A

Dissertation

on

DESIGN OF HIGH SPEED & LOW POWER SRAM DECODER

Submitted towards the partial fulfillment of requirement for the award of degree of

Master of Technology

in

VLSI Design

Submitted by

Shivkaran Jain

Roll No. 601161008

Under the guidance of

Mr. Arun Kumar Chatterjee

Assistant Professor, ECED

Thapar University, Patiala

ELECTRONICS AND COMMUNICATION ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
(Established under the section 3 of UGC Act, 1956)
PATIALA–147

DECLARATION

I hereby declare that the dissertation entitled "**Design of High speed and Low power Decoder**" is an authentic record of my study carried out as requirement for the award of **Master of Technology** in **VLSI Design** at **Thapar University**, Patiala, under the supervision of **Mr. Arun Kumar Chatterjee**, Assistant Professor, ECED.

The matter embedded in this report has not been submitted in any other University/Institute for the award of any degree.

SK Jain

Date: 12-07-2013

Shivkaran Jain
Roll No. 601161008

It is certified that the above statement made by the student is correct to the best of my knowledge and belief.

Arun Kumar Chatterjee.

Date: 12/7/2013

Arun Kumar Chatterjee
Assistant Professor, ECED
Thapar University, Patiala

Countersigned by:

Head
Electronics and communication
Engineering Department
Thapar University, Patiala

Dean of Academic Affairs
Thapar University, Patiala

ACKNOWLEDGEMENT

First of all, I would like to express my gratitude to **Mr. Arun kumar Chatterjee**, Assistant Professor, Electronics & Communication Engineering Department, Thapar University, Patiala for his guidance and support throughout this research work. I am very fortunate to have the opportunity to work with him.

I am thankful to the **Head of Department, Professor (Dr.) Rajesh Khanna** and **PG Coordinator, Dr. Kulbir Singh** (**Associate Professor**) of Electronics & Communication Engineering Department for their encouragement and inspiration for the execution of this research work.

I am also thankful to the entire faculty and staff of Electronics & Communication Engineering Department for the help and moral support which went along the way for the successful completion of this work.

Next, I would like to thank Ms. Madhu Chatterjee, Mr. Anil Singh and my friends for all the good times in the lab and for their help, criticisms and suggestions which make everyday a pleasant one. Thanks so much to all of you for the fun and great memories here at Thapar University.

Finally and above everyone else, my heartfelt thanks and life-long gratitude goes to my parents and my loving sister for their love, affection, constant support and encouragement.

I am also thankful to God who bestowed upon his grace and always with me whenever I felt lonely.

Shivkaran Jain

601161008

ABSTRACT

This dissertation presents the design of an 8 to 256 line decoder for a 256 x 256 6T SRAM array. Here the objective is to design and compare the speed and power efficient memory decoder. In order to achieve this task firstly, various NAND gate architectures have been designed namely conventional NAND gate, NOR style NAND gate and source coupled NAND gate. These architectures are simulated at 180nm technology node on cadence tool [Appendix B]. For schematic entry cadence composer is used, for simulation Cadence Spectre 5.1.4 1_ISR and for layout Cadence Virtuoso 5.1.4 1_ISR is used and its DRC, LVS and RCX has been done using Cadence Assura 3.2.0.

Next, three decoders have been designed using conventional NAND gate(4 to 16), NOR style NAND gate and source coupled NAND 8 to 256 gates and their critical paths have been obtained. The circuit techniques used to reduce the power dissipation and delay have been explored. It has been observed that NOR style NAND gate is 96.34% faster and 93.6 % more power efficient than conventional NAND gate. Source coupled NAND gate has been found to be 96.56% faster and 96% more power efficient than conventional NAND gate and 6.25% faster and 42.86% power efficient than NOR style NAND gate. When these gates are used in decoder designs, it has been observed that source coupled NAND decoder gives the best performance and has better power efficiency as compared to other decoder structures. It has been found that NOR style NAND decoder is 94.4% faster and 67% power efficient, whereas source coupled NAND decoder is 94.6% faster and 81.6% power efficient than conventional NAND gate decoder(4 to 16). As compared to NOR style NAND based decoder, source coupled NAND decoder is 5% faster and 44.2% power efficient.

Finally, delay power product obtained for the three decoders are 390μ $W-$ ns, 72.27μ $W-$ ns and 38.64μ $W-$ ns respectively.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER

1

INTRODUCTION

1.1 Introduction:

To continue to grow according to the Moore's law, it is desired to reduce the cost per transistor or the cost per function performed by a microchip. In search for better alternatives so as to reduce power consumption in a chip, we moved from transistor-transistor logic (TTL) and emitter coupled logic (ECL) to complementary metal oxide semiconductor (CMOS) technology. Initially working with CMOS was excellent since unlike earlier technologies, there was almost negligible static power dissipation. But as we continue to scale down the technology and continue to shrink the transistor sizes, inherent parasitic passive elements have started to surface and disturbed our harmony with standard CMOS logic style. As we continue to reduce both the supply voltage and threshold voltage of a MOSFET, sub threshold leakage is bound to increase. Low power electronics is especially important in the domain of memory design. Since memory is one component found on almost every digital system that involves some processing no matter how small. Currently memories constitute a large portion of the chip area and if we can reduce the power consumption of memories, we will be able to affect a almost all types of systems. Memory is a very important part of a computer system. Modern memory system implements the concept of the memory hierarchy. While a flat memory built using a single technology and single concept is lucrative for its simplicity, a well-implemented hierarchy allows a memory system to approach simultaneously the performance of the fastest component, the cost per bit of the cheapest component, and the energy consumption of the most energy efficient component. A memory in terms of computer hardware is a storage unit. There are many different types of hardware used for storage, such as magnetic hard drives and tapes, optical discs such as CDs and DVDs, and electronic memory in form of integrated memory or stand-alone chips [1].

In modern CMOS circuits, the static random-access memory (SRAM) is widely used as on-chip cache. The design of SRAM is generally divided into two parts, the decoder, and the sense and column circuits. For a read access operation, the decoder delay contributes for up to half of the total access time and a significant fraction of the total SRAM power consumption [2]. An overview of a synchronous SRAM architecture with size of 2^m by N is shown in Figure 1.1 that consists of a row decoder, a storage array, a column decoder, and a sense amplifier (SA) and write-control unit [2].The conventional design for the SRAM storage cell circuit is based on 6 MOS transistors (6-TSRAM) as shown in Figure 1.2 [3].

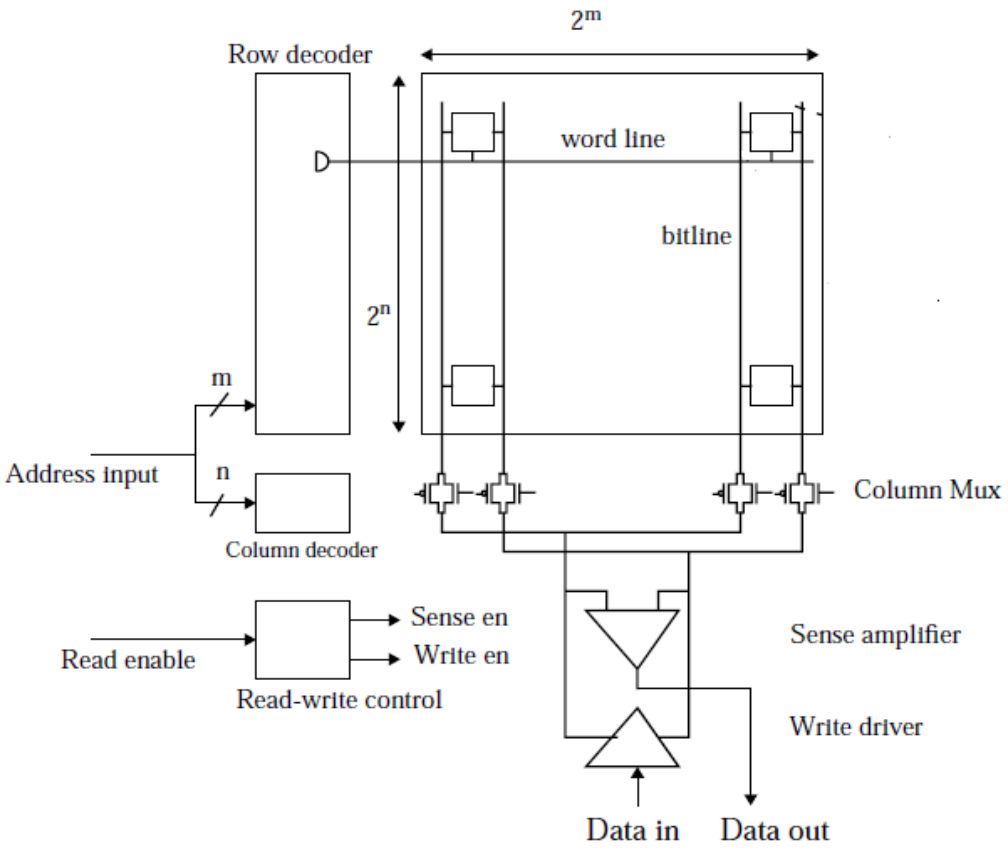

Figure 1.1 SRAM architecture [2]

A typical SRAM cell is made up of six MOSFETs. Each bit in an SRAM is stored on four transistors (M1, M2, M3 and M4) that form two cross-coupled inverters. This storage cell has two stable states which are used to denote 0 and 1. Two additional access transistors serve to control the access to a storage cell during read and write operations. In addition to such six-transistor (6T) SRAM, other kinds of SRAM chips use 4T, 8T, 10T, or more transistors per bit. During the read operation, wordline is activated to turn on M5 and M6

transistors. If the cell is in state 1 the signal on BL is high and the signal on BL' is low. The opposite is true if the cell is in state 0.Thus BL and BL' are always complements of each other.

In computer memory, a sense amplifier is one of the elements which make up the circuitry on a semiconductor memory chip (integrated circuit). It is part of the read circuitry which is used when data is read from the chip. The job of a sense amplifier is to sense the logic levels from a bit line which represents a data bit (1 or 0) stored in a memory cell on the chip, and amplify the small voltage swing to recognizable logic levels so the data can be interpreted properly at the output terminal of the chip. Sense amplifier circuits consist of 2 to 6 (usually 4) transistors. There is one sense amplifier for each column of memory cells, so there are usually hundreds or thousands of identical sense amplifiers on a chip [4].

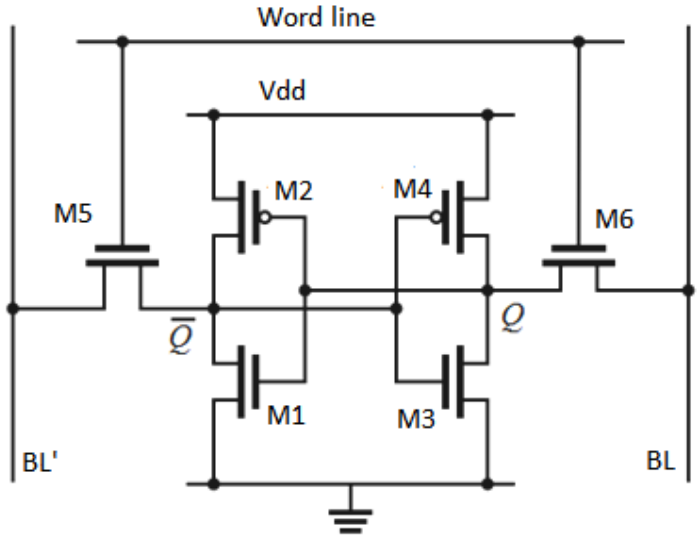

Figure 1.2 SRAM storage cell circuit based on 6 MOS transistors [3]

Decoders are used to assert the word lines based on the input addresses. The decoders are designed to drive the capacitance of the word lines and have small power dissipation. Logically the decoder is equivalent to 2^n n-input AND gates where the AND operation is performed in a hierarchical manner. The decoder structure mainly consists of an initial predecoder stage, where a group of address inputs are decoded to give the predecoder outputs. The column decoder takes m inputs and produces 2^m bit line access signals, of which 1, 4, 8, 16, 32, or 64 may be enabled at one time. The bit selection is done using a multiplexer circuit to direct the corresponding cell outputs to data registers. In total, $2^n X\ 2^m$ cells are stored in the core array.

In this report the main concern is about the designing and optimization of decoder circuit. For the purposes of design and optimization, the access path can be divided into two portions: the row decoder - the path from the address inputs to the word line, and the data path - the portion from the memory cell ports to the SRAM I/O ports. The decoder design problem has two major tasks: determining the optimal circuit style and decode structure and finding the optimal sizes for the circuits and the amount of buffering at each level. In this report we will look at various circuit techniques that have been proposed to speed up the decode path and analyze their delay and power characteristics. This will eventually enable us to sketch the optimal decode structures to achieve fast and low power operation

1.2 Specifications of SRAM

- Size of SRAM array : 256 x 256 (6T SRAM)
- Total number of arrays: 4
- No. of address lines : 10
- Size of row decoder : 8 to 256
- Size of column decoder : 2 to 4
- Size of a SRAM cell : 40λ X 40λ
- Access transistor width : 4λ
- Technology used : 180nm

1.3 Objective:

An 8 to 256 line row decoder for a 256 X 256 6T-SRAM array has been designed according to the following steps:

- Various two input NAND gate architectures have been studied and simulated at 0.18μm CMOS technology.
- The circuit techniques, used to reduce the power dissipation and delay have been explored.
- The NAND, NOR STYLE NAND and NOT gates have been optimized for FO4 loads. The optimized gates are characterized on Cadence.
- The delay and power dissipation of critical path for the multistage decoder using conventional NAND gate(4 to 16), NOR Style NAND and source coupled NAND have been obtained.

- Finally, the results obtained from different styles are compared.

1.4 Organization of dissertation:

This dissertation comprises of five chapters in which first chapter is Introduction which contains a brief history of memory design, specification of SRAM and objective of this dissertation. A brief description of research work done by various people throughout the world has been given in second chapter. Third chapter includes design of various NAND gate architectures and their power and delay calculations. Fourth chapter describes design of proposed decoder and gives their delay power calculations and comparison with various other decoder circuits. Finally the fifth chapter gives a conclusion based on decoder design and future scope.

CHAPTER

BASICS OF DECODER

In a memory, decoders are used to assert the word lines based on the input addresses. The decoders are designed to drive the capacitance of the word lines. Logically, a decoder is equivalent to n inputs and 2^n output lines. A Basic decoder structure may consists of AND gates and chain of inverters. A 2 to 4 line decoder structure using NAND gates is shown in Figure 2.1.

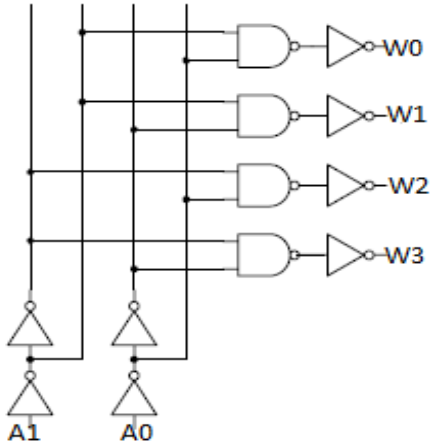

Figure 2.1 Basic 2 to 4 decoder circuit

2.1 Types of decoder:

The decoder structure mainly consists of an initial **predecoder** stage, where a group of address inputs are decoded to give the predecoder outputs. These are then combined at the next stage to drive the next stage of decoders called **postdecoder**. The main reason behind the partition of decoder into two stages is to reduce the logical effort of the complete path. As shown in Figure 2.2 inputs A0 - A3 and their compliments A0' - A3' are combined using a 4 to 16 predecoder. The two sets of 16 predecoded outputs are then combined to give the 256 outputs which drive the word lines. Basically decoder is used to access

memory in systems. Whenever there is a need to use some application input in binary form is given at the input of the decoder (termed as address). According to that address a particular cell from the memory array is accessed. Figure 2.2 shows an 8 to 256 decoder structure [5]. In decoders based on conventional gates one of the output is asserted based on the input address. For next address input the previous output first has to be deasserted and the new decoder output has to be asserted. This operation reduces the speed. This problem can be overcome by using pulsed decoder [6], in which output stays active for a minimum time and then shuts off. Here the gates used are conventional but input is given in the form of pulses of short duration. So before any access all the word lines are off to be guaranteed and the decoder activates one of the word lines.

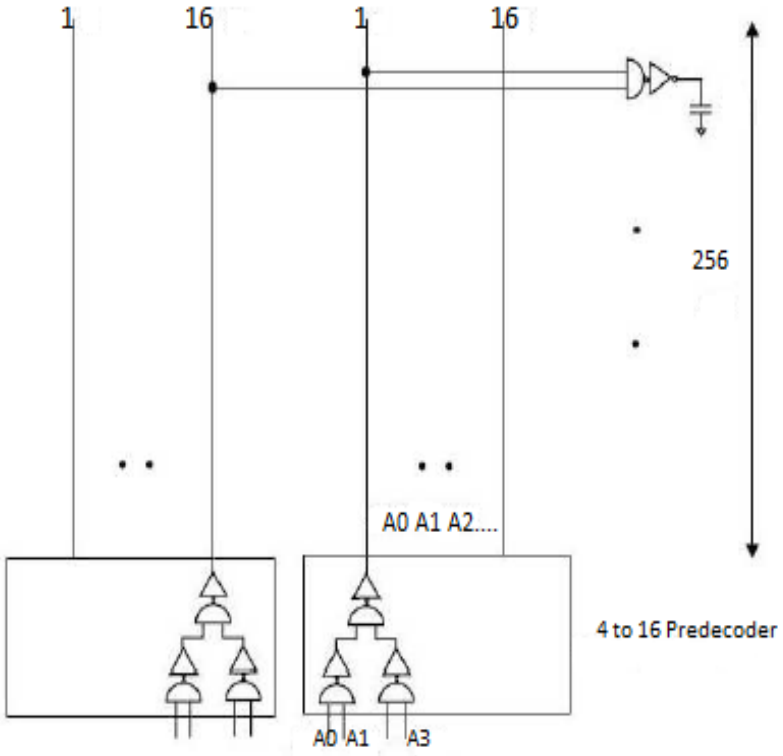

Figure 2.2 An 8 to 256 decoder structure[5]

2.2 Critical path of a decoder:

The path through the logic which determines the ultimate speed of the circuit is called critical path. Critical path for a decoder is the longest path from the address input to the output on the local word line. It is thus important to identify the critical path.

