

Design and Implementation of 4Kb SRAM Array Controller with Optimized Memory Decoders at 28nm Technology

S. Naveen Kumar, K.S.R.Krishna Prasad, Ranjith Gonela Dibyendu Kayal

NIT, Warangal

sriramnk417@gmail.com krish@nitw.ac.inranjith.vlsi.12@gmail.comdiby85@gmail.com

Abstract: Optimization of SRAM (Static Random Access Memory) array design can be done at three domains namely bit cell optimization, sense amplifier optimization and memory decoder optimization. In this paper, we focused on memory decoder optimization. The objective of the paper is to design speed and power efficient memory decoder structure and to implement 4Kb SRAM array controller. We compared four NAND gate based decoder structures at TSMC 28nm technology and OR style NAND decoder structure is found to be efficient. We also implemented 4Kb SRAM array controller which is a byte accessible using binary decoder tree. All the logics have been designed using Cadence Virtuoso schematic editor and simulated using Spectre simulator with operating voltage of 0.9 V.

Keywords: SRAM, decoders, controller, logical effort, sizing.

I. INTRODUCTION

In high-performance SRAM memory arrays, the SRAM cells occupy only around 50% of the total transistor width. Power dissipation—particularly leakage power at the SRAM cell level, has been analyzed extensively and a number of circuit techniques have been proposed to reduce leakage in memory arrays [2], [3]. However, less effort has been made to reduce the power dissipation in memory peripheral circuitry, i.e., the decoders and read and write circuits. In addition to the power dissipation, a significant portion of the total access time is required for word line (WL) decode, particularly as designs migrate to shorter bit-lines due to increasing leakage and performance demands. This paper presents best WL decoder design at 28nm technology that improves access time and simultaneously reduces both active and leakage power. We also discuss about the working principle of 4Kb SRAM array controller.

A. WORDLINE DECODERS

The purpose of decoders is to reduce number of address lines. The logical function of decoder is simple 2^n n-input AND function. But there are number of ways to implement the AND function. The decoder designer has two major tasks: choosing the circuit style and sizing the resulting gates, including adding buffers if needed. Memory designs typically use WL address decoders implemented with static gates. These are high fan-out circuits and logical effort analyses show that a tree of two and three input NAND and NOR gates along with inverters offers the minimum delay [4]. Faster decoders can be implemented by reducing the logical effort.

II. DECODER SIZING

Sutherland and Sproull [8], [9] have proposed an approach called logical effort that allows one to quickly solve sizing problems for more complex circuits. We have used their approach to calculate theoretical delays. The basic delay model they use is quite simple, yet it is reasonably accurate. It assumes that the delay of a gate is the sum of two terms. The first term is called the effort delay and is a linear function of the gate's fan-out, the ratio of the gate's output capacitance to its input capacitance. This term models the delay caused by the gate current charging or discharging the load capacitance. The second term is the parasitic delay. It models the delay needed to charge/discharge the gate's internal parasitic capacitance. Since the parasitic are proportional to

the transistor sizes, this delay does not change with gate sizing or load. Thus using this model, the delay of a gate is simply

$$T_{gate} = le \left(\frac{C_{out}}{C_{in}} \right) + T_{par}$$

Here le is the logical effort of the gate which is the ratio of input capacitance of gate to input capacitance of an inverter with equal drive strengths.

III. DIVIDED WORD LINE ARCHITECTURE

During an access to some row, the word line activates all the cells in that row and the desired sub-word is accessed via the column multiplexers. This arrangement has two drawbacks for macros that have a very large number of columns: the word line RC delay grows as the square of the number of cells in the row, and bit line power grows linearly with the number of columns.

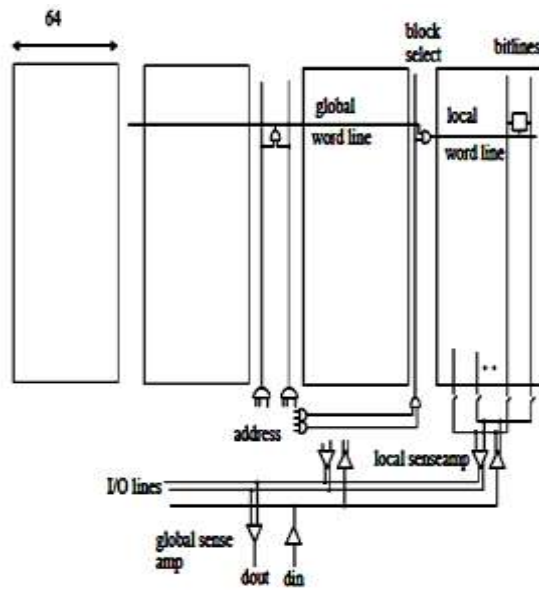


Fig.2 Divided Word Line Architecture

Both these drawbacks can be overcome by further sub dividing the macros into smaller blocks of cells using the Divided Word Line (DWL) technique first proposed by Yoshimoto. In the DWL technique the long word line of a conventional array is broken up into k sections, with each section activated independently thus reducing the word line length by k and hence reducing its RC delay by k^2 .

