Design & Characterization Of HD1P and HS1P Memory Compiler

Major Project Report

Submitted in partial fulfillment of the requirements

for the degree of

Master of Technology

in

Electronics & Communication Engineering

(VLSI Design)

By

Vaghasia Prashantkumar Jayantilal

(13MECV29)

Electronics & Communication Engineering Branch
Electrical Engineering Department
Institute of Technology
Nirma University
Ahmedabad-382 481
December 2014

Design & Characterization Of HD1P and HS1P Memory Compiler

Major Project Report

*Submitted in partial fulfillment of the requirements
for the degree of*

**Master of Technology
in
Electronics & Communication Engineering
(VLSI Design)**

By

Vaghasia Prashantkumar Jayantilal

(13MECV29)

Under the guidance of

External Project Guide:	**Internal Project Guide:**
Mr. Nitesh Gautam	**Prof. Vaishali Dhare**
Manager, R&D,	Assistant Professor (EC Dept.),
Synopsys India Pvt. Ltd.,	Institute of Technology,
Noida.	Nirma University, Ahmedabad.

**Electronics & Communication Engineering Branch
Electrical Engineering Department
Institute of Technology
Nirma University
Ahmedabad-382 481
December 2014**

Declaration

This is to certify that

a. The thesis comprises my original work towards the degree of Master of Technology in VLSI Design at Nirma University and has not been submitted elsewhere for a degree.

b. Due acknowledgment has been made in the text to all other material used.

- Vaghasia Prashantkumar J.

Certificate

This is to certify that the Major Project entitled **"Design & Characterization Of HD1P and HS1P Memory Compiler "** submitted by **Vaghasia Prashantkumar Jayantilal.(13MECV29)**, towards the partial fulfillment of the requirements for the degree of Master of Technology in VLSI Design , Nirma University, Ahmedabad is the record of work carried out by him under our supervision and guidance. In our opinion, the submitted work has reached a level required for being accepted for examination.The results embodied in this major project, to the best of our knowledge,haven't been submitted to any other university or institution for award of any degree or diploma.

Prof. Vaishali Dhare
Internal Guide

Mr. Nitesh Gautam
External Guide

Dr. N.M. Devashrayee
PG co-ordinator

Dr. P.N.Tekwani
Head of EE Dept.

Dr. K Kotecha
Director, IT-NU

Date:

Place: Ahmedabad

Acknowledgements

I would have never succeeded in completing my Thesis without the cooperation, encouragement and help provided to me by various people. Firstly, my sincere thanks to **Memory Design Team** for their help during this training. Their wisdom, clarity of thought and support motivated me to bring this project to its present state.

I am highly indebted to my Group manager **Mrs. Nutan Agarwal**, project manager **Mr. Nitesh Gautam**, my immediate supervisors **Mr Dinesh Gautam** and **M/s Shivangi Mittal** for providing necessary information regarding the project and also for their constant guidance, supervision, kind co-operation, and invaluable support in all aspects. My thanks and appreciations also go to my colleagues and team members in developing the project and for providing me with a lively and energetic work environment.

I would like to express my sincere gratitude to **Dr. Ketan Kotecha** (Director, Nirma University, Ahmedabad) for his continuous guidance, support and enthusiasm. I would take this opportunity to thank **Dr. P. N. Tekwani** (Head of Department, Electrical Engineering), **Dr. N. M. Devashrayee** (Professor and Program Coordinator, M.Tech - EC (VLSI Design)), Internal Guide **Prof. Vaishali Dhare** and all the faculties at Nirma University (VLSI Design), for their vision and relentless effort, support, and encouragement to provide me with this excellent opportunity to carry out my project work in such a highly renowned and esteemed organization, Synopsys Inc. I am equally thankful to **Synopsys Inc.** for providing me the invaluable exposure to the industry and the current market trends.

- Vaghasia Prashantkumar
13MECV29

Abstract

As the development of VLSI technology memory becomes crucial part for the SoC system. Now a days memory area becomes dominant in any SoC system as it takes around 70 percentage of area from total chip area. SRAM is generally used as on chip memory and gives its dominance in static power dissipation while in standby mode. For SoC system design different size of memories are required with different aspect ratio. It is also necessary that same size memory have different options in particular SoC system. Memory compiler is a tool which satisfies these requirements and gives memory instances with different configurations based on applied inputs.A design and characterization flow of memory compiler is shown in this project which explains step by step work of memory compiler development. Mainly focus has been paid to the low power and high performance SRAM design since they are most critical component in Soc and IoT devices. A key to improve performance is to choose optimum size of bit cell. There are tradeoff between area, power dissipation and performance. Therefore based on requirement compiler gives different options like array partitioning, column division to reduce dynamic power dissipation. There are various techniques like supply voltage biasing and dual rail are used to reduce standby power dissipation. In this project high dense and high performance single port 6T SRAM is discussed with its development flow from its basic component and simulation results are shown with appropriate conclusion.

Synopsys Inc. At A Glance

SYNOPSYS®

Accelerating Innovation

- A world leader in providing the semiconductor solutions that help our customers improve quality of life for everyone, both today and in the future
- Among the world's largest semiconductor companies
- A leading EDA (Electronic Design Automation) company serving all electronics segments
- Key strengths in Memory Compilers, semiconductor intellectual property (IP), Synopsys' comprehensive, integrated portfolio of system-level, IP, implementation, verification, manufacturing, optical and field-programmable gate array (FPGA) and EDA
- Chairman and CEO: Aart J. de Geus
- Approximately 9,000 employees
- Approximately 81 support & research & development centers around the globe
- Corporate Headquarters Mountain View, California
- Global presence with sales offices all around the world
- Public since 1992 - shares traded on NASDAQ Stock Market (Nasdaq:SNPS)
- Founded in 1986 by Dr. Aart de Geus and a team of engineers from General Electric's Microelectronics Center in Research Triangle Park, North Carolina

Group Introduction

Synopsys provides a broad portfolio of high-quality, silicon-proven embedded memory and logic library solutions, enabling system-on-chip (SoC) designers to lower integration risk and speed time-to-market.Synopsys provides a broad portfolio of high-quality, silicon-proven embedded memory and logic library solutions, enabling system-on-chip (SoC) designers to lower integration risk and speed time-to-market.

The DesignWare Duet Packages of Embedded Memories and Logic Libraries include memory compilers, ROMs, standard cells, Power Optimization Kits (POKs) and optional overdrive/low voltage PVTs that enable designers to achieve the maximum performance with the lowest possible power consumption for their specific application. The High Performance Core (HPC) Design Kit contains a suite of high-speed and high-density memory instances and logic cells specifically designed to enable SoC designers to optimize their CPU, GPU and DSP cores for maximum speed, smallest area, lowest power or an optimum balance of all three. In addition, the DesignWare STAR Memory System provides an integrated built-in self-test (BIST) and repair solution that improves test quality and manufacturing yield,while the DesignWare STAR Hierarchical System automates hierarchical testing foranalog/mixed-signal IP, digital logic blocks and interface IP on an SoC.

Synopsys also provides a comprehensive family of multiple-time programmable (MTP) and few-time programmable (FTP) non-volatile memory (NVM) IP in standard CMOS process technologies.

- Memory Compilers: DesignWare Memory Compilers are optimized for high performance and high density with advanced power management features. Integrated STAR Memory System for detection and repair of manufacturing faults improves yield. The memory compilers are also a part of the Design Ware Duet Packages and HPC Design Kit.

Contents

List of Figures

Chapter 1

Introduction

Semiconductor memory is used for data storage in any computing architecture either it is desktop or any portable mobile computing device. In modern VLSI design SRAM (Static Random Access Memory) takes its significant position. Now a days SRAM is major component in any electronic chip either it is microprocessor or any general computing device.This type of memory becomes very useful in high performance and low power chip design.

1.1 Objective and Scope Of this Project

In SoC (System on Chip) design, SRAM is most critical component with respect to SoC performance. SRAM generally takes around 70% of area in any SoC design. So focus on SRAM design directly reflects the SoC performance. Here my project work on efficient SRAM compiler design and its performance evaluation. Main concentration is given on low power SRAM design because in modern world use of battery operated gadgets increasing day by day. In VLSI technology miniaturization is necessary while this miniaturization directly affects the power dissipation. Overall battery technology is not mature as its required. So aim to do this project work is to reduce power dissipation with least effect on the performance which will give high speed, small size gadgets with long battery life.

1.2 Thesis Organization

Chapter-1 gives brief introduction about my project work.

Chapter-2 is based on literature survey in which I explained various types of memory used in semiconductor industry. Here I explained SRAM architecture and its basic operation flow. I also included basics of SRAM compiler and its design flow.

Chapter-3 is on Bitcell analysis which gives measurement parameters and methods for its measurement.

Chapter-4 describes various techniques to improve performance of SRAM and its effects on other parameters.

Chapter-5 explain characterization of SRAM instance and its evaluation flow from central database. It also tells about Quality Assurance done for memory compiler.

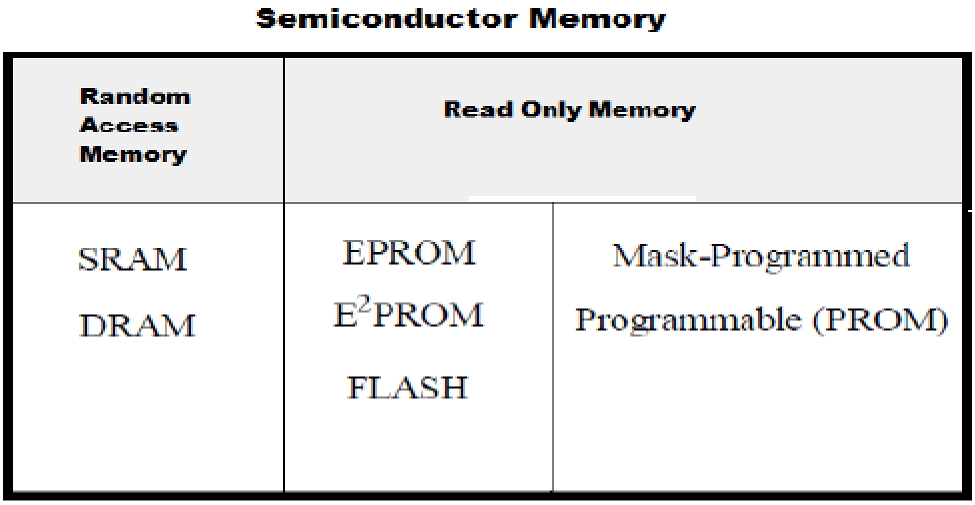
Chapter-6 gives results for project work and thesis ends with conclusion and future scope of this work done.

Chapter 2

Literature Survey

2.1 Classification Of Memory

Semiconductor Memories are mainly classified based on type of data access and the type of data storage mechanism. They are divided in to the following two groups.



Semiconductor Memory

Random Access Memory	Read Only Memory	
SRAM DRAM	EPROM E^2PROM FLASH	Mask-Programmed Programmable (PROM)

Figure 2.1: Classification Of Memory

Non-volatile Memory (NVM) also known as Read-Only Memory (ROM) which retains information when the power supply voltage is off. With respect to the data storage mechanism NVM are divided into the following groups:

Mask programmed ROM. The required contents of the memory is programmed during fabrication.
Programmable ROM (PROM). The required contents is written in a permanent way by burning out internal interconnections (fuses). It is a one-off procedure.
Erasable PROM (EPROM). Data is stored as a charge on an isolated gate capacitor (floating gate). Data is removed by exposing the PROM to the ultraviolet light.
Electrically Erasable PROM (EEPROM) also known as Flash Memory. It is also base on the concept of the floating gate. The contents can be re-programmed by applying a suitable voltages to the EEPROM pins. The Flash Memories are very important data storage devices for mobile applications.

Read/Write (R/W) memory, also known as **Random Access Memory (RAM).** From the point of view of the data storage mechanism RAM are divided into two main groups:
Static RAM, where data is retained as long as there is power supply on.
Dynamic RAM, where data is stored on capacitors and requires a periodic refreshment.[1]

2.2 Memory Architecture

Figure 2.2 shows generic SRAM architecture. This architecture consists array which contains data storage bitcell. There are two types of decoder used to read-write perticular location in array. There are sense amplifier used to resolve data stored in bitcell. There is a control circuitry to generate required signal to synchronous all components.
The RAM architecture consists of the following structures:

SRAM Bitcell , used to store one data bit.

Bit Line Precharge Circuit , precharges bit lines to compensate for voltage drop across pass transistors.

