Challenges in Embedded Memory Design and Test

Erik Jan Marinissen
Philips Research Labs
Prof. Holstlaan 4 – WAY-41
5656 AA Eindhoven
The Netherlands
erik.jan.marinissen@philips.com

Betty Prince
Memory Strategies International
16900 Stockton Drive
Leander, TX 78641
United States of America
bprince@memorystrategies.com

Doris Keitel-Schulz
Infineon Technologies
P.O. Box 800949
81609 Munich
Germany
doris.keitel-schulz@infineon.com

Yervant Zorian
Virage Logic
47100 Bayside Parkway
Fremont, CA 94538
United States of America
yervant.zorian@viragelogic.com

Abstract

Both the number of embedded memories, as well as the total embedded memory content in our chips is growing steadily. Time for chip designers, EDA makers, and test engineers to update their knowledge on memories. This *Hot Topic* paper provides an embedded tutorial on embedded memories, in terms of what is new and coming versus what is old and vanishing, and what are the associated design, test, and repair challenges related to using embedded memories.

1 The Ideal Memory - Yesterday, Today and Tomorrow

Betty Prince – Memory Strategies International

The concept of the 'ideal' memory has changed over the past thirty years from a few high-volume stand-alone standardized MOS memory part types, each with its own manufacturing technology, to embedded memories in CMOS logic processes to potential universal memory technology for use in numerous instances in multimillion transistor logic chips.

1.1 The Stand-Alone Standard Memory Era

From about 1980 to 1990, the ideal MOS memory was a standardized stand-alone part [1]. It had small cell size, good array efficiency, adequate performance, noise and soft error resistance, and met an external I/O standard. A few product types ran in high volume in specialized memory wafer fabs. No single type had all the characteristics of the ideal memory. Three memory types, SRAMs, DRAMs, and Flash EEPROMs, were used in different applications. See Table 1.

The fast stand-alone four-transistor (4T) SRAM cell had two stacked poly load resistors to reduce cell size. Both the DRAM and the Flash used processes that diverged significantly from CMOS logic. DRAMs by 1990 had gone from planar to vertical capacitors – both stacked and trench. Flash memories had double polysilicon floating gates and used several programming mechanisms which various cell types used in different combinations for different application criteria. Flash write was slow and write endurance was limited. Charge pumps were on chip for the high voltage programming.

DRAMs with their small cell size and high density were used in large memory systems where their slower speed was compensated for by fast SRAM cache. These asynchronous DRAMs had a high power consumption associated with charging the high capacitance bit lines in precharge cycles and during refresh. 4T NMOS SRAMs were used in high-speed cache, while 6T CMOS SRAMs, due to ease of use, wide noise margin and low standby power, were used in hand-held systems. Neither application required high density, so the large cell size and high cost of the SRAM was not a problem. Flash memories were used in non-volatile applications such as the BIOS for program code storage in computers.

1.2 The Memory Integrated with Logic Era

A second phase of memory development occurred from 1990 to 2000 [2]. Memories began to have significant amounts of logic integrated onto the chip. Some embedded DRAM and Flash appeared, but were hindered by the historical divergence of the memory and logic technologies. This shift to adding logic on the memory chip was driven by several factors. Submicron geometries both increased logic speed requiring faster memories and reduced cell size providing room for on-chip logic. Lowered power supply voltage made the poly-load 4T SRAMs unstable.

Proceedings of the Design, Automation and Test in Europe Conference and Exhibition (DATE'05)
1530-1591/05 $ 20.00 IEEE

