System-level Power/performance Evaluation of 3D stacked DRAMs for Mobile Applications

Marco Facchini¹², Trevor Carlson¹, Anselme Vignon², Martin Palkovic¹, Francky Catthoor¹, Wim Dehaene², Luca Benini³, Paul Marchal¹

¹IMEC – Interuniversity MicroElectronics Center, Kapeldreef 75, B-3001 Heverlee, Belgium ²ESAT-MICAS Katholieke University Leuven, Kasteelpark Aremberg 10, B-3001 Heverlee, Belgium ³DEIS – Università di Bologna, Viale Risorgimento 2, 40136 Bologna, Italy

Abstract-Convergence of communication, consumer applications and computing within mobile systems pushes memory requirements both in terms of size, bandwidth and power consumption. The existing solution for the memory bottleneck is to increase the amount of on-chip memory. However, this solution is becoming prohibitively expensive, allowing 3D stacked DRAM to become an interesting alternative for mobile applications. In this paper, we examine the power/performance benefits for three different 3D stacked DRAM scenarios. Our high-level memory and Through Silicon Via (TSV) models have been calibrated on state-of-theart industrial processes. We model the integration of a logic die with TSVs on top of both an existing DRAM and a DRAM with redesigned transceivers for 3D. Finally, we take advantage of the interconnect density enabled by 3D technology to analyze an ultra-wide memory interface. Experimental results confirm that TSV-based 3D integration is a promising technology option for future mobile applications, and that its full potential can be unleashed by jointly optimizing memory architecture and interface logic.

I. INTRODUCTION

Mobile devices have come to represent a significant volume of electronics products, with more than 800 million units shipped in 2007. In addition to the dramatic increase of shipping volume year after year, many features have been added to these mobile devices. Convergence of communication, computing and consumer applications has been a recent trend in smart phones and handheld computing devices. To support these features, the mobile devices need to deliver an ever increasing amount of performance. Performance requirements for multimedia applications exceed tens of GOPS [1]. The high performance of core execution drives up the data exchange rate between the core and memory (both on chip and off chip) and leads to the storage of more data. At the same time, the data transfer power consumption must be minimized to guarantee sufficient battery life. Whereas in today's system's on-chip SRAMs are used to cater to the required data rates, the integration of memories, particularly high density ones, in advanced logic chip processes may become prohibitively expensive [2]. Alternative options, such as eDRAM, are not as economical, especially when targeting densities greater than 72Mb. Off-chip DRAM memories provide high density with a lower cost, but has difficulty providing the desired bandwidths due to a restricted number of I/Os. Moreover, off-chip memories are a source of concern because of their higher power consumption. The energy per bit consumed for going off-chip is many times higher than the one required for on-chip accesses. Indeed, complex and power hungry I/O transceiver circuits are needed to deal with the electrical characteristics of interconnections between chips in a conventional package.

The close integration of DRAM memories and logic technology using TSV technology may resolve the above memory challenges. In Figure 1, we depict the targeted chip stack. Both dies are connected with TSVs through the logic die. The TSVs bring two main advantages compared to wire-bonds: (1) they have excellent electrical characteristics, eliminating the need for complex IO driver circuits; (2) they have a very small area footprint. Hence, in contrast to existing packaging solutions, the number of IO circuits can be increased to hundreds and even thousands with a minimal area penalty. With this reduced parasitic load, 3D integration of memory to the core may enable the off chip TSV-connected memory to provide the performance of an on chip memory. Borkar [3] reported that stacked SRAM connected to an 82 core processor test chip demonstrated > 1 TFLOP operation and a 90% reduction of memory access I/O power. The paring of high bandwidth communication with the lower power utilization of 3D integrated memory is an ideal fit for mobile devices.



Figure 1. 3D stacked DRAM

The contribution of this paper is to perform a detailed power assessment of 3D stacked memories for these mobile devices. To this end:

• We propose 3D stacking scenarios that can be realized with minimal change to current DRAM architectures. More revolutionary and power efficient approaches can be envisioned in the longer term. In this paper, we will compare the power utilization of 3D stacks for both a logic die with an existing DDR-SDRAM and a logic die with a DRAM that contains re-designed I/O. More details on the different stacking scenarios will be provided in Section III.

• We present a *detailed micro-architectural power model of the DRAM* for the various stacking scenarios. In contrast to DRAM power models provided by DRAM manufacturers, we can use this model to delimit the importance of the IO power consumption vs. the array power consumption for various interface configurations. The power model will be explained in Section IV.