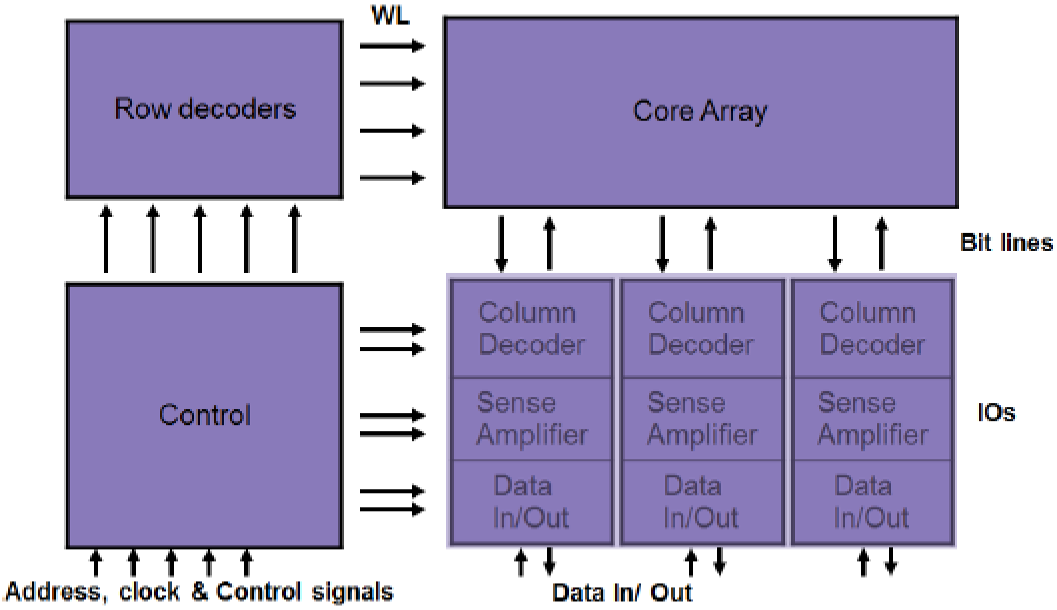


Figure 2.2: Memory Architecture
[9]

Write Buffers , buffers write-data so that it can write on RAM cells.

Sense Amplifier , Generate logic values based on difference on bit-line voltages.

Row & Column Decoders , for address generation to select particular bitcell .

2.3 6T SRAM Cell

Figure shows 6T SRAM cell which consists two cross-copled inverters. Here M1 and M2 are pull down nmos devices and M2 and M5 are pull up pmos devices. M3 and M4 are access transistors. M1, M5, M2 & M6 , combine of all four transistor used for data storage. Main efforts put to reduce this cell area such that millions of storage cells can fit on a chip. Steady state power dissipation is controlled by using larger threshold transistor in the array portion. This SRAM cell layout is highly optimized to reduce overall area in memory. For this some times M5 & M6 are replaced with undoped polysilicon. Such type of configuration called 4T SRAM because in cell it consists only four transistor. For power reduction current through this pull up resistor

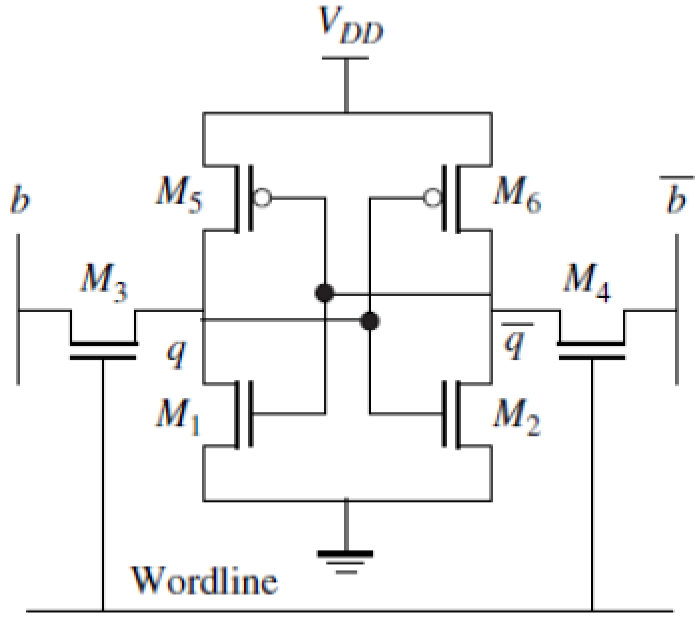


Figure 2.3: 6T SRAM BITcell
[2]

can be reduce by using large size. So there is trade off between area and power. So for this reason 6T is adopted by VLSI industry.[2]

The operation of SRAM array cells as follow. The row decoder selects word lines which run horizontally. All cells connected with that particular word line are available for read or write operation. Only particular row is selected others cells are disconnected from their respective word lines. There are two bit lines for particular cell which run vertically and its function to transfer data. Two column lines bit line and bitline-bar provide differential data path. Some times only one bitline used for data path .So in that architecture only one access transistor therefore called as 5T SRAM but it is not stable as 6T. Because of this symmetrical data paths are used.The word line has a large capacitance, which must be driven by the decoder.It is comprised of two gate capacitances per cell and the wire capacitance per cell. Once the cells along the word line are enabled, read or write operations are carried out. For a read operation, only one side of the cell draws current. As a result, a small differential voltage develops between bit and bitbar column lines. The columns address decoder and multiplexer select the column lines to be accessed. The bit lines will experience a voltage difference as the selected cells discharge one of the two bit lines. This difference is amplified and sent to output buffers. It should be noted that the bit lines also have a very large capacitance due to the large number of cells connected to them. This is primarily due to source/drain capacitance, but also has components due to wire capacitance and drain/source contacts. Typically, a contact is shared between two cells. During a write operation, one of the bit lines is pulled low if we want to store 0, while the other one is pulled low if we want to store 1. The requirement for a successful write operation is to swing the internal voltage of the cell past the switching threshold of the corresponding inverter after fliping word lines must go in reset to ensure single flip .[2]

The design of the cell involves the selection of transistor sizes for all six transistors to guarantee proper read and write operations. Since the cell is symmetric, only three transistor sizes need to be specified, either M1,M3, and M5 orM2, M4, and M6. The goal is to select the sizes that minimize the area, deliver the required performance, obtain good read and write stability, provide good cell read current, and have good soft error immunity.[2]

2.3.1 Read Operation

We now describe the design details of the 6T RAM cell for the read operation using. Assume that logic '0' is stored in the cell. Therefore, M1 is on and M2 is off. Initially, bt and bb are precharged to a high voltage by a pair of column pull-up transistors. The row selection line, held low in the standby state, is raised to VDD which turns on access transistors M3 and M4. Current begins to flow through M3 and M1 to ground. The resulting cell current slowly discharges the capacitance Cbit. Meanwhile, on the other side of the cell, the voltage on remains high since there is no path to ground through M2. The difference between bt and bb is fed to a sense amplifier to generate a valid low output, which is then stored in a data buffer. Upon completion of the read cycle, the word line is returned to zero and the column lines can be precharged back to a high value. When designing the transistor sizes for read stability, we must ensure that the stored values are not disturbed during the read cycle.[2]

2.3.2 Write Operation

The operation of writing 0 or 1 is accomplished by forcing one bit line, either bt or bb, low while the other bit line remains at about VDD. To write 1, is forced low, and to write 0, bt is forced low. The cell must be designed such that the conductance of M4 is several times larger than M6 so that the drain of M2 is pulled below VS. This initiates a regenerative effect between the two inverters. Eventually, M1 turns off and its drain voltage rises to VDD due to the pull-up action of M5 and M3. At the same time, M2 turns on and assists M4 in pulling output to its intended low value. When the cell finally flips to the new state, the row line can be returned to its low standby level. The design of the SRAM cell for a proper write operation involves the transistor pair M6-M4. When the cell is first turned on for the write operation, they form a pseudo-NMOS inverter. Current flows through the two devices and lower the voltage at node from its starting value of VDD. The design of device sizes is based on pulling node below VS to force the cell to switch via the regenerative action. Note that the bit line is pulled low before the word line goes up. This is to reduce the overall delay since the bit line will take some time to discharge due to its high capacitance.[2]

2.4 Introduction To Memory Compiler

Memory compiler is a nothing but simple memory instance generator. Memory designer design memory compiler IP with specific conditions which includes PVT , No of Words , No of Bits etc. IC designer use that memory compiler to generate SRAM for its SOC. IC designer selects on of instance from memory compiler within sprcified range by memory designer. So here build compiler once and use it again and again for same memory instances.

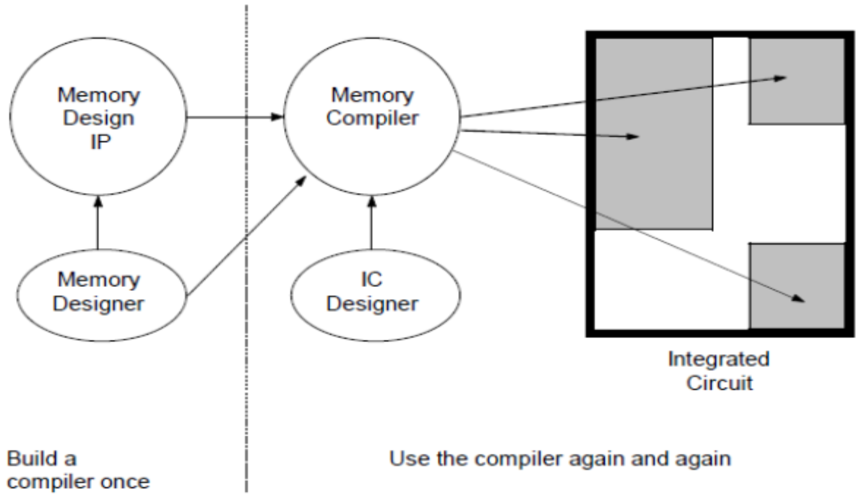
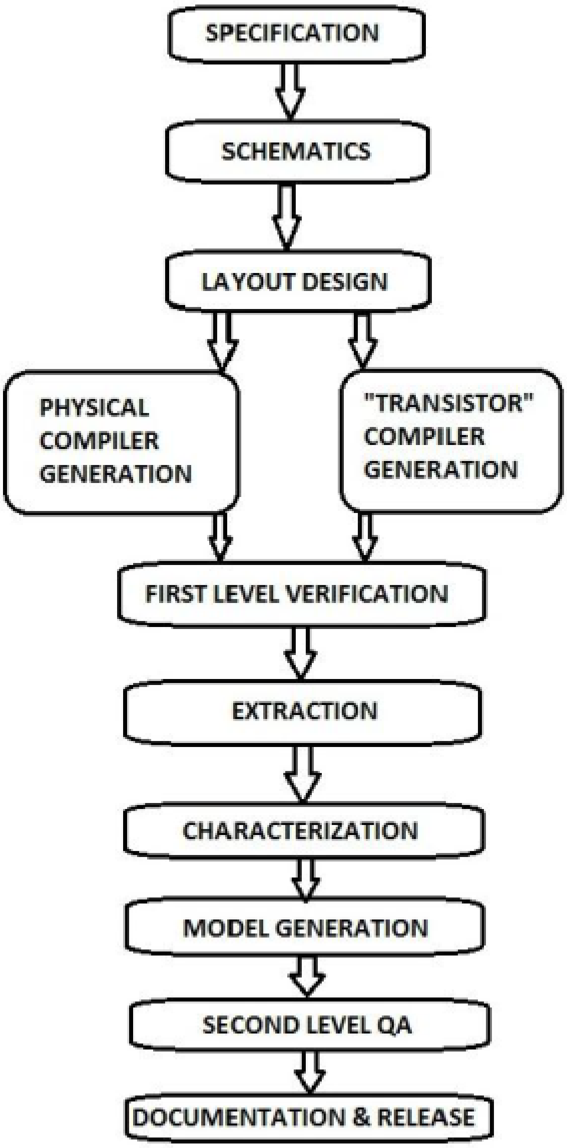


Figure 2.4: Memory Compiler
[9]

2.5 Memory Compiler Design Flow

Figure 2.5 shows memory compiler design flow. Its first start with schematic design from specification. After schematic design layout design and transistor level development works parallel. Then parasitics extracted from layout design and various timing margins , bitcell analysis & sense amp analysis done. After this first level verification done that finalized that all specification are matched. After this characterization of all memory instances has to be done. Then finally QA analysis and documentation done before compiler release.[9]



[htbp]

Figure 2.5: Memory Compiler Design Flow

2.6 Memory Compiler Classification

There are four possible 6T compiler based on transistor sizing and memory instance sizings are possible.

- HS1P :Transistor with higher W/L compared to HD1P
- HD1P :Transistor with lower W/L compared to HS1P
- HD1PRF :Same like HD1P but memory instances are small (with low number of bits & words)
- HS1PRF :Same like HS1P but memory instances are small (with low number of bits & words)

2.7 Basic Features Of Memory Compiler

2.7.1 Power Management

There are following three modes used for power management.

- **PS (Partial Sleep)** - Provides leakage reduction with fine-grained power gating and source biasing.
- **FS (Full Sleep)** - When the DS pin is asserted, integrated periphery power gating with data retention available and the memory outputs are held low.
- **SD (Shut Down)** - When the SD pin is asserted, there is a complete shutdown (both the periphery and array are power gated), with no data retention, and the memory outputs are held low.