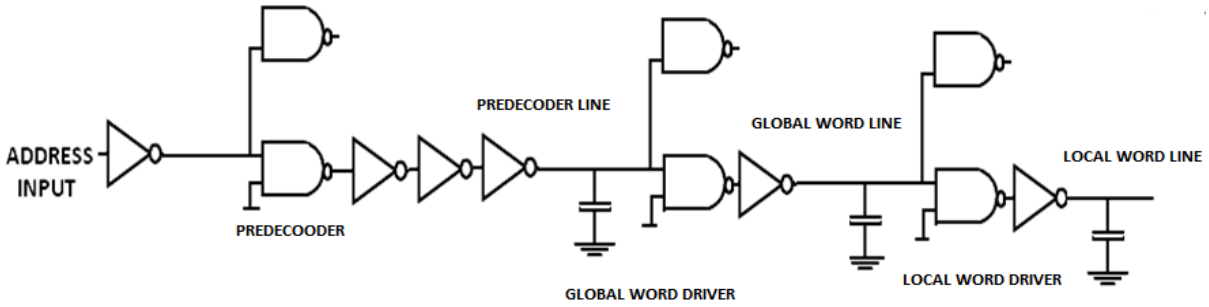

Figure 2.3 Critical path of memory decoder

Input address is applied at the predecoder stage which drives predecoder line after that it goes to global word driver that drives global word line. Finally goes to local word driver that drives local word line.

2.3 NAND gates for decoder:

2.3.1 Conventional Series Stack NAND gate:

In a conventional NAND gate the n-FETs and p-FETs are of same size. The two n-FETs are connected in series stack form (Figure 2.4). To obtain the same drive capability as that of an inverter the n-FETs are made twice as big as in an inverter. In decoders based on conventional gates one of the output is asserted based on the input address. For next address input the previous output first has to be deasserted and the new decoder output has to be asserted. This operation reduces the speed. The above problem can be overcome by using pulsed inputs to the conventional NAND gates so that output stays active for a minimum time and then shuts off.

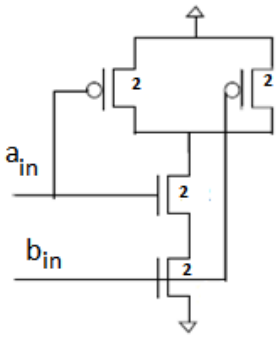

Figure 2.4 Conventional NAND gate [5]

Thus, the inputs are given in the form of pulses of short duration and so, before any access all the word lines are off is guaranteed. The decoder needs to activate one of the word lines and hence the performance is improved.

2.3.2 Nakamura NAND gate:

In this architecture sizes of the PMOS is halved to that of conventional NAND gate. This does not effect rise time because it is guaranteed that the inputs will be deasserted.

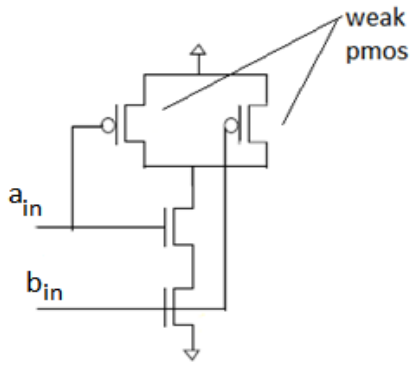

Figure 2.5 Nakamura NAND gate [5]

Problem in all of the above design styles is that if 3 input, 4 input or more input gate are used then charge sharing between output node and the intermediate node capacitance become significant which may degrade the performance and may cause erroneous outputs. Depending upon the value of internode capacitance the charge at output node can fall up to (V_{dd}/2). So depending on threashold value, '1' at output will become '0'.

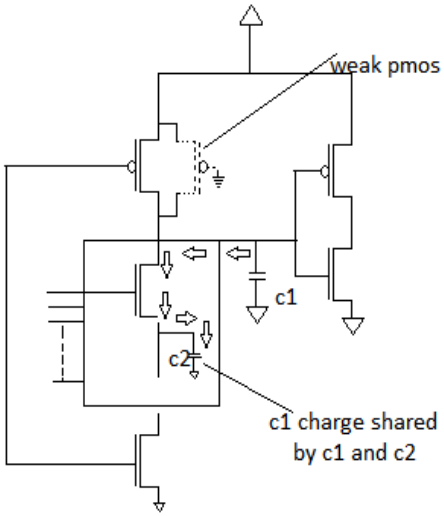

Figure 2.6 Charge sharing problem[7]

Adding a weak PMOS improves the performance but transistor count increases.

2.3.3 Source Coupled NAND gate:

In this architecture to implement two input NAND function only two transistors are needed. So area is very much reduced. NAND functionality is obtained by connecting one of the input to the gate terminal directly while inverted of the other input to the source terminal. One of the special attributes of this circuit is that it can be at least as fast as an inverter if the branching effort of the source input is sufficiently large and if the total capacitance on the source line is much greater than the output load capacitance for the gate. This is shown in Figure 2.8 and 2.9 where L represents output capacitances and an inverter of size w0 drives the inputs in both the circuits. In Figure 2.8, the output capacitance for inverter is a capacitance of value 'C' plus the gate capacitance of 'b' inverters.

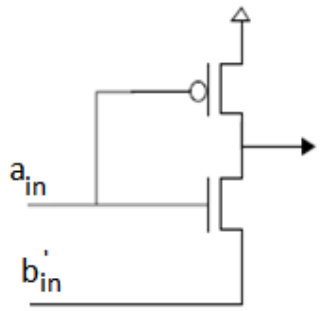

Figure 2.7 Source coupled NAND gate [5]

Table 2.1 Truth table of source coupled NAND gate [5]

Ain	Bin	Bin'	output
0	0	1	1
0	1	0	1
1	0	1	1
1	1	0	0

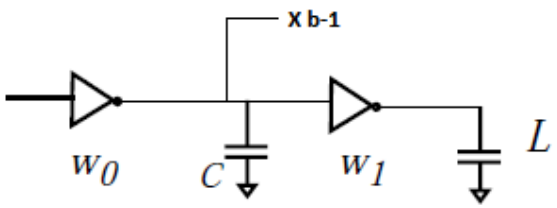

Figure 2.8 Inverter driving load capacitance[5]

The delay 'D' of the circuit is given by

$$D = \frac{C}{w_0} + \frac{b.C_{g.W_1}}{W_0} + \frac{L}{W_1} + parasitic\ delay \qquad (1)$$

Where the parasitic delay is independent of transistor sizes. For the circuit in Figure 2.9, using source coupled NAND gate, the delay can be formulated under two conditions:

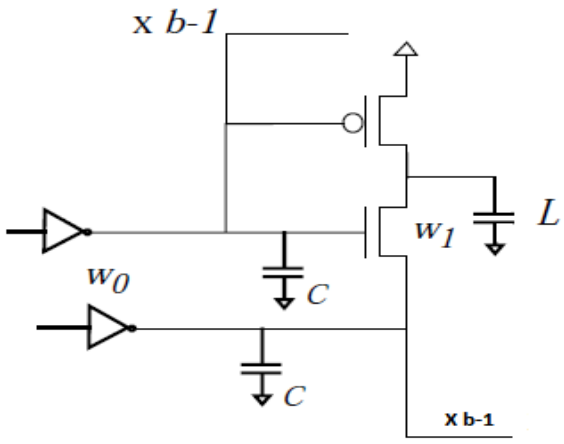

Figure 2.9 Source coupled NAND gate driving load capacitance[5]

1. If source input falls low much before the gate input and the capacitance on source input is more than the load capacitance then the source terminal behaves as virtual short and the circuit behaves as an inverter and its speed is same as that of an inverter.
2. If the gate input arrives before the source input then the delay depends on source input and is calculated with respect to source input and is given by:

$$D = \frac{C+\left(b.\frac{C_j}{6}+C_j\right)w_1+L}{w_0} + r\frac{L}{w1} + parasitic\ delay \qquad (2)$$

The total capacitance seen by the input inverter is the sum of the line capacitance, the junction capacitance of 'b' n-FETs, the junction capacitance of the drain of the n-FETs and the p-FETs of the gate and finally the load capacitance. Here it is assumed that the n-FET source is shared to halve its junction capacitance. Since the n-FET source starts off at a high voltage and then falls low, the n-FET threshold will also be high to begin with due to the body effect and then will reduce which implies that the average drive current through this n-FET is slightly smaller than that in an inverter. This is captured in the factor r in the second term of equation (2). To get some numbers for 'b', Assume r = 1, and L = $4c_g w_1$ in equation (1) and compare it to equation (2). It has been found that the NAND gate path is faster than the inverter if b > 6. The branching effort and the input capacitance for the source input can be large in the case of word drivers due to the interconnect and the loading from the other word drivers. The word drivers can be laid out such that the source area for adjacent drivers can be shared thus halving the junction capacitance.

2.3.4 NOR style NAND gate:

In previous designs as number of inputs increases the transistor in series path also increases, which increase the logical effort of the circuit i.e. the circuit becomes slower and also the problem of charge sharing will become worse. A NOR gate has a large fan in and also has small logical effort because the transistors are connected in parallel. A large fan-in NAND gate can be implemented by doing a NOR of the complementary inputs and is a good candidate in designing high speed predecoder. As the number of inputs increases, n-FETs are connected in parallel thus logical effort remains constant.

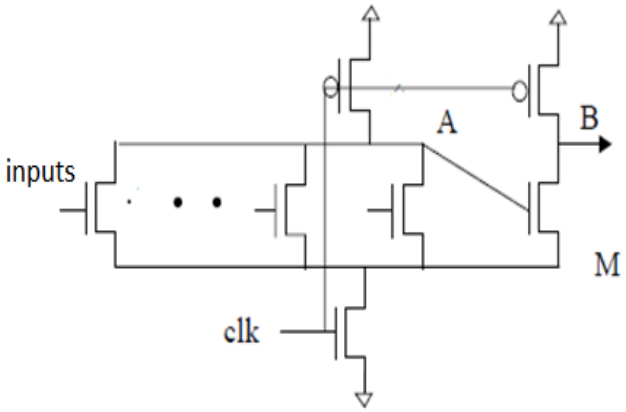

Figure 2.10: n-input NOR style NAND gate [5]

Table 2.2: Truth table for a 2- input NOR style NAND gate

In1	In2	In1b	In2b	output
0	0	1	1	0
0	1	1	0	1
1	0	0	1	1
1	1	0	0	1

An extra n-FET (M) on the output node B [8], shares the same source as the input n-FETs, but its gate is connected to the output of the NOR gate (A). When clock (clk) is low, both nodes A and B are precharged high. When clock goes high, the behavior of the gate depends on the input values. If all the inputs are low, then node A remains high, while node B discharges and the decoder output is selected. If any of the inputs are high, then node A discharges, shutting off M and preventing node B from discharging and hence causing that output to remain unselected. As this situation involves a race between A and B, the gate needs to be carefully designed to ensure robust operation.

2.4 Decoder design:

The decoder design problem has two major tasks: determining the optimal circuit style and decode structure and finding the optimal sizes for the circuits and the amount of buffering at each level. The problem of optimally sizing a chain of gates for optimal delay and power is done by the method of logical effort. Circuit techniques are than explored to

speed up the decode path and analyze their delay and power characteristics. This enabled to sketch the optimal decode structures to achieve fast and low power operation.

2.5 High speed and low power techniques:

2.5.1 Multiple Threshold CMOS(MTCMOS):

James T. Kao et al. [9] showed MTCMOS logic is effective standby leakage control technique, but difficult to implement since sleep transistor sizing is highly dependent on discharge pattern within the circuit block. They showed that dual V_t domino logic style avoids the sizing difficulties and inherent performance associated with MTCMOS. High V_t cells are used where leakage has to be prevented whereas low V_t cells are employed where speed is of concern. Both cells are effectively used in MTCMOS technique.

In active mode of operation the high V_t transistors are turned off and the logic gates consisting of low V_t transistors can operate with low switching power dissipation and smaller propagation delay. In standby mode the high V_t transistors are turned off thereby cutting off the internal low V_t circuitry.

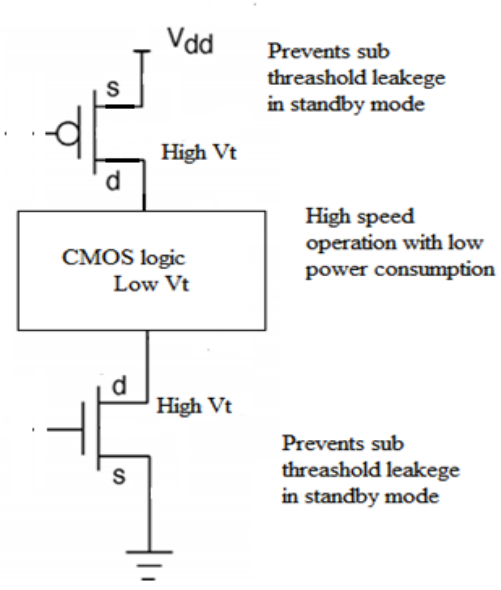

Figure 2.11: MTCMOS technique [8]

2.5.2 Variable Threshold CMOS (VTCMOS)

One of the efficient methods to reduce power consumption is to use low supply voltage and low threshold voltage without sacrificing speed performance. But increase in the lower threshold voltage devices leads to increased sub threshold leakage and hence more standby power consumption. Variable Threshold CMOS (VTCMOS) devices are one solution to this problem. In VTCMOS technique threshold voltage of the low threshold devices are varied by applying variable substrate bias voltage from a control circuitry. VTCMOS technique is very effective technique to reduce the power consumption with some drawbacks with respect to manufacturing of these devices. VTCMOS requires either twin well or triple well technology to achieve different substrate bias voltage levels at different parts of the IC. The area overhead of the substrate bias control circuitry is negligible [7, 8]**.**

2.5.3 Divided word line technique:

Yoshimoto et. al. [10] first proposed the Divided Word Line (DWL) technique, where the macros are further divided into smaller blocks of cells. In the DWL technique the long word line is further divided into n sections and each of these sections can be activated independently. This reduces the word line length by n and the RC delay by n^2. Figure 3.1 shows the DWL architecture where a macro of 256 columns is further divided into 4 parts of 64 columns. Each sub-part has its own set of local word line drivers. A particular row is selected by first activating the global word line driver which selects a particular block, and the local word line drivers then select a particular row within this block. Thus, the local word line is now reduced to 64 columns and this reduces the RC delay by a factor of 4. Even though the global word line is nearly as long as the width of the macro, it still has a lower RC delay than a full length word line since its capacitive loading is smaller. The global word line driver has to drive only 4 local word line drivers, instead of 256 memory cells. The word line RC delay is reduced by another factor of four by keeping the word line drivers in the center of the word line segments thus halving the length of each segment. The column current is also reduced by a factor of four, since only 64 memory cells are activated at a time instead of all 256.

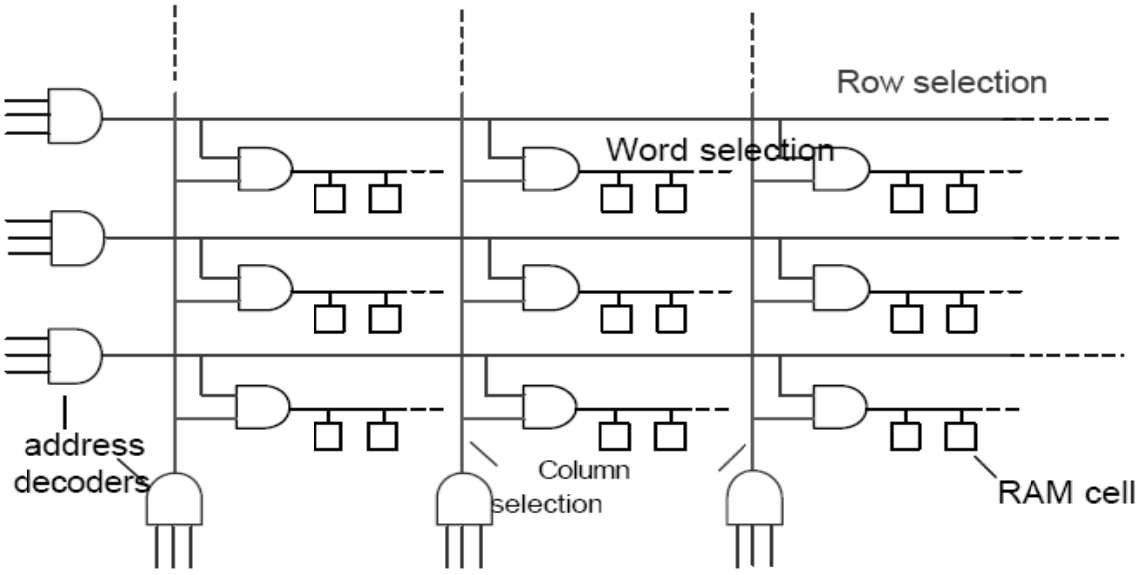

Figure 2.12: Divided word line structure [10]

2.5.4 Logical effort and Branching effort:

The three key features of the decode path which distinguishes it from a simple inverter chain is the presence of logic gates other than inverters, branching of the signal path at some of the intermediate nodes, and the presence of interconnect inside the path. Logic gates and branching are easily handled by using the concept of logical effort and branching effort introduced in. A complex gate like an n-input NAND gate has *n* n-FETs in series, which degrades its speed compared to an inverter. To obtain the same drive strength as the inverter, the n-FETs have to be sized *n* times bigger, causing the input capacitance for each of the NAND inputs to be *(n+2)/3* times bigger than that of an inverter having the same drive strength. This extra factor is called the logical effort of the gate. The total logical effort of the path is the product of logical efforts of each gate in the path.

2.5.4.1 Logical Effort:

The method of logical effort shows how many stages of logic are required for the fastest implementation of any given logic function. The method reveals the proper transistor sizes in each stage to realize the fastest overall operation [11].

Table 2.3. Logical efforts for different gates [11]

Gate type	Number of inputs 1	2	3	4	5	n
INVERTER	1					
NAND		4/3	5/3	6/3	7/3	(n+2)/3
NOR		5/3	7/3	9/3	11/3	(2n+1)/3
MULTIPLEXER		2	2	2	2	2
XOR		4	6-12	16-32		
MAJORITY			4			
MULLER C		2	3	4	5	n

Calculation of logical effort:

To compute the logical effort of an arbitrary logic gate a circuit is designed that performs the intended logic function with the same drive characteristics as an inverter. The logical effort is the ratio of the input capacitance of one of the logic gate's inputs to the input capacitance of the inverter [4]. For an 8 input AND gate (Figure 2.13.1) the logical effort is calculated as follows [11]:

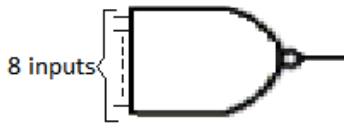

Figure 2.13.1 8 input AND function[11]

For NAND gate (g)=(8+2)/3 =10/3

For NOT gate (g)=1

Logical effort= (10/3)*1 =3.33

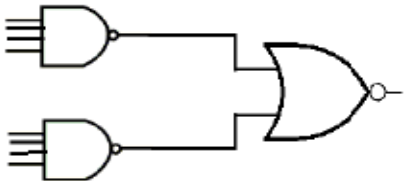

Figure 2.13.2 8 input AND function[11]

For NAND gate (g)=(4+2)/3 =2

For NOR gate (g)=5/3

Logical effort= (5/3)*2 =3.33

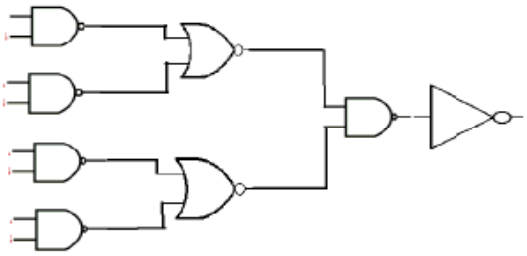

Figure 2.13.3 8 input AND function[11]

For NAND gate (g) =(2+2)/3 =4/3

For NOR gate (g) =5/3

Far NAND again=5/3

Logical effort= (4/3) (5/3)(4/3)*1 =2.96

2.5.4.2 Branching effort:

In the decode path, the signal at some of the intermediate nodes is branched out to a number of identical stages, e.g. the global word line signal in Figure 2.8. The amount of branching at each node is called the branching effort of the node and the total branching effort of the path is the product of all the node branching efforts [12]

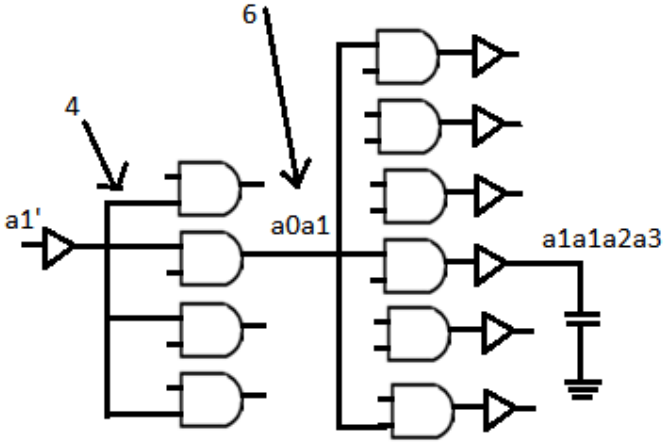

Figure 2.14 Branching effort[12]

In general for a r to 2r decoder, the total branching effort of the critical path from the input or its complement to the output is 2r-1 since each input selects half of all the outputs.