• We account for *the actual usage conditions of the DRAM within mobile systems*. This is important to accurately capture the internal vs. the interface power utilization. In contrast to cache-based architectures of high performance systems, a software-controlled DMA typically transfers the data between DRAM and on-chip local memory using large read/write bursts with a large amount of locality. As a result, the power due to pre-charge/activation cycles is typically lower than in high performance based systems. To accurately capture these usage conditions we have developed a cycle-accurate SystemC model of the DRAM and its controller that is fed by realistic application traces from our in-house memory optimization tool. The statistics collected with this tool are fed into the micro-architecture power model. Our method to collect these statistics is explained in Section V.

• We will present *experimental results, predicting power* benefits up to 60%. They will be presented in Section VI.

First, we present the related work.

II. RELATED WORK

We first provide a short background on the specific 3D technology that will be used within this paper. Thereafter, we explain how 3D technology has been applied for memory stacking in current literature.

A. Through Silicon Via Technology



Figure 2. 3D Stacked IC technology featuring 5um Through Silicon Vias enabling the interconnection of global wires between different dies.

The results in this paper are based on IMEC's *3D-SIC* TSV technology [11]. Similar flavors of this technology have been presented by LETI/Sematech/Intel/IBM/TSMC. The Through Silicon Vias have a 5 μ m diameter and 10 μ m pitch (see Figure 2). Electrical resistance and capacitance, based on TCAD simulations [12], is 40m Ω and 38fF, respectively. This 3D SIC technology provides a high density for 3D interconnects, thereby allowing the close integration of DRAM with logic dies at the level of global back-end-of-line interconnects.

B. Stacking memory

In recent years, several authors have been exploring the benefits of memory and logic stacking. Already in 1996, [4] assessed the performance benefits of tightly integrating DRAMs in a 3D RISC system. By alleviating typical limitations on memory size and bandwidth, they estimated that performance benefits of up to 25% could be realized. A similar estimation for multi-core systems was presented in [5]. Further improvements are possible if DRAMs are redesigned to take advantage of the high vertical interconnect density and heterogeneous technolo-

gies [6]. An interesting research direction is the combination of network-on-chips with highly-banked memories implemented with 3D technology [7]. Our work differs from the above in two aspects. First, we focus on mobile systems. Assessing the power consumption of various stacked scenarios in these types of systems is a strict requirement. We have developed a detailed power model to address these concerns. Secondly, we focus on 3D integration scenarios that can be manufactured in the short term, i.e. we only consider small changes to the DRAM IO circuits rather than globally re-thinking the memory architecture.

As IO circuits play an important role in the design of our 3D stacked DRAM scenarios, we have investigated papers on IO circuits for 3D SiP/SiC solutions. In [8], a high-speed interface for a stacked logic and memory design is proposed using IO bumps. An interesting contribution of this paper is to demonstrate that CMOS drivers can be used to transmit data between different chips if the parasitic load of the inter-chip connection is low. In [9], a more complex interface circuit is proposed for interconnecting multiple dies. The circuit includes a hysteresis buffer for signal integrity, and has a dedicated programmable supply for the transmit/receive circuits. Simulation results indicate that power consumption of 3D IO circuits is many times lower than typical 2D IOs. Unfortunately, the papers above do not describe the precise benefits on the overall power consumption of the DRAM subsystem. Our contribution is to explain in detail how much power can be saved for four specific 3D stacked DRAM scenarios.

Finally, [10] comes closest to our work. This paper presents an innovative DRAM memory. The DRAM is integrated sideby-side with a logic die within a single package. To increase bandwidth, the interface width of the DRAM has been extended. By operating the DRAM at lower frequencies, large power savings can be realized. Nevertheless, the integration scenarios evaluated within this paper are different: we discuss integration scenarios which are closer to existing DRAMs and adapt our interface circuits to use Through-Silicon-Via technologies. In the next section, we will discuss the proposed scenarios for 3D stacked DRAMs in more detail.