	SRAM		DRAM		Flash/EEPROM	
	1980-1990	1990-2000	1980-1990	1990-2000	1980-1990	1990-2000
Read speed	fast (ns)	faster (ns)	moderate (ns)	fast (ns)	moderate (ns)	moderate (ns)
Write speed	fast (ns)	faster (ns)	moderate (ns)	fast (ns)	very slow (ms,s)	very slow (ms,s)
Non-volatile	no	no	no	no	yes	yes
Cell size	4	6	1.5	1.5	1	1
Cell type	NMOS	CMOS	planar	vertical	NOR	NOR & NAND
Density	low	low	high	high	high	high
Supply voltage	5 V	3.3/2.5 V	5 V	3.3/2.5 V	5 V	3.3/2.5 V
Write voltage	5 V	3.3/2.5 V	5 V	3.3/2.5 V	18 V	12 V
Mask adders	none	none	n.a.	8–11	n.a.	8–11
Standardization	I/O	spec.	I/O	spec.	I/O	spec.
Application	cache	PDA	server	PC	BIOS	cell phone

Table 1: Stand-alone MOS memory characteristics – The first 20 years.

Fast standardized SRAMs and DRAMs acquired a synchronous interface, making them digital state machines. Behind the synchronous interface, multiple banks were integrated on the chip increasing speed but reducing array efficiency. Circuits such as Delay-Locked Loop (DLL) and Error-Correcting Circuitry (ECC), and wide buses for multi-word prefetch, were moved behind the synchronous interface. The small DRAM cell size permitted densities up to 1 Gbit, while the larger cell size of the 6T SRAM kept it around 16 Mbit in size. Refresh was moved onto the DRAM chip and with the addition of a pseudo-SRAM interface, the high density P-SRAM (DRAM) became capable of functioning in an SRAM socket. SRAMs, already made in a logic process, began to move onto the logic chip. The low voltage and high speed resulted in lower SRAM transistor thresholds affecting stability and increasing standby power. DRAMs had only a high threshold transistor in the array, so the array leakage was low and high cell capacitance permitted long refresh cycles, which reduced the average standby current. As a result, the P-SRAM (DRAM) had lower standby power than a comparable density SRAM. The reduced stability of the SRAM made it more susceptible to soft errors while the high speed requirement meant that increasing the capacitance of the storage node was not an option in most applications. Flash memory meanwhile began to be required for high density data storage as well as code storage and the higher density NAND flash was developed to fill this application. Applications such as computer and cell phone operating systems continued to use the faster random access NOR flash cell. High voltage programming was moved onto the flash chip and voltage down converters provided the lower voltage standard I/Os.

1.3 The Scaled Embedded Memory Era

From 2000 to 2005, the era of true embedded memory has begun [3]. The capability of integrating 100's of millions of gates and cells on the chip means that large subsystem sections are being integrated and memory is becoming a large part of a chip which is functionally not a memory.

Characteristics of embedded memory are different from stand-alone memory. Wide on-chip buses and parallelism make high speed operation less essential for high bandwidth. Multiple banks of memory and multiple on-chip processors permit even higher bandwidth. Power is reduced by integration of fast I/Os, by segmenting high capacitance lines, and by clocking techniques. Boundary scan (JTAG), Built-In Self Test (BIST) and Built-In Self Repair (BISR) bring test on chip and ECC on chip reduces soft error problems. Commercial IP becomes the on-chip equivalent of memory standardization.

An important criterion now is compatibility with the CMOS logic process. Specialized memory processes, which increase the cost of the logic chip, are not readily accepted. Planar DRAM cells and single polysilicon flash memory cells, which do not add process steps or masks to CMOS logic, but do result in larger cell size, are available from several foundries. The memory macro can be customized for the system rather than needing to be standardized. See Table 2.

	SRAM	DRAM	Flash/EEPROM
Random read speed	very fast	fast	moderate
Soft error	ECC	ECC	n.a.
Interface	custom	custom	custom
Mask adders to CMOS	none	0–8	0–11
Supply/write voltage	1.8/1.1 V	1.8/1.1 V	1.8/1.1 V
Verified macro IP	yes	yes	yes
Test	JTAG, BIST	BIST, BISR	external

Table 2: Embedded MOS memory characteristics – The current era.

Scaled Flash and DRAM substitutes have been developed. As scaled floating gate Flash becomes more difficult to make reliably, some in the non-volatile industry have moved to a nitride storage technology called SONOS (or

MONOS) and others are studying the possibility of breaking the floating gate up into many smaller gates called 'silicon nanocrystals'. Both options could scale with the Flash technology and be embedded with fewer added steps in the CMOS logic process without using exotic materials. Potential scaled DRAM substitutes include several novel gain transistors and a single transistor capacitorless DRAM structure. See Figure 1 for cell size trends.