2.6 Different types of decoder:

2.6.1 Conventional NOR decoder:

The conventional NOR decoder uses one p-type transistor per word line for precharging and several n-type transistors per word line for discharging . Figure 2.15 shows the schematic of the conventional decoder. The variables *m* and *n* are used in all schematics in this paper where *n* represents the number of address bits and *m* indicates the number of address locations. In addition to the decoder circuitry, the transistors that provide the decoder's functionality by charging and discharging the word lines, the conventional decoder also requires peripheral circuitry. Peripheral circuitry includes the inverters and logic gates that generate the signals required to control the decoder circuitry. The conventional decoder's peripheral circuitry includes an inverter for the precharge signal to the T_{pcg} transistors as well as eight NOR gates that control the T_{pd} transistors to eliminate power dissipation when precharging so power consumption is greatly reduced.

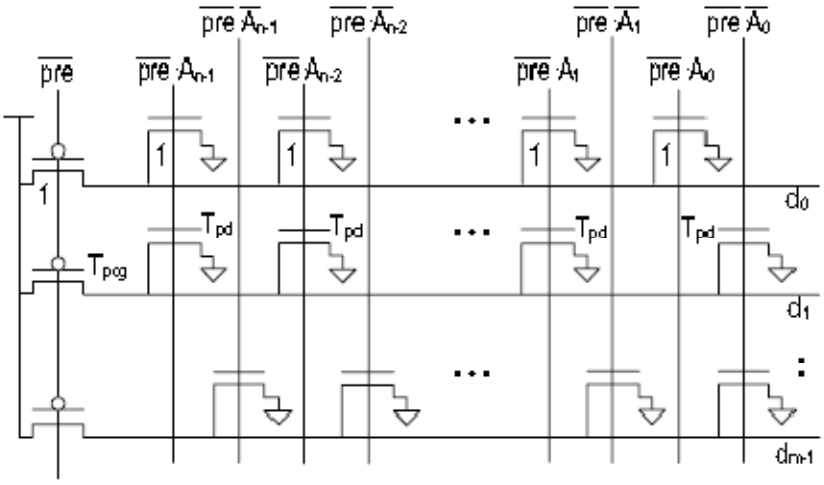

Figure 2.15 Conventional decoder [14]

The current consumption of a conventional decoder word line is measured as the current drawn by T_{pcg}. Each T_{pcg} on each word line draws current since no selective precharging is used [14].

2.6.2 AND decoder:

The AND decoder is a simpler design which uses negative logic via a NAND gate per word line to produce inverted, intermediate word line signals. Each intermediate signal is then inverted to drive each word line. Figure 2.16 shows the AND decoder schematic.

The AND decoder is observed to consume the least decoder current since it has the greatest precharge selectivity. The conventional decoder precharge all word lines non-selectively, so regardless of the address transition, current is consumed fairly uniformly while the performance of the Sense-Amp and AND decoders depends on the address transitions due to selective precharging [14].

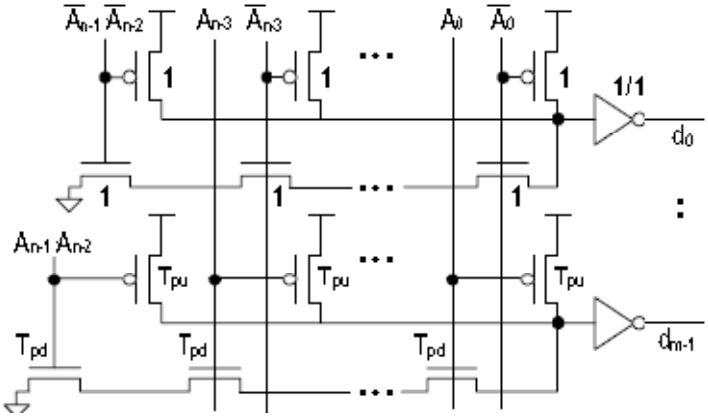

Figure 2.16 AND decoder [14]

2.6.3 AND-NOR decoder:

Novel AND–NOR decoder is a selective approach which only charges and discharges one-quarter of the select-lines each cycle. Here the most-significant bits (MSBs) of the address is used to select the portion of the decoder that will be charged. This design choice benefits from the principle of locality, as changes in the least-significant bits (LSBs) of the address are expected to be more frequent than changes in the MSBs of the address. The AND–NOR decoder design is also similar to the basic selective precharge approach of [15] to precharge the match lines in a content- addressable memory. The basic form of the AND–NOR decoding scheme is shown in Figure. 2.17.

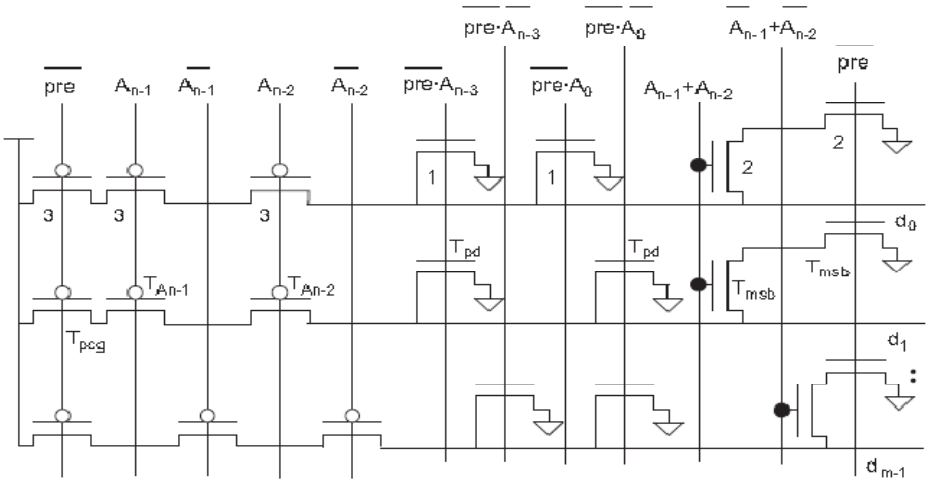

Figure 2.17 AND-NOR decoder [15]

In the precharge stage, the AND–NOR decoder uses Tpcg, TAn _1, and TAn _2 on each select-line to selectively precharge the lines based on the two address MSBs. This requires only one-quarter of the select-lines, the quarter containing the requested address, to be precharged each cycle. In the evaluate stage, the Tpd transistors discharge all non-addressed select-lines. Figure 2.18 shows short-circuit current dissipation between the precharge and evaluate stages.

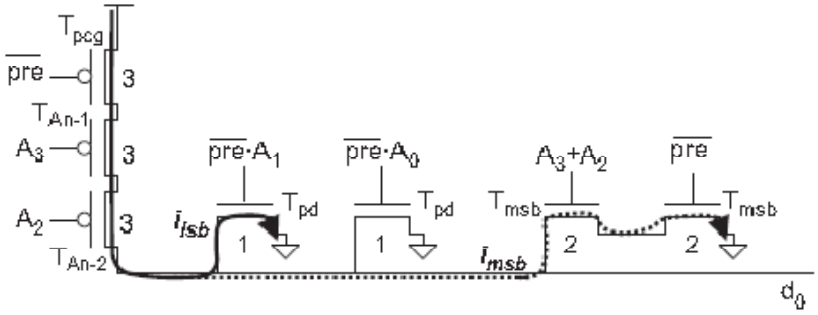

Figure 2.18 Short circuit current of AND –NOR decoder[15]

CHAPTER 3

DESIGN OF DECODER

The objective of this dissertation is to design a 8 to 256 line decoder for 256 x 256 6T-SRAM array. For this firstly various two input NAND gate architectures have been studied and simulated at 0.18µm CMOS technology. The NAND, NOR style NAND, source coupled NAND and NOT gates have been optimized for FO4 loads. The optimized gates are characterized on Cadence tool. The delay and power dissipation of critical path for the multistage decoder using conventional NAND gate(4 to 16), NOR Style NAND and source coupled NAND have been obtained. Finally, the results obtained from different styles are compared.

For this the SRAM array is divided into 4 smaller blocks of cells. The decoder circuit is placed in between the four macros. As a result of this the long word line is further divided into four sections and each of these sections can be activated independently. Figure 3.1 shows the architecture where a macro of 256 columns is further divided into 4 parts of 64 columns. Each sub-part has its own set of local word line drivers. A particular row is selected by first activating the global word line driver which selects a particular block, and the local word line drivers then select a particular row within this block. Thus, the local word line is now reduced to 64 columns and this reduces the RC delay by a factor of 4. Even though the global word line is nearly as long as the width of the macro, it still has a lower RC delay than a full length word line since its capacitive loading is smaller. The global word line driver has to drive only 4 local word line drivers, instead of 256 memory cells. The word line RC delay is reduced by another factor of four by keeping the word line drivers in the center of the word line segments thus halving the length of each segment. The column current is also reduced by a factor of four, since only 64 memory cells are activated at a time instead of all 256 cells.

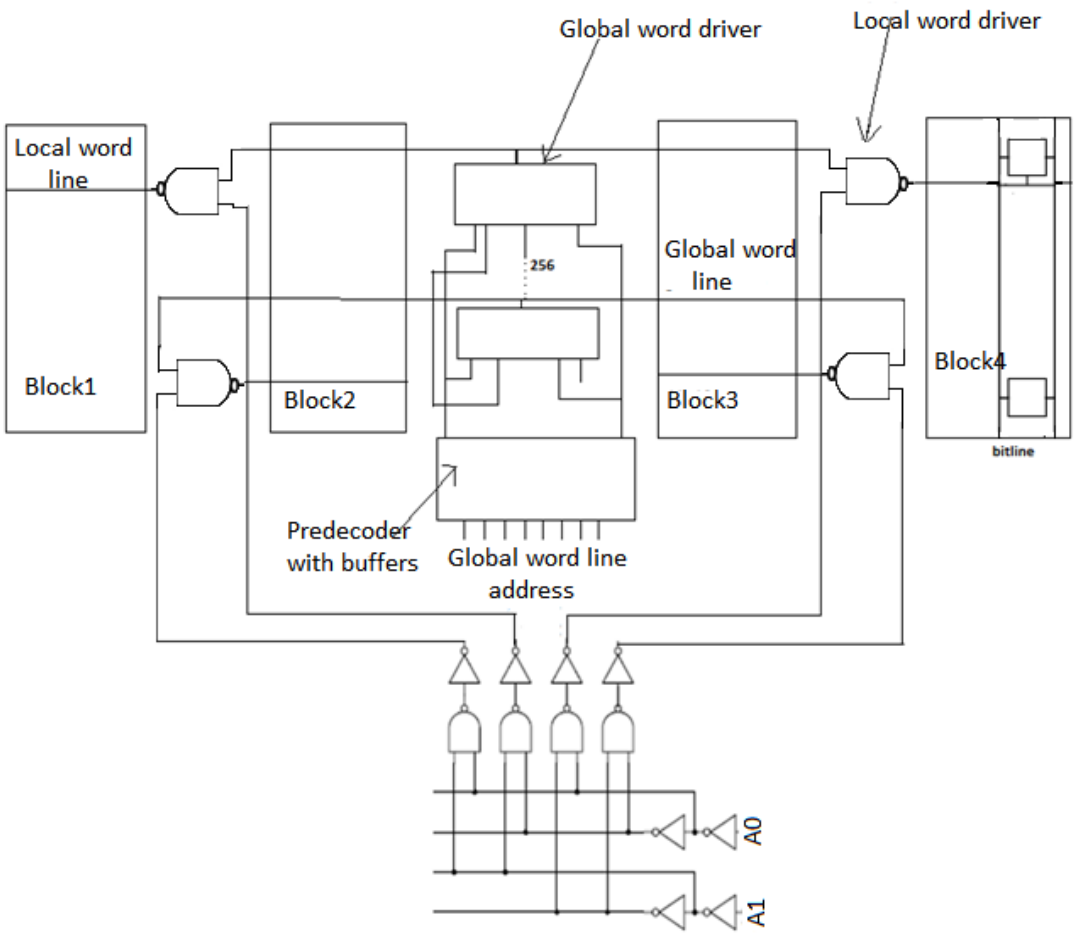

Figure 3.1 Memory array partitioned into subarrays

In order to design the decoder circuit firstly a minimum sized inverter has been designed.

3.1 Reference inverter:

Reference inverter means the inverter of minimum size which is symmetric in nature. This minimum sized inverter works as a unit load in FO4 load.

Fan out = C_{load} / C_{in},

Where C_{load} = total MOS gate capacitance driven by the logic gate under consideration

C_{in} = the MOS gate capacitance of the logic gate under consideration

As a delay metric, one FO4 is the delay of an inverter, driven by an inverter 4x smaller than itself, and driving an inverter 4x larger than itself. Both conditions are necessary since input signal rise/fall time affects the delay as well as output loading [20].

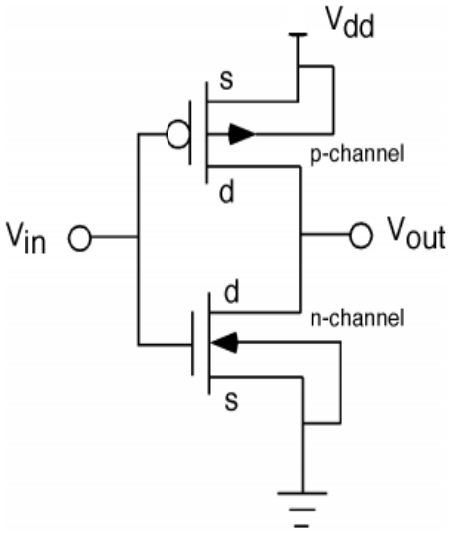

Figure 3.2 CMOS balanced inverter circuit

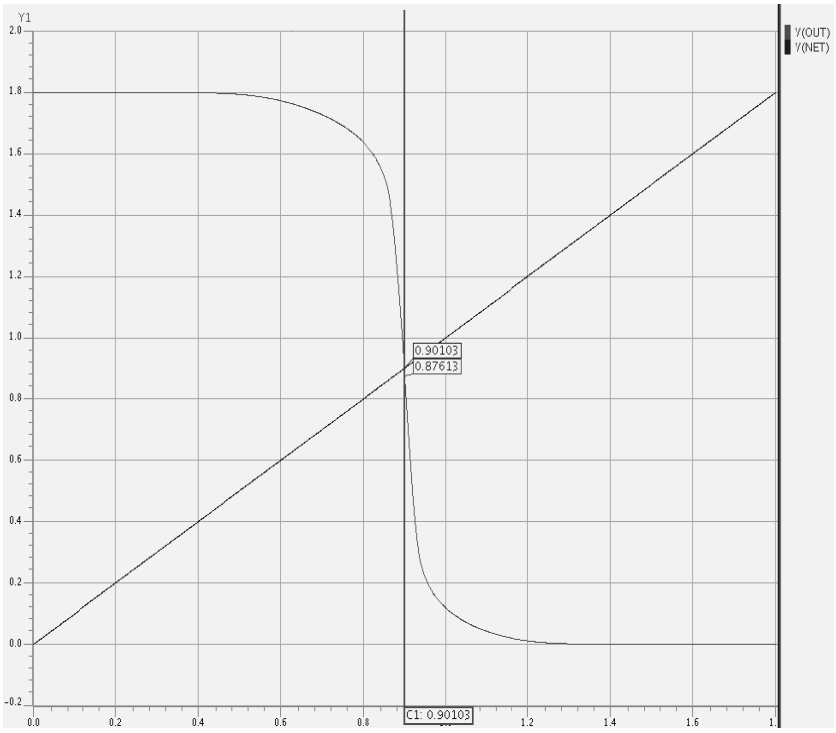

Figure 3.3 Voltage transfer characterstics of balanced inverter

Different parameters used for the reference inverter are given in Table 3.1

Table 3.1 Parameters used for reference inverter

Parameters	**Value**
Supply voltage	1.8 V
Technology used	180 nm
Input capacitance	2.577 fF
Switching threashold voltage	0.9 V
$(W/L)_n$	0.27/0.18
$(W/L)_p$	1.395/0.18

Capacitances are calculated using the formula given as:

$C_{gate}= C_{ox}$WL (3)

Where $W = W_n + W_p$

The model parameters for the nmos and pmos used in the circuit are given in Table 3.2

Table 3.2 Model parameters for NMOS and PMOS

Parameters	**Value**
ϵ_0	8.854 x 10^{-12} F/m
ϵ_{ox}	35.150 x 10^{-12}F/m
t_{ox}	4.1 x 10^{-9}m
L_{ov}	1.7 x 10^{-8}m

$C_{gate}= C_{ox}$WL= 2.577 fF (4)

$C_{overlap}= C_{ox}WL_{ov}$ = .0095 fF

This balanced inverter has been used as FO4 load to test any circuit as shown in Figure

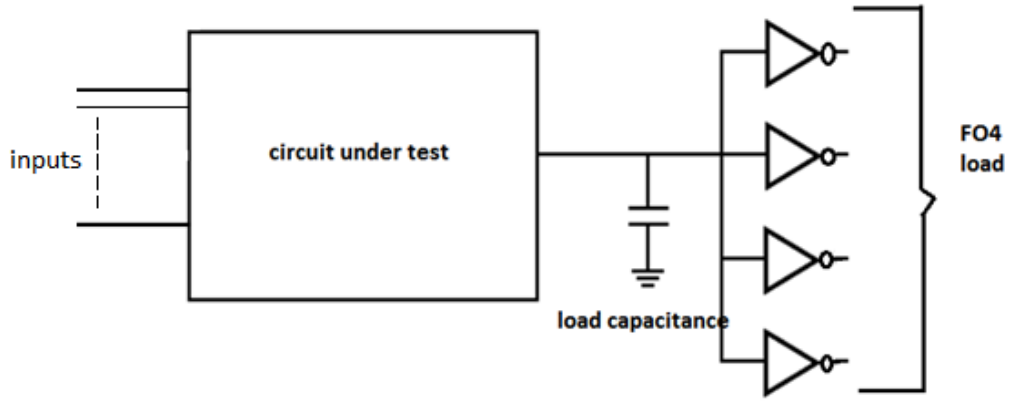

Figure 3.4 Circuit under test with FO4 load

Using this balanced inverter in the FO4 load, an another inverter which drive this load efficiently has been designed. For this the sizes of the transistors again have to be set.