IV. "NAND" GATE STRUCTURES

The n -input AND function can be implemented via different combination of NANDs, NORs, and inverters. Since in current CMOS technology, a pMOS is at least 1.3 times slower than an nMOS, a conventional NOR gate with series pMOS is very inefficient and so the AND function is usually best achieved by a combination of NANDs and inverters. If we use k -input NAND gates with a logical effort of $le(k)$, then we will need $\log_k n$ levels to make the n -input NAND function, resulting in a total logical effort $le(k)^{\log_k n}$. We have used following NAND gate structures along with conventional series stack NAND gate in the decoders.

A. Source Coupled NAND2

It has least logical effort almost equal to that of inverter. One of the inputs is connected to source terminal of nMOS. If B is '0' and A is '1' then Y is less than V_{dd} due to threshold drop of nMOS.

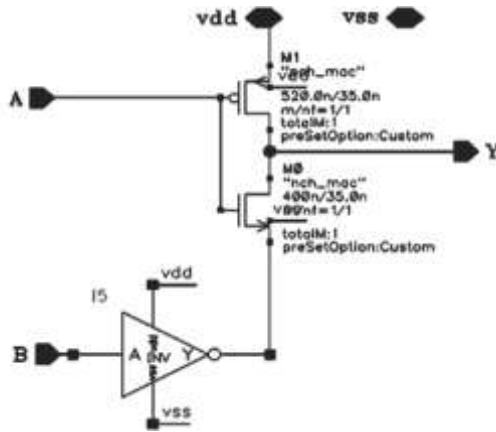


Fig.3: Source Coupled NAND2

B.Dynamic OR style NAND3

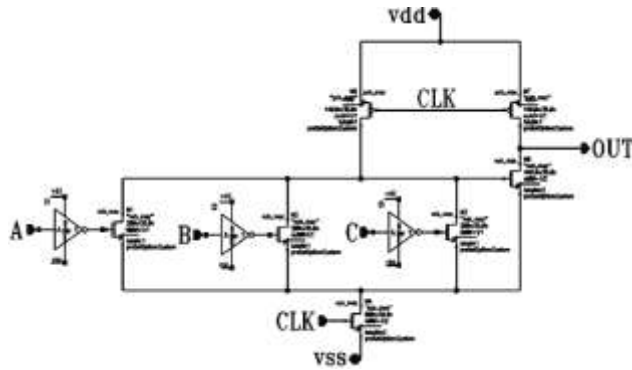


Fig.4: Dynamic OR style NAND3

It has larger fan-in due to OR based style. During pre-charge CLK is '0' and OUT is pre-charged to '1'. During evaluation CLK is '1' and if atleast one input is '0' then OUT is '1' (Vdd). And if all inputs are '0' then OUT is '1' (< Vdd) and if all inputs are '1' then OUT is '0'.

C.Dynamic NAND3

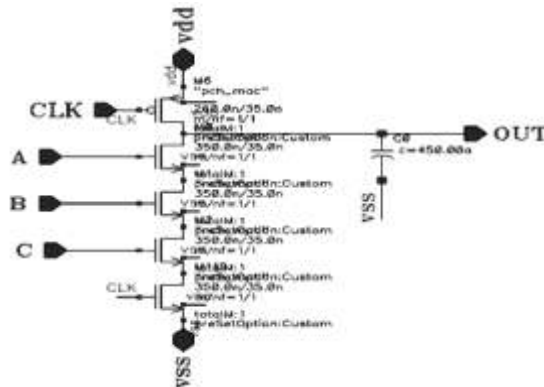
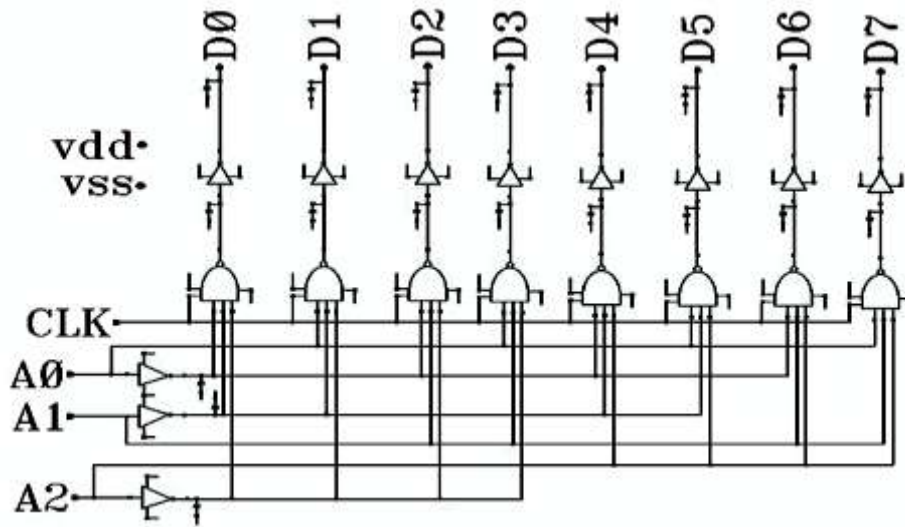


Fig.5: Dynamic NAND3

It consumes less power since short circuit power dissipation is zero. But it has stack of nMOS transistors which gives more delay.

