RC 22118 (99105) **Electrical Engineering**

IBM Research Report

Interconnect-Centric Array Architectures for Minimum SRAM Access Time

Azeez J. Bhavnagarwala, Steven Kosonocky
IBM Research Division
Thomas J. Watson Research Center
P. O. Box 218
Yorktown Heights, NY 10598

James Meindl
Georgia Institute of Technology
Atlanta, GA 30332

Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

Interconnect-Centric Array Architectures for Minimum SRAM Access Time

Azeez J. Bhavnagarwala, Stephen Kosonocky and James D. Meindl[†]
IBM T. J. Watson Research Center, Yorktown Heights NY 10598
[†]*Georgia Institute of Technology, Atlanta GA 30332*

Abstract[‡]

Physical and generic models that analytically couple the array architecture of CMOS SRAMs with the wire lengths and fan-outs along critical paths to decode and sense data are reported. Verified to be accurate with published SRAMs, these models enable the design of optimal array architectures to minimize total access time by balancing communication distance limited wire delays with fan-out and area limited gate delays.

1. Introduction

CMOS SRAM circuit design paradigms that have been in practice for over two decades have taken advantage of improvements in circuit performance obtainable by scaling transistor dimensions [1,2]. While device performance improves with scaling, the distributed RC time constant per unit length of interconnections increases as the square of the scaling factor for minimum feature size [3]. Embedded SRAM cache sizes increase by approximately 2X every generation (Figure 1) to compensate for the increasing performance disparity between processor and main memory, the growing performance cost of off-chip accesses and as demands on its size increase with more sophisticated operating systems and applications [4,5]. The increasing severity of interconnect delays, reverse scaling of metal and insulator dimensions notwithstanding [6], in tandem with larger cache sizes directly translate into longer effective communication distances [7,8,9] across SRAM cache arrays mandating an interconnect-centric methodology for the design of large, high performance cache SRAMs.

Analytical models for SRAM access time previously developed in [10] and extended in [11] were applied to explore the impact of various SRAM cache organizations on its performance. More recently, the models in [10] combined with energy and area models were extended in [8] to project limitations on speed and power scaling of SRAMs. While the critical path descriptions in [10,11] apply well to small cache's used at the time [10] was published, they becomes increasingly incapable of addressing the complexities of two dimensional hierarchical array division, seen more recently in [12,13,14], as SRAM cache array densities evolve along the trends observed in Figure 1.

2. Hierarchical Array Division

Partitioning of an undivided array (Fig 2) into global vertical blocks (Fig. 3) results in shorter Word Lines (WLs). Shorter WLs translate into a smaller WL RC delay [15] but increase the wire lengths driven by stages preceding the WL driver. Further division and subdivision of the array into sub-global and sub-sub-global blocks, referred to as 'hierarchical word line division' in [16], was demonstrated to improve the WL decode delay in subsequent implementations [17] by limiting the increases in wire-lengths and total capacitance along the decode network. Increasing the number of hierarchical divisions beyond a certain point, however, increases the fan-out of decode stages as well as the total area consumed by them [15]. Larger fan-outs and area constraints on the maximum W/L ratios of decode stages imposed by array efficiency requirements, directly translate into an increasing gate delay penalty to decode a WL. Balancing decreasing RC delays due to shorter wire-lengths with increasing gate delays due to larger fan-outs and area limited W/L ratios yields an optimal hierarchical division of the WL for a prescribed area constraint.

Hierarchically dividing the Bit Line (BL) into shorter local BL permits the same differential voltage across a local BL pair to develop sooner due to a smaller local BL capacitance seen by the cell [18]. Improvements in local BL delay obtained from reductions in BL length are limited by increases in the area consumed by local sense amplifiers and the delay of the global BL.

A rigorous analytical model that captures the fan-out, wire-length and area dependence of the decode network and the bit path on the architecture of the hierarchical array division would enable us to quantify the trade-offs and precisely identify optimal array configurations.

3. A Generic Critical Path Model

3.1. Array Architecture

The SRAM array of size **M** bits is assumed to be divided into identically sized **nm** 'macros' (Figure 4) where each macro can be considered an independent RAM [8]. Each macro stores part of the accessed word or a 'subword' with all macros activated simultaneously to yield the entire accessed word, $\mathbf{n_{IO}}$ bits wide. Each macro is assumed to be divided into $\mathbf{n_1}$ global subarrays given by (1a) in the Appendix, where $\mathbf{n_{1x}}$ equals the number of rows of global subarrays and $\mathbf{n_{1y}}$ equals the number of columns of global subarrays as seen in Figure 4. Each global subarray may be further divided into $\mathbf{n_2}$ subglobal subarrays in (1b). Similarly, further division of the subglobal subarray or subarrays further down the hierarchy are modeled in (1c), where the subscript, **i**, indicates the index of the hierarchical division. The smallest building block of this hierarchical division of the macro – the local subarray, is referred to as a **block** and has a width equal to the length of a word line, $\mathbf{N_c}$ cells wide and a height of $\mathbf{N_r}$ cells - equal to the length of a local bit-line, $\mathbf{N_{BL}}$ cells tall, times the number of BL divisions per block, $\mathbf{d_{BL}}$ (2). The total number of blocks in a macro is given by (3). The size, in bits, of each macro is given by (4). The WL decode network selects one of $\mathbf{N_{WL}}$ word lines given by (5a). The column decode network selects one of n_{BL} BL pairs given by (5b).