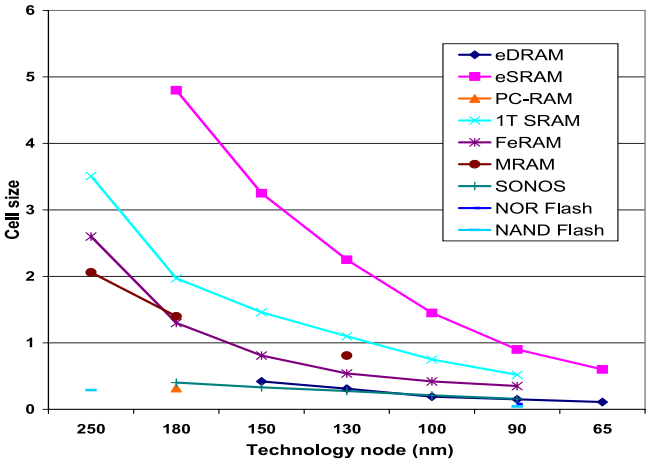


Figure 1: Cell size vs. technology for embedded memories [4].

The CMOS logic process has also changed. Copper wires and new dielectric technologies have entered the mainstream. Materials once considered exotic in the wafer fab can now be handled. A growing intolerance of multiple different memories in the system or chip has developed into a renewed search for a universal memory technology which is now defined as a single RAM type with: fast read/write, low voltage operation, non-volatility, infinite endurance, high reliability, compatibility with the CMOS logic process, and low power consumption. Candidates include: Magnetic RAM (MRAM), Ferro-electric RAM (FeRAM), and chalcogenide memory. See Table 3 for some mask adder comparisons.

Base Process	+ Masks for Memory	Memory Type
Conventional Memory		
CMOS logic	+4 – +8	DRAM/P-SRAM
CMOS logic	+4 – +11	Flash (floating gate)
Scaled Memory		
CMOS logic	+3 – +4	SONOS/MONOS
CMOS logic	+4	Silicon nanocrystal Flash
Emerging Memory		
CMOS logic	+3	MRAM
CMOS logic	+2	FeRAM
CMOS logic	+4	Chalcogenide

Table 3: Mask layers added to CMOS logic for various embedded memory types.

2 Embedded Memories – The Product Perspective

Doris Keitel-Schulz – Infineon Technologies

2.1 Trends in SOC Design

Memory content in SOC was increasing dramatically over the last years. In 2010, about 90% of the silicon area will consist of memories with different functionality as shown in Figure 2.

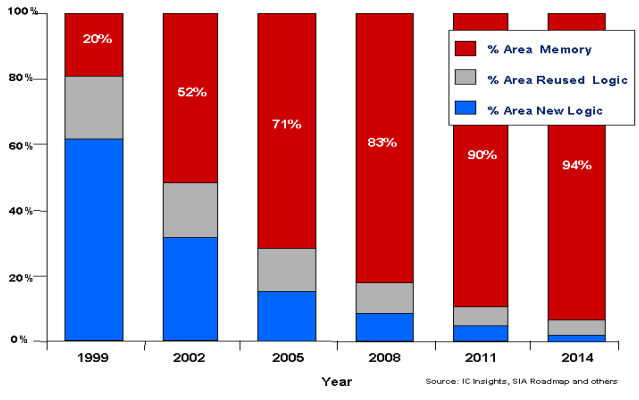


Figure 2: Memory contents in ASIC type SOC.

These memories will be either large junks of 'hand-optimized' blocks or smaller memories generated by compilers. Both, the hand-optimized memories and the compiled memories are validated on test sites before they are released for general use in productive designs [5]. The portion of reused logic blocks will already exceed the portion of newly created functions significantly by 2008. The main reason for reuse of logic blocks and also memories is design stability to achieve first-pass designs and thus meeting the required market window. In addition, the pure 'material cost' for designs in technologies below 100 nm are becoming more and more fundamental as mask costs alone already today exceed one million dollar.