III. SCENARIOS FOR 3D STACKED DRAM

We compare four different interface options between the logic and DRAM dies, as depicted in Figure 3 for a single bit wire. Scenario 1 consists of stacking a logic die and an existing DRAM, but use TSVs rather than wire-bonds to interconnect both dies. I/O transceivers and the termination circuits based on SSTL2 are used to interconnect both dies. These circuits are designed for high parasitic load and coupling, common for interchip connections through package I/Os and/or over PCBs. However, as the TSVs have much better electrical characteristics, the parallel termination circuits can be removed. This results in scenario 2. As the capacitive load is much lower now, we can consider using CMOS buffers to interconnect DRAM and logic die. Therefore, in scenario 3, we replace the IO transceivers with CMOS based buffers. Finally, in scenario 4, we take advantage of the interconnect density of the TSVs, increasing the amount of I/O pins between DRAM and logic die (similar to [10]).

As a worst-case reference, we have added a fifth scenario in which we consider that the DRAM and logic die are integrated



Figure 3. Interconnect scenarios

as different packages on a PCB. This is identical to scenario 1 but with PCB interconnections rather than TSVs.

IV. POWER MODEL FOR 3D STACKED DRAM

Memory vendors conventionally provide aggregated power numbers [13] along with standard formulas [14] to characterize their products' power consumption, as a function of the expected workload [15]. Unfortunately, these models are insufficient to estimate the benefits of 3D stacked DRAM. Indeed, for analyzing the above four scenarios, it is necessary to isolate, at least, the contribution of the DRAM transceivers from the other memory die internal functional blocks. Also, it is required to include the power consumption of the attached data and command bus as a function of the different command states and interface physical configurations (see Figure 3).



Figure 4. Input-output of the micro-architectural power model (a) and detailed steps for internals (b) and bus(c) power calculations

Therefore, we have developed a micro-architectural power model of the DRAM, including the communication bus to the logic die (see Figure 4a).

The model provides a breakdown of power use for the internal functional blocks, taking into account the memory state and workload requirements. It is parameterized to (1) type of interchip interconnect (PCB vs. TSV), (2) presence of SSTL-I terminations on the bus wires, (3) I/O transceiver circuit style (SSTL vs. CMOS), (4) the data bus width of the DRAM and (5) DRAM protocol (DDR vs. SDR). Other inputs include typical usage currents (Iddx) and usage statistics (see Section V on how to collect these numbers). The model consists of two parts: (1) communication bus power estimation and (2) internal power estimation, i.e. all major contributors to power on a DRAM chip. Both are discussed in detail below.

A. Internal power of an SDRAM

The three steps of our power model are depicted in Figure 4b. Our model starts from the currents specified in a typical commercial SDRAM datasheet (the Iddx currents). From here we break up these currents into the contributions of the single internal DRAM components, using distribution percentages from a DRAM manufacturer. We model the following architectural micro-blocks: (1) address registers, (2) command decoding & control logic, (3) row address path, (4) column address path, (5) memory array, (6) data path, (7) supply voltage generators, (8) DLL. Having this detailed power breakdown, it becomes feasible to extrapolate how architectural changes impact the DRAM power. For instance, consider that we want to estimate the activation/precharge power of a memory with twice the page size. For this purpose, we will take the Idd0 current from the data sheet (e.g., 77mA). By applying one possible distribution percentage for the array (40%), we obtain how much current sinks in this component (30.8mA). Scaling this number linearly for larger page sizes, we obtain the final array current of 61.6mA. Similarly, we scale the power consumption for the other internal components of the DRAM. Thereafter, these numbers can be aggregated again into the scaled Idd0 figure and/or used for estimating the power for active-precharge energy (both for static and dynamic power). For the latter, we reuse the formulas presented in [14]. Finally, and again similar to [14], we de-rate these power figures with the usage statistics of the application under test.

B. Wire power across the DRAM bus

In this section, we explain how we have estimated the power consumed in the wires of the memory bus for the four scenarios depicted in Figure 3.

We assume that the bus can be described with a concentrated parameter RC model. Thus I/Os and physical interconnect medium (TSV/PCB) are singularly represented as an (R,C) pair. In Figure 5 joint capacitance and relevant resistance values have been listed. The interconnect capacitance in the TSV scenarios should be higher then the TSV capacitance of 38fF (see Section II). Indeed we suppose a significant capacitive contribution due to the need for a redistribution layer (RDL) to route the TSV signal to the pre-existing DRAM I/O pads (see Section III).