3.2. Decode Critical Path

The CMOS critical path circuit architecture to decode a WL may be completely defined with six parameters: (i) Choice of logic gates in each decode stage (ii) Logic depth (iii) Fan-in of each decode stage (iv) Fan-out of each decode stage (v) Geometries and resistivity of wires driven by each decode stage and (vi) Device widths of N and P channel devices in each gate along the decode path.

Choice of logic gates: Most CMOS SRAMs that implement the decode network with static CMOS use a decode stage

[‡] This work was supported by the Defense Advanced Research Project Agency (Contract: F3361595C1623), the Semiconductor Research Corporation (SJ-374-002) and by IBM Corporation

comprising of a static CMOS NAND gate followed by an inverter [9]. Alternatives to static CMOS such as clocked decoding are commonly used as well. We assume static CMOS gates in this analysis for simplicity.

Logic Depth: The logic depth, n_{cp} along any path of the decode network (6) is given by both - the number of WLs to be decoded, N_{WL} as well as the average fan-in, $\overline{f_{in}}$ rounded to the nearest integer, of the NAND gates along that decode path. Two additional gates are added to account for the double inverter in the address input buffer. The number of address bits required to decode one of N_{WL} word lines is given by (7), where $\mathbf{N_Z}$ is the number of address bits required to decode one of $\mathbf{n_B}$ blocks ad $\mathbf{N_X}$ is the number of address bits required to decode one of $\mathbf{N_r}$ rows in each block. The number of decode stages along an $N_{bits} - N_{WL}$ decode network is given by (8).

Fan-in: A fan-in of two minimizes the delay of the decode network [8,9] (Figure 5) because every additional fan-in for a decode stage not only increases the delay of the NAND gate, but also doubles it's fan-out, increasing the total capacitance encountered along any single path while lowering the number of possible decode paths. Thus, every NAND gate between predecoder and word line is assumed to have a fan-in of two in this analysis.

Fan-out & wire-length: The fan-outs of each decode stage and also the maximum wire-lengths driven by each stage are determined by modeling the number of times a decode stage is replicated horizontally and vertically across the entire macro as a function of how the macro is divided hierarchically in two dimensions. The replica counts of each decode stage are modeled by imposing as boundary conditions, the replica counts of the address input buffers or the replica count of the WL. The replica counts of stage k, ${}^{k}nrep$ with the stage identified by the left superscript, equals the product of its vertical replica count, ${}^{k}nrep_{v}$ and its horizontal replica count, ${}^{k}nrep_{h}$ (9). In Figure 7 (inverters following the NAND gates are not shown for simplicity), for example, gate **A** drives a word line and has a replica count of 32, which is a product of a vertical replica count of 8 and a horizontal replica count of 4. The vertical or horizontal replica counts of each stage is equal to the product of the vertical or horizontal replica counts of the decode stages that drive each of it's inputs (10). Replica counts and their horizontal and vertical components of all gates in the example decode network of Figure 7 are listed in Table I. An address input buffer has a replica count of two since an address and its complement must both be available for a decoder. The address input buffers for address bits that decode one of several horizontal selections such as: N_r, n_{1x}, n_{2x} etc., are thus assumed to replicate twice vertically (11a). Similarly, the replica counts of the address input buffers for address bits that decode one of several vertical selections such as: n_{1y}, n_{2y} etc. are assumed to replicate horizontally twice (11b). *The fan-out of a decode stage is calculated as the ratio of the replica count of the driven stage to the replica count of the stage whose fan-out is desired (12).* The fan-outs of the stages in Fig. 7 are listed in Table I. *The maximum horizontal or vertical wire-length between two decode stages along any decode path equals the difference in horizontal or vertical pitch (14) of the two decode stages* (Figure 6). Since wire networks assume a Manhattan layout, the maximum interconnect length between two decode stages is given by the sum of the maximum horizontal and vertical wire-lengths between these stages. (Wire-lengths of Fig. 7 are listed in Table I in units of cell dimensions, m_h, m_w. This calculation assumes a WL length of 128 cells).

Device widths: The loads driven along the WL decode network are given by (15)-(16) with gate beta ratios and expressions for delay given by (19). Optimal device widths that minimize total delay along the decode path depend on the gate used and the parasitic wiring being driven by each gate [9].The optimal device width is calculated using (20)-(21). Device and Interconnect process parameters assumed in the examples are listed in Table 2.

The model for replica count of each decode stage, the W/L ratio for the NFET devices and the beta ratio of each gate permits us to estimate the area of a stage along all decode paths using an empirical model from [8] given in (22). Although (22) was developed in [8] for WL drivers, it could apply reasonably well to other decode stages since they too, like the WL driver comprise of a 2-input NAND gate followed by an inverter. The total area consumed by decode stages along any decode path is given by (28).

3.3. BL Architecture and delay path model

The local bitline delay is modeled, as in [9] in (23), where ΔV equals the sensitivity of the local sense amplifier, typically 200mV or less [9] and I_{read} equals the read current that flows through the two series connected NFETs in the memory cell. The local BL capacitance, the total capacitance at the output of the multiplexer that drives the input to the local sense amplifier, and the capacitance seen at the input to the local sense amplifier is modeled in (24-26). The local sense amplifier drives its output across the height of the macro along a global BL with a delay given by (27a). The total BL delay is given by the sum of (23) and (27b). This sum captures the dependence of BL delay on the architecture of the horizontal division of the block and the macro.

