Cache Array Architecture Optimization at Deep Submicron Technologies

Annie Y. Zeng, Ken Rose and Ronald J. Gutmann
Center for Integrated Electronics
Rensselaer Polytechnic Institute
Troy, NY 12180-3590 USA
zengy@rpi.edu

Abstract

A cache access time model, PRACTICS (PRedictor of Access and Cycle TIme for Cache Stack), has been developed to optimize the memory array architecture for the minimum access and cycle times of on-chip memory using circuit models based on Cadence simulations. Lumped RC models have been used to approximate the distributed RC interconnect network in the access time models. Both SRAM and DRAM models have been validated with industrial designs. The limited influences of gate fan-out and transistor size on the cache array architecture indicate that interconnect delay is dominant at deep submicron technologies.

1. Introduction

Due to the speed mismatch between the microprocessor and main memory, on-chip memory is a key determinant of microprocessor performance. On-chip cache design requires a balance between delay, area and power consumption. The circuit selection for decoders, bitlines and sense amplifiers, transistor sizing of these circuits, interconnect sizing and memory array architecture are important parameters. Since exploring a large memory design using conventional SPICE circuit simulations is time consuming, simplified models are very valuable and effective. Such models, showing the dependence of the cache access time on the cache parameters, are not only helpful in memory design at current technology nodes, but can be used to predict future performance.

Several authors have developed analytical models for SRAM access time [1-4]. Wada presented an equation for the access time of an on-chip cache as a function of various cache parameters (cache size, associativity and block size), cache architecture and process parameters [1]. Wilton and Jouppi developed a complete cache access time model, CACTI (Cache Access and Cycle Time) [2] based on Wada's model. Amrutur and Horowitz combined these models with energy and area models to address SRAM speed and power scaling issues [3]. Bhavnagarwala and coworkers [4] have developed an analytical model for a hierarchically partitioned SRAM, balancing interconnect delays, fan-out and gate delays. All of these models use lumped wire capacitances to estimate word and bit line delays.

These previous models were generally adequate for large (> 0.5um) technologies and applied well to small cache sizes. For large cache sizes with deep submicron technology, interconnect delays are more important and we will show that lumped capacitance models inaccurately describe interconnects. Previously, results were extended to smaller technology nodes by scaling delay times in proportion to gate length; this linear scaling approach can be quite inaccurate for current cache designs.

In this paper, we extend previous SRAM access analytical models and develop a new access and cycle time model named PRedictor of Access and Cycle TIme for Cache Stack (PRACTICS), which includes both SRAM and DRAM models. Our models are based upon detailed simulations at the 250nm technology node and can be extended to further technologies. We have also incorporated industrial wiring strategies for particular technology nodes. An extension has been developed to predict performance advantages of wafer-level 3D integration of on-chip caches [5].

2. Overview of PRACTICS

In PRACTICS, the worst-case latency of the memory reading operation is specified by the access time. The principal inputs to PRACTICS are memory size and block size. It uses an equation for the access time of on-chip memory as a function of various memory organization parameters, array architecture parameters and technology profiles, with device and circuit models based on Cadence simulations. The access time is estimated by decomposing each component into several equivalent RC circuits and

using the first-order approximation of the Elmore model [6] to estimate the delay of each stage. The access time is minimized by running an exhaustive search algorithm to all variations of the array architecture parameters for a specific technology node and wiring strategy.

In the SRAM model, a 6-transistor SRAM cell is employed. Both tag and data arrays are accessed in parallel. The DRAM model uses a one-transistor dynamic cell. The array architecture is similar to the SRAM model except only the data path is considered.

We used fully independent cache banks to support simultaneous multiple accesses to the cache. The cache is divided into smaller arrays, each mapping a distinct address space. While banking adds decoding overhead, these extra delays become negligible compared with the delay inside the bank as the cache size of each bank increases, and are neglected in PRACTICS.

The cache is pipelined using an approach similar to wave pipelining [7] that effectively realizes five cache accesses at any given time. There are six main delay components in the critical access path. Considering that the wordline should be held when the bitline is being charged, the pipe stage number is defined to be five, since the wordline, bitline and sense amplifier delays are included in the same pipeline stage. The pipeline rate is limited by the slowest pipeline stage.

3. Access time models

In PRACTICS each delay component is estimated individually and then combined to estimate the access and cycle time of the entire cache. The access time is comprised of six main components: address-in routing delay, decoder delay, wordline delay, bitline and sense amplifier delay, internal data bus driving and data output bus delay (global). Based on delay component calculations with different configurations, the program selects the best configuration for the minimum access time by running an exhaustive search algorithm with an execution time less than 30 minutes on a 755MHz PC.