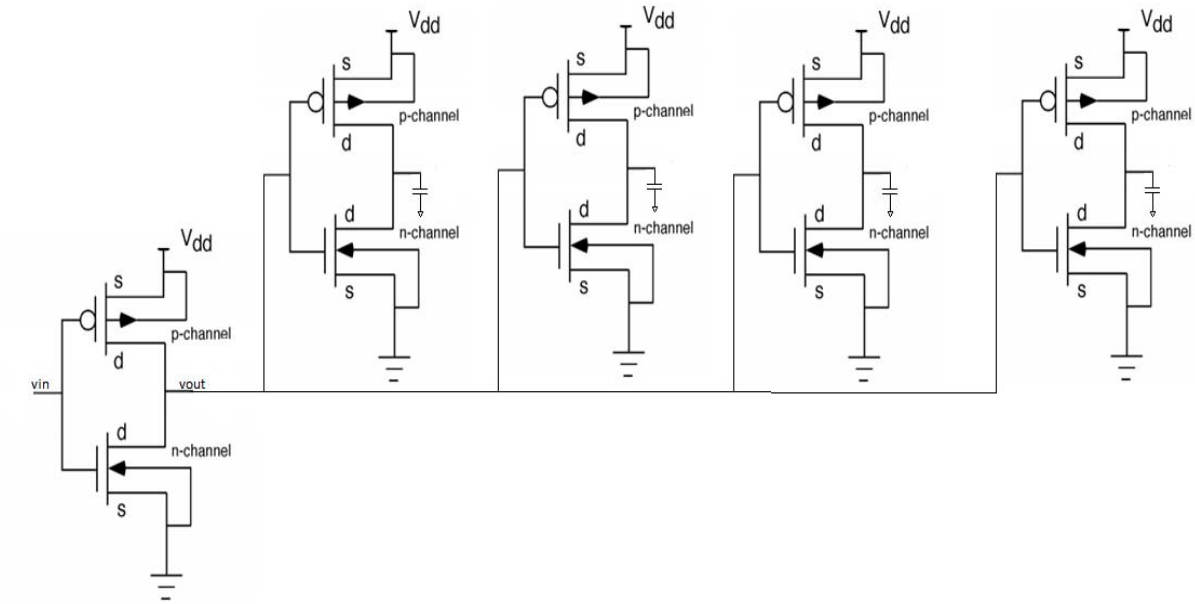

Figure 3.5 Inverter driving FO4 load

Gate capacitance of inverter driving 4 balanced inverters:

C_{gate} =8.573 x 10^{-3} x 1.8 x 10^{-6} x (3.5+2.5) x 10^{-6} =9.2588fF

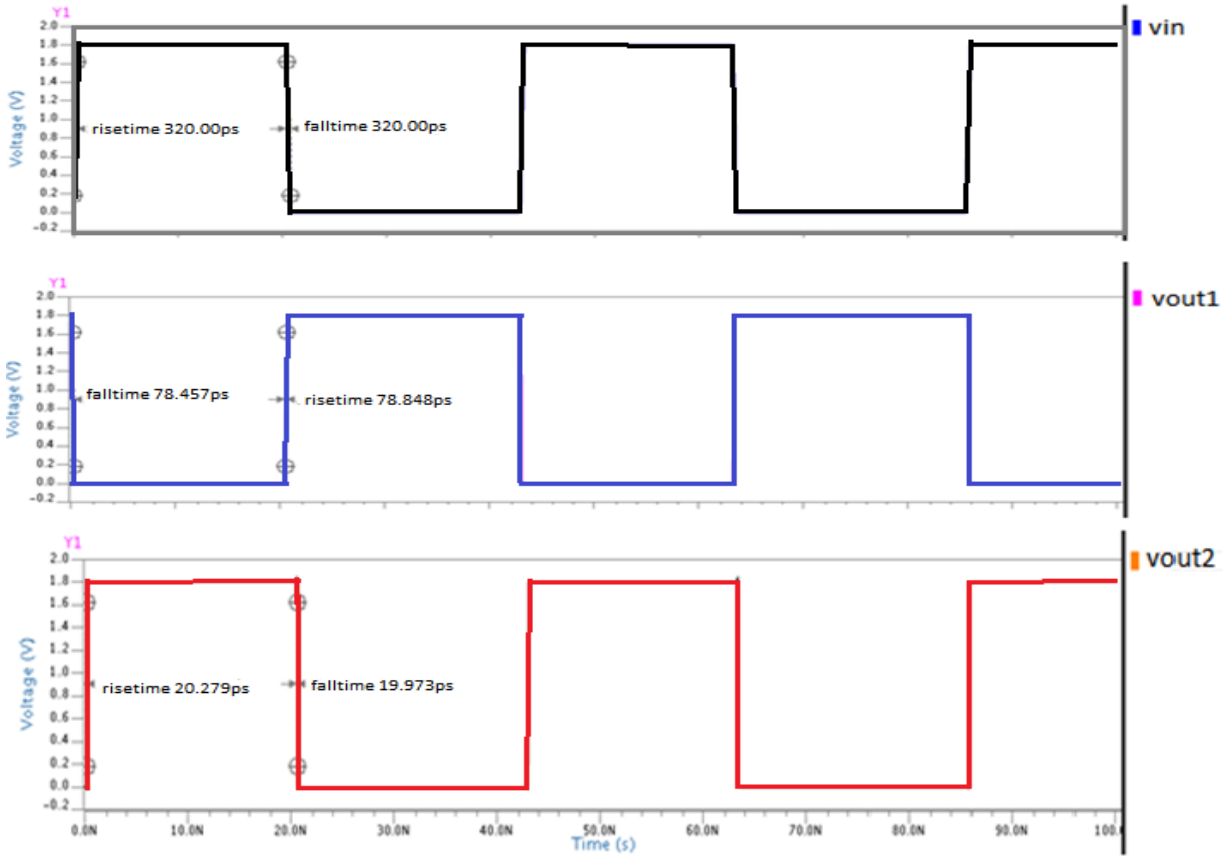

Figure 3.6 Output for inverter with FO4 load

Table 3.3 Different parameters for an Inverter

Inverter parameters	Value
Inverter area	.99 μm x 3.15 μm
Inverter area(lambda terms)	11 λ x 33 λ
Switching point	.88 V
Rise time	78.84 ps
Fall time	78.45 ps
Rising delay	39.8 ps
Falling delay	43.3 ps
Propagation delay	41.55 ps

3.2 Design of NAND gates:

AND function can be implemented with NAND gates, NOR gates and inverters. But as p-FET is atleast two times slower than n-FET, NOR are very inefficient so the AND function is implemented using NAND gates. Next various styles of designing NAND function has been discussed.

All circuits are tested with FO4 load connected at their output terminal

3.2.1 Conventional NAND gate:

In conventional style n-FETs are connected in series while the p-FETs are connected in parallel. As the two n-FETs are connected in series stack form, so to obtain the same drive capability as that of inverter the n-FETs should be twice as big as in an inverter. But in submicron technology because of velocity saturation in inverter FET its current is not as twice as that of series stack. For 180 nm technology it is found that the same current drive

capability is obtained with n-FET in series stack is 1.2 times larger than inverter n-FET [5]. Figure 3.7 shows the conventional NAND gate driving FO4 load.

8.573 x x 1.8 x x (1.2+.42) x

= 2.49989fF

= = .0095 fF

Capacitance for input ‘A’ and input ‘B’ = 2.50939 fF

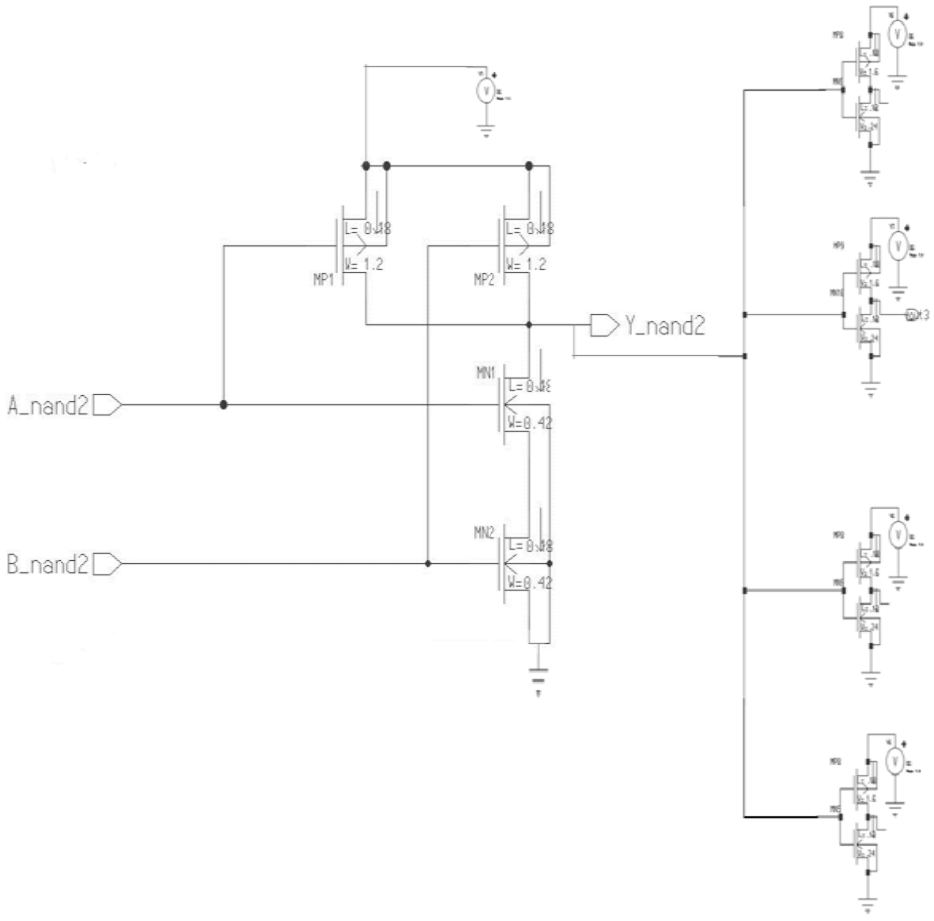

Figure 3.7 2 input Conventional NAND gate

3.2.2 NOR style NAND gate:

The most important design issue in this is the size of lower transistor. It should be strong enough so that when any input becomes '1', node 'A' discharges immediately so that logic 1 at 'B' is maintained. But should also kept in mind that node B should not be discharged that time so size of M should be relatively small. As input is coming through an inverter so total capacitance seen by the gate is the sum of output capacitance of inverter and the gate capacitance of transistor

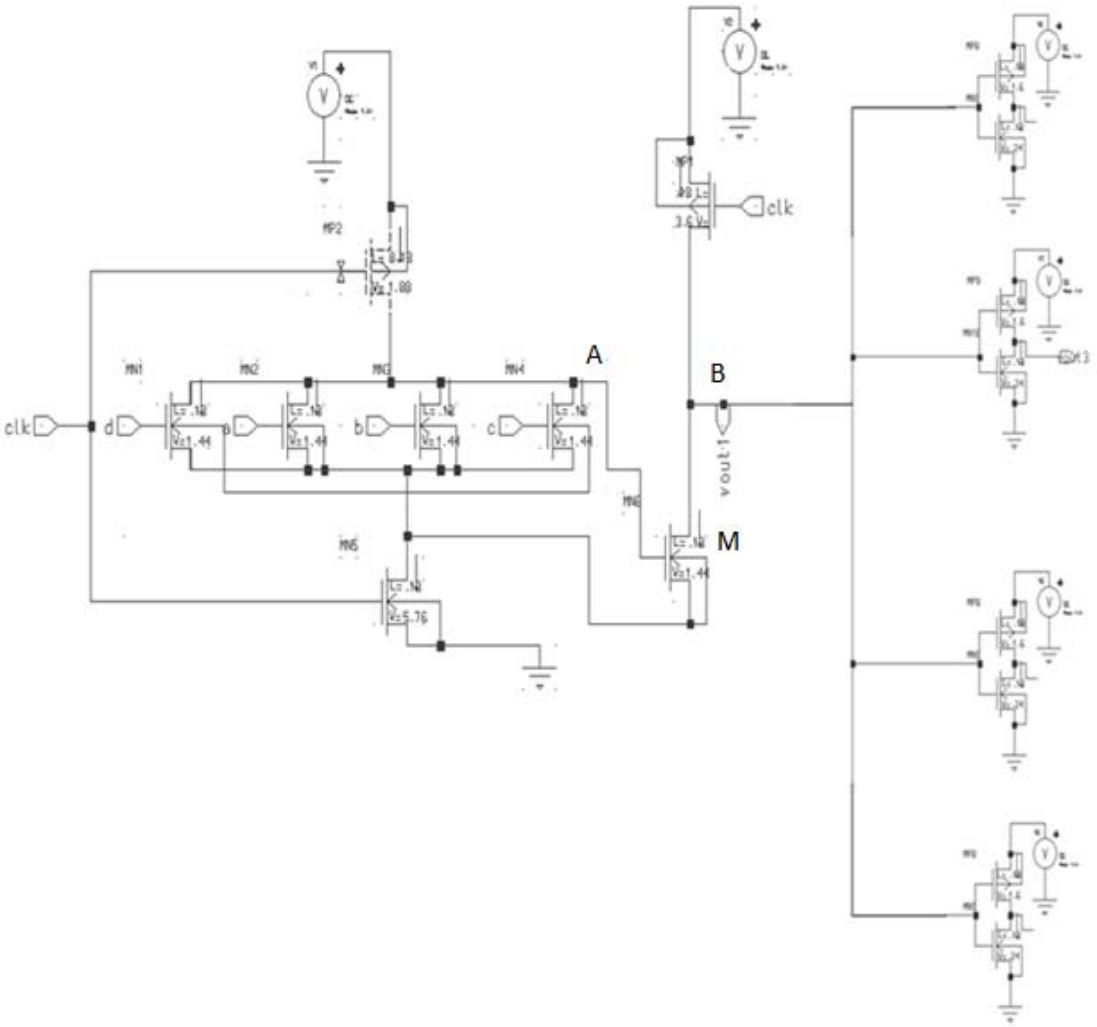

Figure 3.8 4-input NOR style NAND gate

$C_{inverter}$ = 2.5 fF

$C_{clock\ i/p}$ = 16.10474 fF

C_{input} = 2.4fF + 2.5 fF = 4.9 fF

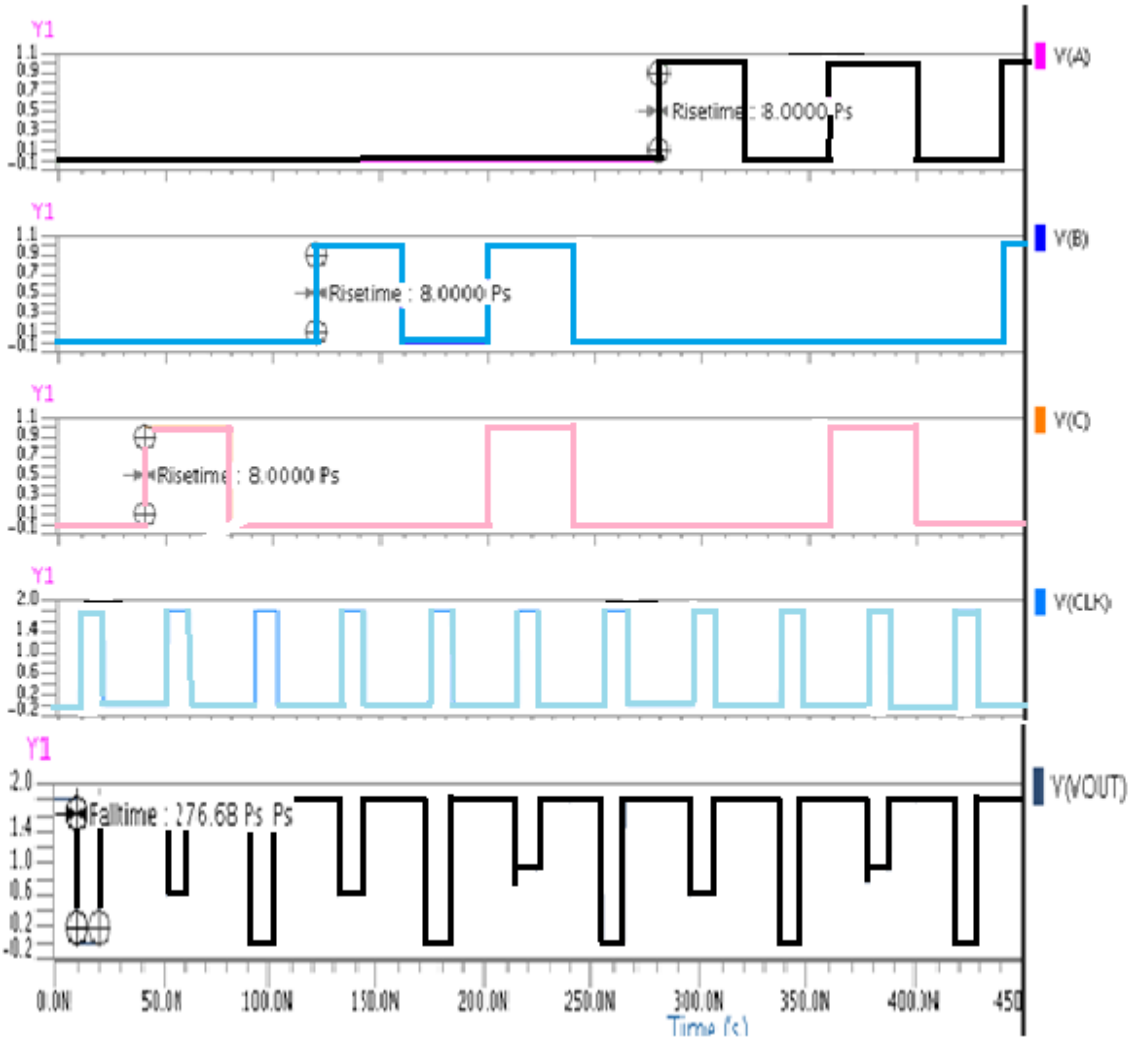

Figure 3.9 Waveforms of Figure 3.8

3.2.3 Source Coupled NAND gate:

The essential requirement of this circuit is that its source terminal capacitance should be larger that its load capacitance so a 100 fF capacitor is added there, also the source input is coming through an inverter so output capacitance of inverter also added in it making it larger than load capacitance.