4. Optimal Array Division

The critical path models for decode and sense operations are verified with published SRAM arrays in Table 4. The device/interconnect parameters reported in these publications are listed in Table 2 and their array architectures in Table 3 (Figures 8-11). The WL RC response time improves with smaller N_c due to a shorter WL length. The BL response time improves with smaller N_r due to a smaller BL capacitance. Shrinking N_c lowers power dissipation as well since a smaller N_c directly translates into fewer cells dissipating read current during an access. Lowering N_r also improves power dissipation since a smaller BL capacitance requires less recovery power after a write operation. Thus, lowering N_c and N_r yields substantial improvements in the energy consumed by a data read operation. *The array architecture defined by its hierarchical division in two dimensions into $\mathbf{n_B}$ blocks (3) primarily determines the overheads in gate and interconnect delays that accompany reductions in N_c and N_r.* Determining each of the horizontal and vertical hierarchical components of $\mathbf{n_B}$ so that the sum of the gate and interconnect delay components are at a minimum leads to an optimal array architecture. Figs. 8c,9c & 11 plot the dependencies of decode and sense delay as N_c and N_r are decreased.

5. Summary and Conclusions

New, physical and generic models for SRAM access time are reported. These couple the hierarchical architecture of the array with the definition of the decode and sense critical paths. Delays predicted by these models were verified with published data. Optimal array architectures that minimize the decode and sense delays by balancing decreasing interconnect delays with increasing fan-out and area limited gate delays are calculated using the proposed models.

6. References

[1] B. Prince, "Semiconductor Memories – A Handbook of Design, Manufacture and Application," 2nd Ed., New York, NY, J. Wiley, 1991, Chapter 4.

[2] S. Flannagan, "Future Technology Trends for Static RAMs, 1988 IEDM Digest Of Technical Papers, Dec. 1988, pp. 40-43.

[3] B. Bakoglu, "Circuits, Interconnects and Packaging for VLSI", Reading MA, Addison-Wesley, 1990, Chapter 1.

[4] D. Burger, "System Level Implications of Processor-Memory Integration", Proceedings of the 24th International Symposium on Computer Architecture, June 1997, pp. 1-10.

[5] B. Prince, "High Performance Memories – New Architecture DRAMs and SRAMs, Evolution and Function," New York, NY, J. Wiley, 1996, Chapter 1.

[6] M. T. Bohr, "Interconnect scaling - the real limiter to high performance ULSI," *Solid State Tech*. Vol. 39 No. 9 pp. 105-108, Sep 1996.

[7] Grant McFarland, "CMOS Technology Scaling and its Impact on Cache Delay," Ph.D. Dissertation, 1998, Stanford University.

[8] B. Amruthur and M. Horowitz, "Speed and Power Scaling of SRAMs", IEEE Journal of Solid State Circuits, Vol. 35, No. 2, pp. 175-185, Feb 2000.

[9] B. Bateman, "High Speed SRAM Design," 1998 ISSCC Tutorial Slides.

[10] T. Wada & S Pryzbylski, "An Analytical Access Time Model for On-chip Cache Memories", IEEE Journal of Solid State Circuits, Vol. 27, No. 5, pp. 1147-1156, Aug 1992

[11] S. Wilton and N. Jouppi, "An Enhanced Cache Access and Cycle Time Model," IEEE Journal of Solid State Circuits, Vol. 31, No. 5, pp. 677-687, May 1996.

[12] H. Pilo, A. Allen, J. Covino, P. Hansen, S. Lamphier, C. Murphy, T. Traver, and P. Yee, "An 833 MHz 1.5-W 18Mb CMOS SRAM with 1.67Gb/s/pin", IEEE Journal of Solid-State Circuits, Vol. 35, No. 11, pp.1641-1647, Nov. 2000.

[13] C. Zhao et al, "An 18Mb, 12.3GB/s CMOS Pipeline-Burst Cache SRAM with 1.54Gb/s/pin", ISSCC Digest of Technical Papers, Feb 1999, pp. 200-201.

[14] J. Hill & J. Lachman, "A 900MHz 2.25MB Cache with On-chip CPU – Now in Cu SOI", ISSCC Digest of Technical Papers, Feb. 2001, pp. 176-177.

[15] M. Yashimoto et al, "A Divided Word-Line Structure in the Static RAM and Its Application to a 64Kb Full CMOS RAM", IEEE Journal of Solid-State Circuits, Vol. SC-18, No. 5, pp. 479-484, Oct. 1983.

[16] T. Hirose et al, "A 20ns 4-Mb CMOS SRAM with Hierarchical Decoding Architecture," IEEE Journal of Solid State Circuits, Vol. 25, No. 5, pp. 1068-1077, Oct. 1990.

[17] K. Itoh et al, "Trends in Low-Power RAM Circuit Technologies", Proceedings of the IEEE, Vol. 83, No. 4, pp. 524-543, Apr. 1995.