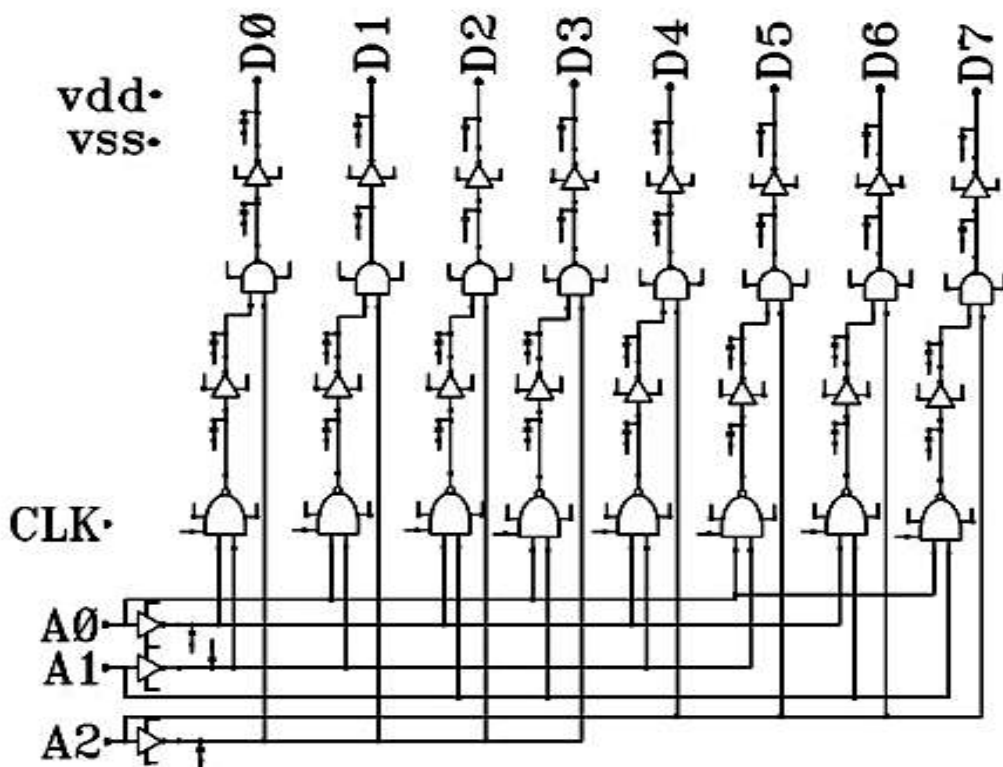
V. SCHEMATIC DESIGNS OF DECODER STRUCTURES