The first step is to derive both static and dynamic contributions for a single bit wire as detailed below.

| interface capacitive | loads (Farad) | | | |
|----------------------|-------------------|-----------------|-------------------|-----------------|
| | Memory I/Os + ODT | | interconnect | Controller I/Os |
| | DQs | CMDs | | |
| Scenario Ref | 4E-12 | 2E-12 | 1.83E-12 | 2.5E-12 |
| Scenarios 1 / 2 | 3.5E-12 | 1.5E-12 | 2.37E-13 | ~1E-15 |
| Scenarios 3 / 4 | ~1.5E-15 | ~1E-15 | 2.37E-13 | ~1E-15 |
| off-chip termination | resistances - f | or terminated s | cenarios (Ref / 1 | <u>)</u> |
| Rt | Rs | Rpin-in | | |
| 50 Q | 22 0 | 28 0 | | |

Figure 5. Capacitances and resistances for the wire power in the different scenarios

Static power. In the reference scenario and in scenario 1, when the signal is driven high or low, the termination scheme produces a static current up to 8.1mA according to the definition of the SSTL2 standard. Even in the high impedance state, the transceivers still consume leakage power. We account for this leakage power in scenarios 2, 3 and 4.

Dynamic energy. The dynamic energy consumption is due to switching of the wire and transceiver capacitances. The precise amount of capacitive load depends strongly on the considered scenario (see Figure 5). In the reference scenario and in scenario 1, the load is operated at reduced swing across the termination resistors.

To estimate the average power of a wire, we combine static and dynamic consumptions according to the SDRAM protocol. E.g., the protocol specifies that a data wire should be driven High-Z when there are no read/write requests, which makes its static power consumption less than the clock wire one. For the latter, the usage statistics of the application under test are applied. Finally, the bus power is the sum of the contribution from each individual wire. In particular for the reference scenario we assume to have 16 data, 15 address, 5 command and 6 clock and synchronization wires.

V. COLLECTING USAGE STATISTICS FOR MOBILE APPLI-CATIONS

A. System architecture

The primary focus is to determine benefits of stacked 3D DRAM for a mobile system. Particularly, in the context of this paper, we target a platform that consists of ADRES cores [16] as the primary processing unit, as it is very efficient when kernels have been optimized and mapped to run on the core. An ADRES processor consists of a highly power-efficient VLIW processor and Coarse-Grained Reconfigurable Array (CGRA).

Furthermore, the platform architecture is designed with scratchpad memories rather than caches and relies on softwarecontrolled DMAs to transfer data power efficiently between the layers. There is one DMA controller attached between each scratchpad memory and the AMBA bus. These software controlled memory architectures are more energy efficient and therefore typically used in mobile applications [17][18]. The DRAM is used as second-level memory (L2). A memory controller translates the bus requests into the JEDEC SDRAM protocol, allowing for communication with the memory. The software mapped on this architecture should be highly optimized in order to efficiently use this memory architecture. In the next paragraphs, we indicate how we have modeled both architecture and the application mapping in order to quantify the benefits of a 3D stacked DRAM (see Figure 6).



Figure 6. Memory statistics generation

B. Software optimization flow

The software optimization flow starts from an application specified in CleanC. CleanC is a subset of C that allows for better automatic analysis and manipulation of the code by our parallelizing and optimizing compilers. Two in-house tools, Memory Hierarchy (MH) and Multi-Processor Assist (MPA) manipulate this CleanC code. Particularly, the MH tool automatically inserts DMA transfers between memory levels for the DMA transfer engines to execute. These DMA transfers are properly timed to have the data arrive when needed, but also not to unnecessarily occupy space in the memory levels closer to the processor. The output from the MH step is then provided to MPA tool. The latter tool parallelizes the code according to the user-specified affinities. It thus partitions the software into a number of tasks, each containing a collection of kernels to be executed on the processors. MPA automatically identifies data and functional parallelism and after the programmer chooses which tasks are to be mapped to which processor, it modifies the code to support the multi-processor implementation by generating a new copy of the code with the proper communication and synchronization methods. By simulating this highly optimized code onto a model of the target platform, we collect the usage statistics for estimating DRAM power.

C. Simulation and memory usage statistics

Rather than executing the above code after compilation on a cycle-accurate model of the target systems, we reduce the complexity of the platform for shorten simulation times.

In fact, our model of the target platform consists of cycleaccurate SystemC model of DRAM, a memory controller and trace-generators. The latter two types of components are interconnected with cycle-accurate model of the AMBA bus. The SystemC simulator records all necessary statistics for our DRAM power model.