Inserting repeaters in a wire can overcome the quadratic increase in delay due to a linear increase in interconnect length. In PRACTICS 1.0, uniform repeater insertion is used as an initial strategy for optimizing repeater size and location.

There are two types of delay components involving long wire delay, i.e. delay components with repeater insertion and without repeater insertion. The first class includes address-in routing, decoder, internal output bus, output bus driving and valid-signal delays. The wordline and bitline delays belong to the second class.

The major concern is the effects of interconnect delay in the cache access time model. The basic model considered, as shown in Figure 1, is a distributed RC line that is driven by an inverter and connected to a load

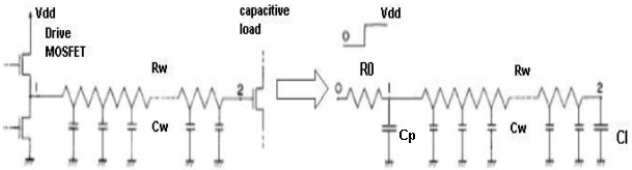

Figure 1. Basic delay model [8]

transistor. Most of the wiring in our model can be reduced to this form. The drive inverter is replaced by an equivalent resistance R0 shunted by an intrinsic capacitance Cp, and the load transistor by a capacitance Cl in the equivalent circuit.

The accuracy of the delay approximation using this basic equivalent circuit depends on the selection of the lumped RC model to approximate the distributed RC line. Sakurai [8] offers a method to select the appropriate ladder RC model for wiring approximation based on the two variables, C_T (= Cl/Cw) and R_T (=R0/Rw). Rw and Cw are the total wiring resistance and capacitance. The smaller C_T and R_T (longer wires), the more complicated ladder RC model required.

Most models in PRACTICS are based on this basic model, varying for different conditions. Though they are based on Cadence simulations at 250nm technology, their extensions to further technology nodes are reasonably accurate because the selection method is based on two ratio values (C_T and R_T).

A π1 lumped RC model has been used to approximate the distributed RC wire network in the wordline and bitline charging models, which have been simulated and verified using Cadence with IBM 6HP, which includes 250nm CMOS and the IBM 6-level aluminum wiring strategy [9]. The transient response of wordline charging model from Cadence simulation, as shown in Figure 2, indicates that the π1 lumped RC model fits the wire distributed RC model very well compared with the pure wire capacitance model used previously [1-4].

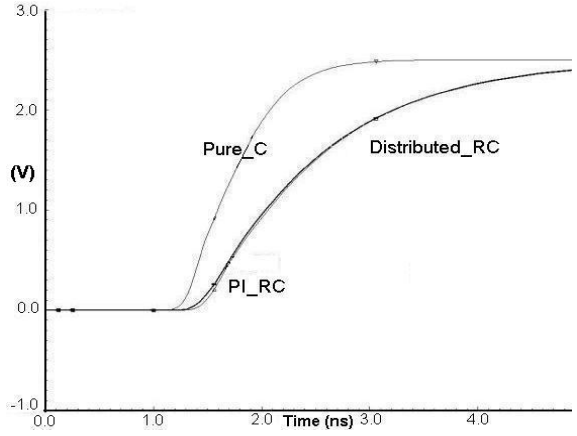

Figure 2. Wordline transient response

4. Validations

The DRAM model of PRACTICS has been validated with a simulation of a 64Mb NEC DRAM [10]. This 64Mb DRAM is made by a 0.25µm embedded DRAM ASIC 5-metal fabrication technology. The memory contains eight 8Mb banks, with each bank composed of eight 1Mb blocks. Each 1Mb block consists of 8 x 16 subarrays, and the block size is 32b [10]. In PRACTICS, the IBM 6HP device parameters and wiring strategy [9] have been applied. Setting these array architecture and technology parameters, the model estimates an access time of 6.78ns, within 1% of the measured value of 6.8ns (see Figure 3). Note that the PRACTICS results closely match the measured values at each stage. In addition, the architecture parameters that give the minimum access time are identical to the original design [10].