A.3:8 Pre-decoder using NAND3OR



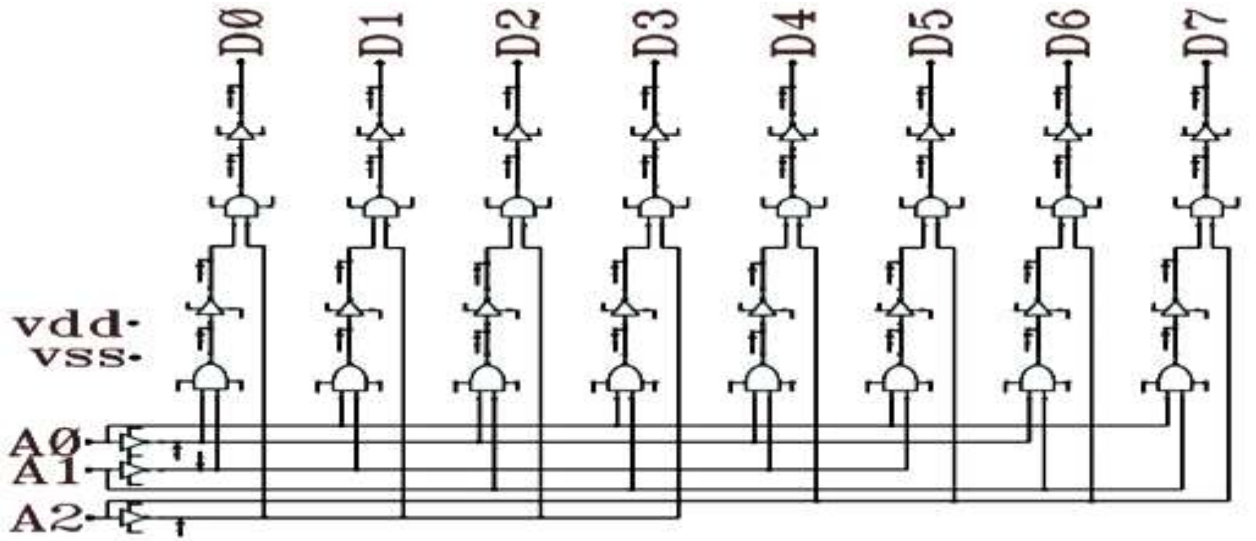
Logic depth: INV-NAND3-INV

B. 3:8 Pre-decoder using NAND2OR+ NANS2SC



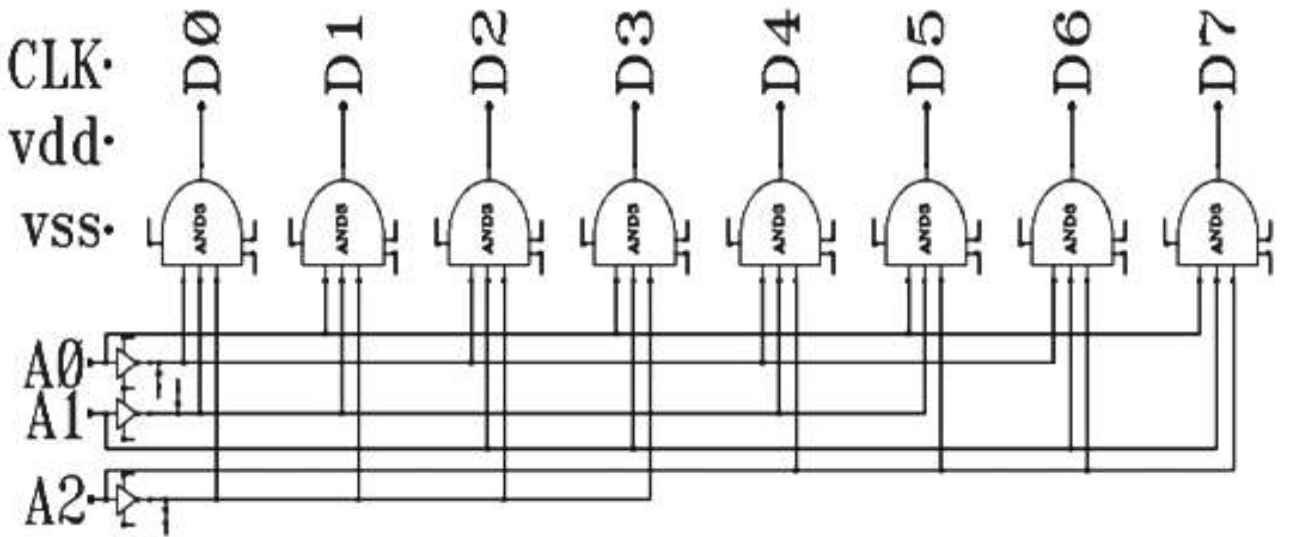
Logic depth: INV-NAND2OR-INV-NAND2SC-INV

C. 3:8 Pre-Decoder using CMOS-NAND2 +NAND2SC



Logic depth:INV-NAND2 (CMOS)-INV-NAND2SC-INV

D.3:8 Pre-decoder using Dynamic NAND3 gate



Logic depth:INV-NAND3 (dyn) +INV (AND3)

E.COMPARISON OF 3:8 PRE-DECODERS FOR $C_L=3.5fF$

Pre_Decoder Type	Avg_Delay(ps)	I_avg(uA)	P_avg(uW)	vdd(V),vss(V),Temp©
Dynamic NAND3OR	18.9	14	12.6	0.9,0,27
Dyn. NAND2OR+NAND2SC	37.8	15.8	14.3	0.9,0,27
Conv.NAND2+NAND2SC	39.9	10.2	9.19	0.9,0,27
Dynamic NAND3	30.8	10.2	9.15	0.9,0,27

We have designed same decoder structures as above for 4:16 pre-decoders for capacitive load of 7fF. The delays and power consumption values for four decoder structures are tabulated below.

F. COMPARISON OF 4:16 PRE-DECODERS FOR $C_L=7fF$

Pre_Decoder Type	Avg_Delay(ps)	I_avg(uA)	P_avg(uW)	vdd(V),vss(V),Temp@
Dynamic NAND4OR	19.5	25.4	22.9	0.9,0,27
Dyn. NAND3OR+NAND2SC	34.4	38.2	34.4	0.9,0,27
Conv.NAND2+NAND2SC	35.1	19.6	17.7	0.9,0,27
Dynamic NAND4	34.9	12.5	11.3	0.9,0,27