2.7.2 Dual Rail Functionality

Separate voltage rails for the array and the periphery may be enabled at the instance level, with level shifters in the periphery. This option enables the array to be held at a safe operating voltage of VDDnominal while the periphery voltage is reduced (upto Nominal -30%) to save power. [9]

2.7.3 BIST Interface

The compiler (without redundancy) and compiler (with redundancy) options create memory instances that include all of the necessary logic to facilitate at-speed Built In Self Test (**BIST**). When the **bist_enable** option is enabled, the generated memory instance includes multiplexers (muxes) for all address, control and data signals as well as comparators and capture logic. All output signals are fully scannable and the data flows synchronous with the external clock. Incorporating this logic into the memory instance reduces the critical path when BIST is enabled and reduces the number of wires that are required to route between the memory instance and the BIST engine. The integrated logic will also enable high performance testing of functional logic surrounding the memory in your designs, using ATPG scan tools. Synchronous Write-through is available when these options are enabled. This allows input data to flow to output pins synchronously with the clock.[9]

2.7.4 Redundancy

The redundancy_enable option enables the memory compiler to generate memory instances that include redundancy for repair. When **redundancy_enable** is activated, additional memory is added to the instance to be used when BIST diagnostics determine that a repair is necessary. 1 element of 4 columns on each side of the center decoder for redundancy.[9]

2.7.5 Self Time Bypass

In this mode , the memory self time circuitry is bypassed. The memory timing is controlled by the external clock signal (**CLK**). Self Time bypass mode is initiated after the rising edge of **CLK** and is terminated by the falling edge. The Self Time bypass mode may be used to determine the margin of the internal self-timed circuitry.[9]

2.7.6 Read Margin Control

The memory array bitlines are pre-charged prior to a memory cell access. After accessing the memory (Read cycle), a differential signal develops between the bitlines (bitline and bitline bar). This differential signal is fed to the input of the bitline sense amplifier to determine what data is stored in the bitcell. Since it takes time for the differential signal on the bitlines to develop, the greater the time delay prior to strobing the sense amplifier, the greater will be the differential signal at the input to the sense amplifier. A low sense amplifier differential signal is susceptible to noise and sense amplifier input voltage offset. Higher input differential voltage results in greater reliability of the sensed data. However, delaying the time when the sense amplifier is strobed results in a longer cycle time, reducing maximum operating speed and increasing access time (Tac and Tcq increase). Hence the trade off of memory speed verses yield/reliability. The longer you wait, the easier it is for the sense amplifier to determine what was stored in the memory cell. Thus the term Robustness. The longer you wait, the longer it takes to access the cell (i.e., access time). Thus, the term Speed Tradeoff. This parameter enables the selection between high yield and high performance READ/WRITE margin settings. All memories are characterized with four **timing_mode** settings that control the compile time options available with the compiler.[9]

- **FAST**: Used for increased performance (speed) for designs with low memory bit count on chip.
- **SLOW**: Used for chips with very high memory bit count, lower operating speed. Can be used for debug purposes.
- **DEFAULT**: Recommended setting, optimum yield.

- **VDDMIN**: This is a licensed option. It is used to operate the memory instance at the foundry specified minimum Vdd condition. Very slow operating speed. Can be used to assist debug.

2.7.7 Synchronous Write Through

Synchronous write-through is used for test coverage improvement. It lets signals travel across the memory and enables the internal memory logic to be visible to the external test circuit. This eliminates the hidden embedded circuit effect of the internal memory logic. The test circuit can be a scan circuit or any other type of circuit. Synchronous writethrough will be controlled with the **DFTMASK** pin. When **DFTMASK=1**, the output latch will follow the pipeline output data. When **DFTMASK=0** the output will remain in the previous state. Synchronous write-though muxes will be placed before the output latches. The clock going to the output latches must be capable of going to Logic-1, during ATPG mode. Synchronous write-though will depend on both clock domains for a 2-port register file assuming that CLKA and CLKB are in different clock domains.[9]

2.7.8 Optional Periphery Transistor Vt Selection

A compile-time option, **periphery_Vt**, is offered to select the periphery transistor threshold voltage implant (Vt). The array transistor threshold implants remain unchanged by this compile-time option.[9]

2.8 Summary

This chapter explain about semiconductor memory and its classification.Basics of memory compiler and its features also included in this chapter.

Chapter 3

Bitcell Perfaormance Evaluation

In this chapter,we will see how bitcell and sense amplifier of SRAM analized.The main parameters for BitCell Analysis are as follows.

- Write Margin
- Leakage Current
- Read Current
- Static Noise Margin

3.1 Write Margin

Write margin is defined as ability to write bitcell.For smaller write margin it is harder to write a bitcell. Write margin mesures in two following domain.

- Write Margin in Time Domain
- Write Margin in Voltage Domain

3.1.1 Write Margin in Time Domain

There are following two types of write margin measurements in time domain.

- **Bit Line Driven Write Margin**

  When RC of bitline is more than RC of wordline , in this case bit line driven write margin measurement is used. Here we used bit line as a control parameter. So called as a bit line driven write margin.

  Figure 3.1 shows setup for bit line driven write margin in time domain.

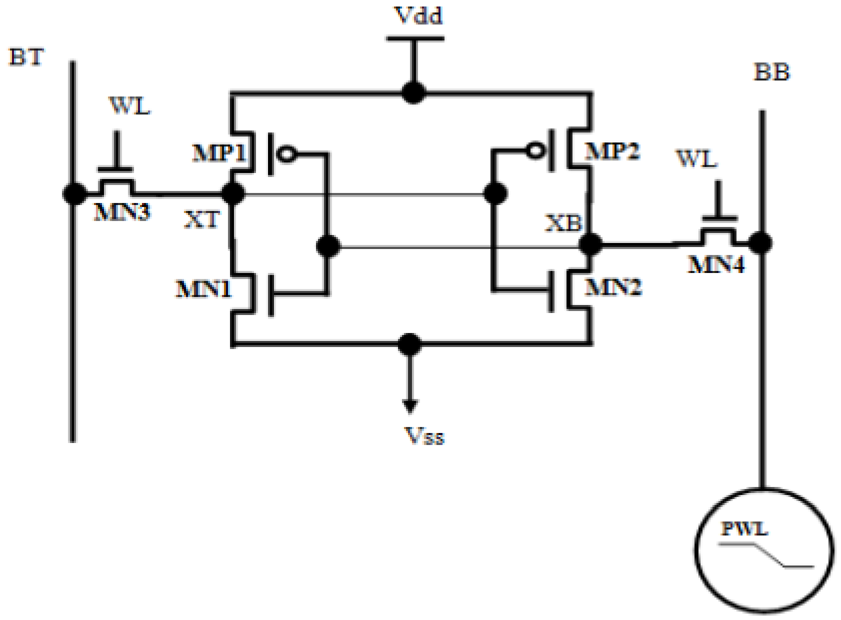


Figure 3.1: Write Margin (BL Driven) Setup
[9]

- **Word Line Driven Write Margin**

  When RC of wordline is more than RC of bitline , in this case wordline driven write margin measurement is used. Here write margin is controlled by wordline so called as a wordline driven write margin.

  Figure 3.2 shows setup for wordline driven write margin in time domain.

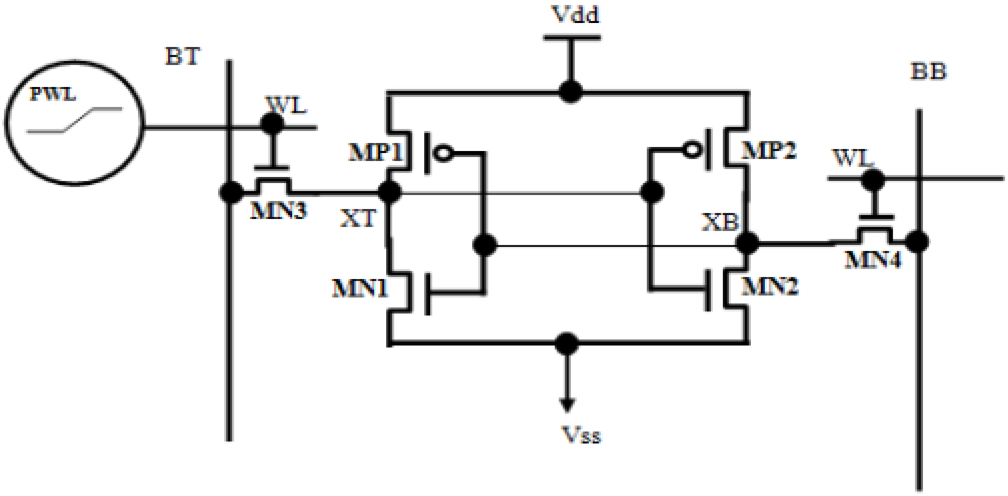


Figure 3.2: Write Margin (WL Driven) Setup
[9]

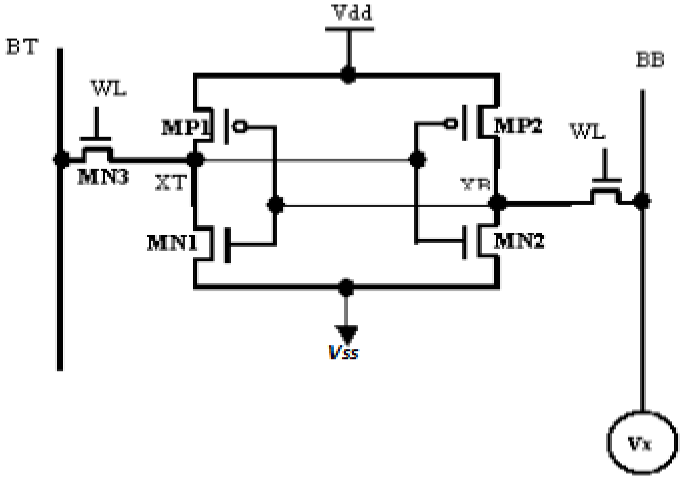


Figure 3.3: Write Margin Voltage Domain Setup
[9]

3.1.2 Write Margin in Voltage Domain

Figure 3.3 shows setup for voltage domain write margin.It is measurement of minimum bitline voltage to change state of bitcell.For smaller write margin , more efforts needed to write bitcell. Write margin must be greater than or equal to 50% of Vdd.

3.1.3 Transisstor Sizing Effects On Write Margin

– **Pass transistor sizing effects on Write Margin**
  Pass transistor (W/L) ratio directly propotional to write margin. By increasing the (W/L) of pass transistor,we can increase write margin. If (W/L) of pass transistor increases then its resistance will decreases and storing '1' at XB node.

– **Pull-Up transistor sizing effects on Write Margin**
  Pull-Up transistor (W/L) inversely proportional to write margin.So, increasing the (W/L) of Pull-Up transistor degrades write margin.

– **Pull-Down transistor sizing effects on Write Margin**
  Pull-Down transistor (W/L) inversely proportional to write.So, increasing (W/L) of Pull-up transistor degrades write margin.

3.2 Leakage Current & Read Current

In normal memory operation only one wordline is on and same way one pair of bit line and its complimentary bitline is on. So , here leakage is one that is produced by other cells which are not activated by its word line and bitline. For every read operation we first precharge bitlines. Therefor it is necessary to measure leakage current for desirable read operation. Read current must be higher than leakage current otherwise it may produce wrong decision during read operation. There is following condition for leakage in memory.

$$I_{leak} < I_{read}/N \tag{3.1}$$

where N is equal to total cells in a column

Figure 3.4 shows setup to measure leakage current.

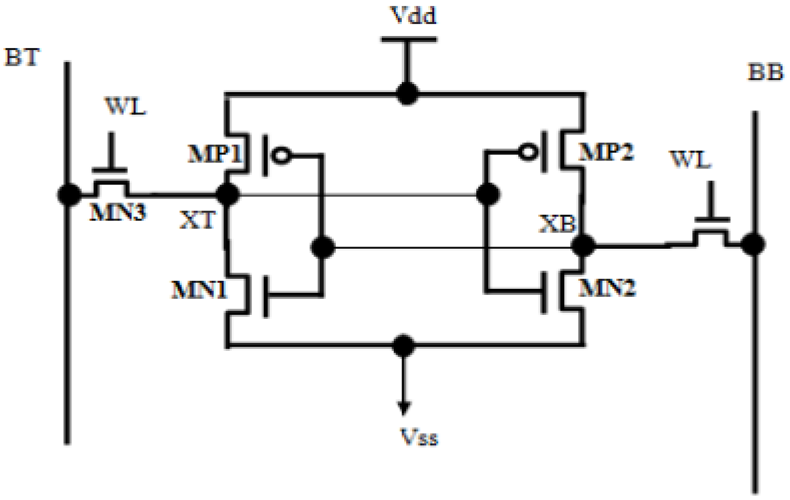


Figure 3.4: Leakage & Read Current Setup
[9]

Read current is calculated as a current passing through read path during read operation while leakage is one that is calculated during idle mode.

3.3 Static Noise Margin

One most important aspect for SRAM cell design is stability of the cell. The cell stability determines the sensitivity of the memory to process tolerances and operating conditions. Stability of the cell is expressed by STATIC NOISE MARGIN.

