Implementation of TCAM using DRAM

A project thesis

submitted by

Shruthi Parvathi R

in partial fulfillment of the requirements
for the award of the degree of

Master of Technology &
Bachelor of Technology

Dept. of Electrical Engineering
IIT Madras
Chennai 600 036

Thesis Certificate

This is to certify that the thesis titled **Implementation of TCAM using DRAM**, submitted by **Shruthi Parvathi R**, to the Indian Institute of Technology, Madras, for the award of the Dual Degree of Master of Technology and Bachelor of Technology, is a bona fide record of the research work done by her under my supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Janakiraman Viraraghavan
Project Guide,
Assistant Professor,
Dept. of Electrical Engineering,
IIT Madras, 600 036

Place : Chennai
Date : June 2020

Acknowledgements

I am greatly indebted to Prof. Janakiraman Viraraghavan for guiding me through the entire course of my Dual Degree project. He always took the time and effort to discuss the problem and to suggest different methods to find the solutions. His valuable remarks always gave new directions to my project. I am also thankful to him, whose courses helped me improve my knowledge in Digital Circuits through the course of the five years of my Dual Degree program.

I am thankful to all the professors whose courses helped me improve my knowledge in Integrated Circuits and Systems through the course of the five years of my Dual Degree program. Their classes always inspired me to think beyond classrooms into more practical scenarios.

A special thanks to Dasari Shirisha, who was my fellow associate in doing this project. This project would not have been possible without their contributions and insightful observations. A special thanks to my family members and my friends for their continuous moral support and encouragement.

Abstract

This project proposes a new DRAM architecture for Ternary Content Addressable Memory (TCAM). The main motive is to perform a fast search operation without increasing the area of the memory array. DRAM array is lesser in area compared to a SRAM array. TCAM in SRAM is popular as the addition of extra logic transistors is not expensive. The available TCAM in DRAM architectures does not conserve much area due to the additional area expense of adding extra transistors to a thick oxide access transistor.

We have proposed a new DRAM architecture that is area effective and takes a lesser number of cycles for a search operation, and also does not require extra peripheral circuitry to do a match operation. It can also perform fundamental DRAM operations such as Read, Write, and Refresh. The challenges and analytic solutions have been discussed in detail.

Contents

List of Figures

List of Tables

Chapter 1

Introduction to TCAMs in SRAM

1.1 Content Addressable Memory

A Content Addressable Memory (CAM) compares its search input data with every word stored in the memory and returns the address location of the matching words. This parallel multi-data search makes a CAM an essential component for computer networking devices to provide faster lookup in routing tables, high-associative caches, and register renaming. CAMs are much faster than RAMs in data search applications. When a user supplies the memory address in a RAM, it returns the word stored in the address. In comparison, CAM accesses the words based on the content itself. A lookup of an entry in a CAM can be performed in a single clock cycle, whereas a RAM module requires multiple clock cycles to make a single memory fetch. CAMs are used in a wide variety of applications requiring high search speeds, such as Artificial Neural Networks, Database engines, network routers, etc.

CAM compares the input search data against the table of stored data and returns the address of matched data. [1]

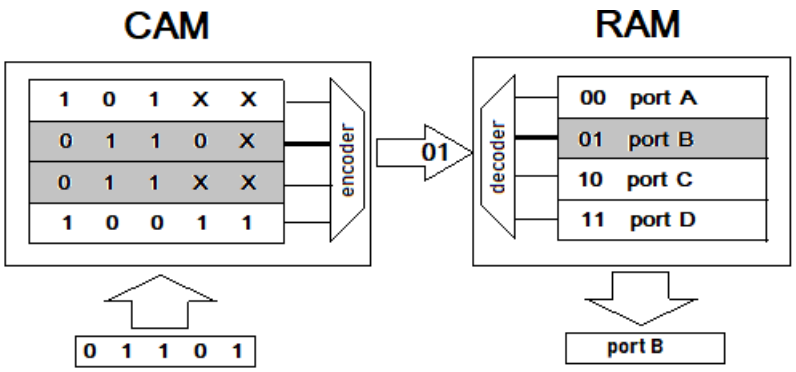


Figure 1.1: CAM-based implementation of the routing table

The parallel comparisons of CAM consume huge power and amount to

increase in Silicon area.

1.2 CAM Architecture

There are two types of CAM- Binary CAM (BCAM) and Ternary CAM (TCAM). A Binary CAM performs binary lookup and returns either 0 or 1. A single bit is enough to store the data. A TCAM stores 0,1, as well as a don't care state. The "Don't Care" is stored at an additional by adding an extra bit ("care" or "don't care" bit) to every memory cell, and hence 2 bits are required to store a single data. The input to the CAM system is a search word, which is fed on the search lines of the stored data.

