly sub-optimal solutions.

4 DRAM Performance

In the past the row and column access times in a DRAM core have declined by roughly only 10%/year whereas the peak device memory bandwidth has increased over the last couple of years by two orders of magnitude [10]. This was achieved by:

- intelligent synchronous interfacing and protocols;
- exploiting the fact that an active row can act as a cache. In some memory structures additional row caches are even implemented on the memory device;
- using prefetching and pipelinig techniques; and
- using multiple internal memory banks.

Hence large commodity memory devices with several Gbit/s of peak memory bandwidth are available today [10]. However:

- The size of PC memory systems has grown by only half the rate of single DRAM devices for many years. As the growth of bandwidth requirements has kept pace with that of the memory systems, the interface width of DRAMs should thus have been growing as fast as the size of single DRAM devices. This has not happened for packaging reasons. Instead granularity has decreased, often inducing unnecessary but unavoidable extra memory.
- The increased bandwidth must be paid with increased latencies and burst lengths.
- The peak bandwidth is a theoretical quantity; in practive several memory clients have to read and write data which introduces page misses and overhead. Hence the sustainable bandwidth can be much lower than the peak bandwidth.

From a system designer's point of view embedded DRAM offers the possibility to avoid these drawbacks by:

- Adapting the memory size to the exact system requirements;
- Adapting the memory bus interface to the system requirements;
- Optimizing the memory structure (page length, number of banks, word width) to system requirements.

Let us discuss two examples to illustrate issues related to memory granularity and bandwidth: an MPEG2 video decoder [11] and the so-called "Processor-Memory Performance Gap."

4.1 MPEG2 Video Decoder

The majority of MPEG2 video decoders are tuned to work with 16 Mbits of memory as this is a standard size. In fact the MPEG standardization group expressly modified the standard to make 16 Mbits sufficient for decoding both the PAL and NTSC video formats (by disallowing a certain flag for so-called PAL B-pictures). MPEG2 decoder manufacturers relying on external DRAM would otherwise have had to move to 20 Mbits (4×4 Mbit) or 32 Mbits (2×16 Mbit).

An MPEG2 video decoding pipeline contains three large memory blocks: an input buffer for storing the incoming compressed data stream, two full frame buffers for bidirectional picture reconstruction, and an output buffer for progressive-to-interlaced conversion. Memory can be saved only in the output buffer. Specifically, about 3 Mbit can be saved at the expense of doubling the throughput of the decoding pipeline as well as the memory bandwidth of the motion compensation module. However, adequate memories of sizes smaller than 16 Mbits are not available

(three 4-Mbit memories are insufficient), and if they were available, they would not be able to provide the minimum required bandwidth. Here eDRAM comes to the rescue.

In general, video applications need a large amount of memory and bandwidth to store and access frames. Standard commodity sizes are usually not a multiple of the frame memory size: a PAL frame, for example, in 4:2:0 format needs 4.75 Mbit, whereas an NTSC frame requires 3.96 Mbit. In these cases eDRAM enables implementations with minimum overhead.

4.2 The Processor-Memory Performance Gap

There is an increasing gap between processor and DRAM speed: processor performance increases by 60% per year in contrast to only a 10% improvement in the DRAM core. Deep cache structures are used to alleviate this problem, albeit at the cost of increased latency which limits the performance of many applications. Merging a microprocessor with DRAM can reduce the latency by a factor of 5-10, increase the bandwidth by a factor of 50 to 100 and improve the energy efficiency by a factor of 2 to 4 [9].

Developing memory is a time-consuming task and cannot be compared with a high-level based logic design methodology which allows fast design cycles. Thus a flexible memory concept is a prerequisite for a successful application of eDRAM. Its purpose is to allow fast construction of application-specific memory blocks which are customized in terms of bandwidth, word width, memory size and the number of memory banks, while guaranteeing first-time-right designs accompanied by all views, test programs, *etc.*

5 A Flexible Embedded DRAM Concept

In this section we describe a powerful eDRAM concept [12] which permits fast and safe development of embedded memory modules. The concept, developed by Siemens for its customers, uses a 0.24μm technology based on its 64/256Mbit SDRAM process. Key features of the concept include:

- Two building-block sizes, 256 Kbit and 1 Mbit. Memory modules with these granularities can be constructed.
- Large memory modules, from 8-16 Mbit upwards, achieving an area efficiency of about 1 Mbit/mm^2.
- Embedded memory sizes up to at least 128 Mbits.
- Interface widths ranging from 16 to 512 bits per module.
- Flexibility in the number of banks as well as the page length implemented.
- Different redundancy levels, in order to optimize the yield of the memory module to the specific chip.
- Cycle times better than 7 ns, corresponding to clock frequencies better than 143 MHz.
- A maximum bandwidth per module of about 9 Gbyte/s.
- A small, synthesizable BIST controller for the memory (see next section).
- Test programs, generated in a modular fashion.