VI. SCHEMATIC DESIGNS OF MEMORY CONTROL CIRCUITS

A. Single Port 6-T SRAM bit cell

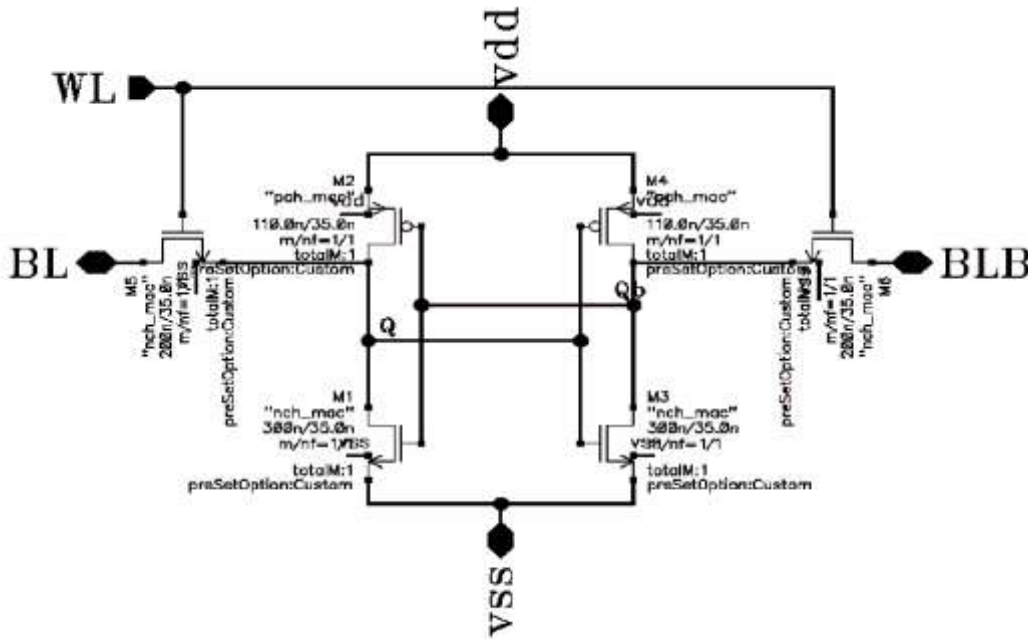


Fig.6 Single Port 6-T SRAM bit cell

For read stability, $cell\ ratio = \frac{W_{1,3}}{W_{5,6}} \cong 1.5$

For write ability, $pull\ up\ ratio = \frac{W_{2,4}}{W_{5,6}} \cong 0.55$

Transistor	Width(nm)	Length(nm)
Access(M5,M6)	200	35
Pull-up (M2,M4)	110	35
Pull-down(M1,M3)	300	35

B. WORD LINE CONTROLLER OF 4Kb SRAM ARRAY

WL controller controls enabling WLs of 4*64 array. It has two internal 2:4 decoders which generate WRE and RDE signals for each row. The outputs of external decoders are given to WL generate circuit. Then the Block control circuit will enable WLs of each row of 4*64 array using two outputs of WL generate circuit.