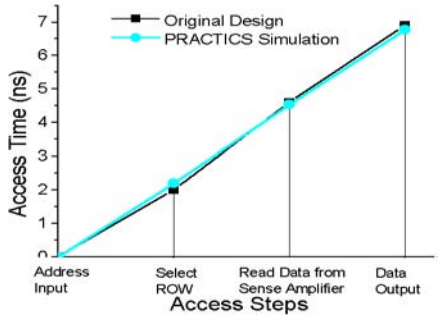

Figure 3. DRAM model verification results

The SRAM model has been validated by the simulation of a 4-way 18Mb Intel SRAM cache [11]. The 18Mb Intel SRAM cache uses a 5.6µm^2 (2.22µm x 2.52µm) 6-transistor cell and is fabricated on 0.18 µm 6-metal-layer, 1.3-1.5V CMOS. The design separates the global bitlines into read and write bitlines on Metal 4 and Metal 6, with local bitlines on Metal 2 [11]. The cache is split into four banks and each bank is composed of 18 global subarrays.

Using the device RC values from Hodges, et al. [13] and the Intel 180nm wiring strategy [14] in PRACTICS, specified organization, array architecture and technology parameters are exploited to obtain an access time of 2.82ns. This value is within 9% of the measured access time of 2.6ns[11]. Figure 4 shows the individual delay time comparison between the measured results and PRACTICS simulations. Note that with the same setting, the linear scaling approach from CACTI [12] gives an access time of 6.13ns, a factor of 2.3 greater than the measured and PRACTICS value.

5. Cache design spaces in PRACTICS

Besides the cache size, block size and associativity, two types of parameters are used to optimize the array architecture for minimum access time, i.e. array architecture parameters and technology profiles. Table 1 summarizes the memory array architecture parameters being used in PRACTICS. The two groups of technology profile are summarized below.

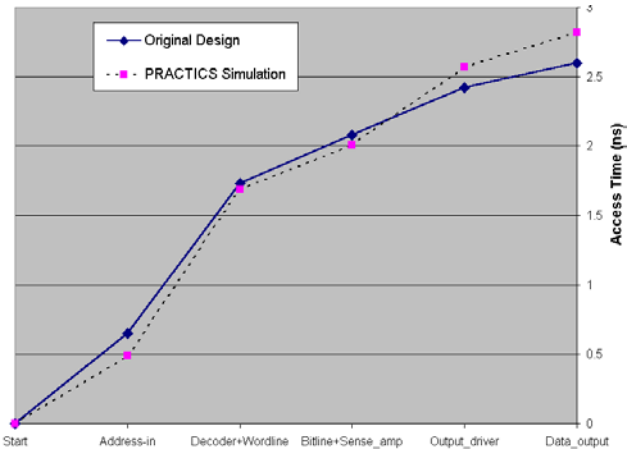

Figure 4. SRAM model verification results

Table 1. Memory array architecture parameters

Symbols	Meanings	Parameters
Layer	Layer of wafer stack	1,2,3,4…
NumBank	Number of banks	1, 2, 4…
Assoc	Associativity	1, 2, 4…
Ndwl	Number of segments per word line (data)	1, 2, 4…
Ndbl	Number of segments per bit line (data)	1, 2, 4…
Nspd	Number of sets mapped to a single wordline (data)	1, 2, 4…
Ntwl	Number of segments per word line (tag)	1, 2, 4…
Ntbl	Number of segments per bit line (tag)	1, 2, 4…
Ntspd	Number of sets are mapped to a single wordline (tag)	1, 2, 4…

The first technology profile is on the circuit level. This includes memory cell sizes and circuit design parameters, such as the circuit styles of decoder and sense amplifiers, and transistor sizing of these circuits. Table 2 lists the memory cell sizes at different technology nodes, where the values at the 90nm node are extrapolated from previous technologies.

Table 2. Intel SRAM cell size [15]

	250nm	180nm	130nm	90nm
SRAM cell size (um x um)	10	5.6	2	1
Width (um)	3.01	2.22	1.22	(0.87)
Height (um)	3.41	2.52	1.64	(1.16)
Aspect ratio (H/W)	1.13	1.135	1.34	(1.33)

Notes: parameters in parenthesis are estimated.

In PRACTICS 1.0, the circuit parameters are fixed. We are using a three-stage CMOS static NAND + NOR + Inverter decoder architecture. The first stage is the address-in inverter driving the NAND gates of the 3-to-8 row address pre-decoder. The second stage is NAND gates driving several NOR gates for each row. These 3-to-8 codes are combined using NOR gates in this stage. The final stage is a NOR gate driving the inverter before the wordline driver. The decoder architecture is shown in Figure 5. The transistor sizes of these circuits are based on the 250nm technology node, and are applied to simulations at smaller technology nodes using a linear scaling approach. The fan-in of each NAND gate is set as 3, and fan-out of each NAND gate is 4. The size of all repeaters is 32x minimum gate length. A two-stage differential sense amplifier [1] is employed in order to achieve the desired full-swing signal.