Hence, for obtaining valuable memory statistics, it is imperative to feed the trace-generators with a trace file containing all DMA transfers across the AMBA bus to the DRAM within the optimized code. We generate this trace-file by executing the code on our high-level simulator (HLS). The HLS emulates the parallelized code on the host workstation, thereby modeling synchronization and timing of the target platform and outputting the resulting trace-files. Similar to many high-level simulators, the timing data is approximated with profile data, collected by running the kernels standalone on the target processors ISS.

The memory statistics collected in this way are representative for the targeted multi-core application.

VI. EXPERIMENTAL RESULTS

In order to have a meaningful exploration, the memory traces have been gathered by running common end-user applications on our memory system simulation environment. Three heterogeneous traces have been used to present our results (see Figure 7): MPEG-4 Part 2 Simple Profile encoder on QCIF, artificial traces that show particularly locality characteristics and QSDPCM.



Figure 7. Benchmark memory access traces characteristics

In Figure 8, we show the experimental results for the three scenarios with the same interface data-bus width. A first observation is that DRAM power for multimedia application is important. The MPEG-4 encoder consumes up to 290mW of power even at such a low frame resolution. Power consumption is likely to increase for high definition applications. Secondly, the figure indicates that for the reference scenario a large portion of the energy is consumed by the exchange of data between memory and logic die. The power consumption of the bus varies between 50% and 60% of the total of the memory system. The internal memory power consumption is only slightly effected from changes in the I/Os (scenario 3). On the other hand the bus power is drastically reduced while changing scenarios. Replac-



Figure 8. Power of the DRAM for the scenarios of Figure 3.

ing the PCB interconnects with TSVs does not significantly help (scenario 1). However, by removing the termination circuits (scenario 2) we obtain more than a 2x reduction of the power consumed in the interconnects. In this scenario, almost all the power consumed by the bus is attributed to driving the transceiver's load. Indeed the largest savings are obtained in scenario 3. Herein, we use simple CMOS transceivers on both the logic and DRAM dies, designed for low power.

A. Energy in an interconnect wire

To better understand what causes these energy savings, we decompose the energy consumption necessary for sending one bit across the DRAM bus for the four different scenarios (see Figure 9). The power of the first two scenarios is mainly dominated by the static power consumed by the SSTL termination scheme when driving a valid data. The dynamic energy consumption of an SSTL signal is on the other hand relatively low as it sends data at a reduced voltage swing. Therefore, reducing the interconnect capacitance by replacing PCB wires with TSVs does not result in as large of an energy savings as resulted from the migration from the reference scenario to scenario 1.

However, as TSVs tend to be much shorter and have better electrical characteristics than PCB wires, there is no longer a need for these termination circuits. After eliminating them, we find that dynamic switching of the line and transceiver capacitance contributes to the next largest power use (scenario 2). Replacing the SSTL2 transceivers (typical ~2-3pF capacitance) with simple CMOS drivers with lower capacitance (only a few fF in capacitance) strongly improves the power consumption (scenario 3).



Figure 9. Energy of a single data wire assuming single bit toggling at $200 MHz/2.6V V_{DD}$

B. Pros/cons of a wider interface

An even more significant consequence of the transition to TSVs is that the data-bus width between the DRAM and the memory controller can be increased to 128 bits while consuming still less power than our reference scenario. Hence, we can trade-off the energy savings with data throughput. The power-performance trade-off is illustrated in Figure 10. To obtain this graph we used the same activity of the mpg4 application, thus we increased the throughput proportionally to the width of the interface.



Figure 10. Trade-off between throughput and total memory system power consumption.



Figure 11. Energy required from DRAMs with CMOS transceivers and different data-bus width (MPEG4 encoder application traffic).

In Figure 11, we show the results when the memory statistics of the MPEG4 encoder application are updated for the correct interface dimension (i.e., after rerunning our memory Simulator with a wider interface). The numbers indicate that significant gains are achieved in terms of both reading and writing energy (respectively 33% and 28% savings) by increasing the data-bus width from 16 to 64 bits. Also, with a 64-bit wide interface the encoding application runs in fewer cycles. Therefore, the background energy consumption is reduced whenever the data-path is not active. In contrast, the background energy consumption increases by 14% when the data-path is active (ACT_STBY state). The reason for this is that the data-path includes I/Os, gating circuits, multiplexer and registers that multiply according with the interface width.