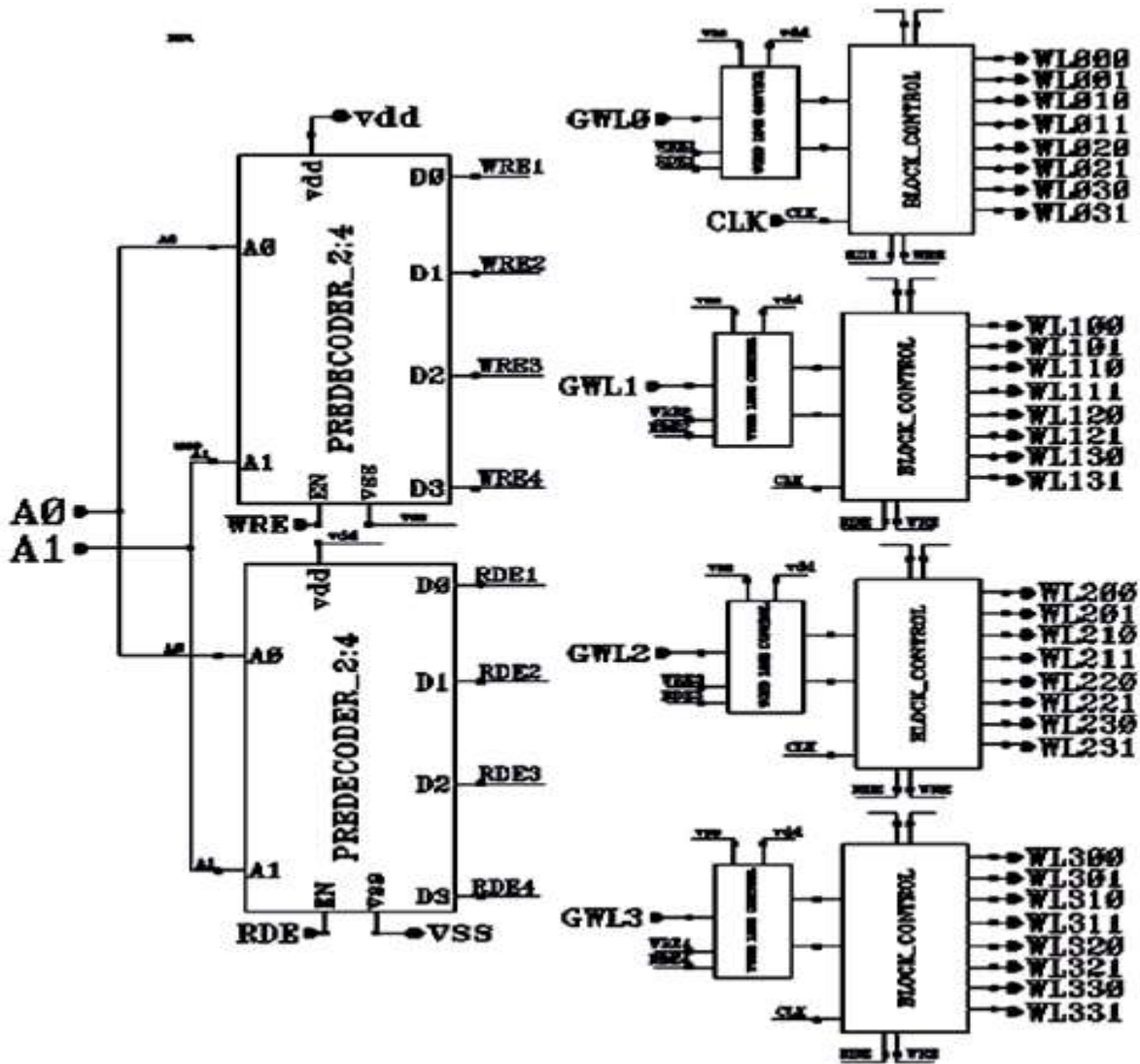


Fig.7 Word line controller of 4kb SRAM array

C. BIT LINE CONTROLLER OF 4Kb SRAM ARRAY

It has pre-charge circuitry, input circuitry and output circuitry connected to bit line pairs. Operations of all these circuits are mutually exclusive. The pre-charge signal is $PC=CLK+WRE$. So in data retention mode bit lines are pre-charged to VDD. In the output circuitry, there are sense amplifier circuits connected to each bit line pair. During write operation only input circuit is ON. During read operation, only output circuit is ON. There are data latch circuits in input circuitry to drive input data onto complementary bit lines. Bit lines are pre-charged in between two read operations. Output circuitry has latch based sense amplifiers. Each sense amplifier is connected to each pair of complementary bit lines. We calculated a wire capacitance of 9.7fF on each bit line and the calculated device capacitance is 32fF per bit line. So total bit line capacitance is about 42fF.