[18] R. Guo, T. Su, and C. Chao, "A 500MHz 1Mb On-Chip Cache Design Using Multi-Level Bit Line Sensing Scheme", 1996 Symposium on VLSI Circuits Dig. Of Tech. Papers, Jun. 1996, pp. 130-131.

[19] T Sakurai & R Newton, ''Delay models for series connected MOSFET structures'' IEEE Journal of Solid State Circuits, Vol. 28, No. 1, pp 40-48, Jan 1993.

[20] A. Bhavnagarwala, B. Austin, K. Bowman and J. Meindl, "A Minimum Power Methodology for Projecting Limits on CMOS GSI", IEEE Transactions on VLSI Systems, Vol. 8, No.3, pp. 235-251, June 2000.

[21] 1999 International Technology Roadmap for Semiconductors, SIA Handbook

[22] C. Mead and L. Conway, " Introduction to VLSI Systems", Reading PA, Addison Wesley, 1980, Chapter 4.

[23] T. Sakurai, 'Approximation of Wiring Delay in MOSFET LSI', *IEEE Journal of Solid State Circuits*, Vol. SC-18, No. 4, pp. 418-426, Aug 1983

Table 1: Replica counts, fan-outs and maximum wire-lengths in units of cell dimensions driven by decode stages for example in Figure 8

Gate	$nrep$	$nrep_v$	$nrep_h$	f_{out}	$L\max$	$L\max_v$	$L\max_h$
A	32	8	4	1	128	0	128
B	8	8	1	4	384	0	384
C	4	1	4	8	7	7	0
Z	2	1	2	2	128	0	128
X	2	2	1	4	3	3	0

Table 2: Please see next page

Table 4: Comparison of references with new model

	[15]	New model	[16]	New model	[13]	New model
t_{WL}	-	9.5ns	7.8ns	8.6ns	.95ns	1.1ns
t_{BL}	-	2.1ns	2.0ns	2.3ns	-	.16ns
n_{1y}	8	4	4	4	8	8
n_{1x}	1	(1)	1	(1)	1	(1)
n_{2y}	1	(1)	8	8	1	(1)
n_{2x}	1	(1)	1	(1)	8	8
N_c	32	32	128	128	256	256
N_r	256	(256)	1024	(1024)	256	256

Table 3: Array Architecture parameters for Examples in Figures 9-11

Symbol	Parameters as listed in the References		
	1	2	3
	[15]	[16]	[13]
M	64Kb	4Mb	18Mb
n_{IO}	8b	4b, 1b	72b
$nm_x \times nm_y$	1x1	1x1	2x2
$n_{1x} \times n_{1y}$	1x8	1x4	1x8
$n_{2x} \times n_{2y}$	1x1	1x8	8x1
$n_{3x} \times n_{3y}$	1x1	1x1	1x1
$n_{4x} \times n_{4y}$	1x1	1x1	1x1
n_B	8	32	64
d_{BL}	1	1	2
N_r	256	1024	256
N_c	32	128	256
N_{WL}	2048	32768	16384
N_{straps}	None	None	8
m_w	21.0µm	3.5µm	2.22µm
m_h	24.5µm	5.3µm	2.52µm
${}^{WL}L_{frac}$	1.0	1.0	0.5

Table 2: Device & interconnect parameter inputs for Figures 9-11. Parameters in parenthesis are calculated/estimated

Symbol	Parameter	[15]	[16]	[13]
L, Leff	Channel length	2μm	0.6 μm	0.18, 0.14 μm
t_{ox}	Gate oxide thickness	400 A	150 A	30 A
V_{dd}	Supply voltage	5 V	3.3 V	1.5 V
V_{to}	Device threshold voltage	(1.0 V)	(0.8 V)	0.32 V
c_{go}	Min. geom.. device Gate oxide capacitance	(3.45 fF)	(0.828 fF)	(0.373 fF)
c_{do}	Min. geom.. device junction capacitance	(2.0 fF)	(0.51 fF)	(0.145fF)
$I_{dsatn,0}$	Min. geom.. N device Drain sat. current	(290 μA/μm)	(505 μA/μm)	0.95mA/μm
$I_{dsatp,0}$	Min. geom. P device Drain sat. current	(104 μa/μm)	(278 μA/μm)	410μA/μm
V_{dsat}	Maximum NFETDrain saturation voltage	(1.93 V)	(1.15V)	(0.684V)
N_a	NFET substrate doping concentration	(2.98×10^{16} cm^{-3})	(1.466×10^{17} cm^{-3})	($3.8\times10^{17}cm^{-3}$)
ρ_{WL}, ρ_n	Resistivity of metal level n	Poly, Salicide	Salicide, Al	Salicide, Al
ζ_n	Capacitance per unit length	2.1pF/cm		
T_n	Thickness of metal level n	(2μm)	(1.0μm)	250nm (poly WL)
				700nm (M3 BL, WL)
				1600nm (M5 GBL, GWL)
Λ_n	Aspect ratio of metal level n	1	2	(1) poly WL
				2.2 (M3: WL,BL)

Appendix:

(1a) $n_1 = n_{1x} \times n_{1y}$; (1b) $n_2 = n_{2x} \times n_{2y}$ (1c) $n_i = n_{ix} \times n_{iy}$ (2) $N_r = N_{BL} \times d_{BL}$ (3) $n_B = n_1 \times n_2 \times ..\times 1 = \prod_i n_i$ (4) $\frac{M}{nm} = n_B \times N_c \times N_r$