= WL= 2.577 fF

= = .0095 fF

Gate capacitance of inverter driving 4 balanced inverter:

8.573 x x 1.8 x x (5.5+2) x +2.5 f =14 fF

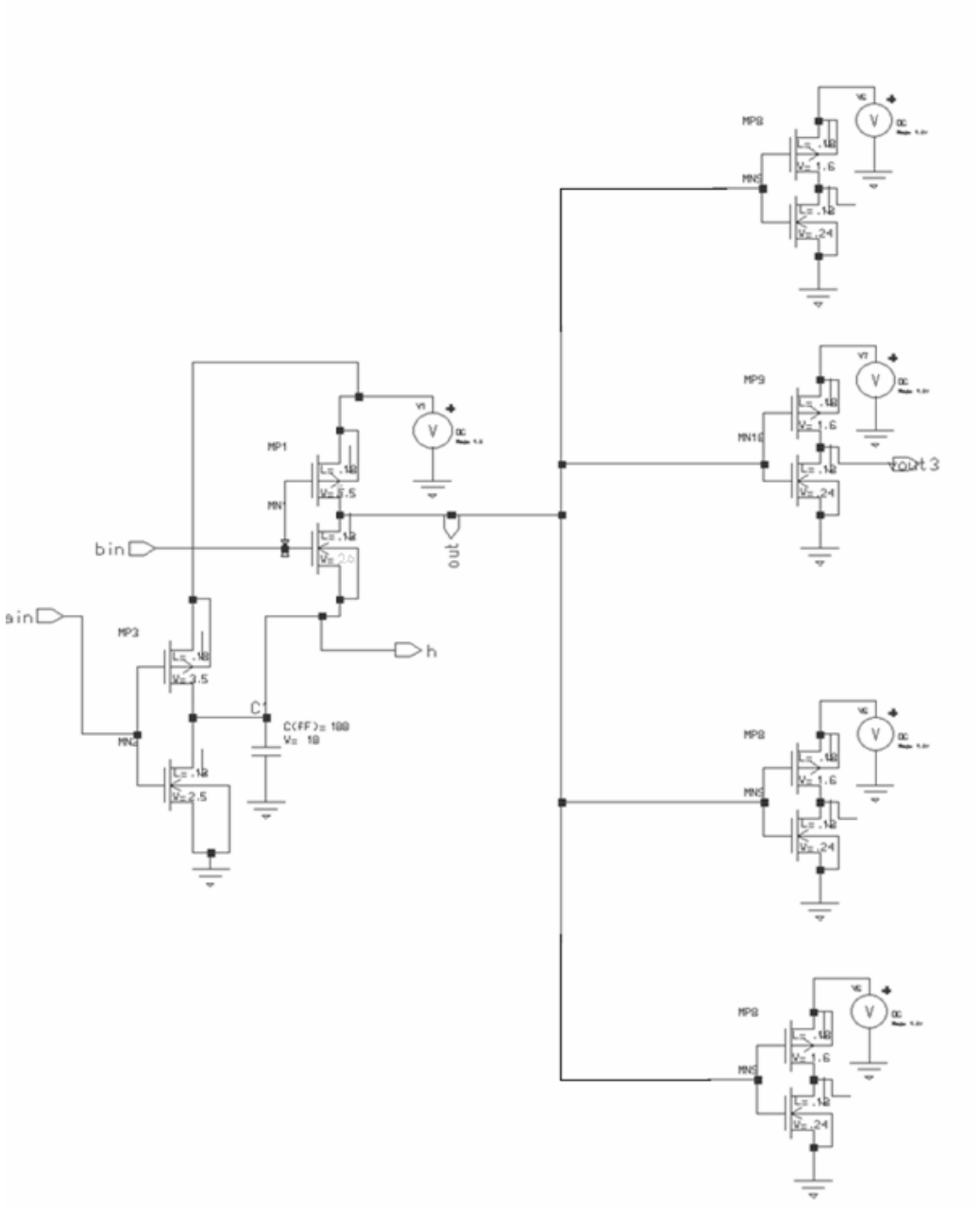

Figure 3.10 2 input source coupled NAND gate

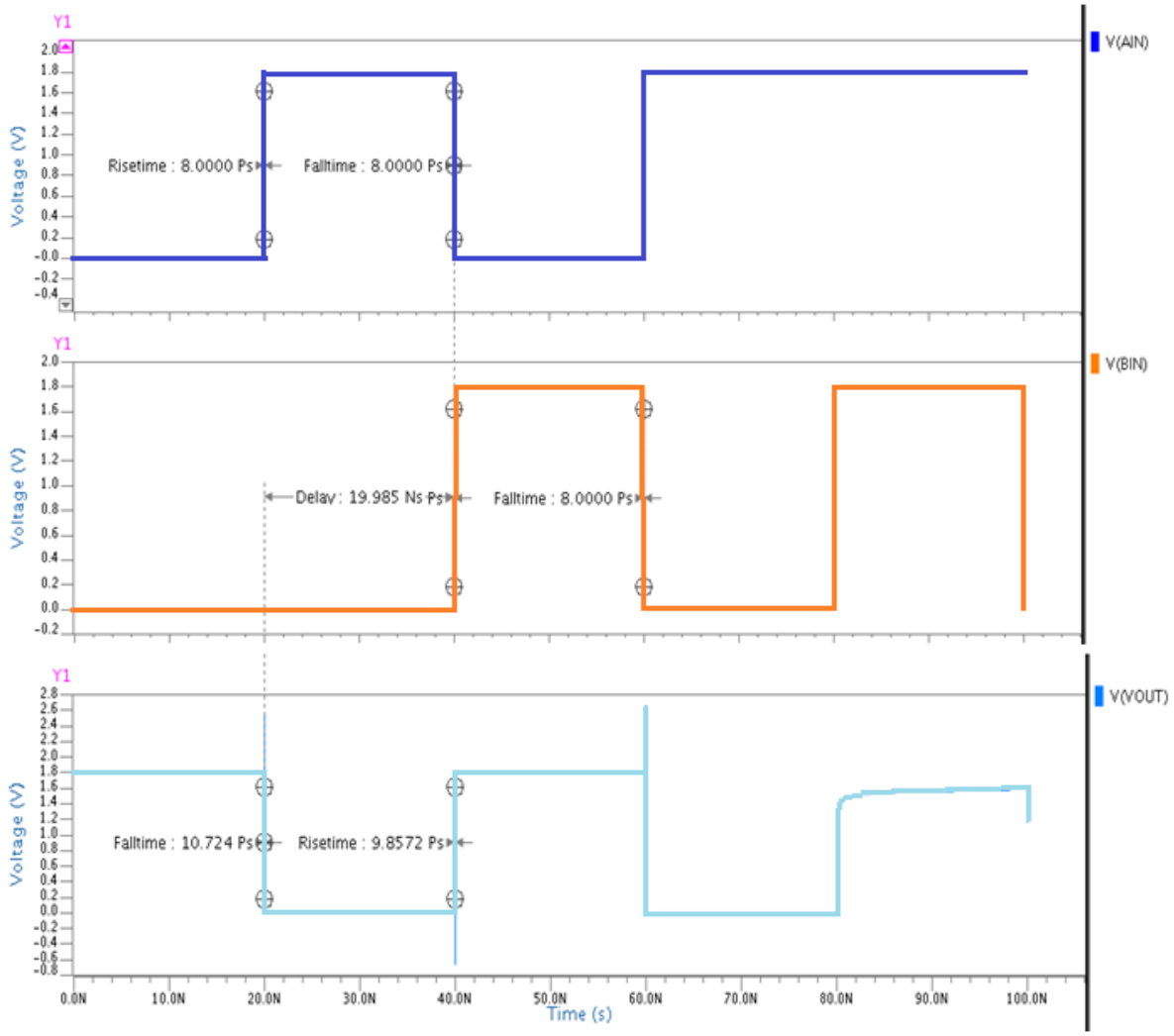

Figure 3.11 Input – output waveforms for source coupled NAND without FO4

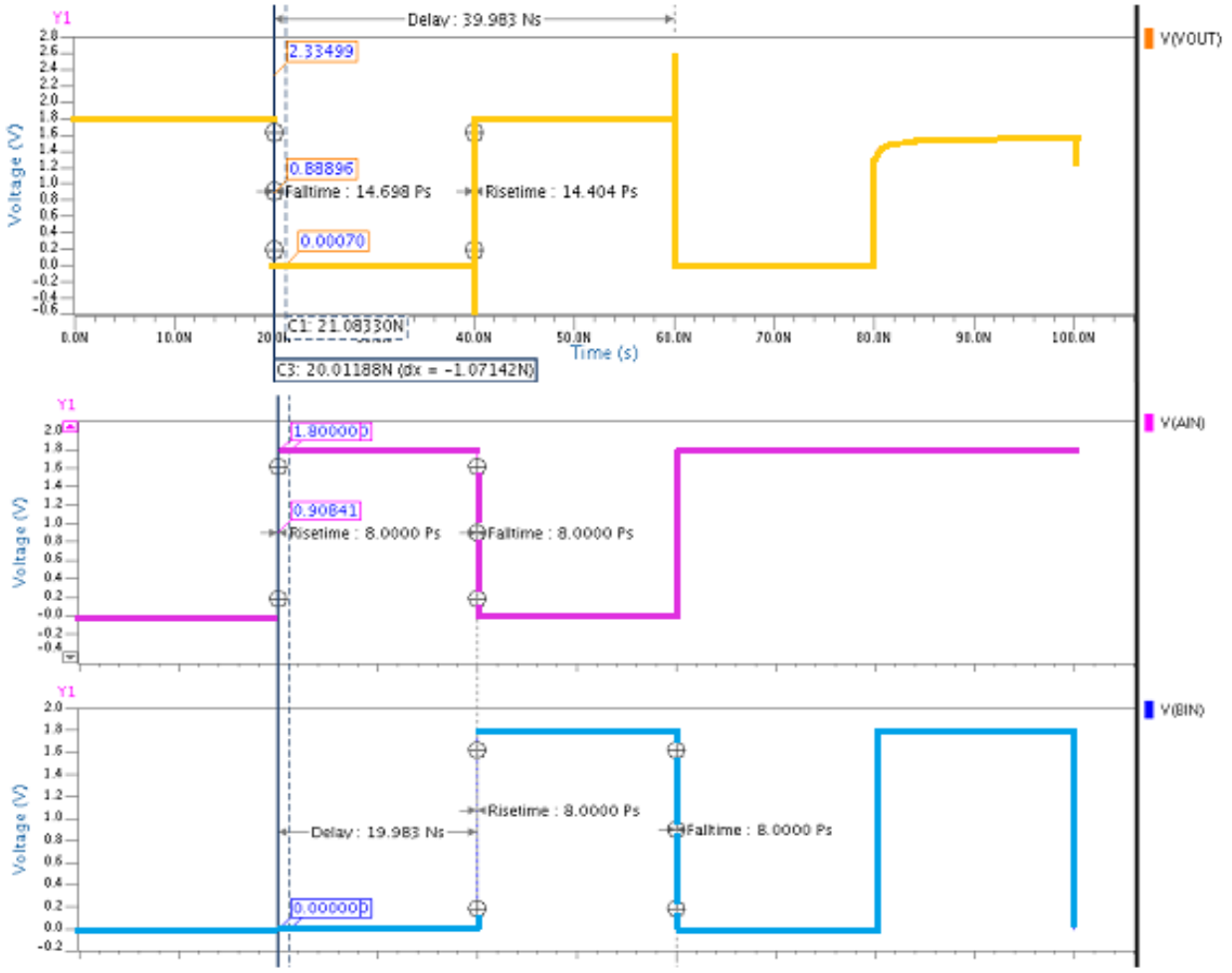

Figure 3.12 Input – output waveform for Figure 3.10

Table 3.4 Gate capacitances for different architectures

Capacitance	Value
CMOS inverter FO4 load	2.57704 fF
CMOS inverter driving FO4 load	9.25884 fF
2 input conventional NAND gate	5.01878 fF
2 input conventional NAND gate with FO4	17.8557 fF
4 input NOR style NAND gate(for Clock input)	18.66474 fF
4 input NOR style NAND gate(for input)	5.00 fF
Source coupled NAND gate (for source input)	13.89 fF

3.3 Design of 8 to 256 line decoder using different NAND gate architectures:

The below designs are illustrating the critical path of a decoder

3.3.1 Decoder design using Conventional NAND gate:

The predecoder of Figure 2.2 has been designed using conventional NAND(4 to 16) gate only as shown in Figure 3.13.

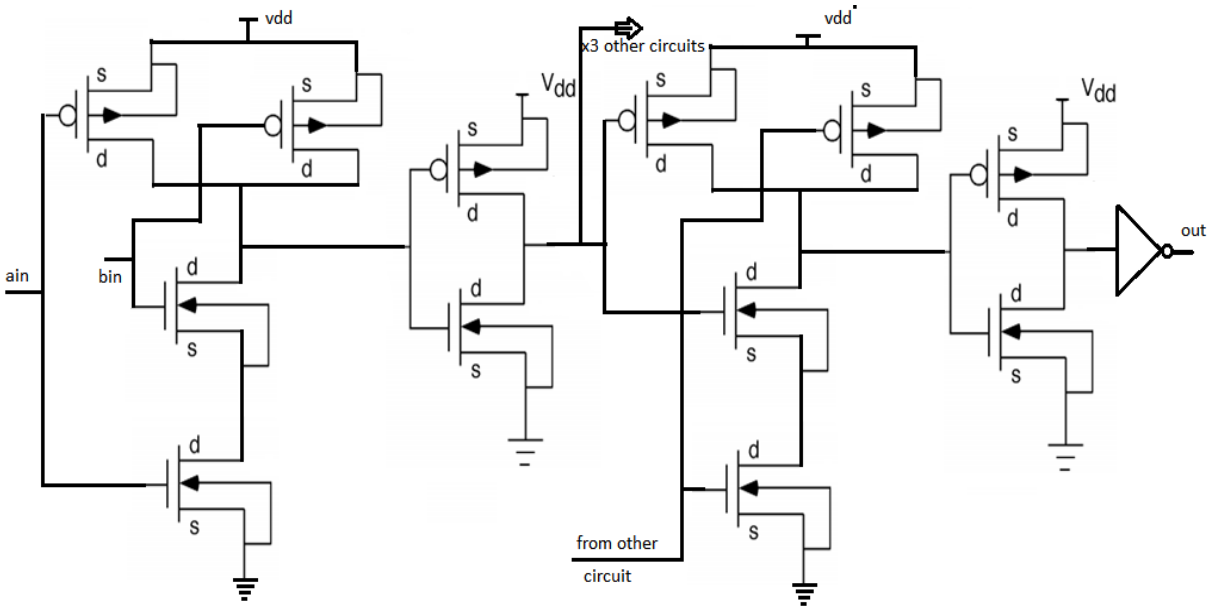

Figure 3.13 Conventional NAND based 4 to 16 predecoder

3.3.2 Nor style NAND decoder:

There are 16 NOR style NAND gates as shown in Figure 3.14. The evaluate n-FET at bottom is chosen to be larger than other gates for good drive strength.

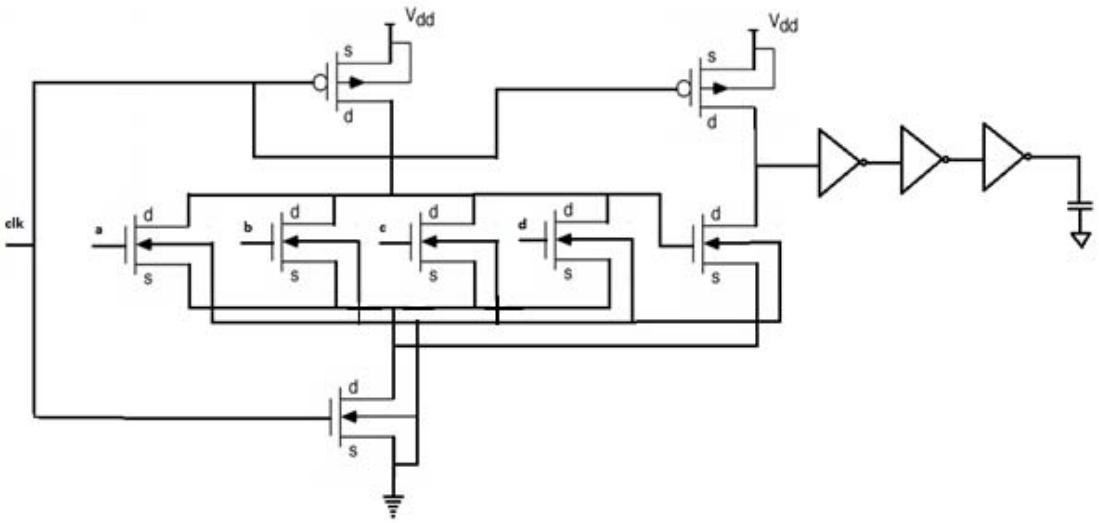

Figure 3.14 NOR based NAND 4 to 16 line predecoder

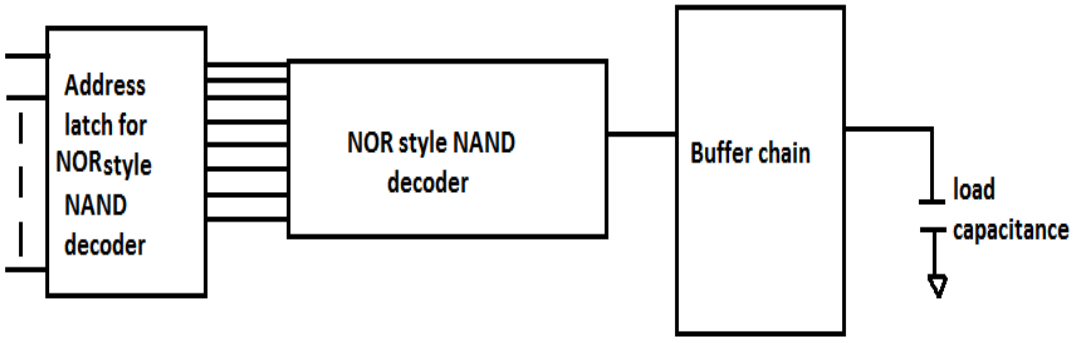

Figure 3.15 Critical path using NOR style NAND gate

3.3.3 Source coupled NAND decoder:

To design critical path using source coupled NAND style 3 gates or 6 transistors plus buffer chain is used. In this also address is decoded in basically two stages. To make gate as fast as inverter its source terminal capacitance should be larger than load capacitance and the branching at source terminal itself makes the terminal capacitance to be larger than load capacitance.