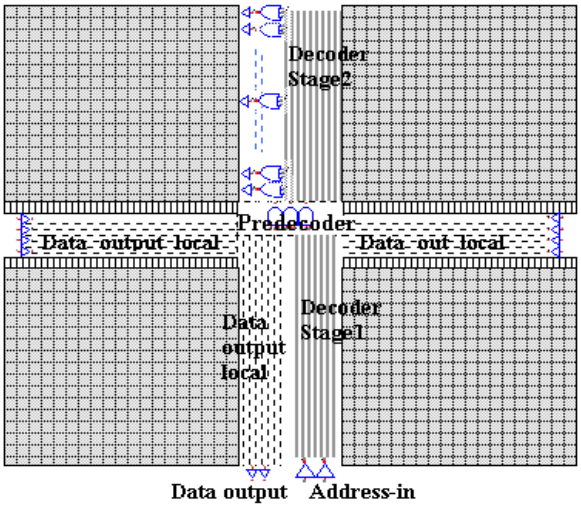

Figure 5. Decoder architecture

The second group of technology parameters is on the device level. The analytical models used in PRACTICS are obtained by decomposing both the active devices and distributed wires into equivalent ladder RC circuits. The active models include transistor equivalent resistance models, and input/output capacitance models. In the transistor equivalent resistance models, the pull-down/up resistances are given by Rpd = Rn_unit / W; Rpu = Rp_unit / W, where W is the transistor gate width and the unit equivalent resistances Rn_unit and Rp_unit are technology dependent. We used constant values for Rn_uint and Rp_unit obtained from either simulation or reference literature. The input capacitance includes the gate capacitance and the overlap capacitance, while the output capacitance is the drain diffusion capacitance. Both are proportional to the transistor geometry and the unit capacitance values obtained from either simulation or reference literature.

The RC estimates for interconnects are assumed to be independent of the applied voltages. The formulas used to calculate the wiring delay are given below:

$Rw\_unit = \rho_{eff} / (H_{int} \times W_{int})$;

$Cw\_unit = 2\ \varepsilon_{eff} \times \varepsilon 0 \times W_{int} \times (1/T_{ILD} + A/S_{wire})$.

where A is the wire aspect ratio, defined as H_{int} / W_{int}, L_{int} is the line thickness, W_{int} is the line width, S_{int} is the line spacing, and T_{ILD} is the dielectric thickness.

In PRACTICS 1.0, a simple device model is used at 180nm and 130nm nodes [13], as only a small error is anticipated with deep submicron technologies. Intel's 250nm 5-level aluminum [16], 180nm 5-level aluminum [15] and 130nm Cu dual damascene [17] wiring strategies have been used.

6. Parameter sensitivity of optimal array architecture

In PRACTICS, the memory array has been partitioned into a set of subarrays with different values of Ndwl, Ndbl and Nspd. Increasing the subarray number makes the wordline and bitline shorter, which shortens these delays. But it also increases the fan-out of decoder gates as well as the total area consumed by them. Large fan-outs and area constraints on the maximum transistor size of decoder gates imposed by array efficiency requirements, result in an increasing gate delay penalty for decoders. Optimal memory array partitioning has been discussed [4], in order to balance the decoder logic delay, wire RC delays and area constraints to decoder gate sizes. In this section, we show the effects of fan-out and decoder gate sizes on the selection of optimal array architecture parameters and performance prediction at deep submicron technology nodes using PRACTICS.

We take a direct-mapped 16M SRAM cache design as an example to demonstrate these effects. Because both the data and tag array are accessed in parallel, and the data path delay is dominant, only the data path delay is discussed here. Table 3 lists the PRACTICS simulation results of conventional 2D implementations for different technology nodes, where the block size is 128B. The distributions of individual delays have been illustrated in Figure 6, which shows the shift from gate delay dominant stages such as the decoder delay to interconnect dominant stages such as the address-in routing, data output bus driving delays. Note that this shift is clear between the

Table 3. PRACTICS simulations of 16MB cache

Nodes (nm)	250	180	130	90
NumBank	8	32	256	8
Ndwl/Ndbl/Nspd	512/ 512/4	512/ 512/2	256/ 128/1	512/ 512/4
Access time (ns)	7.66	5.34	2.36	2.14
Cycle time (ns)	2.50	1.99	1.07	0.69