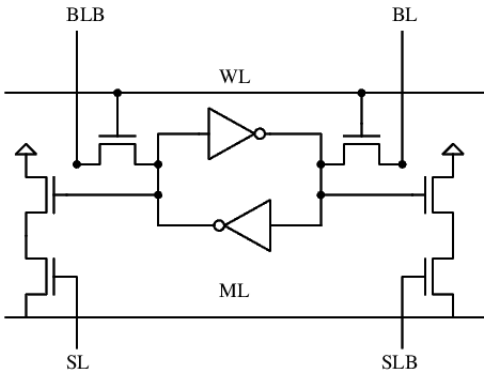


Figure 1.2: Binary CAM 10T bit cell

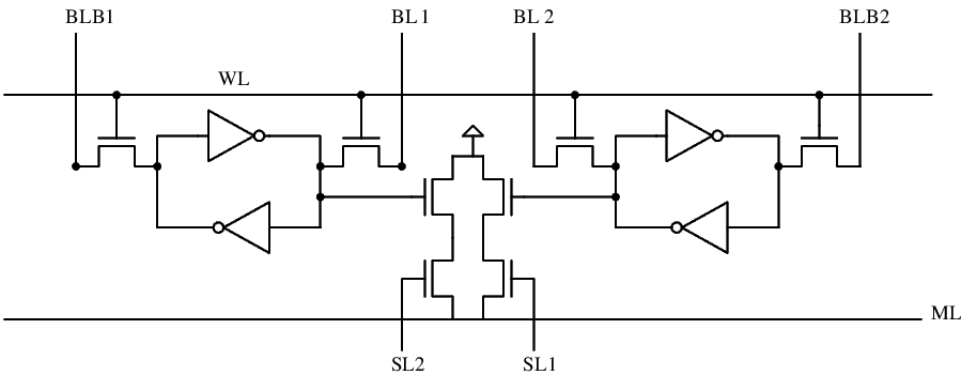


Figure 1.3: Ternary CAM 16T bit cell

1.2.1 NOR based SRAM TCAM

The NOR cell performs [1] the comparison between the stored bit, D, and $\overline{D}$ on search lines, SL and $\overline{SL}$, using four comparison transistors, M_1, M_2, M_3, and M_4. SL and D are the inputs to dynamic XNOR logic gates. Only during a mismatch, one of the two pull-down paths M_1/M_3 and M_2/M_4 from

the matchline activates, connecting the matchline (ML) to ground. During a match condition, SL and D disable both pull-down paths, disconnecting ML from the ground.

The cell is said to be in a Don't care when both D and $\overline{D}$ equal to logic "1". This disables both pull-down paths, and irrespective of the inputs in the search line, the Matchline doesn't discharge.

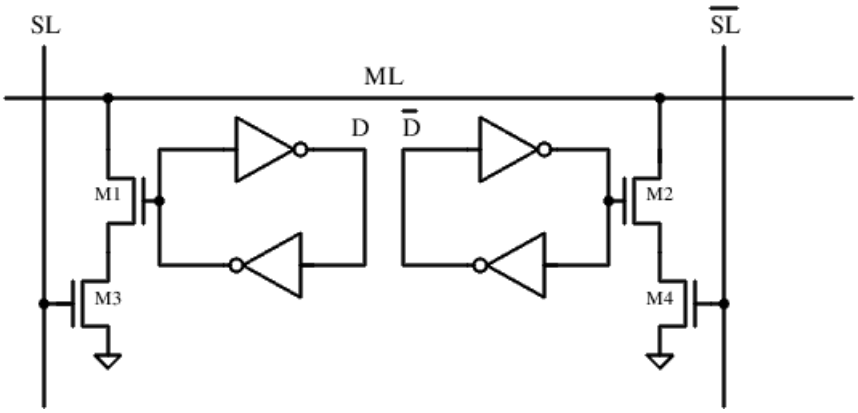


Figure 1.4: NOR based SRAM TCAM

1.2.2 NAND based SRAM TCAM

The NAND cell performs [1] the comparison between the stored bit, D, and $\overline{D}$ and the corresponding search data on the search lines, SL, and $\overline{SL}$, using three comparison transistors M_1, M_D, and $M_{\overline{D}}$. In the case of a match when both $SL = 1$ and $D = 1$, Pass transistor M_D is turned ON and passes the logic '1' on the SL to node B. Node B is the bit-match node which is logic '1' if there is a match in the cell. M_1 is also turned ON in the other match case when $SL = 0$ and $D = 0$. In this case, the transistor $M_{\overline{D}}$ passes a logic HIGH to raise node B. The remaining cases, where $SL \neq D$, result in a mismatch condition, and accordingly, node B is logic '0', and therefore the transistor M_1 is OFF.

To store a Don't care state, the mask bit is set to '1'. This forces the transistor M_{mask} to turn ON, regardless of the value of D, ensuring that the cell always matches.

To implement a mask condition, both the search lines SL, and $\overline{SL}$ are set to logic '1', enabling at least one of the two transistors M_D or $M_{\overline{D}}$ to pass the logic '1' to node B.

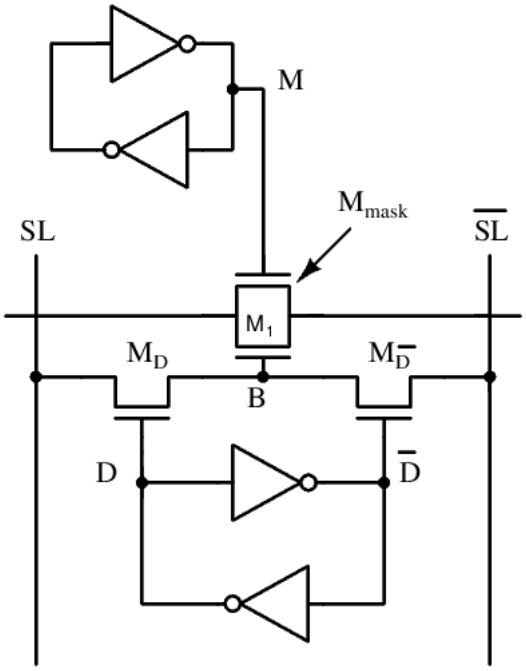


Figure 1.5: NAND based SRAM TCAM

An advantage of the NOR based SRAM TCAM cell is that it provides a full rail voltage V_{DD} at the gates of all comparison transistors. On the other hand, a disadvantage of the NAND based SRAM TCAM cell is that it provides only a reduced logic "1" voltage at node B, which may reach only $V_{DD} - V_{tn}$ when the search lines are driven to V_{DD} (where V_{DD} is the supply voltage and V_{tn} is the nMOS threshold voltage).