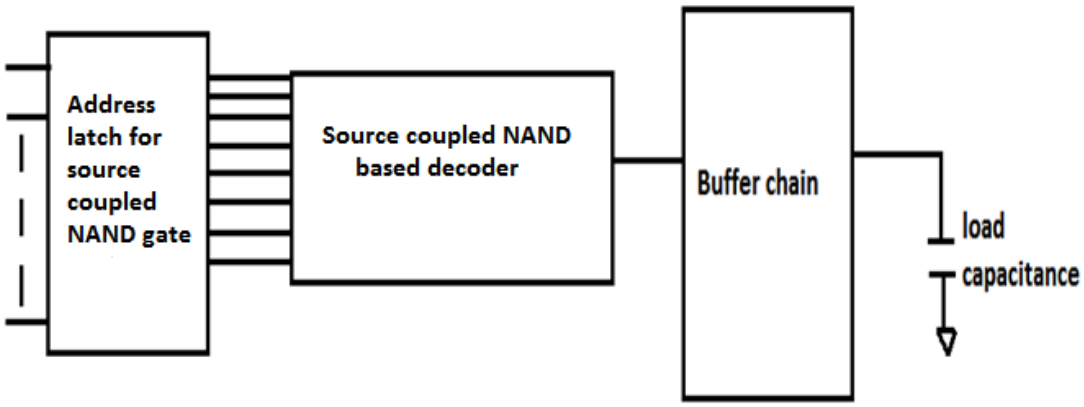

Figure 3.16 Critical path using source coupled NAND gate

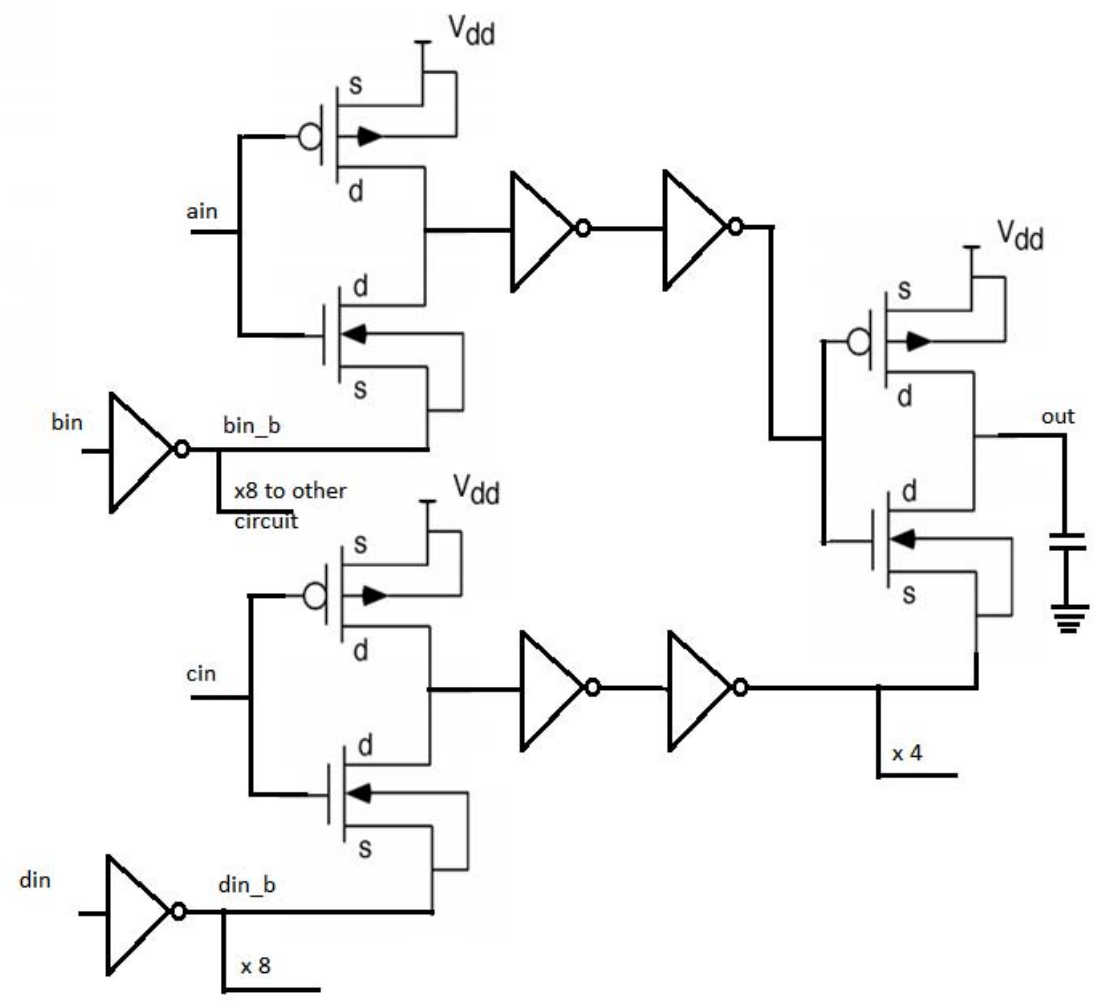

Figure 3.17 Source coupled NAND 4 to 16 line predecoder

So there is no need of adding extra capacitance, the structure of circuit itself is providing the necessary capacitance.

When input at gate terminal is logic '1' and the source input is also logic '1' the circuit behaves as pass transistor and output is one V_t drop below power supply voltage. So inverters are needed at output to get full swing. But as to drive second stage of critical path inverter chain is used, this inverter chain do this additional task of getting signal back from $(V_{dd} - V_t)$ to V_{dd}. So no extra inverter is needed.

CHAPTER 4

RESULTS AND COMPARISON

This dissertation work presents the design of 8 to 256 line decoder for an SRAM array. For this firstly various two input NAND gate architectures have been studied for decoder designs and simulated at 0.18µm CMOS technology, level 53 model file on Cadence tool. For schematic entry cadence composer is used, for simulation Cadence Spectre 5.1.4 1_ISR is used, for layout Cadence Virtuoso 5.1.4 1_ISR is used and DRC, LVS and RCX is done using Cadence Assura 3.2.0.

A conventional NAND gate, NOR style NAND gate, source coupled NAND gate and a NOT gate have been optimized for FO4 loads. Next, these optimized gates have been used in the decoder designs. Mainly three decoders are designed i.e., conventional multistage NAND gate decoder, NOR Style decoder and the decoder using source coupled NAND gates. The delay and power dissipation of critical path for these decoder circuits have been obtained. Finally, the results obtained from different styles are compared.

Table 4.1 gives the comparison of various NAND gate architectures in terms of delay and power. The gates are simulated at 0.18µm CMOS technology.

Table 4.1 Power delay comparison of NAND architectures

NAND Gate Architecture	Power Dissipation (μW)	Delay(ps)	Delay-Power Product (μW- ps)
Conventional NAND Gate	19.67	2623	51594.41
Nakamura's NAND Gate	11.32	512	5795.84
NOR Style NAND Gate	1.26	96	120.96
Source coupled NAND gate	0.72	90	64.8

As discussed in Chapter 2, a two input source coupled NAND gate gives the best performance for a decoder circuit as compared to other two input NAND gate architectures. This fact is clearly seen from the above table. A Conventional NAND gate may not suitable for implementing a decoder circuit due to its large delay and large power dissipation. A NOR style NAND gate is beneficial when the number of inputs to the gate increases. A delay power product of 64.8 μW-ps has been achieved for the source coupled NAND gate.

This suggests that the gate is most efficient and suitable for decoder design giving least power dissipation and better performance in comparison to other NAND gate architectures.

The layouts of the NAND gate are shown in figure 4.1, 4.2 and 4.3. Table 4.2 gives the area comparison of various NAND gate architectures. It has been observed that source coupled NAND gate occupies minimum area of 36 μm^2 resulting in a compact decoder circuitry.

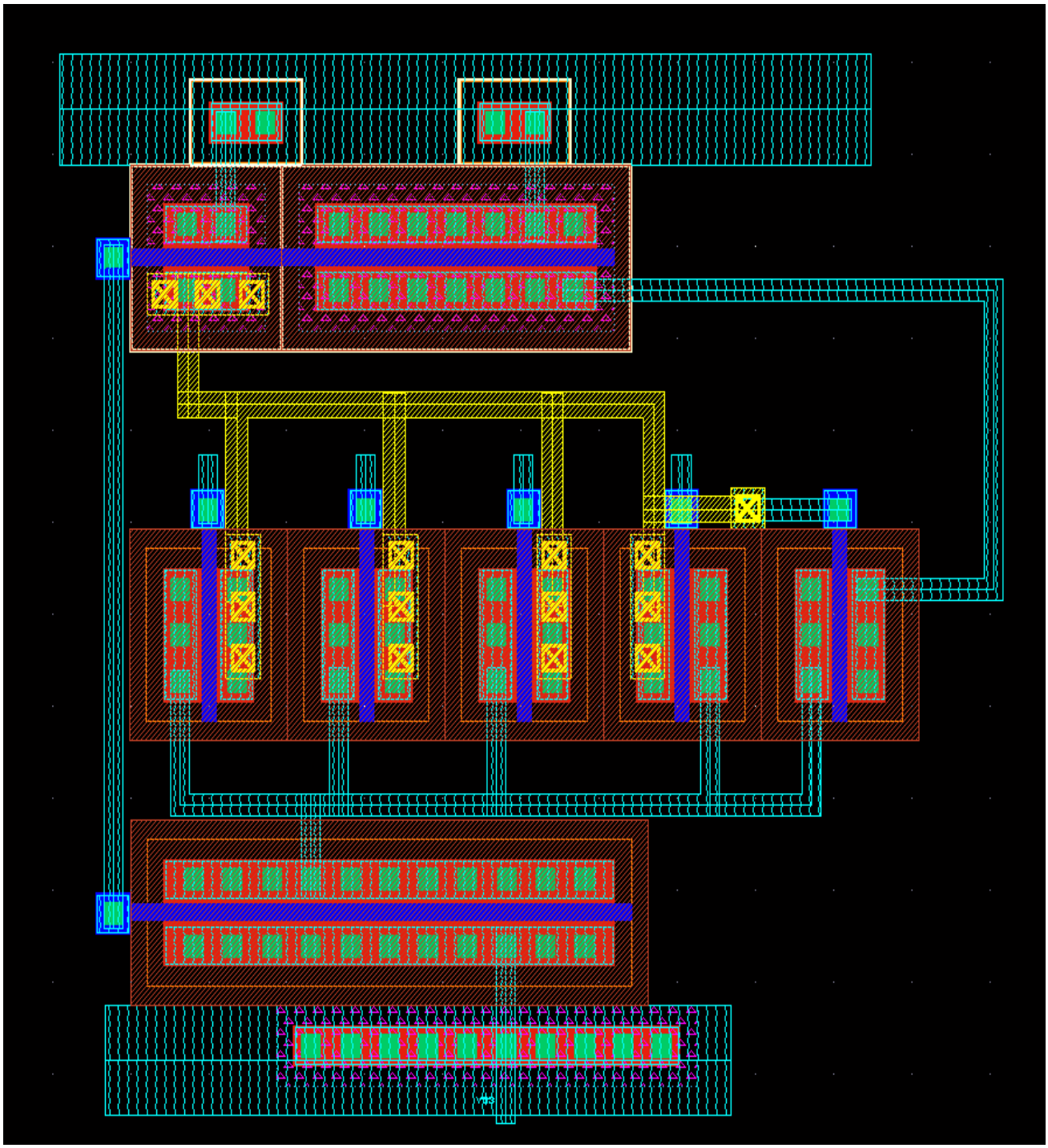

Figure 4.1 Layout of NOR based NAND gate

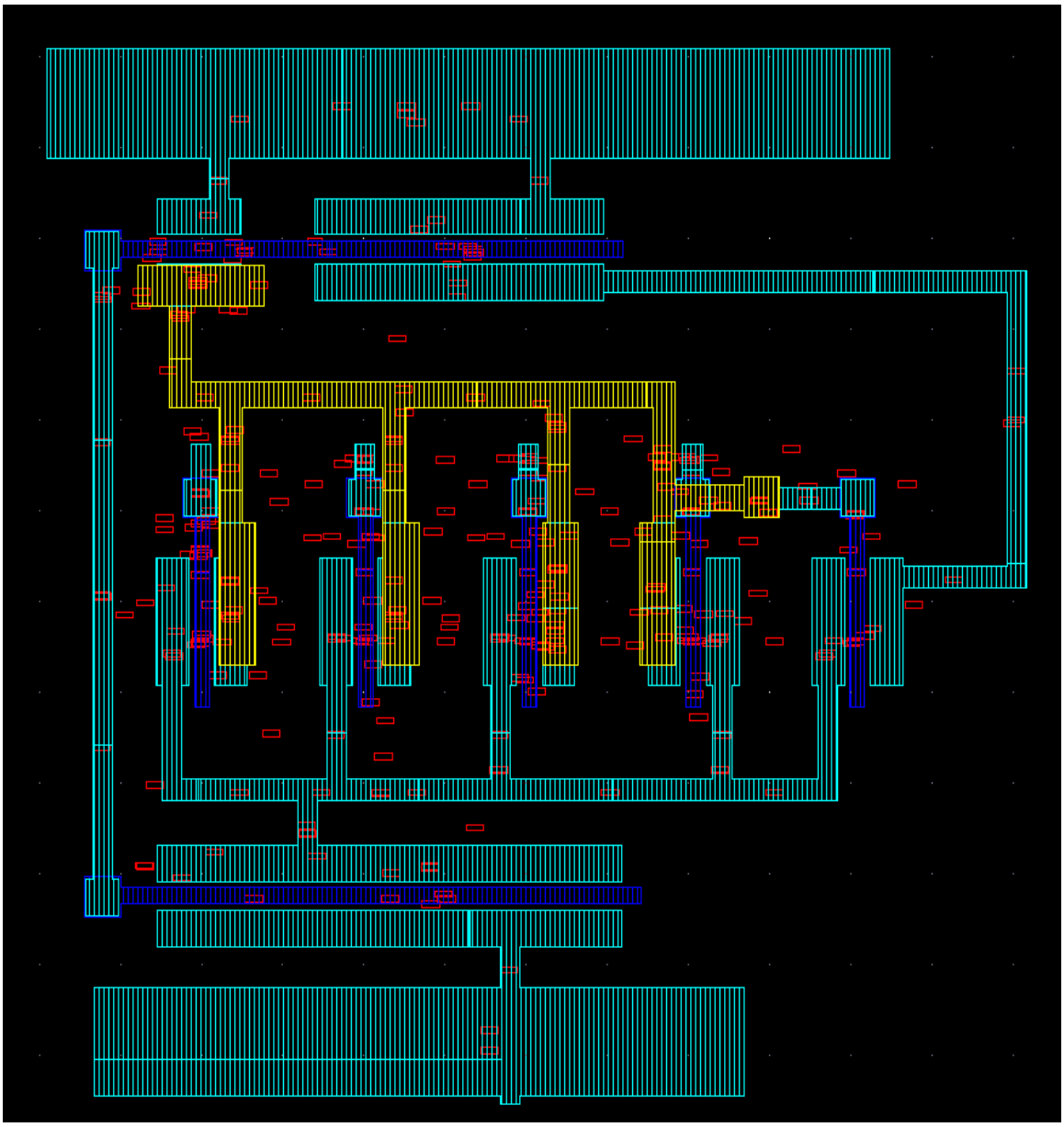

Figure 4.2 RC extraction view of NOR based NAND gate

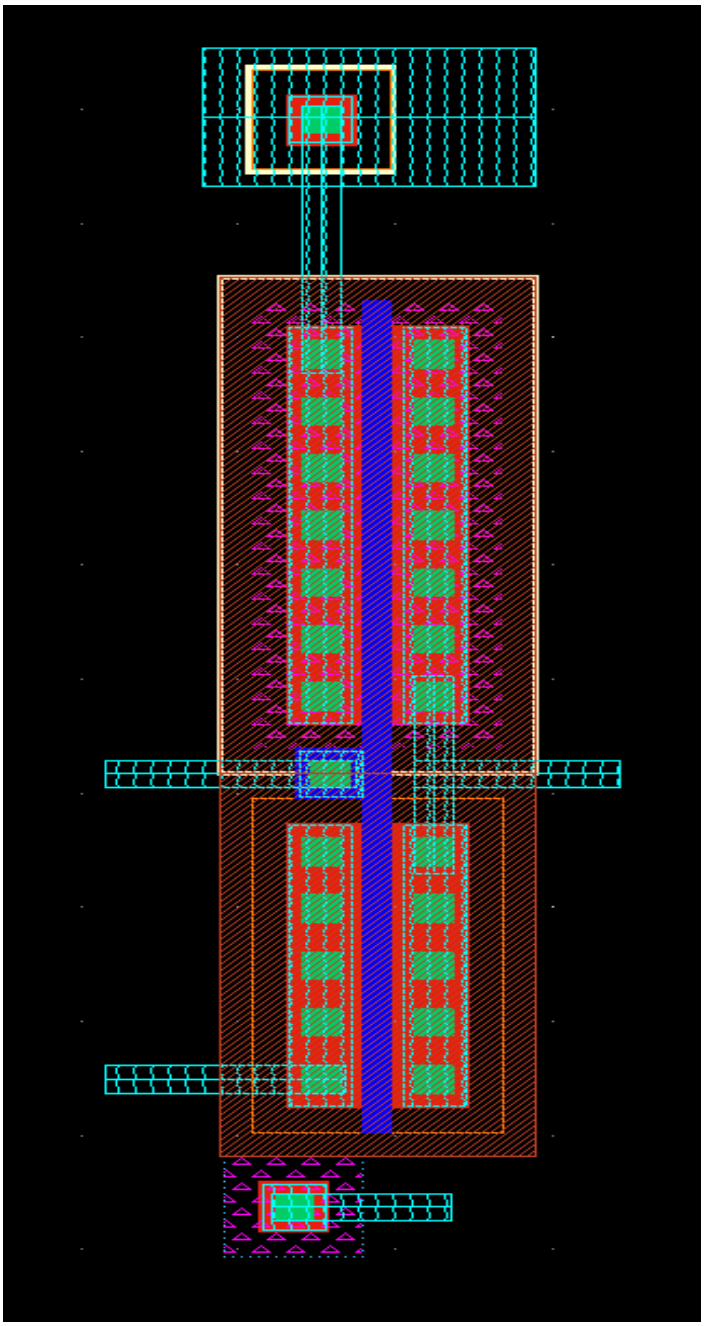

Figure 4.3 Layout of source coupled NAND gate

Table 4.2 Area comparison of NAND architectures

NAND Gate Architectures	**Area (μm^2)**	**Area (λ terms)**
Conventional NAND gate	2.34 x 9.27	26 x 103
NOR Style NAND Gate	12.02 x 11.5	133 x 128 (approx)
Source coupled NAND gate	10.62 x 3.29	188 x 37 (approx)

Table 4.3 Post layout simulation results

NAND Gate Architecture	**DRC (design rule check)**	**LVS (layout v/s schematic)**	**Percentage change in results after post layout simulation**
NOR style NAND	No DRC error	Schematic matched	9%
Source coupled NAND	No DRC error	Schematic matched	4%

Table 4.3 gives the post layout simulation results. It is seen that the percentage change in the results from pre-layout simulation to the post layout simulation is 9% for NOR style NAND gate and 4% for source coupled NAND gate. This indicates that the optimized gates are suitable for fabrication as well.