Further increase of the data-bus width does not provide additional benefits, because only 4% of the memory requests issued from this run of the MPEG4 encoder are longer than 128 bits. Therefore, additional energy is used to drive the extra wires while the application is not benefiting from the extra data throughput.

VII. CONCLUSIONS

In this paper, we have estimated the benefits of 3D stacked DRAM. By exploiting the excellent electrical characteristics of TSVs, it becomes possible to replace the typical transceiver circuits on DDR with more power efficient CMOS transceivers. Results show that we can save up to 98% of the interface power consumption or 60% of the total memory system power consumption by modifying the I/O transceivers and bus designs. In this scenario, the DRAM interface could be widened without

incurring a significant power penalty. However, the precise benefits of 3D technology strongly depend on the data locality and memory traffic produced by the application under consideration. This can be evaluated with our simulation environment and micro-architectural power model of the 3D stacked DRAM.

These results show promise that with additional software modifications, an even higher improvement could be seen. In addition, internal DRAM structure modifications to take advantage of these wider widths could also provide opportunities for an overall reduction in energy use.

REFERENCES

- K. Uchiyama., "Power-Efficient Heterogeneous Parallelism for Digital Convergence", VLSI Circuit Digest of Technical Papers, IEEE p 6-9, June 2008
- [2] S. Q. Gu et al., "Stackable Memory of 3D Chip Integration for Mobile Applications", IEDM, December 2008
- [3] S. Borkar, International 3D System Integration Conference, May 2008, p.1-1
- [4] M.B. Kleiner et al., "Performance improvements of the memory hierarchy of RISC-Systems by application of 3D-Technology", in Transactions on Components, Packaging and Manufacturing Technology, Part B: Advanced Packaging, vol. 19, no.4, Nov. 1996, pp. 709-718
- [5] T. Kgil et al., "PicoServer: Using 3D Stacking Technology to Enable a Compact Energy Efficient Chip Multiprocessor," Proc. Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS 06), ACM, Press, 2006, pp. 117-128
- [6] C. C. Liu et al., "Bridging the Processor-Memory Performance Gap with 3D IC Technology," IEEE Design & Test of Computers, vol. 22(6), pp. 556–564, 2005
- [7] F. Li et al., "Design and Management of 3D Chip Multiprocessors Using Network-in-Memory," Proc. 33rd Ann. Int'l Symp. Computer Architecture (ISCA 06), IEEE CS Press, 2006, pp. 130-141
- [8] T. Ezaki et al, "A 160Gb/s Interface Design Configuration for Multichip LSI", Proc. ISSCC, 2004
- [9] S. Alam et al, "Inter-strata Connection Characteristics and Signal Transmission in Three-dimensional (3D) Integration Technology", Proc. ISQED, 2007
- [10] K. Kumagai , "System-in-Silicon Architecture and its Application to H.264/AVC Motion Estimation for 1080HDTV", Proc. ISSCC, 2006
- [11] B. Swinnen et al., "Hybrid bonding" in "Wafer Level 3-D ICs Process Technology", Springer (in print)
- [12] P. Marchal et al., "3D Technology Assessment: Path-finding the technology/Design Sweet-spot" proc. IEEE, January 2009
- [13] "Memory Power Consumption DRAM IDD", RAMpedia, 6 September 2008, http://www.rampedia.com/index.php/ae2a>
- [14] "System power calculator", Micron, 6 September 2008, http://www.micron.com/support/part_info/powercalc.aspx
- [15] Wang D. et al., "DRAMsim: A memory-system simulator." SIGARCH Computer Architecture News, vol. 33, no. 4, pp. 100-107. September 2005.
- [16] B.Mei, et a;. "ADRES: an architecture with tightly coupled VLIW processor and coarse-grained configurable matrix", Proc. IEEE Conf. on Field-Programmable Logic and its Applications (FPL), Lisbon, Portugal, pp.61–70, Sep. 2003.
- [17] R.Banakar, S.Steinke, B.Lee, M.Balakrishnan, P.Marwedel, "Scratchpad memory: design alternative for cache on-chip memory in embedded systems", Proc. on the tenth international symposium on Hardware/software codesign (CODES), pp.73—78, 2002.
- [18] L.Benini, A.Macii, E.Macii, M.Poncino, "Synthesis of Application-Specific Memories for Power Optimization in Embedded Systems", Proc. of the 37th conference on Design automation (DAC), Los Angeles, CA, USA, pp.300–303, 2006