Chapter 2

Introduction to TCAMs in DRAM

2.1 Dynamic TCAM implementations

Dynamic CAM has an inherent advantage over static CAM since it can store three states of CAM in a small area.

2.1.1 4T Dynamic CAM cell

The four transistors (4T) Dynamic CAM cell [2] shown in Figure 2.1 consists of two transistors T_{C0} and T_{C1}, to perform XOR operation of the data presented at Bit and NBIT and the data stored in the cell. The gates of these transistors serve as dynamic storage elements and are labelled as S_{b1} and S_{b0}.

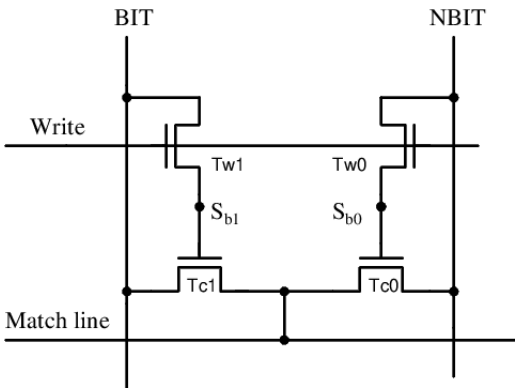


Figure 2.1: 4T DTCAM

S_{b1}	S_{b0}	State
0	0	Don't care
0	1	0
1	0	1
1	1	Not allowed

Figure 2.2: DRAM cell stored states

The output of the XOR operation is connected to the match line, so in the case of a match, the Matchline is not discharged. The Table 2.1 gives information on results on various cases of a search operation.

Stored data	Sb1	Sb0	bit	NBit	Match condition
0	0	1	0	1	Match
0	0	1	1	0	Mismatch
1	1	0	1	0	Match
1	1	0	0	1	Mismatch
X	0	0	0	0	Match

Table 2.1: Table for match operation in 4T DCAM

2.1.2 6T Dynamic CAM cell

Dynamic TCAM [3] has 6T structure storing a trit i.e., logic 0 for 01 (data=0, mask=0), logic 1 for 10 (data=1, mask=0) and don't care for 00 (data=x,mask=1)

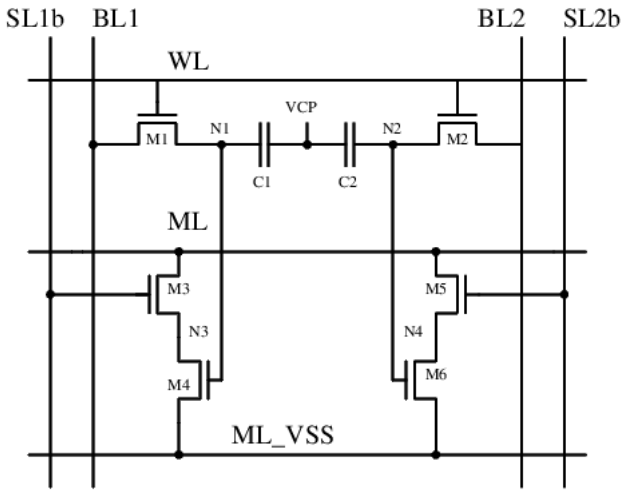


Figure 2.3: 6T Dynamic TCAM

Search operation in the above architecture is performed by pre-charging the Matchline to a value below Vdd, and the search lines are held at ground. Then the pre-charge of Matchline is released, and the inverted input bits are placed on the search line. If the search data matches the cell data or is masked, the matchline would not discharge. The matchline is discharged in a case of mismatch.
The transistors M3, M4, and M5, M6 implement the NAND operation of search input and the contents of the cell. For example, if '1' is stored in a cell (10), placing '0' on the search line (SL1b=0, SL2b=1), the NAND stack is turned OFF, and the Matchline would not discharge. If '1' is placed on search lines (SL1b=1, SL2b=1), there is a discharge path from ML to ML_VSS from M3, M4, thereby implying a mismatch.

Chapter 3

Area Analysis of TCAM in eDRAM

3.1 Problems with implementing TCAM in DRAM over SRAM

In SRAM based implementation, the provision for comparison is through XOR logic by adding four extra transistors, M1, M3, M2, and M4, per SRAM cell. The gates of these extra transistors are connected to the contents of the cell as in M1 and M2, and the input search data as in M3 and M4, like in Fig 3.1. The comparison is performed without turning on the Wordline.