(5a) $N_{WL} = n_B \times N_r$ (5b) $n_{BL} = \frac{N_c \times n_B \times nm \times d_{BL}}{n_{IO}}$ (6) $n_{cp} = 2 \times \log_{f_{in}} \left| \log_2 \left| N_{WL} \right| \right| + 2 = 2 \times \log_{f_{in}} \left| \log_2 \left| n_B \times N_r \right| \right| + 2$

(7) $N_{bits} = \log_2 \left| N_{WL} \right| = \log_2 \left| n_B \times N_r \right| = \log_2 \left| n_B \right| + \log_2 \left| N_r \right| = N_Z + N_X$ (8) $N_{stages} = \log_{f_{in}} \left| N_{bits} \right|$ (9) ${}^k nrep = {}^k nrep_v \times {}^k nrep_h$

(10a) ${}^k nrep_v = \prod_{k f_{in}} {}^{k-1} nrep_v$ (10b) ${}^k nrep_h = \prod_{k f_{in}} {}^{k-1} nrep_h$ (11a) ${}^1 nrep_v = 2$, ${}^1 nrep_h = 1$ (11b) ${}^1 nrep_v = 1$, ${}^1 nrep_h = 2$

(12) ${}^{WL} nrep_v = n_{1x} \times n_{2x} \times ...\times 1 \times N_r$, ${}^{WL} nrep_h = n_{1y} \times n_{2y} \times ..1$ (13) ${}^k f_{out} = \frac{{}^{k+1} nrep}{{}^k nrep}$ (14a) ${}^k L\max_v = n_{1x} \times n_{2x} \times ....\times 1 \times N_r \times m_h \times \left(\frac{1}{{}^k nrep_v} - \frac{1}{{}^{k+1} nrep_v} \right)$

(14b) ${}^k L\max_h = n_{1y} \times n_{2y} \times ....\times 1 \times N_c \times m_w \times \left(\frac{1}{{}^k nrep_h} - \frac{1}{{}^{k+1} nrep_h} \right)$ (14c) ${}^k L\max = {}^k L\max_v + {}^k L\max_h$

(15a) ${}^{WL} C_{total} = N_c \left(2c_{go} + m_w \times \left[{}^{WL}\zeta_n + {}^{WL}\zeta_p \right] \right)$ (15b) ${}^{WL}\Re_{total} = \frac{{}^{WL}R_n \times {}^{WL}R_p \times m_w \times N_c}{\left({}^{WL}R_n + {}^{WL}R_p \right) \times N_{straps}}$

(16) ${}^p C_{total} = {}^{p+1}k_n c_{go} \left(1 + {}^{p+1}\beta \right) \times {}^p f_{out} + {}^p k_n c_{do} \left(1 + {}^p\beta \times {}^p f_{in} \right) + {}^p L\max \times {}^p\zeta_n$ (17) ${}^p\beta = \frac{I_{dsatn,0}}{{}^p f_{in,eff} \times I_{dsatp,0}}$

(18a) $f_{ineff} = 1 + \frac{2\left(f_{in} - 1\right) V_{dsat} \left(1 - \frac{1}{\sqrt{2}} \right) (1 + \kappa)}{\left(V_{dd} - V_{th} \right) - \frac{V_{dsat}}{2}}$ (18b) $\kappa = \frac{1}{C_{ox}} \sqrt{\frac{q N_a \varepsilon_{si}}{4\phi_f}}$ (19) ${}^p t_{pd} = \frac{{}^p C_{total} V_{dd}}{{}^p k_n \times I_{dsatn,0}} \times {}^p f_{in,eff}$

(20) $\left({}^p k_{n,opt} \right)^2 = \frac{{}^{p-1}k_{n,opt} \left[{}^{p+1}k_{n,opt} \times \left(1 + {}^{p+1}\beta \right) \times {}^p f_{out} \times c_{go} + {}^p k_n c_{do} \left(1 + {}^p\beta \times {}^p f_{in} \right) + {}^p L\max \times {}^p\zeta \right]}{\left(1 + {}^p\beta \right) \times {}^{p-1} f_{out} \times c_{go}} \times \frac{{}^p f_{in,eff}}{{}^{p-1} f_{in,eff}}$

(21) $\left({}^{n_{cp}} k_{n,opt} \right)^2 = \frac{{}^{n_{cp}-1}k_{n,opt} \left[{}^{WL} C_{total} \right]}{\left(1 + {}^{n_{cp}}\beta \right) \times {}^{n_{cp}-1} f_{out} \times c_{go}} \times \frac{{}^{n_{cp}} f_{in,eff}}{{}^{n_{cp}-1} f_{in,eff}}$ (22) ${}^k A = \left[7.5 \times {}^k k_n \left(1 + {}^k\beta \right) + 124.25 \right] L^2$

(23) $t_{lBL} \cong \frac{\left(C_{lBL} + C_{mux} + C_{LSA} \right) \times \Delta V}{I_{read}}$ (24) $C_{lBL} = \frac{N_r}{d_{BL}} \left[c_{do} + \zeta_{BL} \times m_h \right]$ (25) $C_{mux} = \frac{N_c \times nm \times d_{BL}}{n_{IO}} \times \left(1 + \beta_{mux} \right) \times c_{do} \times {}^{mux} k_n$