Siemens has made eDRAM since 1989 and has a number of possible applications of its eDRAM concept in the pipeline, including TV scan-rate converters, TV picture-in-picture chips, modems, speech-processing chips, hard-disk drive controllers, graphics controllers, and networking switches. These applications cover the full range of memory sizes (from a few Mbits to 128 Mbits), interface widths (from 32 to 512 bits), and clock frequencies (from 50 to 150 MHz), which demonstrates the versatility of the concept.

6 Testing Aspects

Testing DRAMs is very different from testing logic. In the following, the main points of notice are discussed:

- The fault models of DRAMs explicitly tested for are much richer; they include bit-line and word-line failures, cross-talk, retention time failures *etc.*
- The test patterns and tester equipment are correspondingly highly specialized and complex. As DRAM test programs include a lot of waiting, DRAM test times are quite high, and test costs are a significant fraction of total cost.
- As DRAMs include redundancy, the order of testing is (1) pre-fuse testing, (2) fuse blowing, (3) post-fuse testing. There are thus two wafer-level tests.

The implication on eDRAMs is that a high degree of parallelism is required in order to reduce test costs. This necessitates on-chip manipulation and compression of test data in order to reduce the off-chip interface width. For instance, Siemens offers a synthesizable test controller supporting algorithmic test pattern generation (ATPG) and expected-value comparison (partial BIST).

Another important aspect of eDRAM testing is the target quality and reliability. If eDRAM is used for graphics applications, occasional “soft” problems, such as too short retention times of a few cells, are much more acceptable

than if eDRAM is used for program data. The test concept should take this cost-reduction potential into account, ideally in conjunction with the redundancy concept.

A final aspect is that a number of business models are common in eDRAM, from foundry business to ASIC-type business. The test concept should thus support testing the memory either from a logic tester or a memory tester, so that the customer can to memory testing on his logic tester if required.

7 Conclusion

Embedded DRAM opens new possibilities for system designers which go far beyond the simple integration of logic and standard commodity memory. Parameters of commodity DRAMs which designers have been forced to take for given, including size, interface width, and organization, are now available as design parameters. In addition, trade-offs between logic and memory are possible.

Designers and chip architects must exploit these new degrees of freedom to develop new system solutions. Flexible memory concepts are necessary to allow fast implementation. Memory allocation, appropriate access schemes and data/memory mappings are keys to optimizing the overall system architecture. New design-for-testability techniques are necessary which consider the special need of memory testing. Furthermore the transistor-oriented memory and high-level based design methodology must be merged in the physical design phase.

Acknowledgments

Thanks to our colleagues from the University of Kaiserslautern and Siemens AG.

References

[1] J. Borel. Technologies for Multimedia Systems on a Chip. In *1997 International Solid State Circuits Conference, Digest of Technical Papers*, volume 40, pages 18–21, February 1997.

[2] H. De Man. Education for the Deep Submicron Age: Business as Usual? In *Proceedings of the 34th Design Automation Conference*, pages 307–312, June 1997.

[3] O. Kimura (Moderator). DRAM + Logic Integration: Which Architecture and Fabrication Process. Evening Discussion Session at the 1997 International Solid State Circuits Conference, February 1997.

[4] N. Dutt (organizer). How will Memory Issues Impact Synthesis for Embedded Systems-on-Silicon? Panel Discussion at the 10th International Symposium on System Synthesis, September 1997.

[5] R. Salters. Embedded memories and embedded logic. Invited Talk at the 10th International Symposium on System Synthesis, September 1997.

[6] S. A. Przybylski. *New DRAM Technologies: a comprehensive analysis of the new architectures.* Report, 1996.

[7] K. Murakami, S. Shirakawa, and H. Miyajima. Parallel Processing RAM Chip with 256Mb DRAM and Quad Processors. In *1997 International Solid State Circuits Conference, Digest of Technical Papers*, volume 40, page 228, February 1997.

[8] T. Shimizu et al. A Multimedia 32b RISC Microprocessor with 16 Mb DRAM. In *1996 International Solid State Circuits Conference, Digest of Technical Papers*, volume 39, pages 216–217, February 1996.

[9] D. Patterson et al. Intelligent RAM (IRAM): Chips that Remember and Compute. In *1997 International Solid State Circuits Conference, Digest of Technical Papers*, volume 40, pages 224–225, February 1997.

[10] B. Prince. *High Performance Memories.* John Wiley & Sons, 1996.

[11] S. Hein and N. Wehn. Design of Multimedia Systems: a Case Study - Anatomy of an MPEG2 Decoder. In *Hardware Software Codesign for Telecom and Multimedia Systems*. Tutorial, 33rd Design Automation Conference, Las Vegas, June 1996.

[12] SIEMENS Semiconductor Homepage. `http://www.siemens.de/Semiconductor`, 1997.