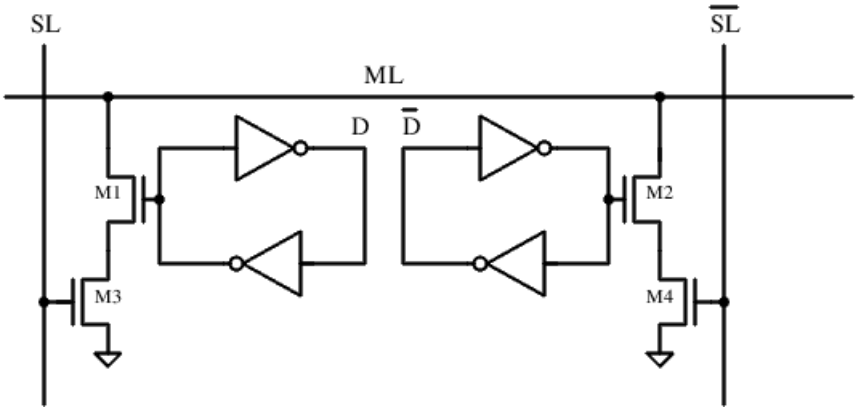


Figure 3.1: NOR based SRAM TCAM

In SRAMs, all transistors are logic transistors. So adding these extra transistors to the NAND stack will not require any additional isolation area. Hence the area overhead will be minimum.
DRAMs store their contents on a capacitor rather than in a feedback loop like SRAM. Deep Trench Capacitors (DTC) form the primary storage element in DRAM and have the advantage of offering higher capacitance per unit area. But for the cell architecture in Fig 3.2, if the gate of the MOSFET M4 and M6 have to be connected to the capacitor, contact has to be made for metal

connection, affecting the density of eDRAM.

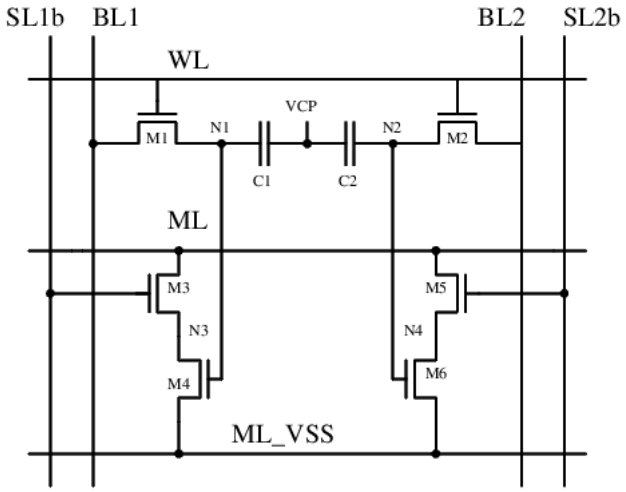


Figure 3.2: 6T Dynamic TCAM

3.2 Area analysis of 6T DRAM TCAM

The access transistor in DRAM is a thick oxide device to reduce the sub-threshold leakage and higher voltage swing. The transistor in the NOR stack can be a thick oxide or a thin oxide device. The layout was done in Cadence in 28nm Technology. The dimension of the data cell (2 DRAM units) without the stack is 2.6μm X 1μm. The area of this single unit is 2.6μm^2. The area analysis of adding the extra transistors in the NOR stack is done below.

3.2.1 Adding thick oxide devices in the NAND stack

In Fig 3.3, the NAND stack is implemented using thick oxide devices. In a typical DRAM array, the deep trench capacitor is in the axis perpendicular to the DRAM array. Hence it's not shown in the figure. In TCAM DRAM, an extra metal contact should be made between the deep trench capacitor and the access transistor to connect node N1 to M4.

The source and drain of M3, M4, and M5, M6 are shared. They are diffused together to make the area compact. The length and width of each thick oxide transistor are 150nm and 320nm, respectively. From the figure, it can be noted that area of the NAND stack is bigger than a single DRAM cell. The dimension of the single TCAM DRAM structure is 3μm X 1.75μm. The area is 5.25μm^2, which is 2X times the DRAM data cell. This is not an

optimal approach for the TCAM structure.