Figure 3 shows possible memory and logic partitions for different technology nodes. Already now, 20 Mb of SRAM can be integrated easily with 6 million logic gates in productive designs. For 50 nm nodes, the portion of memory can go up to more than 100 Mb of highly optimized SRAM and more than 12 million logic gates. By now, typical devices in production include several mega-bit of

memory which consists of more than 100 different memory tiles. Concerning these memories, we find everything from large caches to small local storage elements in various flavors, like high speed, low power, and ultra low power [2]. In addition, different functions and architectures like ROM, scratch-pad SRAM, single-port RAM, multi-port RAM, and CAM are available.

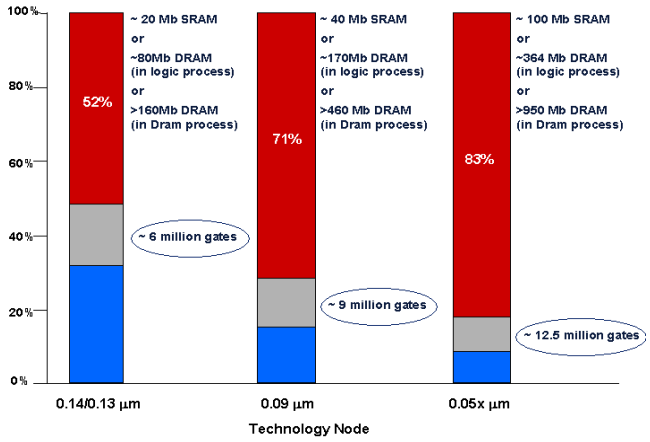


Figure 3: Possible usage of a core area of 120 mm^2.

2.2 Advantages and Challenges to Embed Memories

What justifies embedding all these memories into a device? There can be huge advantages if the different memories are exactly tailored to the specific needs in the design. These advantages are mainly

- improved performance,
- lower power consumption,
- on-demand memory activation with refined stand-by modes,
- exact granularity and organization,
- higher possible bandwidth and lower power than using external devices,
- general form factor and board space advantages,
- package cost reduction, and
- overall cost.

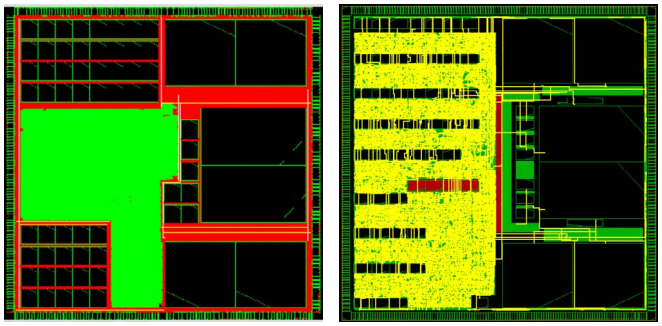
On the other hand, there are also challenges and even possible drawbacks compared to a multi-device or multi-chip solution, like

- development complexity increases,
- memories need to be area and yield optimized,
- new methodologies during the design phase have to be introduced/in place,
- new methodologies for analysis and testing have to be in place [6],
- higher process complexity if additional mask layers are necessary,
- higher mask cost,
- overall test cost and tester requirements have to be taken into account,
- effort for ramping one complex product,
- decreased flexibility and extendibility,
- flexible redundancy concepts necessary, and
- yield limitations.

2.3 Implementation of Embedded Memory Devices

Looking at the above, the real big challenge is how to partition the system and thus implement a cost-optimized system solution in time [7]. Figure 4 shows a communication device which first was built out of several memories and one logic device. The required features for the integrated device have been

- doubling the ‘functionality’,
- performance gain of around 30% compared to the multi device solution, and
- optimization of power, that the device still fits into the former package.



(a) Conventional approach (b) Optimized approach

Figure 4: Methodology development for embedded memory devices.