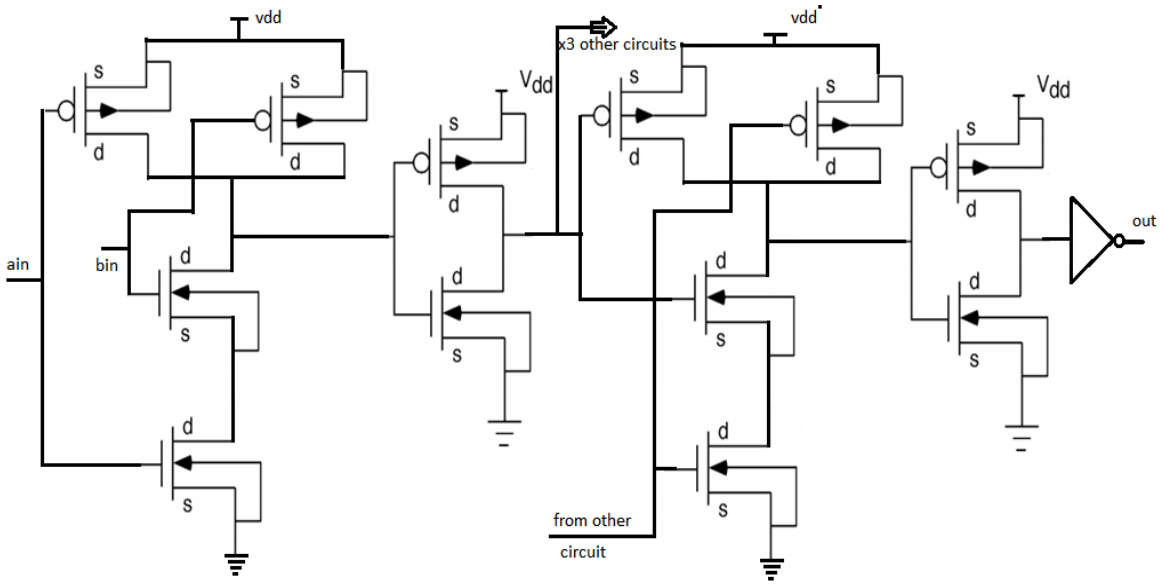

Figure 4.4 Critical path for a 4 to 16 multistage conventional NAND gate decoder

Figure 4.4 shows the critical path for a 4 to 16 line conventional NAND gate decoder. The delay and power obtained for this circuit are 3.9 ns and 1000 µW leading to a large power delay product of 390 µW-ps. this indicates that a larger sized decoder (8 to 256) if designed using conventional NAND gate will not be feasible for a large memory array.

Figure 4.5 shows the critical path for a 8 to 256 line NOR style decoder. The circuit starts with the address latch followed by a chain of inverters to drive the gate capacitances of 256, 8 input NOR style NAND gates. A NOR style NAND gate in the decoder circuit is selected when all the inputs to a gate is high. This is achieved according to the address bits. The selected gate needs to drive the load of 4 local word drivers and hence next chain of buffers is connected in the circuit . Now, a local word driver is selected according to the block selected signal. The local word driver drives the load of 64 cells in a row instead of all the 256 cells. The RC delay is reduced resulting in better performance. The optimized circuit gives the delay power product of 72.17 µW-ns. Figure 4.6 shows the output waveform for the decoder circuit using NOR style NAND gate.

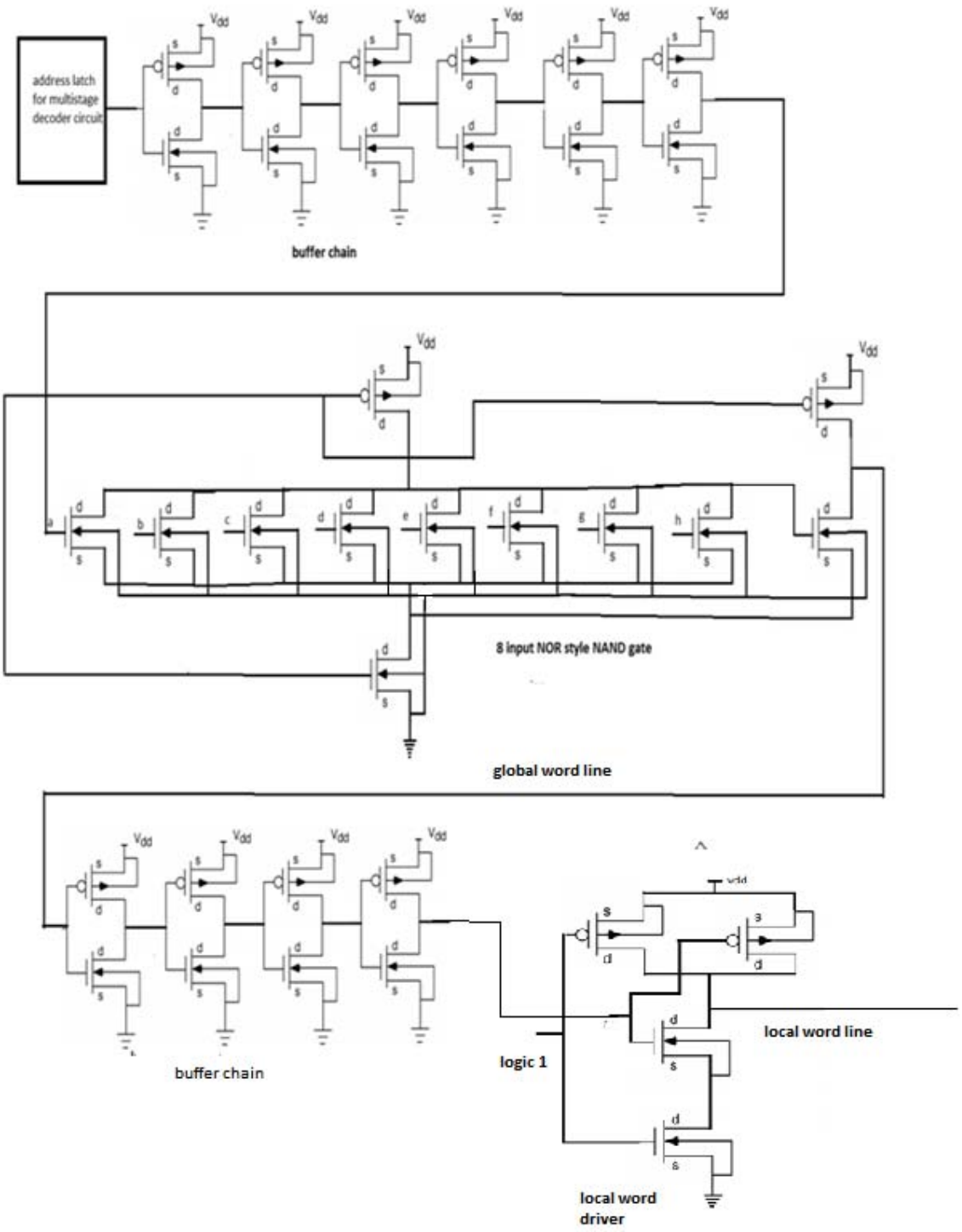

Figure 4.5 Critical path of an 8 to 256 decoder using NOR style NAND gate

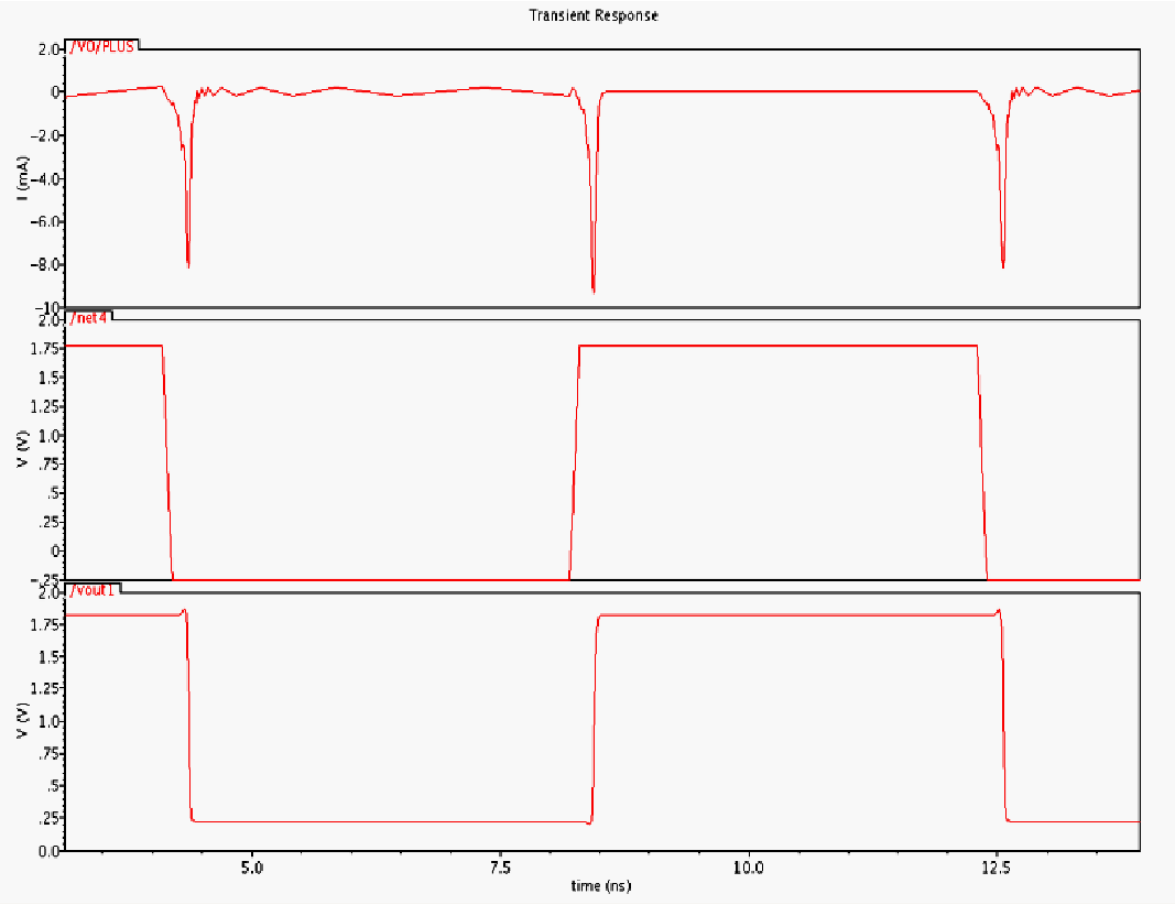

Figure 4.6 Output waveforms for the circuit in Figure 4.5

From the waveforms of decoder based on NOR style NAND gate the falling edge delay is found to be 219 ps and power dissipation 330µ W.

Figure 4.7 shows the critical path for a 8 to 256 line source coupled NAND decoder. The circuit starts with the address latch followed by a chain of inverters to derive the 16 gates and source capacitances of 2 input source coupled NAND gates at the predecode stage. The output of this predecoder is than given to set of buffers to drive the load of 256 source coupled NAND gates of the post decoder stage. In irder to derive the 4 local word drivers, a set of inverters is again constructed after post decoder.The local word driver drives the load of 64 cells in a row instead of all the 256 cells. The RC delay is reduced resulting in better performance. The optimized circuit gives the delay power product of 38.64 µW-ns. Figure 4.8 shows the output waveform for the decoder circuit using source coupled NAND gate.

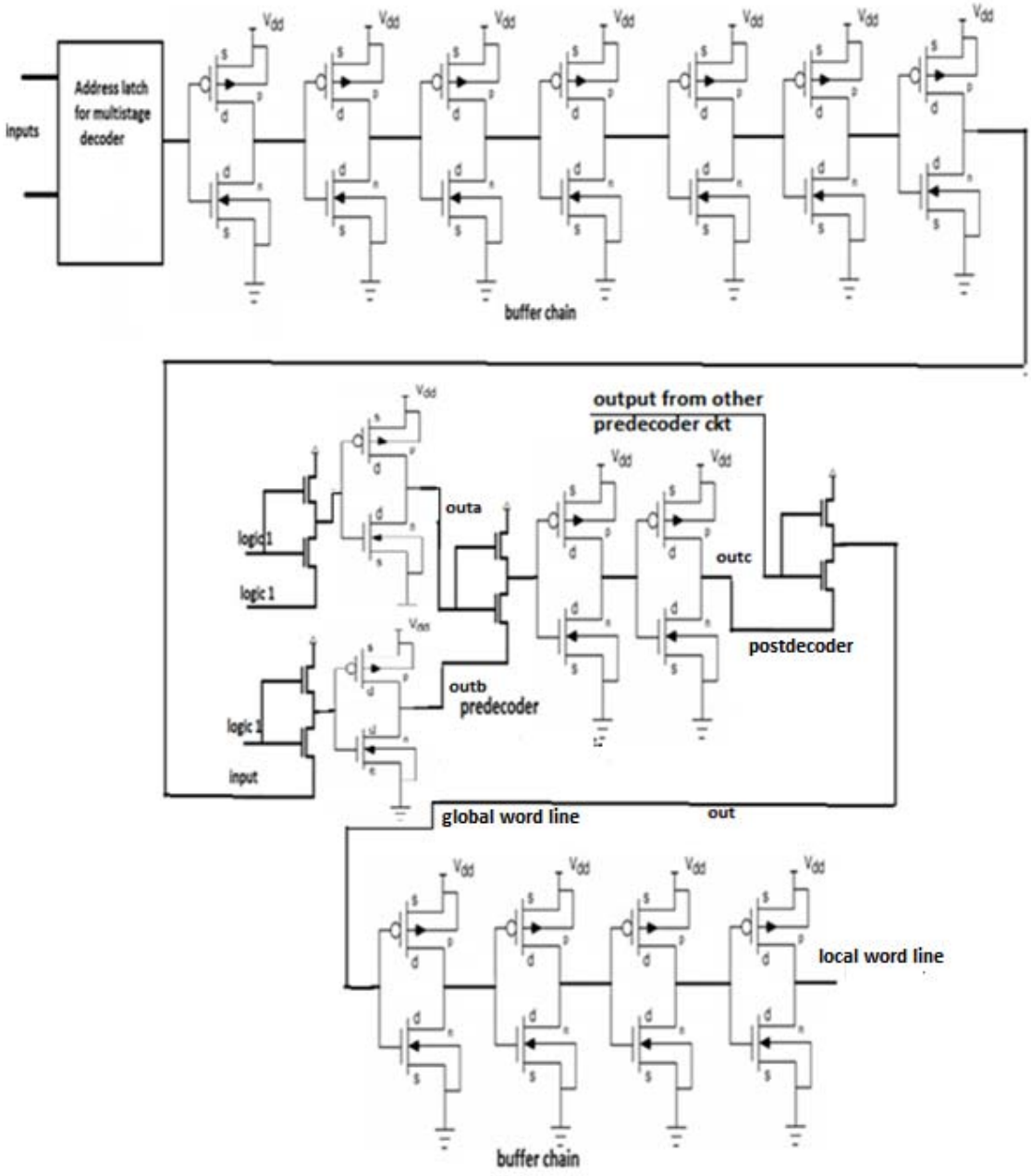

Figure 4.7 Critical path of an 8 to 256 decoder using source coupled NAND gate

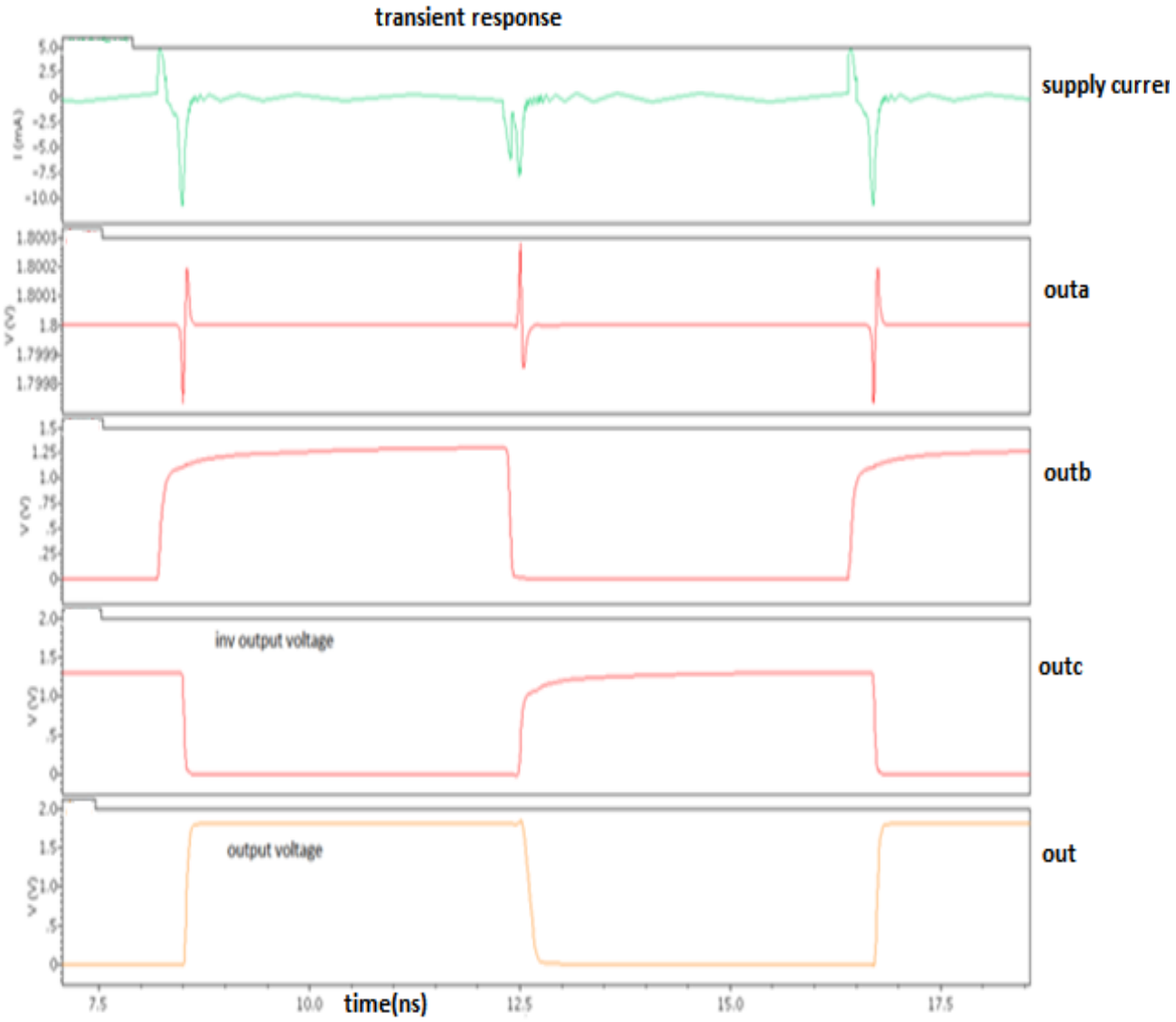

Figure 4.8 Output waveforms for the circuit in Figure 4.7

From the waveforms of decoder based on source coupled NAND gate the falling edge delay is found to be 210 ps and power dissipation 184µW.

Lastly, Table 4.4 gives the comparison of NOR style decoder and source coupled decoder in terms of delay and power dissipation. It has been observed that the delay power product for source coupled decoder is far much better than NOR style decoder and hence this confirms that a source coupled NAND gate is a prime candidate for efficient decoder design.