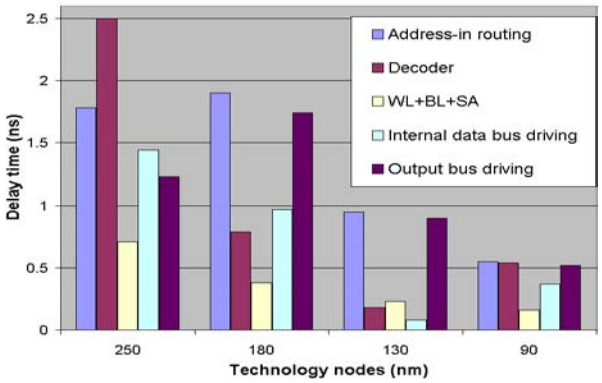

Figure 6. Delay times distributions for 16MB cache

simulations at 90nm and 250nm nodes since they have the same architecture configurations.

6.1. Effects of large fan-out

We consider the effects of fan-outs of decoder gates on the optimal array architecture selections and performance predictions. As listed in Table 4 (a - d), beyond a certain technology node the optimal array architecture configurations change as well. In order to make the performance comparisons reasonable, the performance simulations have also been done using the same optimal configurations as shown in the shaded columns. From Table 4 (a - d), we can see that with the active device shrinking, the effect of fan-out on the optimal array architecture configurations has been alleviated, observing that the configurations at smaller technology nodes are more similar to each other, especially at the 90nm node. Furthermore, this effect is weak, with the variation of the performance within 15% for different fan-outs through all technology nodes.

Table 4. PRACTICS simulations with various fan-out

(a) At 250nm node

Fan_out	4	16	256	512	512
NumBank		8		128	8
Ndwl		512		128	512
Ndbl		512		2	512
Nspd		4		8	4
Access time (ns)	7.66	7.68	8.03	8.36	8.39
Cycle time (ns)	2.50	2.52	2.87	3.05	3.23

(b) At 180nm node

Fan_out	4	16	256	512	512
NumBank		32		256	32
Ndwl		512		64	512
Ndbl		512		2	512
Nspd		2		4	2
Access time (ns)	5.34	5.36	5.67	5.87	6.00
Cycle time (ns)	1.99	1.99	1.99	2.34	1.99

(c) At 130nm node

Fan_out	4	16	256	512	512
NumBank		256		256	256
Ndwl		256		64	256
Ndbl		128		1	128
Nspd		1		8	1
Access time (ns)	2.36	2.37	2.55	2.57	2.74
Cycle time (ns)	1.07	1.07	1.07	1.07	1.07

(d) At 90nm node

Fan_out	4	8	16	128	512
NumBank			8		
Ndwl			512		
Ndbl			512		
Nspd			4		
Access time (ns)	2.14	2.14	2.15	2.21	2.39
Cycle time (ns)	0.69	0.69	0.69	0.69	0.79

6.2. Effects of decoder transistor sizing

The effects of transistor sizes of decoder gates have been considered. We set the transistor sizes used in the PRACTICS program as the unit size set, which has been validated with two industrial designs previously. Table 5 (a-d) lists the PRACTICS simulation results with varying

Table 5. PRACTICS simulations with varying decoder transistor sizes

(a) At 250nm node

Decoder size	1x	2x	5x
NumBank	8	8	8
Ndwl	512	512	512
Ndbl	512	512	512
Nspd	4	4	4
Access time (ns)	7.66	7.00	6.25
Cycle time (ns)	2.50	2.01	1.82

(b) At 180nm node

Decoder size	1x	1x	2x	5x
NumBank	32	16	16	16
Ndwl	512	256	256	256
Ndbl	512	512	512	512
Nspd	2	2	2	2
Access time (ns)	5.34	5.50	5.11	4.84
Cycle time (ns)	1.99	1.68	1.68	1.69

(c) At 130nm node

Decoder size	1x	1x	2x	5x
NumBank	256	256	256	256
Ndwl	256	512	512	512
Ndbl	128	512	512	512
Nspd	1	2	2	2
Access time (ns)	2.36	2.47	2.26	2.18
Cycle time (ns)	1.07	1.03	0.99	0.99

(d) At 90nm node

Decoder size	1x	2x	5x
NumBank	8	8	8
Ndwl	512	512	512
Ndbl	512	512	512
Nspd	4	4	4
Access time (ns)	2.14	2.05	1.99
Cycle time (ns)	0.69	0.68	0.68

decoder transistor sizes. Also in order to make performance comparisons reasonable, the performance simulations have also been done using the same optimal configurations shown in the shaded columns. From Table 5, the optimal array architectures vary with the changes of decoder gate sizes, but the effects are getting weaker as the feature sizes shrink. The variances of performance are also getting smaller monotonically from a 20% offset at the 250nm node to a 7% offset at the 90nm node.