Without embedding all the memories, there was no solution. Multi chip failed due to power reasons, as the off-chip driver for the three larger memories would have absorbed already 30% of the available power budget. Integrating SRAMs and even DRAM was the only viable solution. During implementation, the limits of typical design methodologies were clearly discovered. The conventional approach used a typical floor plan which separates the logic and the memory areas and used generated standard SRAMs. In the optimized approach, significant work was done to understand and improve the data flow through the memories and logic and to strip down all features of the compiled SRAMs which have not been necessary. Thus, the increase in performance could be achieved and additional power saving was possible, due to optimized routing and SRAM power consumption. As the methodology for such an approach is not implemented in typical CAD tools and flows, some parallel work was necessary during the design implementation phase to reach this result [8]. The conclusion from this and other design projects clearly is, that the optimization potential using embedded memories is tremendous and makes sense, if the functionality of the device benefits significantly.

3 Embedded Memory Test and Repair – Trends and Challenges

Yervant Zorian – Virage Logic

Today's SoCs are moving from logic-dominant chips to memory-dominant ones. Today, usage of embedded memories is more than half of the die area for a typical SoC.

3.1 Embedded Memory Yield

Embedding large number of memory bits per chip creates a more powerful SoC that adapts better to today's memory-hungry applications. But it brings with it the problem of large die size and poor yields. Because embedded memories are designed with aggressive design rules, they tend to be more prone to manufacturing defects and field reliability problems than any other cores on the chip. The overall yield of an SoC relies heavily on the memory yield. Hence, securing high memory yield is critical to achieving lower-cost silicon. Traditionally, embedded memories were made testable, but not repairable. Similar to stand-alone memories, yield improvement can be obtained by offering redundancy in embedded memories, i.e., spare elements. Determining the adequate type and amount of redundant elements for a given memory requires both memory design knowledge and failure history information for the process node under consideration [9]. While this is a challenge by itself, providing the right redundant elements does not solve the whole problem. The know-how of how to detect and locate the defects in a memory, and how to allocate the redundant elements requires manufacturing know-how in terms of defect distributions [10]. In order to optimize yield, one needs to utilize test and repair algorithms that contain this know-how, see Figure 5.

In addition to manufacturing repair via redundancy, embedded memories often contribute to a solution for another major challenge, namely the process yield improvement. New systematic defects are often manifested as yield-limiting faults resulting from shrinking geometries and introduction of new material into the fabrication process. To discover the root causes of the yield-limiting factors, adequate diagnosis and failure analysis is needed and accordingly process improvement steps are performed.

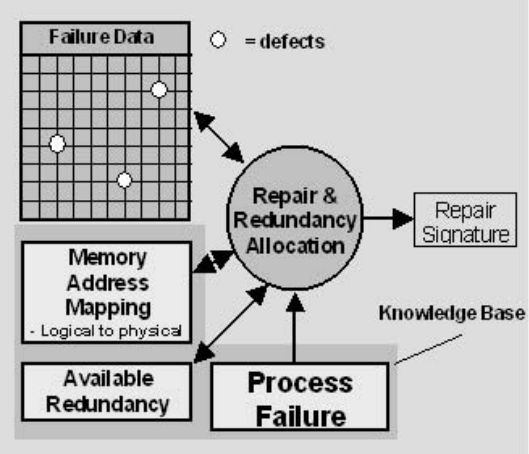


Figure 5: Redundancy allocation process.

Furthermore, the very deep submicron technologies made devices more susceptible to a range of post manufacturing reliability failures. This challenge may be resolved by allowing periodic field-level repair and power-up soft repair for the embedded memories. This type of repair may utilize the remaining redundant elements, if any.

3.2 Manufacturing Cost

The traditional approach to perform memory repair is using external test and repair methods. Since these external test and repair methods rely on extensive use of equipment, this constitutes as much as 40 % of the overall manufactur-

ing cost of a semiconductor chip. Therefore, keeping these expenses down is key to lowering the cost of manufacturing. This is especially important for high-volume consumer electronics or networking products and any cost-sensitive applications.