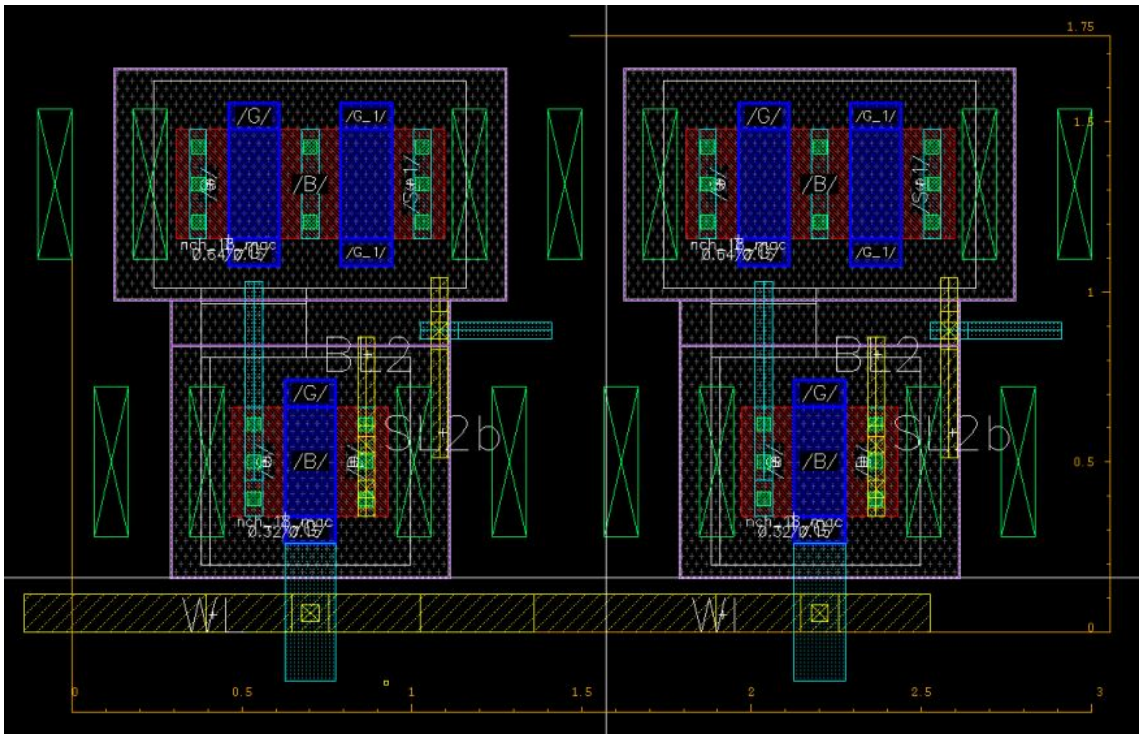


Figure 3.3: 6T Dynamic TCAM with thick oxide transistors in NAND stack

3.2.2 Adding thin oxide devices in XOR stack

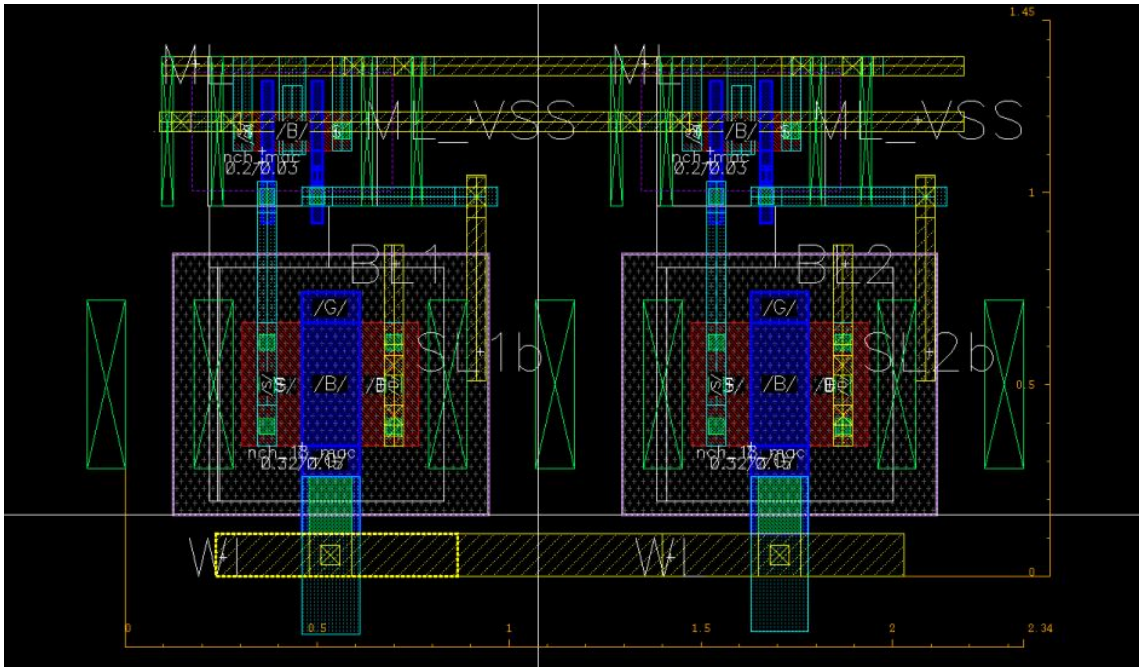


Figure 3.4: 6T Dynamic TCAM with thin oxide devices in NAND stack

In Fig 3.4, two thin oxide transistors have been added to the NAND stack. The width and the length of the transistors in the stack are 100nm and 30nm, respectively. The dimension of the DRAM TCAM data cell, as seen in Fig 3.3 is 2.34μm x 1.45μm. The area is 3.93μm^2, which is 1.35X times that of the DRAM data cell.

Though the area, in this case, is less, the yield of the chip while manufacturing is very low. This is due to the huge difference in the poly Silicon pitches of thin and thick oxide devices, due to which we need to have many dummy cells in the array, which deeply impact the yield during Lithography. So we can't implement the NAND stack using thin oxide devices.

The area overhead in TCAM SRAM was only 25%. But the area overhead in TCAM DRAM is 100%. So, there is a need to develop a technique to implement TCAM in DRAM without changing the unit structure of the DRAM data cell.

Chapter 4

Parallelism of Search Operation

4.1 Implementing TCAM in DRAM

The goal was to implement TCAM in DRAM without changing the eDRAM array. For TCAMs, two bits are required to store a single data. Hence the only change that was made in such a TCAM based eDRAM array from the conventional eDRAM array is that a single data would be represented with two bits.

4.1.1 Pass transistor Logic in DRAM

In the case of a SRAM based TCAM, the search operation was implemented in a stack based XNOR CMOS logic. Such a stack is not desired in DRAM. Hence the XNOR logic was implemented using pass transistor logic.

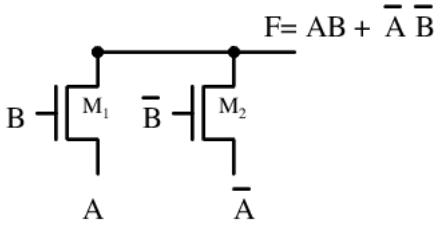


Figure 4.1: XNOR logic using pass transistor logic

A	B	F
0	0	1
0	1	0
1	0	0
1	1	1

Figure 4.2: Truth table of XNOR gate

The eDRAM array consists only of nmos access transistors, and the bit is stored in a capacitor. Each cell is represented by 2 bits. The cell can store 01 or 10 or 00 (Don't care). State 11 is not allowed in the eDRAM array. Table 3.1 gives information on cell state representation for TCAM. The conventional TCAM has matchlines that run across the Wordlines. In the proposed eDRAM based TCAM, the Bitlines act as the matchlines.