(26) $C_{LSA} = c_{do} \times {}^{LSA} k_n$ (27a) $t_{gBL} \cong \frac{\left(C_{gBL} + C_{GSA} + C_{Gmux} \right) \times \Delta V}{I_{LSA}}$ (27b) $C_{gBL} = {}^{gBL}\zeta_n \times \left[\left(n_{1x} \times n_{2x} \times ..\times 1 \right) - 1 \right] \times N_r \times m_h$

(28) $A_{total} = \sum_k {}^k A \times {}^k nrep = L^2 \sum_k \left[7.5 \times {}^k k_n \left(1 + {}^k\beta \right) + 124.25 \right] \times {}^k nrep$

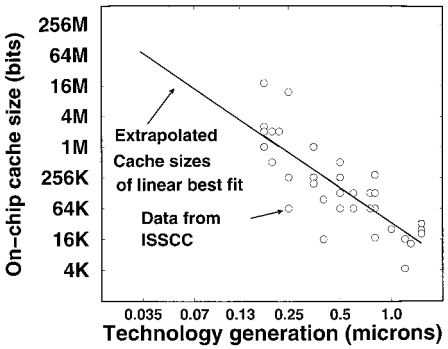


Figure 1: Historical trends in embedded SRAM cache density.

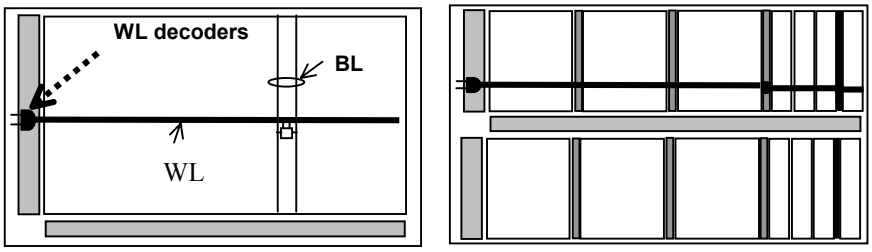


Figure 2,3: Undivided and divided arrays.

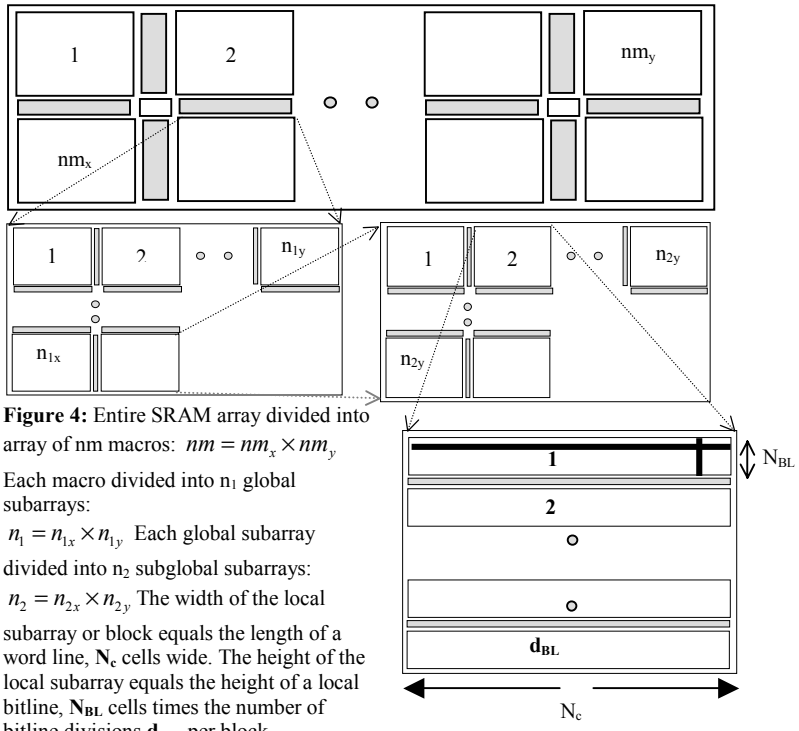


Figure 4: Entire SRAM array divided into array of nm macros: $nm = nm_x \times nm_y$ Each macro divided into n_1 global subarrays: $n_1 = n_{1x} \times n_{1y}$ Each global subarray divided into n_2 subglobal subarrays: $n_2 = n_{2x} \times n_{2y}$ The width of the local subarray or block equals the length of a word line, $\mathbf{N_c}$ cells wide. The height of the local subarray equals the height of a local bitline, $\mathbf{N_{BL}}$ cells times the number of bitline divisions $\mathbf{d_{BL}}$, per block

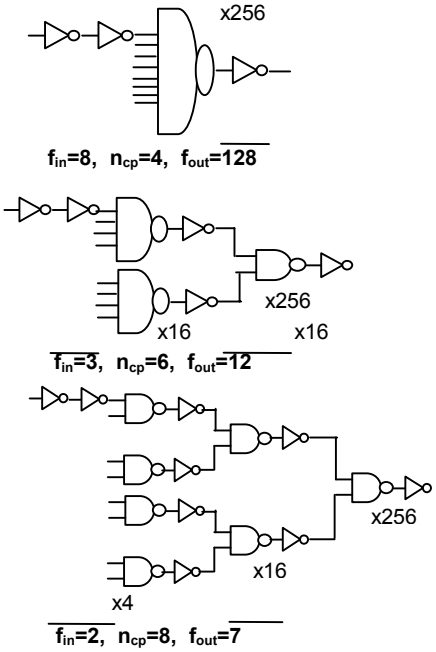


Figure 5: Logic depth (n_{cp}) and average fan-out dependence on fan-in of decode network