3.3 Time-to-Volume

The continuous increase in SoC complexity and the simultaneous increase in time-to-market pressure force SoC providers to look into volume production as the most critical challenge. Time-to-Volume (TTV) is comprised of two periods [11]: SoC design time and production ramp-up time. Reducing SoC design time has been a topic of discussion for a long time. Reusing pre-designed F-IP cores and ensuring ease of integration (interoperability) is a viable way to address the growing SoC design gap. Obtaining embedded memories from IP providers is a common practice today. However, this is not sufficient to ensure a minimal SoC design time. One has also to obtain all the necessary views and models of a given memory to simplify SoC integration and minimize design time. Traditionally, the yield optimization is done during the ramp-up period, i.e., following the design stage. During ramp-up, the yield problems are detected, analyzed, and corrected. As a result, the yield slowly ramps up to a mature level, after which, usually, the volume production starts.

Due to time-to-market pressure, the ramp-up period of an SoC may start before achieving the traditional defect densities, and hence prior to reaching the corresponding yield maturity levels. Because the yield is not sufficiently mature and the SoC is typically complex, the traditional ramp-up can take considerably longer. However, the ramp-up period can be reduced, and hence the TTV, if the yield optimization effort starts before the ramp-up period. This can be realized for embedded memories, if a memory IP provider performs the yield optimization effort at the IP design and characterization phases, prior to SoC ramp-up.

- First: by fabricating memory IP test chips, characterizing them, and applying knowledge from the fabrication process to improve the yield of the memory IP block. This results in silicon-proven IP before the SoC production ramp-up starts.
- Second: by designing into the memory IP and the SoC all necessary memory repair functions in advance. Using this memory repair augments the volume of fault-free SoCs, and hence simplifies the ramp-up effort.
- Third: by designing into the memory IP all necessary diagnosis and failure analysis functions based on which to perform process improvement during ramp up period.

In summary, to address these three challenges, today's embedded memories require solutions capable of addressing the yield and reliability needs: repair at manufacturing level; diagnosis for process improvement; and field repair capabilities. At the same time, these solutions need to minimize the manufacturing cost and reduce TTV.

References

[1] B. Prince. *Semiconductor Memories: A Handbook of Design, Manufacture and Application*. John Wiley & Sons, New York, NY, USA, 2nd edition, 1992.

[2] B. Prince. *High Performance Memories: New Architecture DRAMs and SRAMs – Evolution and Function*. John Wiley & Sons, New York, NY, USA, 1999.

[3] B. Prince. *Emerging Memories: Technologies and Trends*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002.

[4] B. Prince. *Emerging Memories: Applications Device and Technology*. Memory Strategies International, Leander, TX, USA, 2004.

[5] S. Iyer and H. Kalter. Embedded DRAM Technology: Opportunities and Challenges. *IEEE Spectrum*, 36(4):56–64, April 1999.

[6] A. Shubat. Perspectives: Moving the Market to Embedded Memory. *IEEE Design & Test of Computers*, 18(3):5–6, May-June 2001.

[7] D. Keitel-Schulz and N. Wehn. Embedded DRAM Development: Technology, Physical Design, and Application Issues. *IEEE Design & Test of Computers*, 18(3):7–15, May-June 2001.

[8] F. Catthoor et al. *Custom Memory Management Methodology*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998.

[9] S. Shoukourian, V. Vardanian, and Y. Zorian. An Approach for Evaluation of Redundancy Analysis Algorithms. In *Proceedings IEEE Intnl. Workshop on Memory Technology, Design, and Testing (MTDT)*, pages 51–55, 2001.

[10] J. Segal et al. Determining Redundancy Requirements for Memory Arrays with Critical Area Analysis. In *Proceedings IEEE Intnl. Workshop on Memory Technology, Design, and Testing (MTDT)*, pages 48–53, 1999.

[11] Y. Zorian. Embedded Infrastructure IP for SOC Yield Improvement. In *Proceedings ACM/IEEE Design Automation Conference (DAC)*, pages 709–712, June 2002.