3.3.1 SNM Dependencies

- Transistor width modulation
- Word line modulation
- Bit line modulation
- Power supply voltage modulation
- Temperature

3.3.2 SNM Analysis

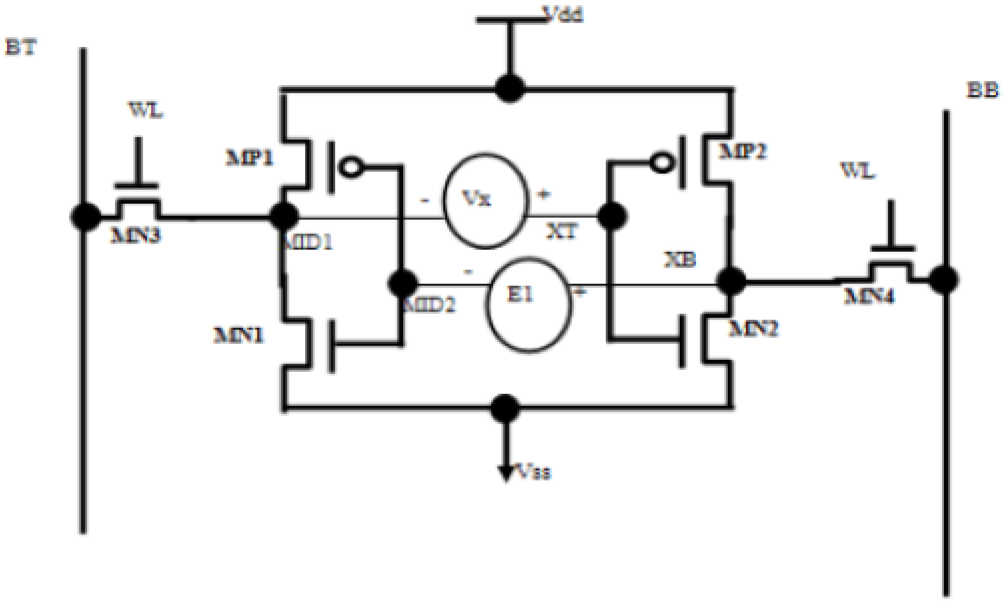


Figure 3.5: SNM Calculation Setup
[9]

Figure 3.5 shows setup for Static Noise Margin calculation. SNM is calculated by applying predetermined noise using voltage control voltage source. Initialize

XT to 0 and XB to Vdd. Bit lines (BT, BB) and Word line (WL) are at Vdd. Introduce noise in the form of voltage sources Vx and E1 (voltage coupled to Vx). Slowly increase Vx from 0 and monitor the points MID1 and XB to see when the cell flips VMID2 + VNOISE less than VtMN1 (Threshold Voltage of Pull down Transistor). Till this criterion is satisfied the memory cell will not flip. As soon as VA+VNOISE becomes more than the threshold voltage of the pull down transistor, the memory cell get flipped. Thus, care must be taken in deciding the transistor sizes while designing the memory cell.

3.3.3 Transistor Sizing effects on SNM

- **Pass Transistor sizing effects on SNM**

  Pass transistor (W/L) ratio and Static Noise Margin are inversely proportional. By increasing the (W/L) ratio of pass transistor, Static Noise Margin decreases.

- **Pull-Up Transistor sizing effects on SNM**

  Pull-Up transistor (W/L) and Static Noise Margin are directly proportional to each other. For SNM improvement we have to increase (W/L) of Pull-Up transistor. Assuming the bitcell as a resistive n/w we can see that resistance will decrease by incresing the (W/L) of Pull-Up transistor so voltage drop will be small and VDD will be maintained on B node.

- **Pull-Down transistor sizing effects on SNM**

  Pull-Down transistor (W/L) and Static Noise Margin are directly proportional to each other. By increasing (W/L) of Pull-up transistor,we can increase Static Noise Margins.

3.4 Summary

This chapter tells about SRAM bitcell critical parameters and various setup to calculate it.

Chapter 4

Performance Improvements Of SRAM

Higher speed with low power consumption is most prior requirement in modern VLSI design. To achieve this requirement we have to adopt some techniques. There are following techniques used for SRAM timing and leakage improvement.

4.1 Timing Improvement Techniques

There are main two parts in SRAM chip called array and periphery. SRAM performance can be boosted by increasing transistor size but we have limitations on area also. So we have done some architectural changes to improve timing of SRAM.

4.1.1 Bank Structure

In SRAM chip there are mainly two capacitances word line capacitance and bit line capacitance. Both are responsible for 'clock to data' timing path.

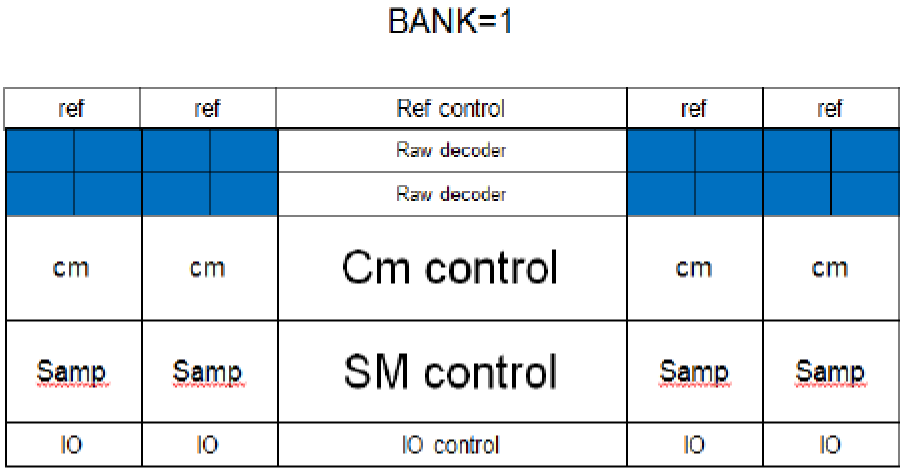


Figure 4.1: BANK-1 Memory Architecture

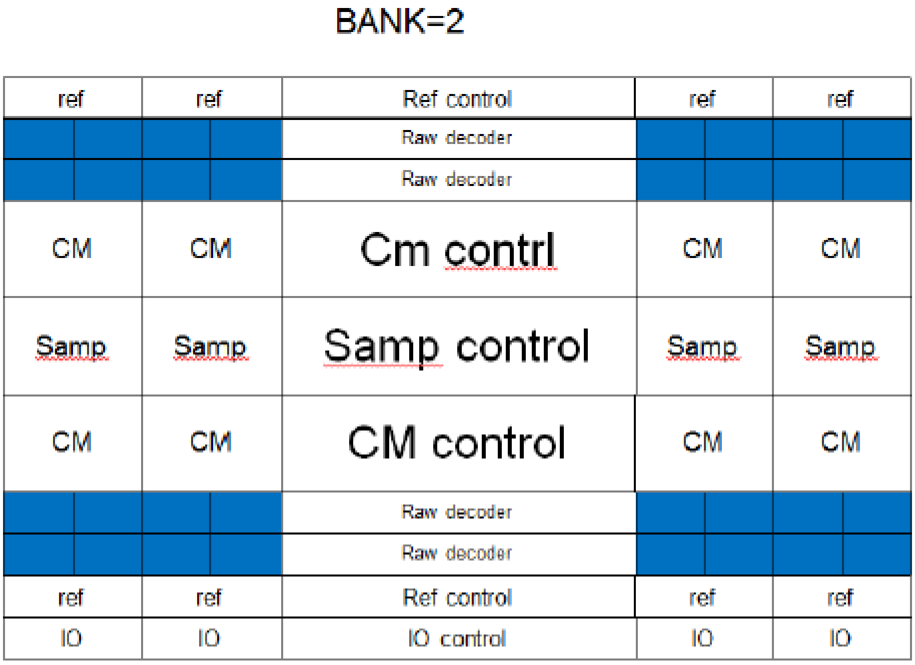


Figure 4.2: BANK-2 Memory Architecture
[9]

In tall memories physical rows are larger than physical columns. So bitlines have larger capacitance which means large RC delay. It will increase bitcell to sense amplifier delay, ultimately increase in clock to data path timing and it will degrade performance of SRAM.we reduce height of memory than RC of bitline reduces and hence improve performance. For this we have to make architectural changes. One possible change in architecture to introduce Bank structure. In bank structure physical rows divide in sections and each section controlled by separate control block which results in area overhead.

4.1.2 Central Decoding

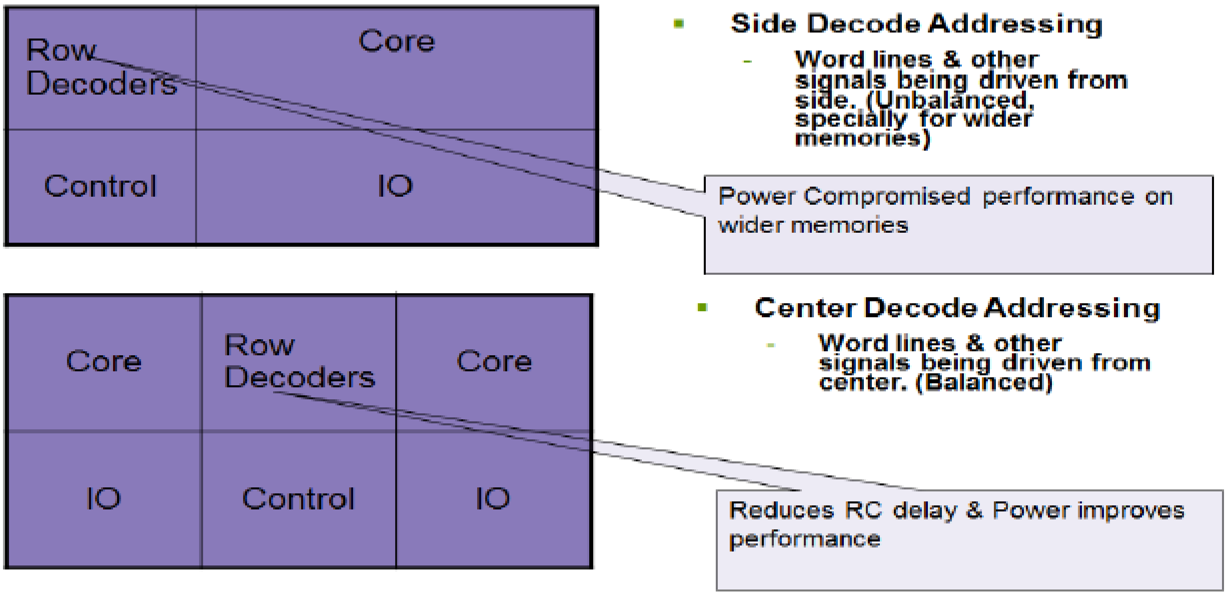

Row Decoders
Core
Control
IO
Side Decode Addressing
Word lines & other signals being driven from side. (Unbalanced, specially for wider memories)
Power Compromised performance on wider memories
Core
Row Decoders
Core
IO
Control
IO
Center Decode Addressing
Word lines & other signals being driven from center. (Balanced)
Reduces RC delay & Power improves performance

Figure 4.3: Central Decoding Scheme
[9]

For wider memories, no of physical columns are more than physical rows. So parasitics of word line are more which result in degradation of rising slope of word line which means there is a larger access time for the bit cell. So we must reduce length of word line which reduce RC on word line. Because of this we use central decoding for wider memories in which length of word line is divide by two. In central decoding technique we put control block and row decoders in middle rather than side. Figure shows the difference between conventional side decoding and central decoding.

4.1.3 Use Of Low Vt Devices In Periphery

There is another way to increase access speed is to reduce threshold of periphery devices. Therefore clock to data time reduces but there is a power consumption problem comes in to picture. For low Vt devices leakage current is increase. So, here timing performance can be improved at the cost of increase in power consumption.

4.2 Power Gating For Leakage Improvement

Now a days technology is advanced in scaling of devices, device dimensions becomes smaller and smaller which results in second order effects become dominant. The subthresold leakage current increases which results in higher leakage. This is major problem in planner devices with reduce in channel length.

In power gating we use memory in three modes.

- Partial Sleep (Core Biasing)
- Full Sleep (Periphery off with data retention)
- Shut Down(Array also off with no data retention)

But you have to think about area of sleep transistors. Routing becomes complex as well as IR drop in sleep transistor. Here mainly focus given on wake up latency and rush current flow through sleep transistor.

4.2.1 Core Bias

Drain and source junctions are kept in reverse in normal MOSFET operation. There is a leakage current due to minority charge carriers near these reverse biased junction called junction leakage. The other source of leakage current in short channel MOSFET due to small distance between drain and soure junction. Therefore drain voltage also control channel and some leakage causes on small gate voltage less than threshold. So to educe this leakage current we have to reduce Vds. There is a technique to reduce Vds called Core Bias.

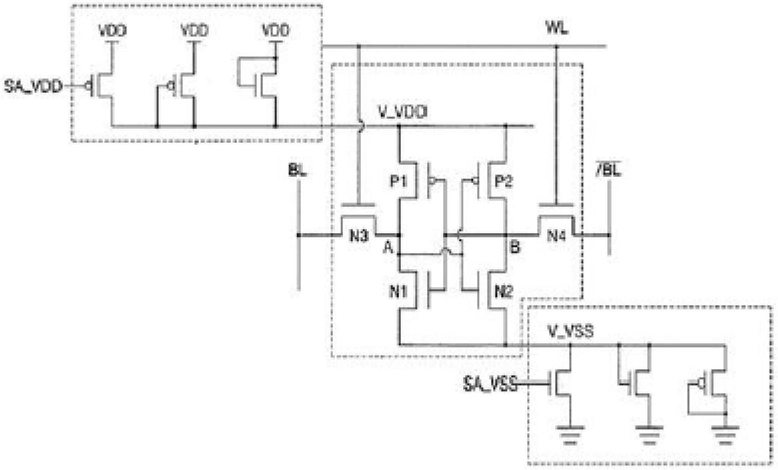


Figure 4.4: Core Bias
[9]

Core bias circuit is shown in figure 4.4. We can reduce Vds either reducing Vdd or increasing Vss. Here two MOSFET act like diode (shorting source and gate of MOSFET) connected parallel to each other are with small size compared to other MOSFETs. These diodes are in reverse biased and based on signal SA_VDD and SA_VSS which values are Vdd and Vss respectively increase in V_Vss and decrease in V_Vdd occurs. So finaly Vds reducing by above circuit which results in leakage improvement.