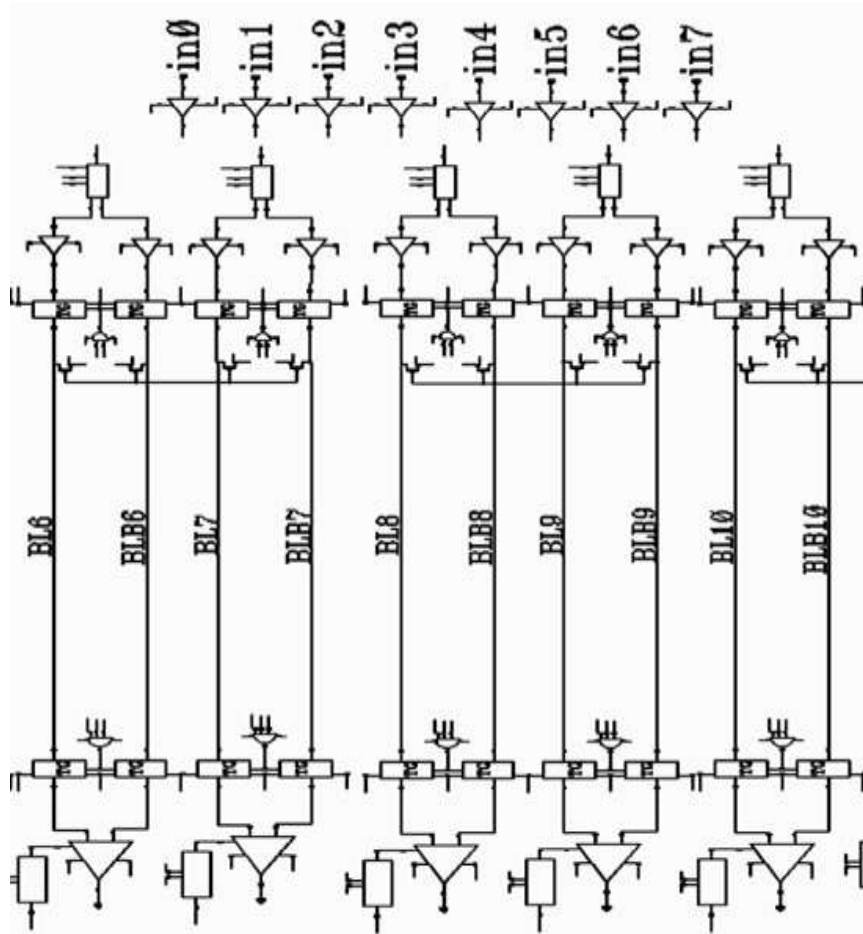


Fig.8 Bit line controller of 4Kb SRAM array

D. WRITE TIMING WAVEFORMS

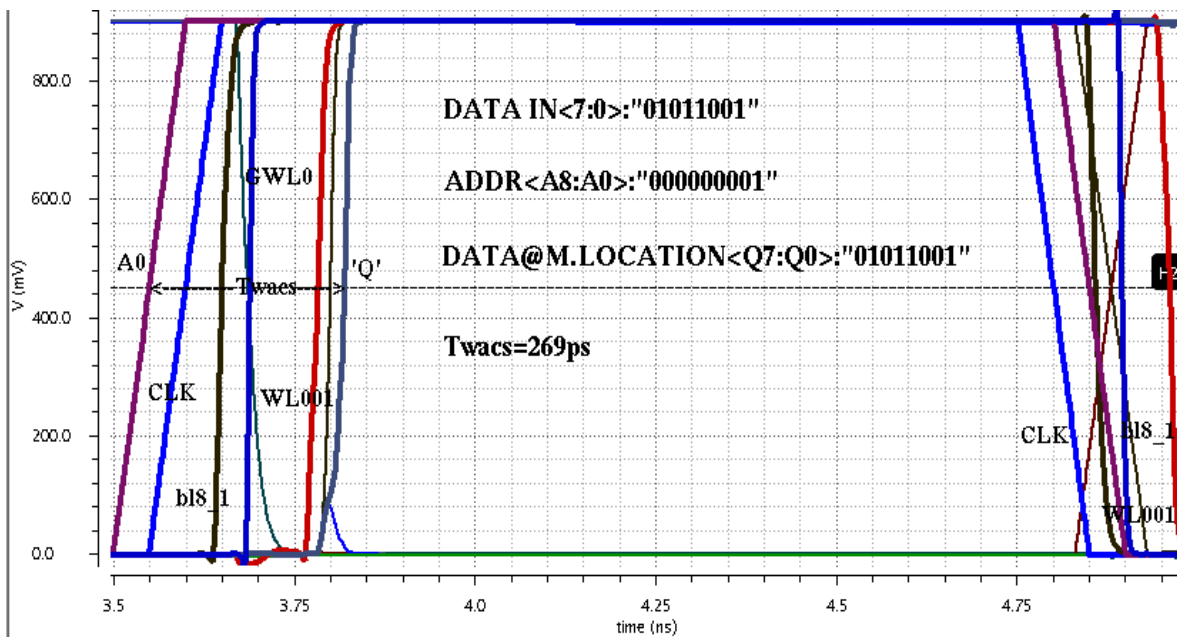


Fig.9 Write access time of 269ps for Address of "00000001"

E.READ TIMING WAVEFORMS

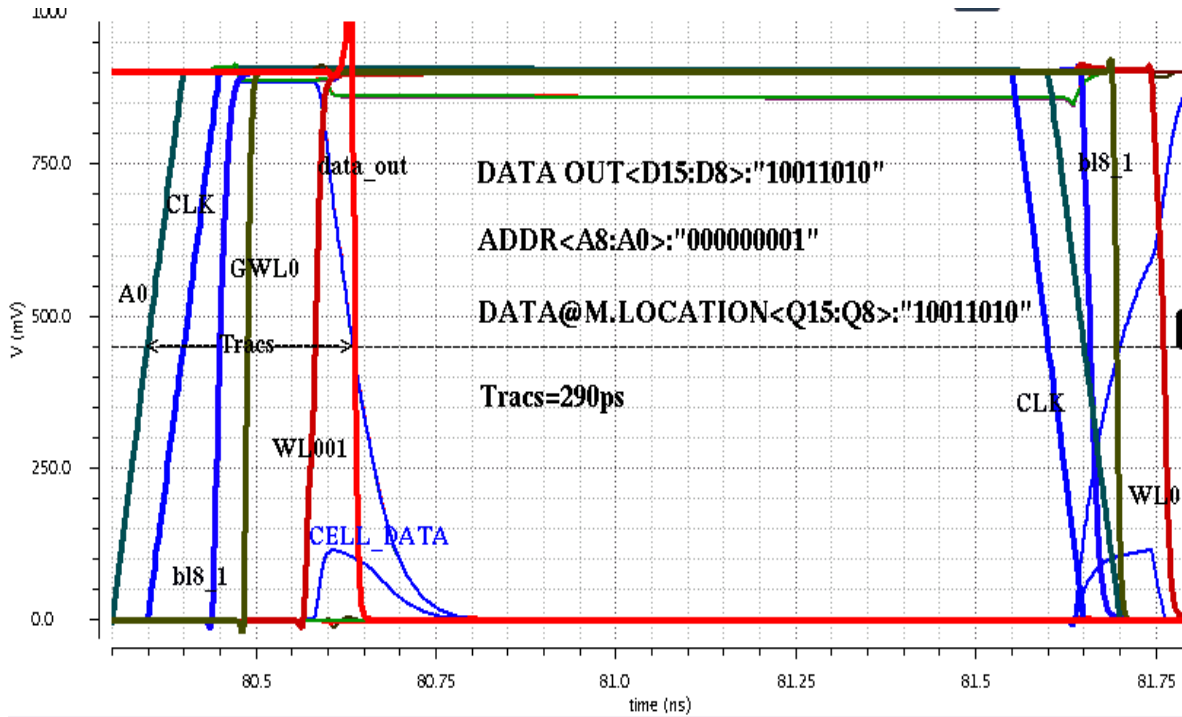
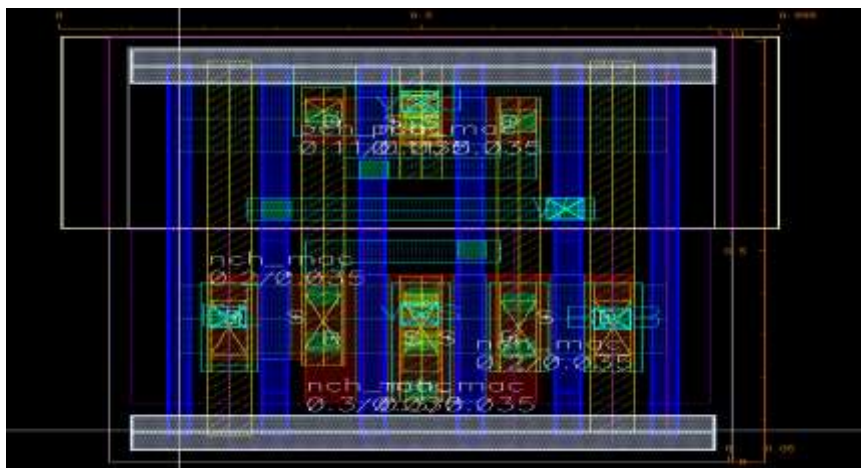