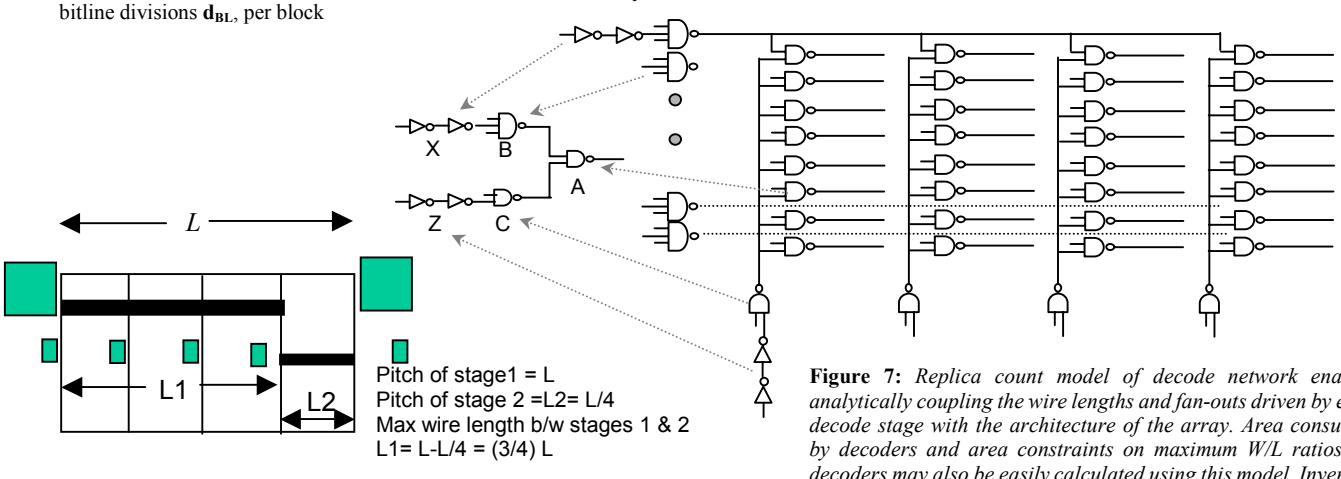
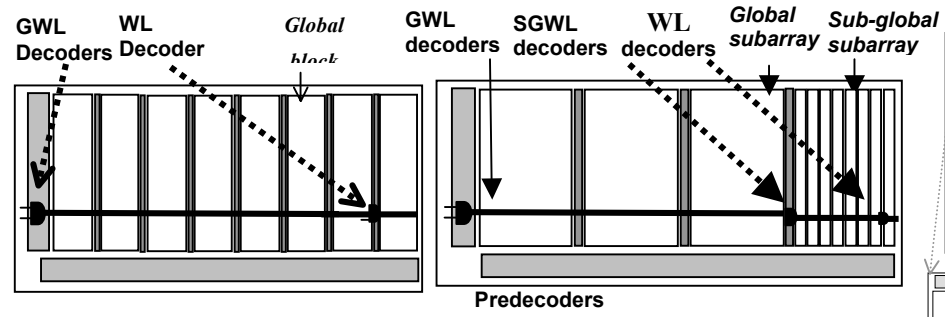
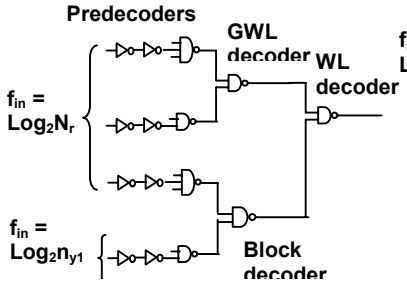
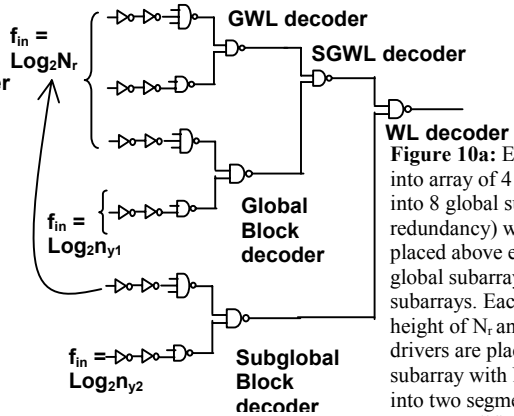


Figure 6: *Maximum wire length between decode stages equals their difference in pitch*

Figure 7: *Replica count model of decode network enables analytically coupling the wire lengths and fan-outs driven by each decode stage with the architecture of the array. Area consumed by decoders and area constraints on maximum W/L ratios for decoders may also be easily calculated using this model. Inverters following NAND gates are not shown for simplicity.*







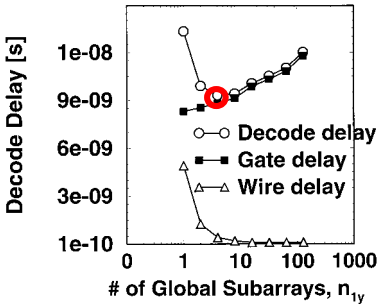


Figure8a,b: Array architecture assumed in from [15]. Decode network realizing this array architecture. Inverters following NAND gates not shown for simplicity.
Figure 8c: Decode delay dependence on number of global blocks. Decode delay decreases initially as the number of blocks increases due to smaller interconnect delays but begins increasing with increasing gate delays as fan-out in an area-constrained environment begins to increase. Optimal number of global subarrays corresponds to the minimum in total decode delay.