4.2.2 Daisy Chain for Power Gating

In full sleep mode we turn off periphery supply to reduce leakage in SoC.This prevents data stored by an array.In shut down mode we also turn off supply from array portion so here it lost its data. But we have to think about wake up.In power gating , large no of sleep transistors being on simultaneous.So a very large rush current flows which make large IR drop causes functional errorand memories corrupted.So we have following possible solution called daisy chain:

There are two types of daisy chain used single daisy chain and dual daisy chain for wake up current and latency control.

- **Single Daisy Chain**

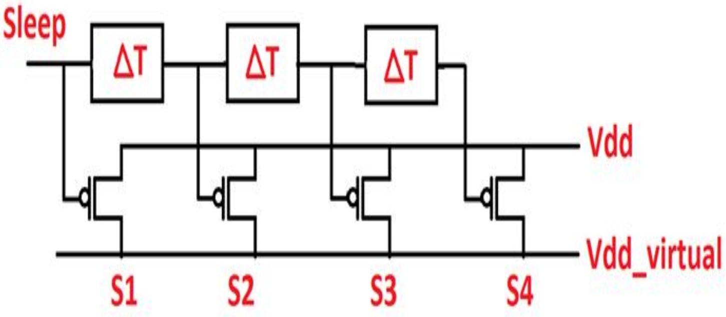


Figure 4.5: Single Daisy Chain

Figure shows single daisy chain used to control large rush current during wake up. Here delay between two stages selected such that it make balance for latency and rush current. Long delay make circuit slow while small may cause large rush current which results large IR drop.

- **Dual Daisy Chain**

Another method to use dual daisy chain which is more effectively work.Here trickle chain is used to reduce rush current which contains small size of transistor. It slowly rise Vdd(virtual) to Vdd. Then after level detector is

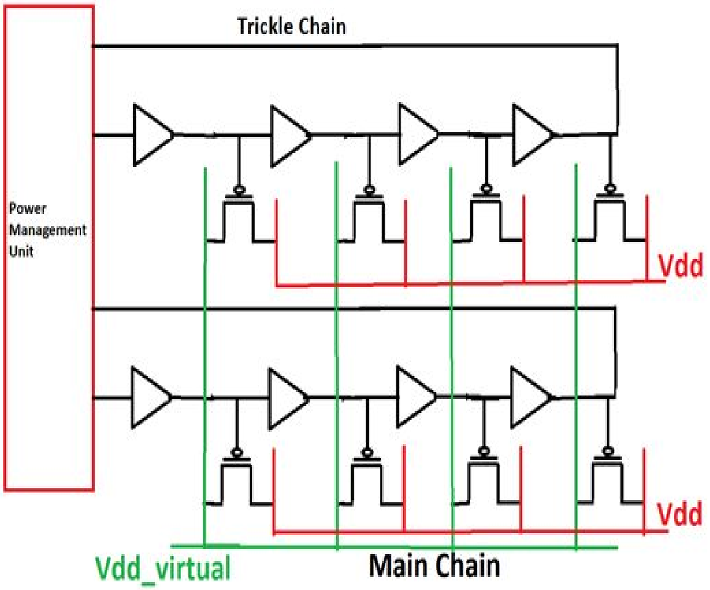


Figure 4.6: Dual Daisy Chain

used to turn on main chain which rise Vdd(virtual) to Vdd in small time and reduce wakeup latency of waking circuit.

4.2.3 Use of High Threshold Voltage Devices

By using high threshold voltage devices subthresold leakage will reduce which results in leakage performance improvement.

4.3 SNM Improvement Technique

Higher SNM (Static Noice Margin) memories always desirable because it has higher noise immunity. For higher SNM noice effect should be less. So a circuit called Read Assist(RA) is used to improve SNM.

In the circuit shown in figure 4.7 a big inverter used to handle WL and two pmos connected in parallel with drain is connected to WL and source is connected to Vss.There are four possible combinations possible because of RA0 and RA1

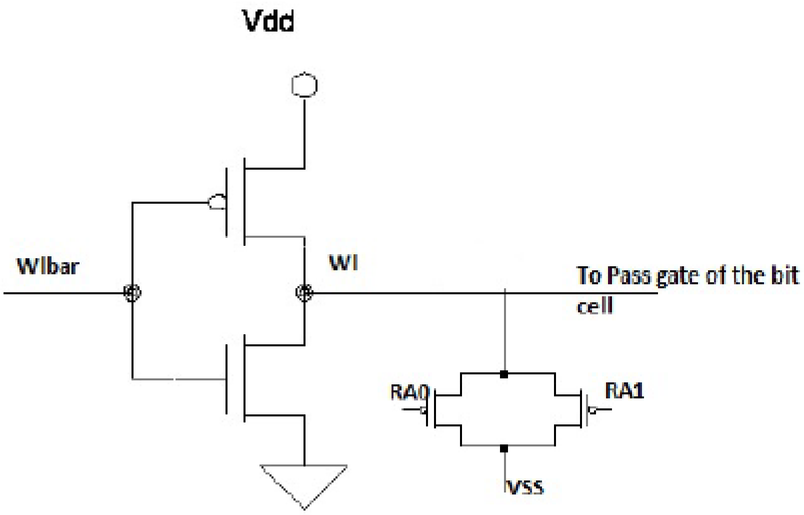


Figure 4.7: Read Assist
[9]

value.When any or both pmos on it tries to pull down WL to Vss but it is not possible because pmos sizes are small. So, it pulls down WL about 10% to Vdd. So when WL goes low small current pass through nmos and it reduces possibility to cell flip. But here tradeoff between DC write margin and SNM because of this circuit increses efforts to write bitcell. So based on four possible combination of RA0 and RA1 ,optimum combination is choosen.

4.4 Summary

This chapter includes various techniques to improve performance of SRAM memory. There are various techniques explained in this chapter for SNM , leakage and timing improvements.

Chapter 5

Compiler Characterization & Evaluation

There is a central database file which includes all timing and power related information. This file is called as a compiler.cdb file. There are two process in memory compiler design.

- Characterization
- Evaluation

5.1 Characterization

Characterization: To describe the actual behavior of an existing piece of software or a circuit in Real time. In Real time condition following factors affected:

- Process
- Voltage
- Temperature and many more

In characterization, simulation done on particular instance and all timing and power related measurements are done. This information related to power & timings are dump in to compiler.cdb file. This simulation is on particular instance. So, called as instance characterization. Below figure explain about instance characterization.

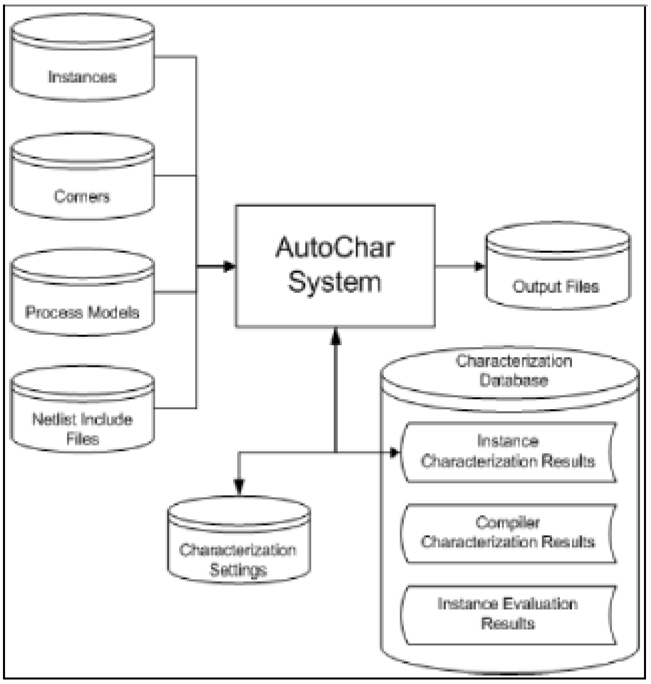

Instances
Corners
Process Models
Netlist Include Files
AutoChar System
Output Files
Characterization Database
Instance Characterization Results
Compiler Characterization Results
Instance Evaluation Results
Characterization Settings

Figure 5.1: Instance Characterization
[9]

Before QA analysis compiler's whole instances are characterized. So, these type of simulation in which whole compiler is characterized called Compiler characterization. Below figure explain about compiler characterization.

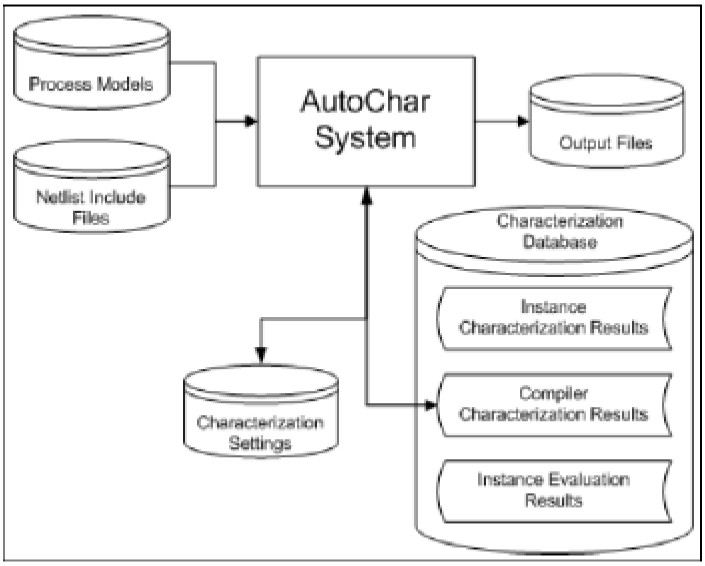

Process Models
Netlist Include Files
AutoChar System
Output Files
Characterization Database
Instance Characterization Results
Compiler Characterization Results
Instance Evaluation Results
Characterization Settings

Figure 5.2: Compiler Characterization
[9]

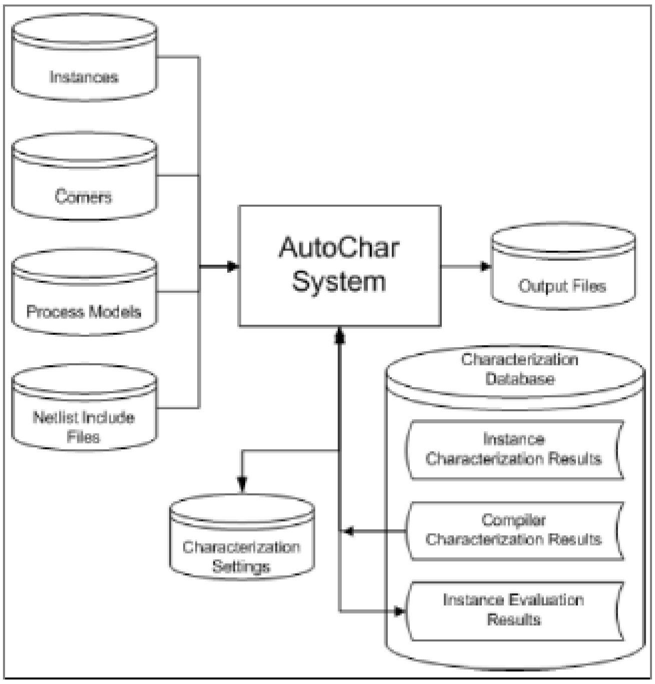

Instances
Corners
Process Models
Netlist Include Files
AutoChar System
Output Files
Characterization Database
Instance Characterization Results
Compiler Characterization Results
Instance Evaluation Results
Characterization Settings

Figure 5.3: Instance Evaluation
[9]

5.2 Evaluation

In instance evaluation output is coming from compiler characterization results. There is only evaluation based on compiler characterization information. Below figure shows instance evaluation flow. If information about particular instance is not present in compiler characterization results ,it gives results from interpolation by taking of near by instance's characterization results.

5.3 Quality Assuarance Of Memory Compiler

There are following QA checks done on compiler before release to check quality of designed compiler.