Fig.10 Read access time of 290ps for Address of “000000001”

Table 1. Component delay contribution of 4Kb array controller

Operation	Decoder delay(ps)	WL controller delay(ps)	Cell delay(ps)	Total delay(ps)
Writing I Location	139	95	25	259
Writing II Loc.	139	94	36	269
Reading I location	139	105	56	300
Reading II Loc.	139	91	60	290

F.LAYOUT OF 6-T SRAM BITCELL



Memory bit cell has to occupy less area. So its layout has to be done compactly. Here Area of the bit-cell layout is about 1 sq.µm. Each bit line length is 0.885µm. Adjacent bit line spacing is 0.07µm. Width of bit line is 0.06µm. So, bit line wire and contact capacitance calculated from tsmcN28 parasitic RC model is 0.15fF. Here the WL is routed horizontally and complementary bit lines are routed vertically.

II. CONCLUSION

On comparing four NAND based decoder structures at 28nm technology, dynamic OR style NAND based decoder structure is faster with moderate power consumption. The 4Kb array controller is giving write access time of 263ps and read access time of 295ps on an average. The decoder delay is 139ps at 28nm. The power consumption for a write and read access is about 0.64mW.

ACKNOWLEDGMENT

The proposed work was carried out in Startup Company working on ASIC based designs in Hyderabad. This work was greatly supported by National Institute of Technology, Warangal.

REFERENCES

- [1] Samson, G.; Ananthapadmanabhan, N.; Badrudduza, S.A.; Clark, L.T., "Low-Power Dynamic Memory Word Line Decoding for Static Random Access Memories," *Solid-State Circuits, IEEE Journal of*, vol.43, no.11, pp.2524,2532, Nov. 2008
- [2] H. Mizuno and T. Nagano, "Driving source-line cell architecture for sub-1V high-speed low-power applications," *IEEE J. Solid-State Circuits*, vol. 31, no. 3, pp. 552–557, Mar. 1996.
- [3] K. Zhang, "SRAM design on 65-nm CMOS technology with dynamic sleep transistor for leakage reduction," *IEEE J. Solid-State Circuits*, vol. 40, no. 4, pp. 895–901, Apr. 2005.
- [4] Amrutur, B.S.; Horowitz, M.A., "Fast low-power decoders for RAMs," *Solid-State Circuits, IEEE Journal of*, vol.36, no.10, pp.1506,1515, Oct 2001
- [5] Park Sooil; Song Jae Yeol; Park Seok Hwi; Jung Ji Hoon, "Design of memory controller Design of general purpose memory controller," *SoC Design Conference, 2008. ISOC '08. International*
- [6] Design & Analysis of 1-Kb 6T SRAM Using Different Architecture; *KumkumVerma Sanjay Kumar JaiswalDheeraj Jain VijendraMaurya* 2012 Fourth International Conference on Computational Intelligence and Communication Networks.
- [7] Design, Simulation and Characterization of Memory Cell Array for Low Power SRAM using 90nm CMOS Technology. *HirdayaNarain Mishra, and Yashwanta Kumar Patel*
- [8] I. E. Sutherland and R. F. Sproull, "Logical effort: Designing for speed on the back of an envelope," *Advanced Res. VLSI*, pp. 1–16, 1991.
- [9] I. Sutherland *et al.*, *Logical Effort: Designing fast CMOS circuits*, 1st ed. San Mateo, CA: Morgan Kaufmann, 1999.