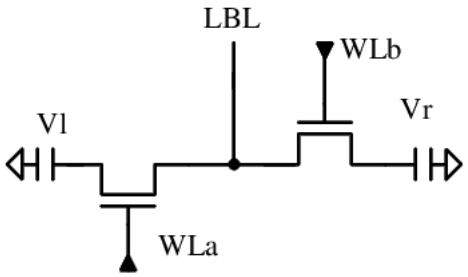
Cell Value	Logic
01	0
10	1
00	Don't care
11	Forbidden

Table 4.1: Cell states for TCAM

This results in area reduction and discards the need for extra external circuitry to detect a match. The word is stored along the column.

4.2 Match operation in DRAM

The match operation is done by placing the complimentary bits or mask bits on the Wordlines and detecting any change in the Bitline through the already available read circuitry.



The Bitlines (BLs) are pre-discharged to the ground. In the case of a match and mask, the Bitline is left in the same pre-discharged state (gnd), and in case of a mismatch, the Bitline charge shares with the cell, and it is pulled to a weak '1'. During a search operation, it was assumed that only the Wordlines of one cell (2 bits) are activated, and the remaining Wordlines in the Bitline are turned off.

Vl	Vr	WLa	Wlb	Match Condition
0	1	1	0	Match
1	0	0	1	Match
0	1	0	1	Mismatch
1	0	1	0	Mismatch
0	0	X	X	Match
X	X	0	0	Mask

Table 4.2: XNOR operation in DRAM

4.2.1 Detecting a match

In the case of a match, the search bits are placed on the Wordlines. The search bits are complimentary to the bits stored in a cell. When the bit is storing a '0', the Wordline is turned on, and when the bit is storing a '1', the Wordline is turned off. This leaves the Bitline in the original pre-discharged state. The read circuitry detects the voltage on the Bitline and resolves it as a match.

4.2.2 Detecting a mismatch

In the case of a mismatch, the Wordlines and the bits in the cell have the same value. When the bit is storing a '0', the Wordline is off, and when the bit is storing a '1', the Wordline is on. Charge sharing between the Bitline and the cell capacitor takes place only when the bit is storing a '1' and the Wordline is turned on. The two bits in a cell are placed on the same Bitline. So, the charge sharing will definitely take place in case of a mismatch as the Wordline of the bit storing a '1' is turned on. The read circuitry detects this increase in voltage of the Bitline, and resolves it as a mismatch. The state 11 in a cell is forbidden.

4.2.3 Don't care and mask condition

The cell can be in the don't care state when both the bits are storing a 0. In this case, turning on the Wordline of either bit will not charge the Bitline.
The proposed eDRAM TCAM allows masking any number of bits in the word. In the case of a mask, both Wordlines are turned off. As a result of this, the Bitline won't charge irrespective of any value that is stored in the cell.
In the above two conditions, as the Bitline remains in its pre-discharged state, the read circuitry resolves it as a match.

4.3 Parallelism during a match operation

The main advantage of a TCAM is that it can do a search operation in a very less period of time. In the proposed eDRAM TCAM, the word is stored along the column. To increase parallelism, multiple Wordlines should be turned on at once.

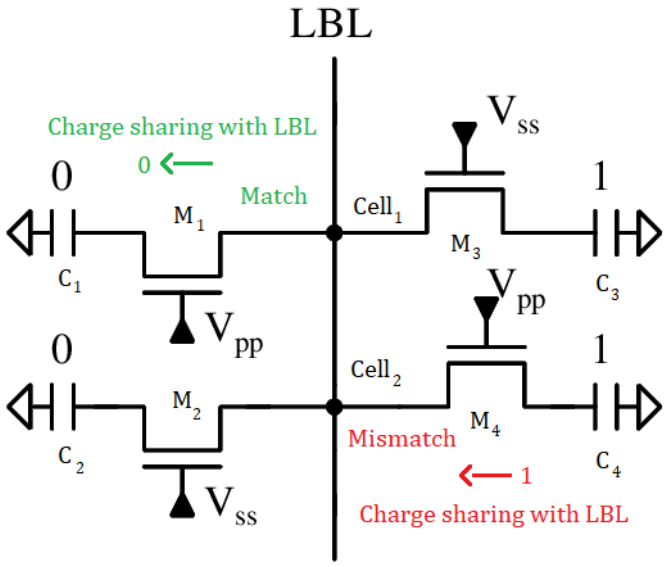


Figure 4.3: Conflict while turning on multiple WLs

Turning on multiple Wordlines at once might result in a charge conflict. In fig 3.3, 4 wordlines are activated together, which evaluates two cells $Cell_1$ and $Cell_2$, at once. Each cell stores 2 bits. $Cell_1$ is a match, whereas $Cell_2$ is a mismatch. Access transistors M1 and M4 are turned ON, and M2 and M3 are turned OFF. The bit corresponding to access transistor M1 is storing a '0', and the bit corresponding to M4 is storing a '1'. As both MOSFETs are turned on, charge sharing takes place between the cell capacitors and the Bitline. The resulting voltage on the Bitline will be as follows:

$$V_{BL\_final} * [2 * C_{cell} + C_{BL}] = (V_{C_1} C_{cell}) + (V_{BL\_in} C_{BL}) + (V_{C_4} C_{cell}) \quad (4.1)$$

where,
V_{BL_final} = Final Bitline voltage after charge sharing
C_{cell}, C_{BL} = Cell and Bitline capacitance
V_{C_1}, V_{C_4} = Voltage on C_1 and C_4
V_{BL_in} = Pre-discharged Bitline voltage
V_{C_1} and V_{BL_in} is close to ground and can be ignored. So,

$$V_{BL\_final} = [V_{C_4}] * \frac{C_{cell}}{2 * C_{cell} + C_{BL}} \quad (4.2)$$