5.3.1 Primetime

Primetime check is used for timing analysis. It is a verilog model verification. This command checks the Synopsys library syntax and ensures the timing arcs defined in the Synopsys library exist in the Verilog model. The Synopsys PrimeTime suite, including PrimeTime, PrimeTime SI, PrimeTime PX and PrimeTime VX, provides a single, golden, trusted signoff solution for timing, signal integrity, power and variation-aware analysis.[9]

5.3.2 Libscreen

Libscreen check is to check monotonicity of the liberty file data. This check verifies the slopes of the timing arcs.[9]

5.3.3 Timever

Timever is a timing verification tool. Timing verification is the process of determining that a given design can be operated at a specific clock. frequency without errors caused by a signal arriving too soon or too late . For example, if the data input of a latch arrives after the closing edge of the clock (a setup violation), or if the data input changes before the closing edge of the previous clock (a hold violation), the latch may not store the data correctly. This check adds timing parameters to the Verilog and checks for read/write setup/hold , tcc on the Verilog.[9]

5.3.4 LibCompare

Libcompare Compare datasheets between compiler versions given in tcl file. It compares timing power and area between two compilers and it reports the percentage deviation in the above mentioned parameters in XLS format.[9]

5.3.5 Celtic

Celtic is a noise analysis tool provided by cadence. It is used to perform signal integrity sign-off . It identifies nets with low noise immunity to avert potential noise-related problems and lethal silicon failures before tapeout. The CeltIC analyzer accurately calculates the impact of noiseon both the delay and functionality of cell-based designs. It performs SoC noise analysis and generates repairs back into place-and-route.[9]

5.3.6 Ccsn

Ccsn is composite current source for noise. This model is used for noise analysis. It enables you to generate CCS Noise for accurate noise calculation and validate the noise immunity of the cells in Synopsys compilers.[9]

5.3.7 Ccst

Ccst is composite current source for time. This model is used for time analysis. The CCS timing model is an open source model which is part of the Liberty format specification. Characterization guidelines, development tools, and library validation and correlation tools are available to speed the library characterization and qualification process. To expediteand simplify CCS library qualification, Synopsys provides a new library QA capability part of Library Compiler that can be used to check the completeness and accuracy of all acquired CCS timing models in a library. In addition, Library Compiler can be used in correlation modeto verify the accuracy of the characterization.[9]

5.3.8 Ecsm

Effective current source modeling. This check is used for Ecsm verification. ECSM is a delay calculation method that uses a current-based cell driver model and a variable pin capacitance receiver model for highly accurate cell-based delay calculation. The model is particularly good at predicting the effect of non-linear waveforms on high impedance interconnects.The model requires additional cell characterization data such as the output current profile for the active transitions of each cell in the library and a variable input pin capacitance table.[9]

5.3.9 Espcv

It is formal equivalence check between Verilog model and a structural model created by tool to verify functionality. ESP-CV is a symbolic simulation-based formality verification tool intended to perform custom equivalence(EQ) checking and provide functional verification coverage for fullcustom IC design.[9]

5.3.10 Funcver

This is Verilog Functional verification. Synopsys VCS tool is used for this. VCS offers industry-leading performance and capacity, complemented by a complete collection of advanced testbench, bug-finding, coverage and assertion technologies. VCS multicore technology delivers a 2x verification speed-up and cuts down verification time by running the design, testbench, assertions, coverage and debug in parallel on machines with multiple cores.[9]

5.3.11 Redhawk

Redhwak is a power integraty solution. It is an IRPower Analysis tool that enables more accurate static and dynamic analysis for on state and ramp up mode. We can easily utilize different utilities for Redhawk to run IR/Power analysis.[9]

5.3.12 Prescreen

Prescreen is to create LVS and DRC report for corner instances. It helps in generating the Prescreen information (Like Mini QA for BE & Mini QA on Functional & timing checks and much more) by generation , verification and reporting for Synopsys compiler.[9]

5.3.13 FamilyVerify

This check reports for all the files and corresponding errors. Also check the structure of the compiler. Compiler family validation tests are intended to assure quality of a compiler before it is released. Normally, you run family validation if you want to: Validate that an existing compiler conforms to one or more family definitions Update an existing compiler to force it to conform to one or more family definitions Create a new compiler that is already conforming to one or more family definitions.[9]

5.3.14 IQA

IQA is the integrated QA. It checks full compiler. For corner instances it checks Lib VS db. Pins transition etc. It ensure the quality of instances before the compilers are released. The IQA check also include the capability to ensure that antenna diodes are always present. Transistor recognition is performed so that a given piece of diffusion geometry can be recognized as a source or drain to a gate. If it is not a sourcedrain, it is a diode and is recorded appropriately.[9]

5.4 Summary

This chapter is about of how characterization and evaluation are carried out in the memory compiler design. This chapter also includes various quality checks which are carried out after whole compiler design and prior to hand over customer to maintain reliable relationship with customer.

Chapter 6

Results & Conclusion

6.1 Bitcell Analysis For HD1P

For HD1P compiler, bitcell transistor used with smaller W/L compared to HS1P.

Following are the W/L ratioes for particular transistor in bitcell.

- Pass Gate Transistor W/L = 1.8571
- Pull Up Transistor W/L = 1.4228
- Pull Down Transistor W/L = 2.7142

The results of various design parametrs of bitcell for HD1P compiler are as below which are simulated with various PVT.

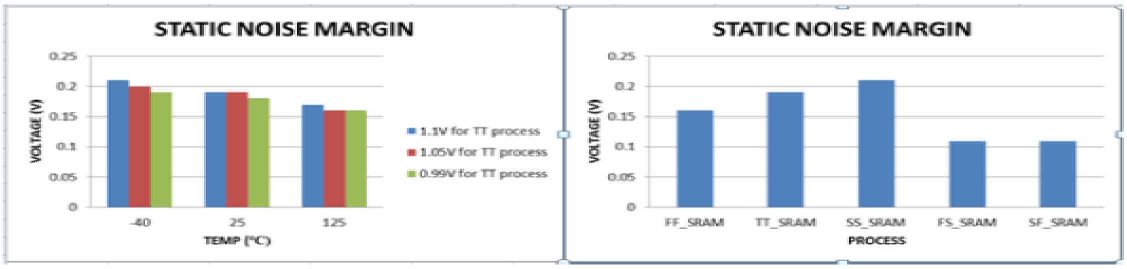


Figure 6.1: SNM Results (HD1P)

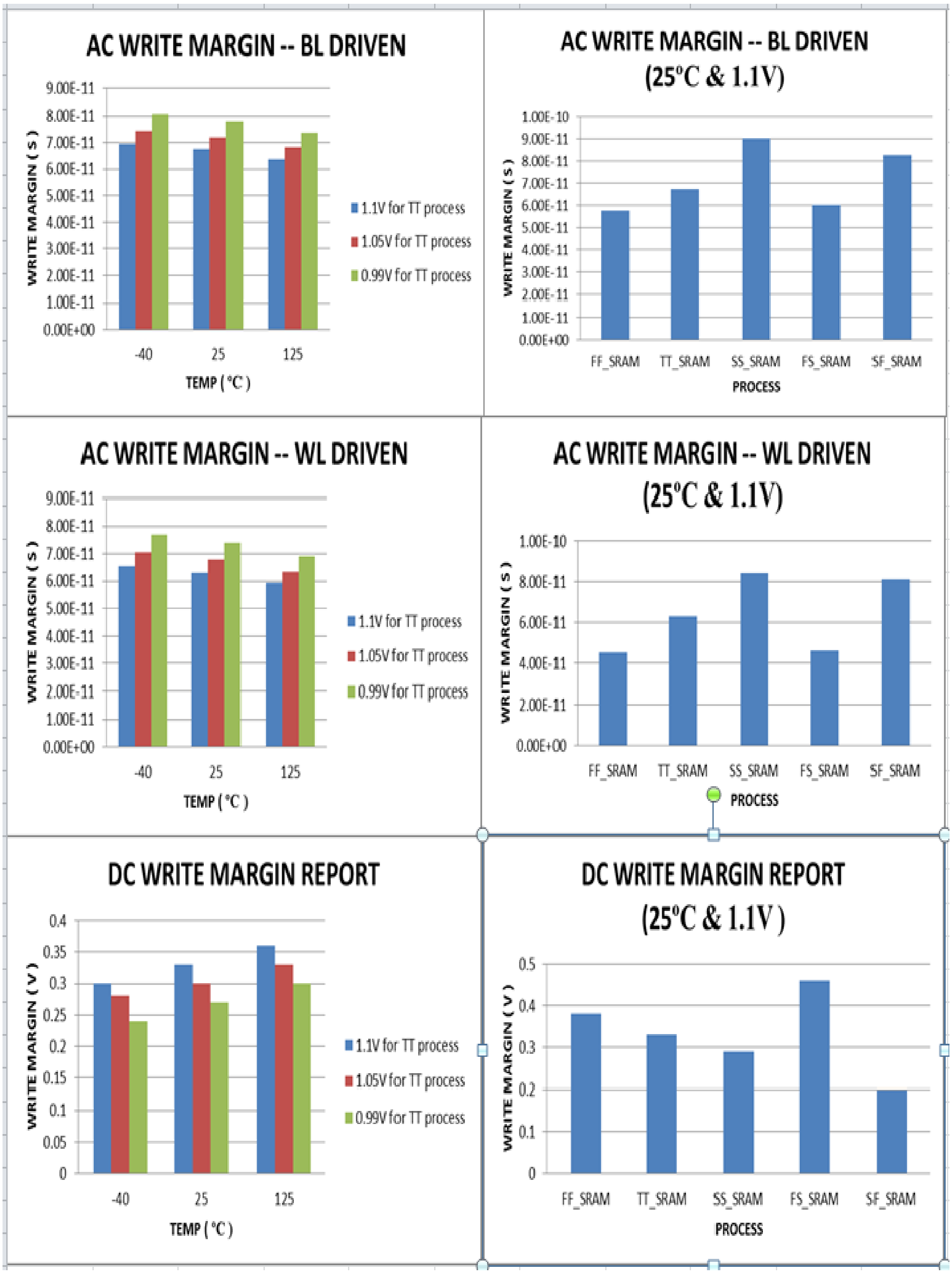


Figure 6.2: Write Margin Results(HD1P)

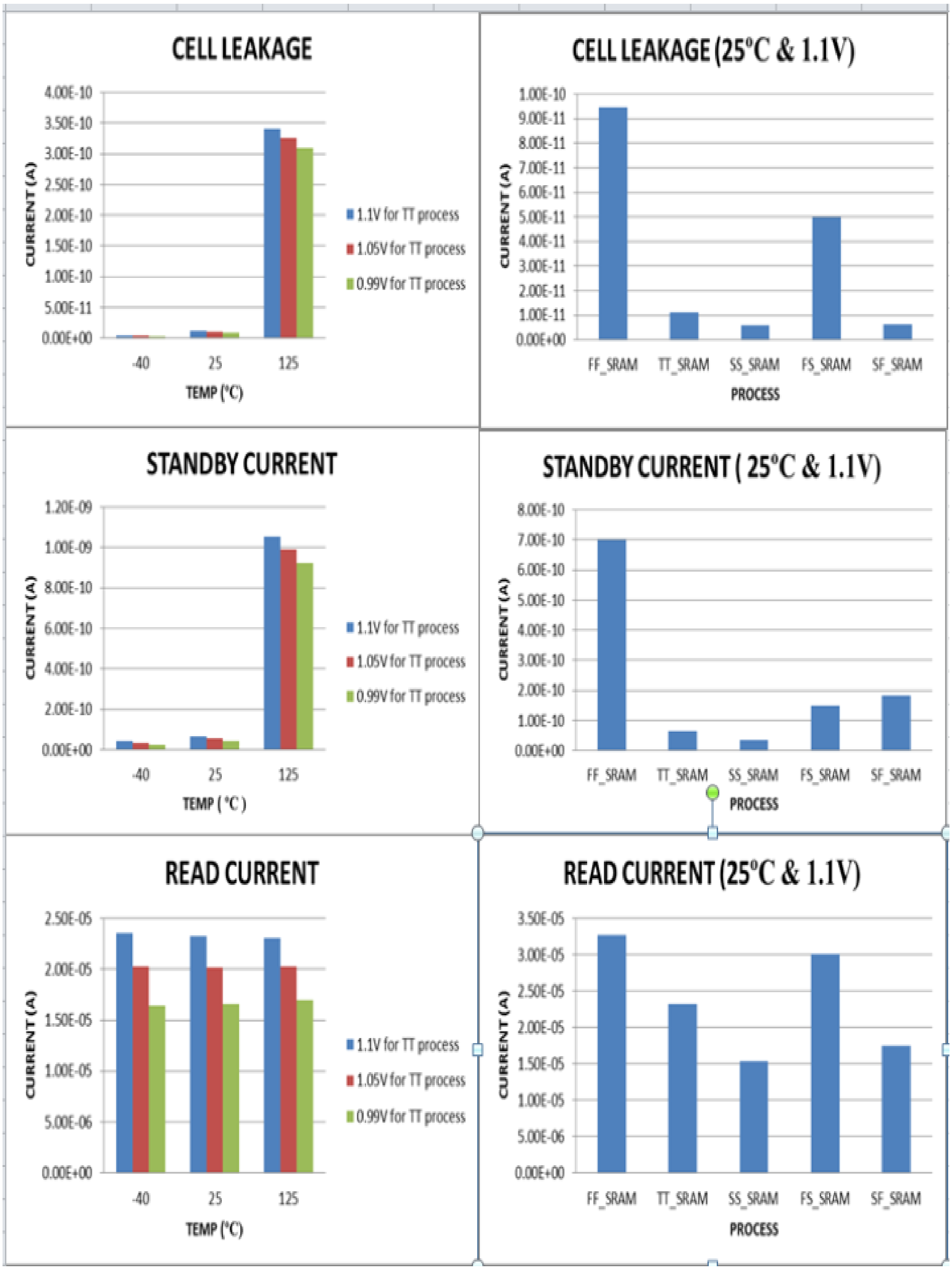


Figure 6.3: Current Results (HD1P)

6.2 Bitcell Analysis For HS1P

For HS1P compiler, bitcell transistor used with large W/L compared to HD1P. Following are the W/L ratioes for particular transistor in bitcell.