7. Conclusions

This paper extends previous SRAM access time models and develops a new SRAM and DRAM access time model, PRACTICS. In addition, both pipelining and repeaters are included, which makes the cache structure more closely represent real caches. RC device models and lumped RC wiring delay models are shown to be appropriate at deep sub-micron technology nodes.

PRACTICS performance has been validated with several reported ICs. When designing a real cache, many different circuit styles could be applied to optimize certain stages in the critical path, such as the design style of decoder or sense amplifier. From the analysis of the effects of fan-out and transistor sizing of decoder gates, the PRACTICS simulations illustrate that with active devices scaling down, for large size caches, the interconnect delay determines the optimal cache array architecture for minimum access time at deep submicron technologies.

8. Acknowledgements

This research is partially supported by the Interconnect Focus Center for Hyper-integration, funded by MARCO, DARPA and NYSTAR.

9. References

[1] T. Wada, S. Rajan, and S. A. Przybylski, “An analytical access time model for on-chip cache memories,” IEEE Journal of Solid-State Circuits, Vol. 27, August 1992, pp. 1147-1156.

[2] S. J. E. Wilton and N. P. Jouppi, “CACTI: An Enhanced Cache Access and Cycle Time Model”, IEEE Journal of Solid-State Circuits, Vol. 31, No. 5, May 1996, pp. 677-688.

[3] B. S. Amrutur and M.A.Horowitz, “Speed and Power Scaling of SRAM’s”, IEEE Transactions on Solid-State Circuits, Vol. 35, No. 2, February 2000, pp. 175-185.

[4] Azeez J. Bhavnagarwala, Stephen Kosonocky and James D. Meindl, “Interconnect-Centric Array Architectures for Minimum SRAM Access Time”, ICCD 2001, pp. 400 - 405.

[5] A. Y. Zeng, J.-Q. Lu, R.J. Gutmann, and K. Rose, “Wafer-level 3D Manufacturing Issues for Streaming Video Processors”, ASMC 2004, pp. 247-251.

[6] W. C. Elmore, “The transient response of damped linear networks with particular regard to wideband amplifiers,” *J. Appl. Phys.*, vol. 19, pp. 55–63, 1948.

[7] C. T. Gray, W. Liu and R. K. Cain III. “Wave Pipelining: Theory and CMOS Implementation”, Kluwer Academic Publishers, Norwell, MA, 1993.

[8] T. Sakurai, “Approximation of wiring delay in MOSFET LSI”, IEEE Journal of Solid-state Circuits, Vol. SC-18, No. 4, August 1983, pp. 418-426.

[9] IBM BiCMOS6HP Design Manual.

[10] T. Kimura, et al., “64Mb 6.8ns Random ROW Access DRAM Macro for ASICs”, 1999 IEEE International Solid-State Circuits Conference, pp. 416-417.

[11] C. Zhao, et al., “ An 18Mb, 12.3GB/s CMOS Pipeline-Burst Cache SRAM with 1.54Gb/s/pin”, 1999 IEEE International Solid-State Circuits Conference, pp. 200-201.

[12] http://research.compaq.com/wrl/people/jouppi/cacti3.pdf

[13] David A. Hodges, Horace G. Jackson and Resve A. Saleh. “Analysis and Design of Digital Integrated Circuits in Deep Submicron Technology”, Third Edition, McGraw Hill Publishers, New York, NY, 2004.

[14] S. Yang, et al., “A High Performance 180nm Generation Logic Technology”, Electron Devices Meeting, 1998. IEDM'98 Technical Digest., International , 6-9 Dec.1998, pp. 197 - 200.

[15] Stefan Rusu, “Trends and challenges in VLSI technology scaling toward 100nm”, Intel corp. Sep 2001.

[16]http://www.intel.com/technology/itj/2002/volume06issue02/art01_130nmlogic/p03_scaling.htm

[17] S. Thompson, et al., “130nm Logic Technology Featuring 60nm Transistors, Low-K Dielectrics and Cu Interconnects”, Intel Technology Journal, May 16, 2002.