The Bitline voltage when 'm' cells are evaluated for a match (2*m Wordlines are activated) while having 'n' mismatch cases and 'm-n' match cases is:

$$V_{BL\_final} = [V_{cell_1}] * \frac{n * C_{cell}}{m * C_{cell} + C_{BL}} \quad (4.3)$$

where V_{cell_1} is the voltage on the capacitor storing bit '1' in the mismatched data cells.
Note: In a match and mismatch case of a cell, one Wordline is turned on, and the other Wordline is turned off. In a mask condition, both Wordlines are turned off, and so no cell capacitor participates in charge sharing. Here, 'm-n' match cases do not include any mask condition.

The Bitline voltage when 'm' cells are evaluated while having 'n' mismatch cases, 'p' mask conditions, and 'm-n-p' match cases is:

$$V_{BL\_final} = [n * V_{cell_1}] * \frac{n * C_{cell}}{(m-p) * C_{cell} + C_{BL}} \tag{4.4}$$

Example: In the case where 32 data cells are activated, the worst case scenario is when there is only a single mismatch in 32 data cells (1 mismatch cell and 31 match cells),

$$V_{BL\_final} = [0.8V] * \frac{C_{cell}}{32 * C_{cell} + C_{BL}} \tag{4.5}$$
$$C_{cell} = 10fF, C_{BL} = 6.4fF$$
$$V_{BL\_final} \approx 0.025V$$

This voltage cannot be detected by the Sense Amplifier as a '1'. This is also the final voltage on the bit that was initially storing a '1'. The bit flips.
The Bitline can sense a mismatch only when,

$$\begin{aligned} V_{BL\_final} &\geq 350mV \\ 0.35 &\leq [0.8] * \frac{n}{32} \\ n &\geq 14 \end{aligned}$$

The Bitline can sense a mismatch only when,

$$\begin{aligned} V_{BL\_final} &\geq 350mV \\ 0.35 &\leq [0.8] * \frac{n}{32} \\ n &\geq 14 \end{aligned}$$

From the above example, it is noted that turning on multiple Wordlines in a Bitline at once will result in a different Bitline voltage depending on how many match, mismatch, and mask conditions are present. Hence, only one cell per Bitline is evaluated, which corresponds to turning on atmost one Wordline per Bitline. To obtain parallelism, multiple local Bitlines should be present so that many cells can be evaluated at once.

4.4 Transfer Ratio

During a read of a '1' or a mismatch, the bitline and the cell charge share and reach a voltage which is determined by the transfer ratio.

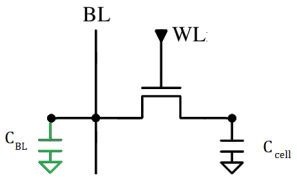


Figure 4.4: Cell and Bitline Capacitance

The initial charge on C_{cell} before a read is equal to the final charge on any one of the two capacitors C_{cell} or C_{BL}, because they are in parallel. The ratio of total capacitance before and after charge sharing is called the Transfer ratio.

$$TR = \frac{C_{cell}}{C_{cell} + C_{BL}} \tag{4.6}$$

The final Bitline voltage is,

$$V_{BL} \; or \; V_{cell_f} = V_{cell_i} * TR \; ;$$

where,
V_{cell_i} = Initial voltage on the cell storing a '1'
V_{cell_f} = Final voltage on the cell storing a '1'

$$V_{BL} \; > \; V_{trip}$$

As the number of cells per BL increases, C_{BL} increases which reduce the transfer ratio. This is plotted below.

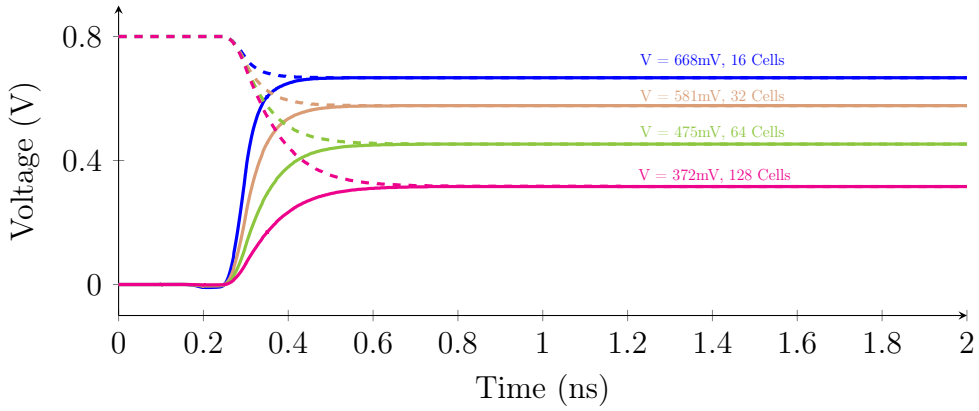


Figure 4.5: Cell & Bitline Voltage after charge sharing

Chapter 5

Sense Amplifier

5.1 Choice of Sense Amplifier

The proposed Architecture should have many Local Bitlines to increase parallelism. There are two choices for the Sense Amplifier, Gated Feedback Sense Amplifier, and 3T Micro Sense Amplifier. They are discussed below.