- Pass Gate Transistor W/L = 3
- Pull Up Transistor W/L = 1.4285
- Pull Down Transistor W/L = 4

The results of various design parametrs of bitcell for HS1P compiler are as below which are simulated with various PVT.

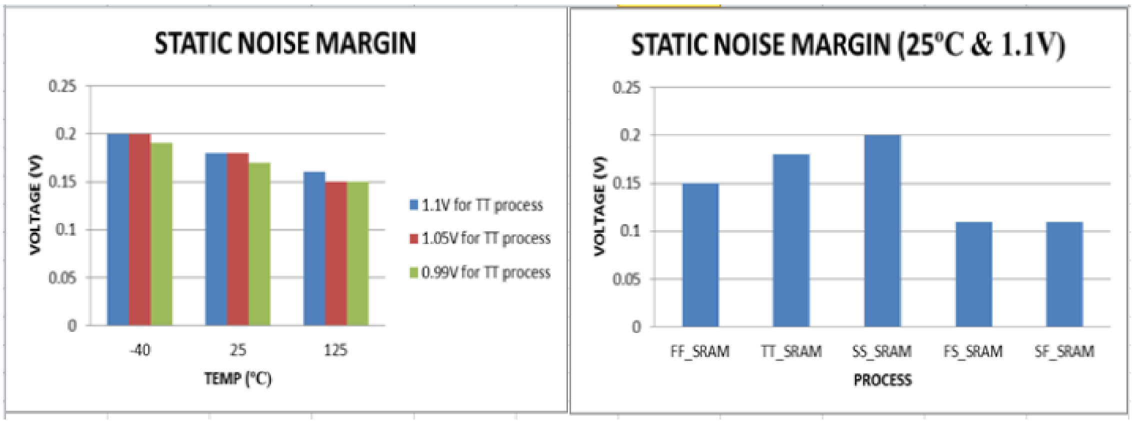


Figure 6.4: SNM Results (HS1P)

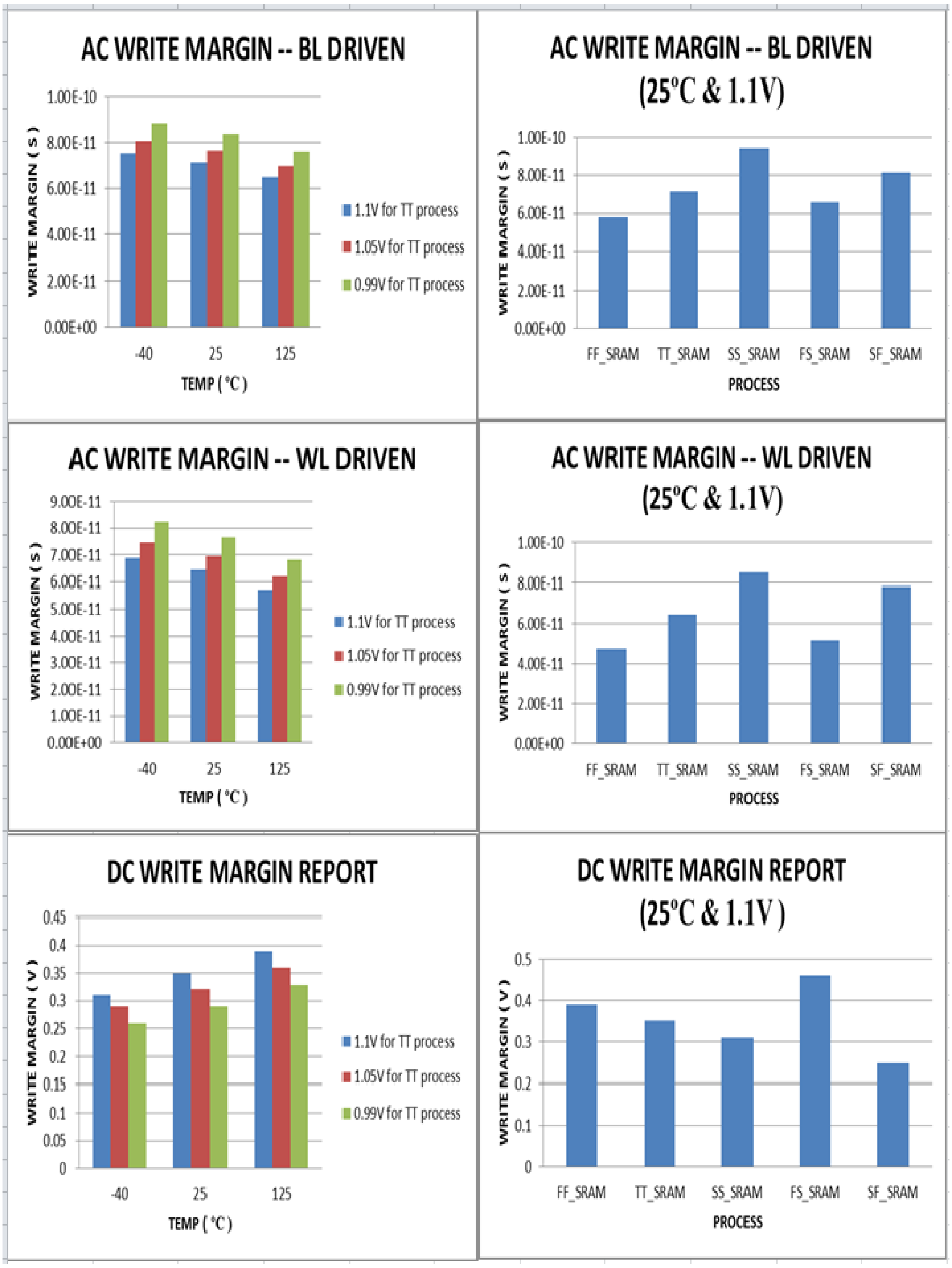


Figure 6.5: Write Margin Results(HS1P)

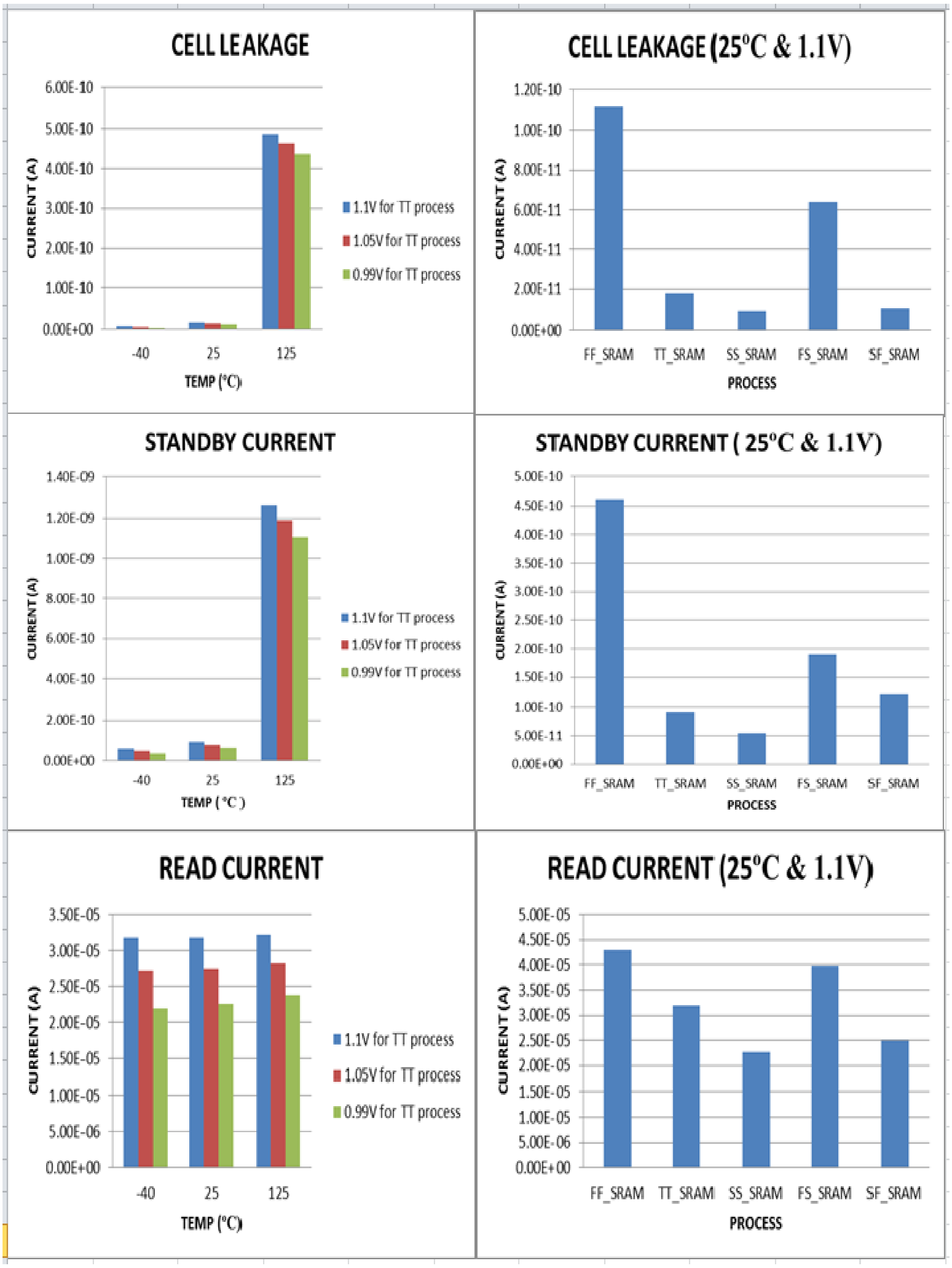


Figure 6.6: Current Results (HS1P)