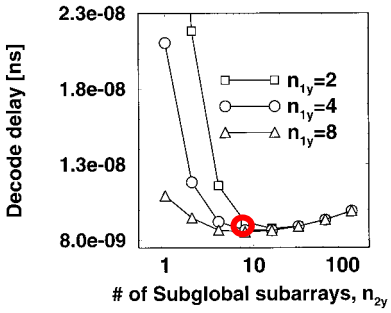


Figure 9a,b: Array architecture and decode network from [16]. Inverters following NAND gates not shown for simplicity. **Figure 9c**: Decode delay dependence on number of global (n_{1y}) and subglobal (n_{2y}) subarrays in Example 2. Dotted lines correspond to a constant total number of blocks.

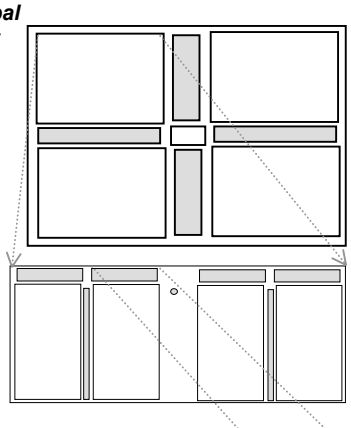
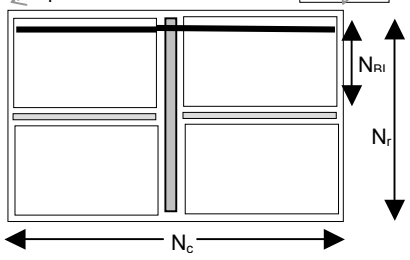


Figure 10a: Entire SRAM in [13] divided into array of 4 macros. Each macro divided into 8 global subarrays (not including redundancy) with global sense amplifiers placed above each global subarray. Each global subarray divided into n_2 subglobal subarrays. Each subglobal subarray has a height of N_r and width of N_c. The WL drivers are placed at the center of the subarray with N_{frac} = 2. The BL is divided into two segments with d_{BL}=2 with local sense amplifiers placed between these.



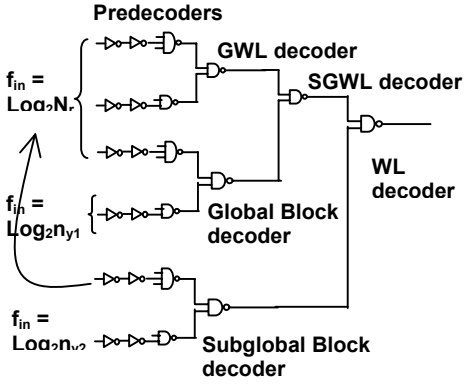
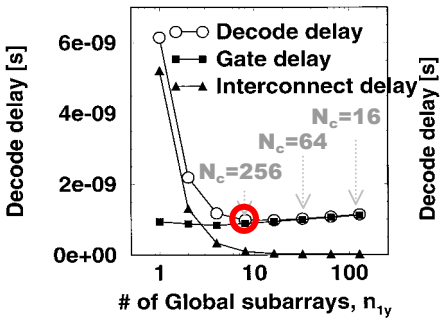
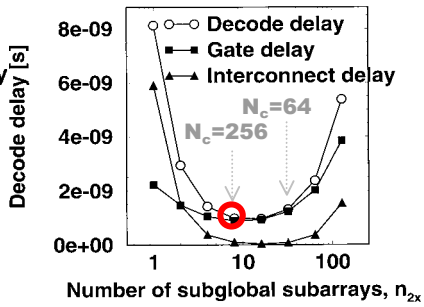


Figure 10b: Decode network of the SRAM array from [13] (Inverters following NAND gates are not shown for simplicity)





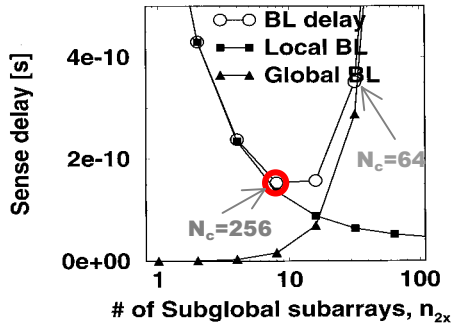


Figure 11a: Decode delay dependence on number of global subarrays, n_{1y} in Fig. 11
Figure 11b: Decode delay dependence on the number of subglobal subarrays n_{2x} in Fig. 11
Figure 11c (at right): Sense delay dependence on the number of subglobal subarrays n_{2x} in Fig. 10.