Table 4.4 Comparison of decoder circuits

Type of Decoder	**Power Dissipation (μW)**	**Delay (ns)**	**Power-Delay Product (μW − ns)**
NOR Style Decoder(using NOR Style NAND Gates)	330	0.219	72.27
Source Coupled Decoder using (source coupled NAND gates)	184	0.210	38.64

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

- In a decoder circuit branching takes place at various points, as a result of which capacitance increases at these points. In the critical path of a decoder using source coupled NAND gate the source terminal sees the capacitance of 16 transistors at the predecoder stage leading to the increase in capacitance at the source terminal. The $\frac{C_{out}}{C_{in}}$ becomes very small and the gate becomes faster than an inverter. Hence source coupled NAND gate takes the advantage of the structure itself and contribute very small power dissipation to the decoder circuit. Here the source capacitance is greater than its load capacitance but less than the gate capacitance leading to no switching power dissipation.

  In other NAND gate decoders the source terminal is grounded and input is given to the gate of the n-FETs and p-FETs. The capacitance seen by input is sum of gate capacitance of n-FETs and p-FETs. Hence load capacitance for predecoder circuit increases the switching power dissipation of the predecoder stage and contributes large power dissipation in the decoder circuit.
- Divided word line technique and memory banking have been used which further improves the performance.
- NOR style NAND gate gives better performance when number of inputs increases because n-FETs are connected in parallel. So logical effort remains constant.
- A multistage structure has been used for source coupled NAND decoder which decreases the overall capacitances hence the speed of the decoder circuit improves.

- The percentage change in the simulation results from pre layout simulation to post layout simulation for NOR style NAND gate and source coupled NAND gate are found to be less than 10%. This confirms that the optimized gates are suitable for fabrication.

Future scope:

A decoder consumes almost 30% of the total power in a memory circuit and hence it becomes mandatory to optimize a decoder circuit in the memory architecture. The main feature of the present work is to optimize the decoder designs in order to achieve better speed and power performance. This work can be extended by using various mixed design styles like MTCMOS, VTCMOS, gating technique etc. in the present architectures so as to get advanced decoder circuits in the field of memory design.

Paper Published

- Shivkaran Jain and Arun kr. Chatterjee, “NAND gate architectures for memory decoder”, IJCT, ISSN 22773061, vol 7, no. 2, pp 610-614, IF 1.341.

REFERENCES

[1] B.S. Amrutur and M.A. Horowitz, "Fast low-power decoders for RAMs", *IEEE Journal of Solid State Circuits*, vol. 36, pp. 1506 -1515, 2001.

[2] Xiao Chen, Dimitrios Velenis, "Effects of Parameter Variations on Low-Power SRAM Decoder", *International Review of Electrical Engineering (I.R.E.E.)*, Vol. 1, no.2, 2006.

[3] Shen-Fu Hsiao, Yo-Chi Chen, Ming-Yu Tsai and Tze-Chong Cheng, "Novel Memory Organization and Circuit Designs for Efficient Data Access in Applications of 3D Graphics and Multimedia Coding", *IEEE International Workshop on Memory Technology, Design, and Testing (MTDT'06)*, Taipei, Taiwan, pp. 6 – 42, 2006.

[4] John M. Raebey, A. Chandrakasan and B. Nicolic, "Digital Integrated Circuits", *PHI publication*, 2nd edition, 2003.

[5] Bharadwaj S. Amrutur, "Design and Analysis Of Fast Low Power SRAMs*", P.h.D Thesis*, Stanford University, 1999.

[6] K. Bernstein, K.M.Carrig, C.M.Durham, P.R. Hansen, D. Hoggenmiller, E.J.Navak and N.J. Rohrar," High Speed CMOS Design Style", *IBM Micro electronics, Kluver Publisher*, 2001.

[7] Sung-Mo Kang and Yusuf Leblebici, "CMOS Digital Integrated Circuits analysis and design", *Tata MCgraw Hill publication*, Third edition, pp. 386-388, 2003.

[8] H. Nambu, K. Kanetani, K. Yamasuka, K. Higeta, M. Usami, T. kusunaki, K. Yamaguchi and N. Homma"1.8ns access, 550mhz 4.5mb cmos sram", *IEEE International Solid State Circuits Conference, Digest of Technical Papers*, Vol:33 Issue:11,1998.

[9] James T. Kao and Anantha P. Chandrakasan, "Dual-Threshold Voltage Techniques for Low-Power Digital Circuits", *IEEE Journal Of Solid-state Circuits*, Vol. 35, No. 7, pp.1009-1018, 2000.

[10] Masahiko Yoshirnoto, Kenji Anarni, Hirofumi Shinohara, Tsutomu Yoshihara, Hiroshi Takagi, Shigeo Nagao, Shinpie Kayano, Takao Nakeno, "A 64kb CMOS

RAM with divided word line structure", *IEEE International Solid State Circuits Conference, Digest of Technical Papers*, Vol 26, pp:58-59, 1983.

[11] I. E. Sutherland and R. F. Sproull, "Logical effort: Designing for speed on the back of an envelope," *proceeding of the 1991 university of California/santa cruz conference on Advanced research in VLSI., pp:1-16, 1991*

[12] A. Kabbani, "Logical effort based dynamic power estimation and optimization of static CMOS circuits", *Elsevier, INTEGRATION, the VLSI journal* ,vol 43,pp: 279–288, 2010.

[13] G. Braceras, D. Evans, J. Conner," A 350mhz 3.3v 4mb sram fabricated in a 0.3mm cmos process", *IEEE International Solid State Circuits Conference, Digest of Technical Papers*,1997.

[14] Michael A. Turi and José G. Delgado-Frias, "High-Performance Low-Power AND and Sense-Amp Address Decoders with Selective Precharging", *IEEE International Symposium on Circuits, Devices & Systems*, vol 55, pp.: 1464 – 1467, 2008.

[15] Michael A.Turi, Jose´ G. Delgado-Frias, "Decreasing energy consumption in address decoders by means of selective precharge schemes", *Microelectronics Journal,* vol 40, pp:1590–1600, 2009.

[16] Chandrakasan, A. P., S. Sheng, and R. W. Brodersen, "Low-power Digital CMOS Design," *IEEE Journal of Solid State Circuits*, vol 24, pp. 473-484, 1992.

[17] Chandrakasan, A. P. and R. W. Brodersen, "Minimizing Power Consumption in Digital CMOS Circuits," *Proceedings of the IEEE*, vol 83, pp. 498-523, 1995.

[18] Qazi, M., M. E. Sinangil, A. P. Chandrakasan, "Challenges and Directions for Low-Voltage SRAM," *Design & Test of Computers, IEEE* , vol.28, no.1, pp.32-43, 2011.

[19] T. Chappell, Barbara A. chappell, Stanley E. Schuster, James W. Allan, Stephen P. Klepner, Rajiv V. Joshi and Robert L. Franch, "A 2-ns cycle, 3.8-ns access 512-kb cmos ecl sram with a fully pipelined architecture", *IEEE Journal of Solid State Circuits*, vol 26, pp:1577-1585, 1991.

[20] http://en.wikipedia.org/wiki/FO4.

[21] Neil H.E. Weste and K. Eshraghian, "Principle Of CMOS VLSI Design", *Addison Wesley Publishing Company*, 2nd edition,1988.

[22] Masato Matsumiya, Shoichiro Kawashima, Makoto Sakata, Masahiko Ookura, Toru Miyabo, Tom Koga, Kazuo Itabashi, Kazuhiro Mizutani, Hiroshi Shimada, and Noriyuki Suzuki., "A 15-ns 16-Mb CMOS SRAM with interdigitated bit-line

architecture", *IEEE Journal of Solid State Circuits*, vol. 27, no. 11, pp. 1497-1502, 1992.

[23] R. C. Jaeger, "Comments on An optimized output stage for MOS integrated circuits", *IEEE J. Solid State Circuits*, vol. SC-10, pp. 185–186, June 1975.

[24] http://www.ece.ucdavis.edu, Lecture 7 Gates with Lower LE RC Model Limitations Boris Murmann Center for Integrated System, Stanford University.

APPENDIX A

LAYOUT DESIGN RULES

The basic design rules are summarized below:

- Metal 1 to metal 1 spacing 0.24 μm
- Minimum contact size $(0.24*0.24)\mu m^2$
- Poly to poly spacing 0.24 μm
- Poly to metal spacing 0.28/0.00 μm
- Contact overlap to p+ diffusion 0.1 μm
- Metal 1 width 0.24 μm
- Poly extension beyond active 0.22 μm
- Minimum contact spacing 0.26 μm
- N well overlap p+ diffusion 0.43 μm
- Diffusion contact to poly spacing 0.15 μm
- Minimum p+ implant overlap p+ diffusion 0.22 μm
- Poly width 0.18 μm
- Minimum poly extension on to field region 0.22 μm
- Poly contact to diffusion edge spacing 0.18 μm
- Minimum poly overlap contact 0.1 μm
- Minimum metal area 0.1764 μm*μm
- Minimum metal2 width 0.28 μm
- Metal1 and metal2 overlap over via 0.08 μm
- Minimum equal potential N-well spacing 0μm or >=0.9μm
- Minimum non equal potential 1.8 V N well spacing 2μm

APPENDIX B

A.1 PTM level 53 model

This model is used for simulation of equivalent circuit model on Cadence.

.model nmos level = 53

+version = 4.0 binunit = 1 paramchk= 1 mobmod = 0

+capmod = 2 igcmod = 1 igbmod = 1 geomod = 1

+diomod = 1 rdsmod = 0 rbodymod= 1 rgatemod= 1

+permod = 1 acnqsmod= 0 trnqsmod= 0

+tnom = 27 toxe = 6.5e-010 toxp = 4e-010 toxm = 6.5e-010

+dtox = 2.5e-010 epsrox = 3.9 wint = 5e-009 lint = 1.35e-009

+ll = 0 wl = 0 lln = 1 wln = 1

+lw = 0 ww = 0 lwn = 1 wwn = 1

+lwl = 0 wwl = 0 xpart = 0 toxref = 6.5e-010 xl = -9e-9

+dlcig = 1.35e-009

+vth0 = 0.3692 k1 = 0.2 k2 = 0 k3 = 0

+k3b = 0 w0 = 2.5e-006 dvt0 = 1 dvt1 = 2

+dvt2 = 0 dvt0w = 0 dvt1w = 0 dvt2w = 0

+dsub = 0.078 minv = 0.05 voffl = 0 dvtp0 = 1e-011

+dvtp1 = 0.1 lpe0 = 0 lpeb = 0 xj = 7.2e-009

+ngate = 1e+023 ndep = 1.2e+019 nsd = 2e+020 phin = 0

+cdsc = 0 cdscb = 0 cdscd = 0 cit = 0

+voff = -0.13 nfactor = 2.3 eta0 = 0.0045 etab = 0

+vfb = -1.058 u0 = 0.0181 ua = -5e-010 ub = 1.7e-018

+uc = 0 vsat = 200000 a0 = 1 ags = 0

+a1 = 0 a2 = 1 b0 = 0 b1 = 0

+keta = 0.04 dwg = 0 dwb = 0 pclm = 0.06

+pdiblc1 = 0.001 pdiblc2 = 0.001 pdiblcb = -0.005 drout = 0.5

+pvag = 1e-020 delta = 0.01 pscbe1 = 2.0e+009 pscbe2 = 1e-007

+fprout = 0.2 pdits = 0.01 pditsd = 0.23 pditsl = 2300000
+rsh = 5 rdsw = 60 rsw = 30 rdw = 30
+rdswmin = 0 rdwmin = 0 rswmin = 0 prwg = 0
+prwb = 0 wr = 1 alpha0 = 0.074 alpha1 = 0.005
+beta0 = 30 agidl = 0.0002 bgidl = 2.1e+009 cgidl = 0.0002
+egidl = 0.8 aigbacc = 0.012 bigbacc = 0.0028 cigbacc = 0.002
+nigbacc = 1 aigbinv = 0.014 bigbinv = 0.004 cigbinv = 0.004
+eigbinv = 1.1 nigbinv = 3 aigc = 0.0213 bigc = 0.0025889
+cigc = 0.002 aigsd = 0.0213 bigsd = 0.0025889 cigsd = 0.002
+nigc = 1 poxedge = 1 pigcd = 1 ntox = 1
+xrcrg1 = 12 xrcrg2 = 5
+cgso = 7e-011 cgdo = 7e-011 cgbo = 0 cgdl = 7.5e-013
+cgsl = 7.5e-013 clc = 1e-007 cle = 0.6 cf = 1.1e-010
+ckappas = 0.6 ckappad = 0.6 vfbcv = -1 acde = 1
+moin = 15 noff = 1 voffcv = 0
+kt1 = -0.154 kt1l = 0 kt2 = 0.022 ute = -1.1
+ua1 = 1e-009 ub1 = -1e-018 uc1 = -5.6e-011 prt = 0
+at = 33000
+fnoimod = 1 tnoimod = 0 noia = 6.25e+041 noib = 3.125e+026
+noic = 8.75e+009 em = 41000000 af = 1 ef = 1
+kf = 0 tnoia = 1.5 tnoib = 3.5 ntnoi = 1
+jss = 1.2e-006 jsws = 2.4e-013 jswgs = 2.4e-013 njs = 1
+ijthsfwd= 0.1 ijthsrev= 0.1 bvs = 10 xjbvs = 1
+jsd = 1.2e-006 jswd = 2.4e-013 jswgd = 2.4e-013 xjbvd = 1
+pbs = 1 cjs = 0.0018 mjs = 0.5 pbsws = 1
+cjsws = 1.2e-010 mjsws = 0.33 cjswgs = 2.1e-010 cjd = 0.0018
+cjswd = 1.2e-010 mjswd = 0.33 pbswgd = 1 cjswgd = 2.1e-010
+mjswgd = 0.33 tpb = 0 tcj = 0 tpbsw = 0
+tcjsw = 0 tpbswg = 0 tcjswg = 0 xtis = 3
+dmcg = 0 dmci = 0 dmdg = 0 dmcgt = 0
+dwj = 0 xgw = 0 xgl = 0
+rshg = 0.4 gbmin = 1e-010 rbpb = 5 rbpd = 15
+rbps = 15 rbdb = 15 rbsb = 15 ngcon = 1

.model pmos level = 53

+version = 4.0 binunit = 1 paramchk= 1 mobmod = 0

+capmod = 2 igcmod = 1 igbmod = 1 geomod = 1

+diomod = 1 rdsmod = 0 rbodymod= 1 rgatemod= 1

+permod = 1 acnqsmod= 0 trnqsmod= 0

+tnom = 27 toxe = 6.7e-010 toxp = 4e-010 toxm = 6.7e-010

+dtox = 2.7e-010 epsrox = 3.9 wint = 5e-009 lint = 1.35e-009

+ll = 0 wl = 0 lln = 1 wln = 1

+lw = 0 ww = 0 lwn = 1 wwn = 1

+lwl = 0 wwl = 0 xpart = 0 toxref = 6.7e-010 xl = -9e-9

+dlcig = 1.35e-009

+vth0 = -0.25399 k1 = 0.2 k2 = -0.01 k3 = 0

+k3b = 0 w0 = 2.5e-006 dvt0 = 1 dvt1 = 2

+dvt2 = -0.032 dvt0w = 0 dvt1w = 0 dvt2w = 0

+dsub = 0.1 minv = 0.05 voffl = 0 dvtp0 = 1e-011

+dvtp1 = 0.05 lpe0 = 0 lpeb = 0 xj = 7.2e-009

+ngate = 1e+023 ndep = 4.4e+018 nsd = 2e+02 phin = 0

+cdsc = 0 cdscb = 0 cdscd = 0 cit = 0

+voff = -0.13 nfactor = 2.3 eta0 = 0.0037 etab = 0

+vfb = -1.058 u0 = 0.0023 ua = -5e-010 ub = 1.6e-018

+uc = 0 vsat = 78000 a0 = 1 ags = 1e-020

+a1 = 0 a2 = 1 b0 = 0 b1 = 0

+keta = -0.047 dwg = 0 dwb = 0 pclm = 0.1

+pdiblc1 = 0.001 pdiblc2 = 0.001 pdiblcb = 3.4e-008 drout = 0.6

+pvag = 1e-020 delta = 0.01 pscbe1 = 2e+009 pscbe2 = 9.58e-007

+fprout = 0.2 pdits = 0.08 pditsd = 0.23 pditsl = 2300000

+rsh = 5 rdsw = 60 rsw = 30 rdw = 30

+rdswmin = 0 rdwmin = 0 rswmin = 0 prwg = 0

+prwb = 0 wr = 1 alpha0 = 0.074 alpha1 = 0.005

+beta0 = 30 agidl = 0.0002 bgidl = 2.1e+009 cgidl = 0.0002

+egidl = 0.8 aigbacc = 0.012 bigbacc = 0.0028 cigbacc = 0.002

+nigbacc = 1 aigbinv = 0.014 bigbinv = 0.004 cigbinv = 0.004

+eigbinv = 1.1 nigbinv = 3 aigc = 0.012731 bigc = 0.00115

+cigc = 0.0008 aigsd = 0.012731 bigsd = 0.00115 cigsd = 0.0008
+nigc = 1 poxedge = 1 pigcd = 1 ntox = 1
+xrcrg1 = 12 xrcrg2 = 5
+cgso = 7e-011 cgdo = 7e-011 cgbo = 0 cgdl = 3e-011
+cgsl = 3e-011 clc = 1e-007 cle = 0.6 cf = 1.1e-010
+ckappas = 0.6 ckappad = 0.6 vfbcv = -1 acde = 1
+moin = 15 noff = 1 voffcv = 0
+kt1 = -0.14 kt1l = 0 kt2 = 0.022 ute = -1.1
+ua1 = 1e-009 ub1 = -1e-018 uc1 = -5.6e-011 prt = 0
+at = 33000
+fnoimod = 1 tnoimod = 0 noia = 6.25e+041 noib = 3.125e+026
+noic = 8.75e+009 em = 41000000 af = 1 ef = 1
+kf = 0 tnoia = 1.5 tnoib = 3.5 ntnoi = 1
+jss = 2e-007 jsws = 4e-013 jswgs = 4e-013 njs = 1
+ijthsfwd= 0.1 ijthsrev= 0.1 bvs = 10 xjbvs = 1
+jsd = 2e-007 jswd = 4e-013 jswgd = 4e-013 xjbvd = 1
+pbs = 1 cjs = 0.0015 mjs = 0.5 pbsws = 1
+cjsws = 9.4e-011 mjsws = 0.33 cjswgs = 2e-010 cjd = 0.0015
+cjswd = 9.4e-011 mjswd = 0.33 pbswgd = 1 cjswgd = 2e-010
+mjswgd = 0.33 tpb = 0 tcj = 0 tpbsw = 0
+tcjsw = 0 tpbswg = 0 tcjswg = 0 xtis = 3
+dmcg = 0 dmdg = 0 dmcgt = 0 xgw = 0
+xgl = 0
+rshg = 0.1 gbmin = 1e-012 rbpb = 50 rbpd = 50
+rbps = 50 rbdb = 50 rbsb = 50 ngcon = 1