6.3 Comparison Between HS1P & HD1P Bitcell

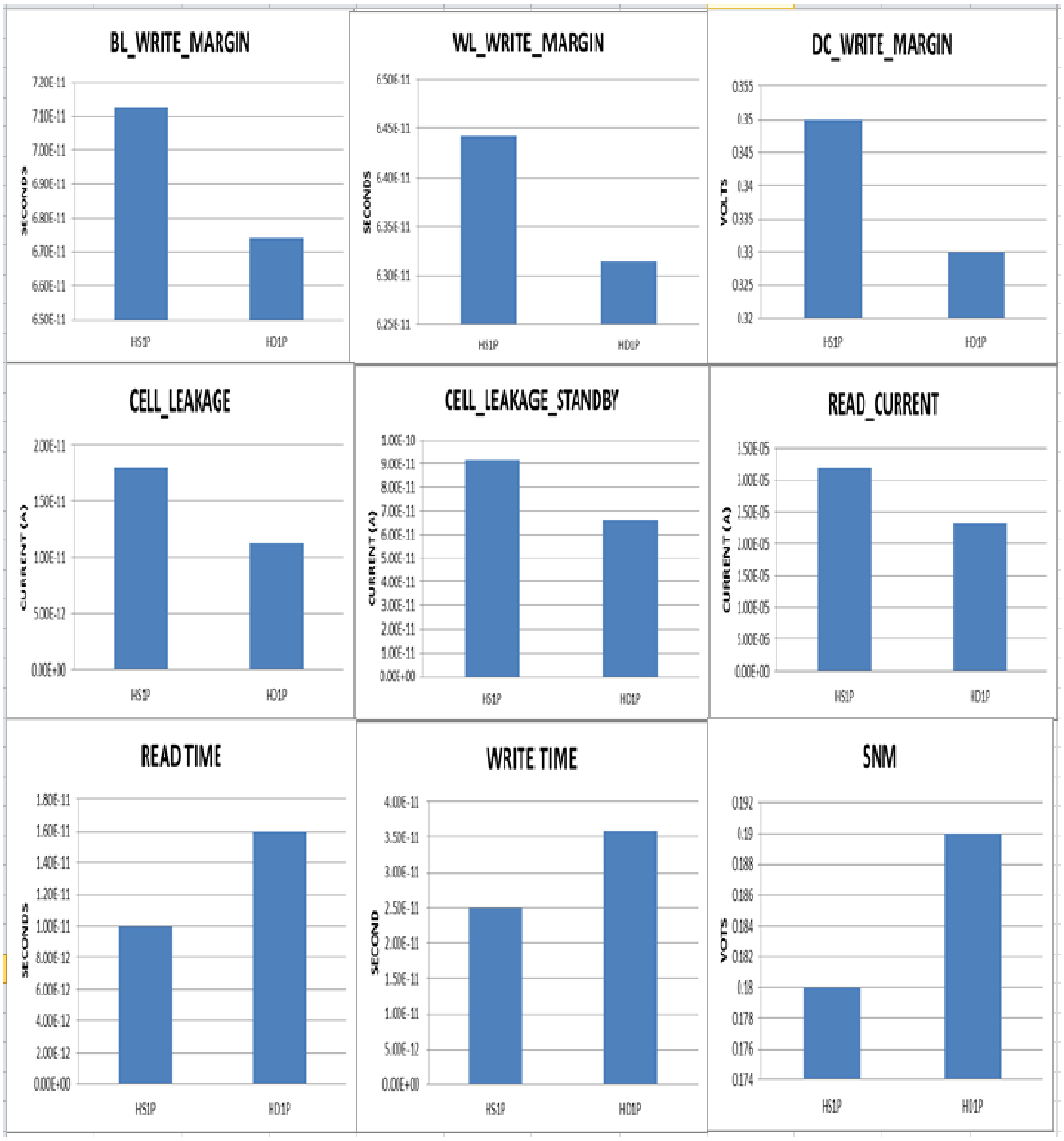


Figure 6.7: HS1P vs HD1P Results

6.4 Central Decoding Architecture

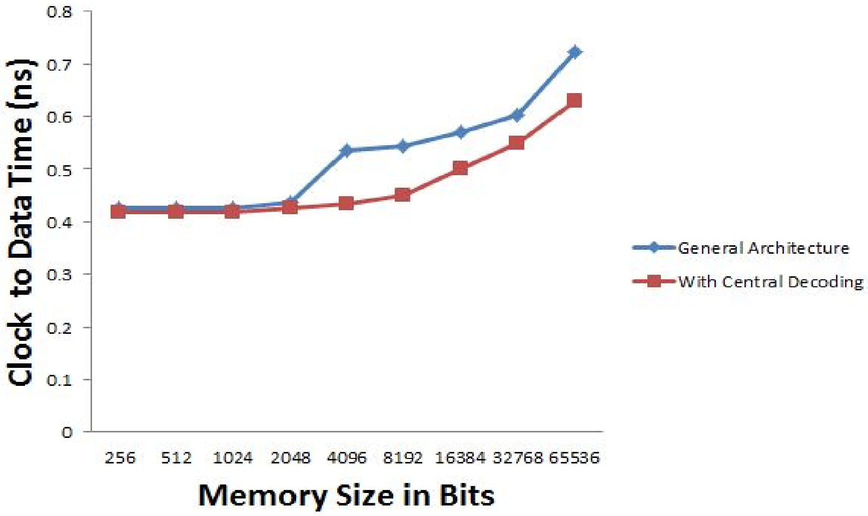


Figure 6.8: Central Decoding Timing Benifit

6.5 Bank Architecture

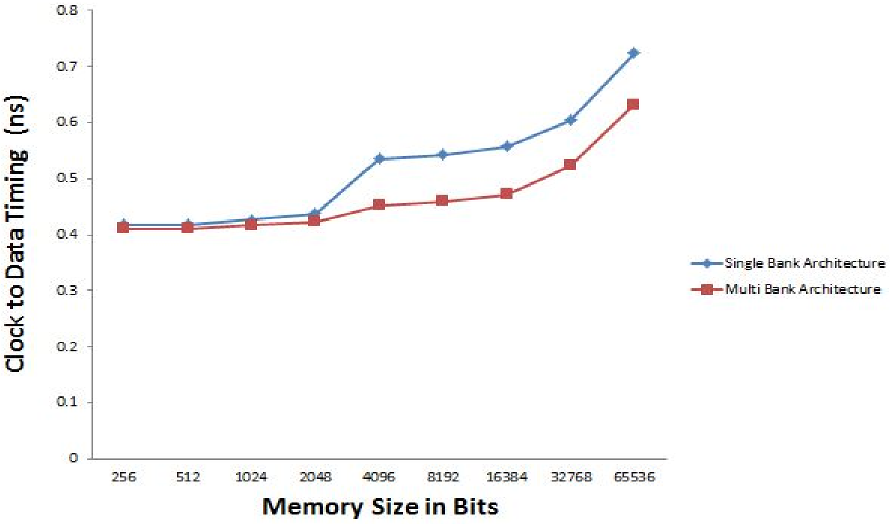


Figure 6.9: Bank Architecture Timing Benifit

6.6 Column Division Architecture

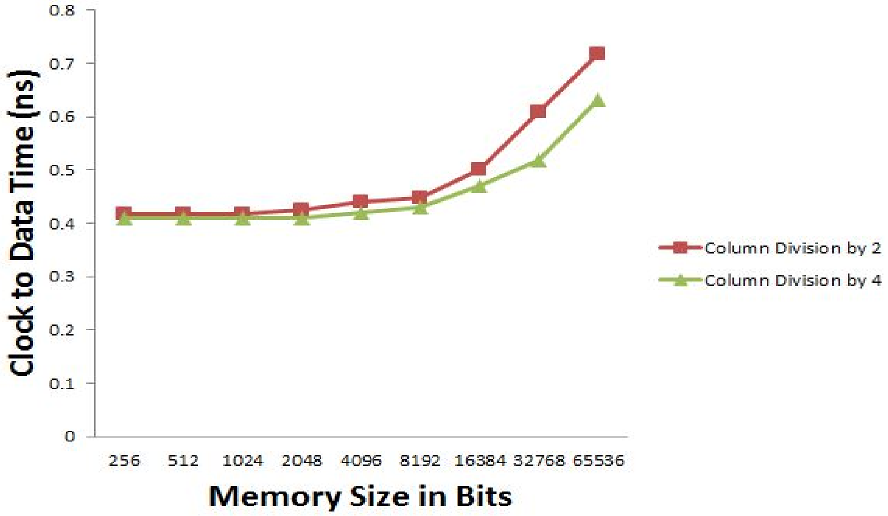


Figure 6.10: Column Division Timing Benifit

6.7 Overall Timing Improvement

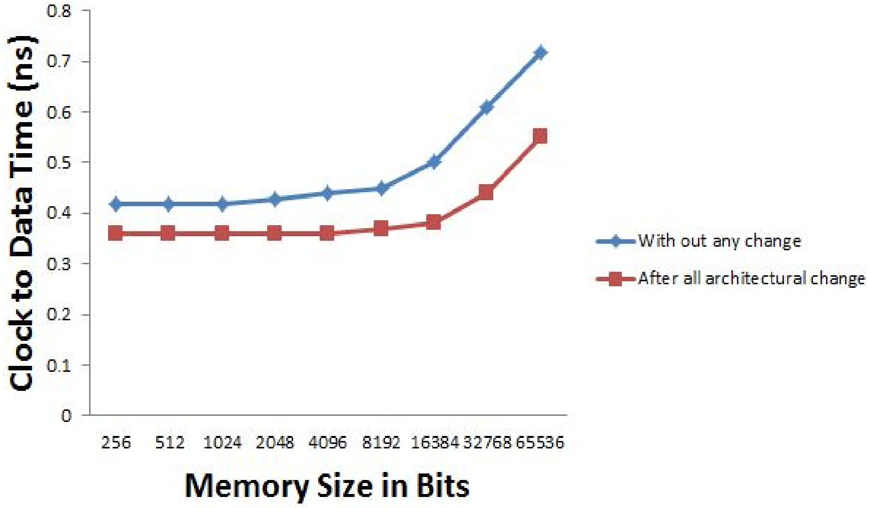


Figure 6.11: Current Results (HS1P)

6.8 Leakage Improvement by Power Gating

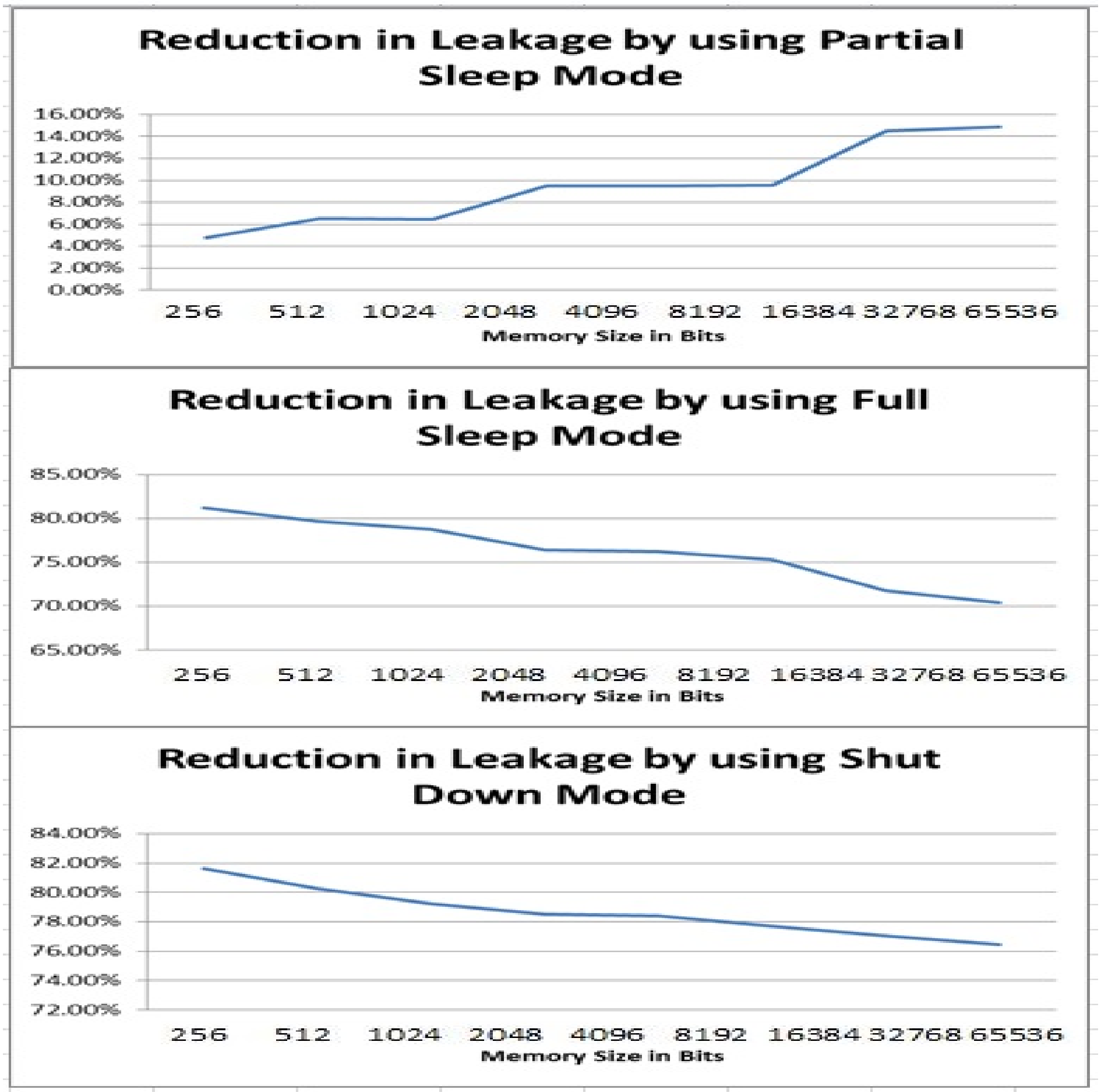


Figure 6.12: Leakage Improvement

6.9 Conclusion

- There are some trade off between area and performance. We get better performance at the cost of area.
- Write margin for HD1P is lesser than HS1P. So, it is harder to write bitcell in HD1P compared to HS1P. Therefore write time is increases in HD1P compared to HS1P. Read Current is more in HS1P compared to HD1P. Therefore read time is less for HS1P compared to HD1P. There is a penalty in SNM and leakage due to high speed in HS1P.
- By properly making various design change like cetral decoding, bank architecture and column division ,we can boost SRAM performance at the cost of area.
- Here we achieved noticeable leakage reduction by using various power gating techniques with the balance of IR drop and latency of wake up circuit.

6.10 Future Scope

We can achieve more leakage reduction by making further design change in power gating circuit. Characterization of HD1P and HS1P SRAM will provide guidelines for IoT and SoC designer to choose optimum SRAM with the proper balance of performance and area.

References

[1] S. Kang, Y. Leblebici, “CMOS Digital Integrated Circuits”, New York, McGraw-Hill, 2003. pp. 402-430

[2] David A. Hodges, “Semiconductor Memories”. pp. 359-376

[3] R. Baker, H. Li, and D. Boyce, “CMOS: Circuit Design, Layout, and Simulation”, New York, IEEE Press, 1999

[4] Lingbo Kou, Robinson, W.H., “Impact of Process Variations on Reliability and Performance of 32-nm 6T SRAM at Near Threshold Voltage”, VLSI (ISVLSI), 2014 IEEE Computer Society Annual Symposium , pp. 214 - 219, July 2014

[5] S. K. Singh, S. V. Singh, B. K. Kausik, C. Chauhan, T. Tripathi, “Characterization and improvement of SNM in deep submicron SNM design”, IEEE International conference on Signal Processing and Integrated Networks (SPIN), pp. 538-542, February 2014

[6] Y.Nakagome, K. Itoh, M. Isoda, K. Takeuchi, and M.Aoki, “Architecture and Design of a High Performance SRAM for low power Applications”, Symposium on VLSI Circuits, IEEE International Digest of Technical Papers, pp. 82 - 83, June 2002

[7] Chung-Hsien Hua, Tung-Shuan Cheng and Wei Hwang, "Distributed Data-Retention Power Gating Techniques for Column and Row Co-Controlled Embedded SRAM",2005 IEEE International Workshop on Memory Technology, Design, and Testing,PP 129-134, August 2005

[8] S. K. Singh, S. V. Singh, B. K. Kausik, C. Chauhan, T. Tripathi, "Improved Power Gating Technique for Leakage Power Reduction",International Journal Of Engineering And Science Vol.4, Issue 10 (October2014), PP 06-10, February 2014

[9] http://solvnet.synopsys.com