5.2 Gated Feedback Sense Amplifier

Gated Feedback Sense Amplifier (GFSA) [4] is the currently used Sense Amplifier in IBM's P9 Processor. Each GFSA controls two sets of Local Bitlines. Each Local Bitline has 64 Wordlines. GFSA can finish a read operation fast as it reads a '1' by default.

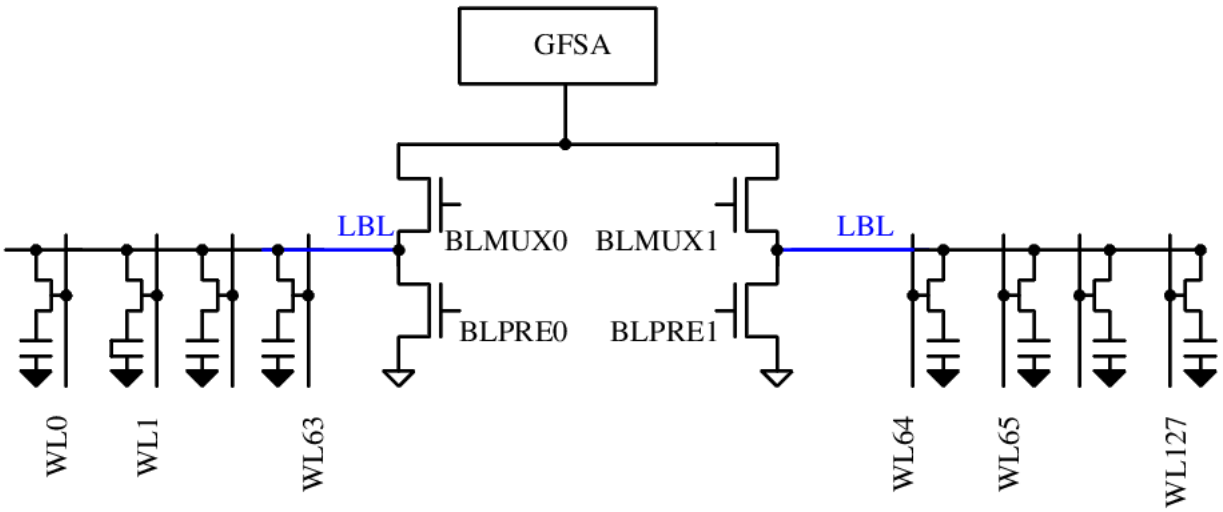


Figure 5.1: Gated Feedback Sense Amplifier

The write-back of '0' and '1' is controlled through timed signals. This Sense Amplifier is not an optimal choice for the Architecture because of its size and

fewer Local Bitlines per Sense Amplifier.

5.3 3T Micro Sense Amplifier

As the name suggests, the 3T Micro Sense Amplifier [5] has only 3 transistors. Each μSA controls one Local Bitline. Each Local Bitline has 32 Wordlines.

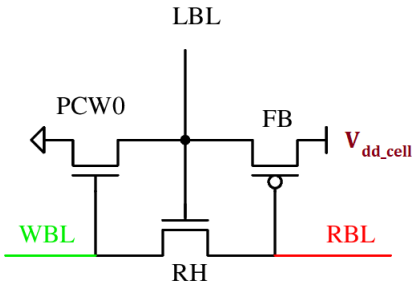


Figure 5.2: 3T Micro Sense Amplifier

The μSA is also connected to two Global Bitlines, WBL, and RBL. The Hierarchy of 3T μSA is given below.

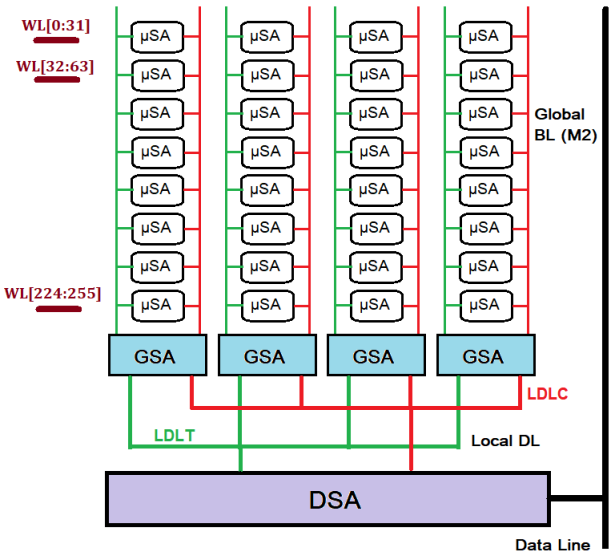


Figure 5.3: 3T Micro Sense Amplifier Hierarchy

The Global Bitline RBL floats at V_{dd} before a read operation. It remains floating at V_{dd} if the bit read is '0'. When the bit read is '1', RBL is discharged to ground. The write-back of a '0' is done through a timed signal which controls WBL. The write-back of '1' happens through a feedback mechanism. During a search operation, RBL discharges only during a mismatch. The Global Bitlines are connected to a Global Sense Amplifier (GSA). The number of

3T μSA per Global bitline is eight. There are four GSAs connected to a Data Sense Amplifier (DSA). The output of the DSA is the Dataline, which is also the matchline in the proposed Architecture.
This is the better choice for the Architecture because of its compact size and the number of Local Bitlines per Global Sense Amplifier.

5.4 Proposed DRAM Architecture

The number of Wordlines per local bitline should be chosen so that the time taken to finish the entire search operation should be minimum.

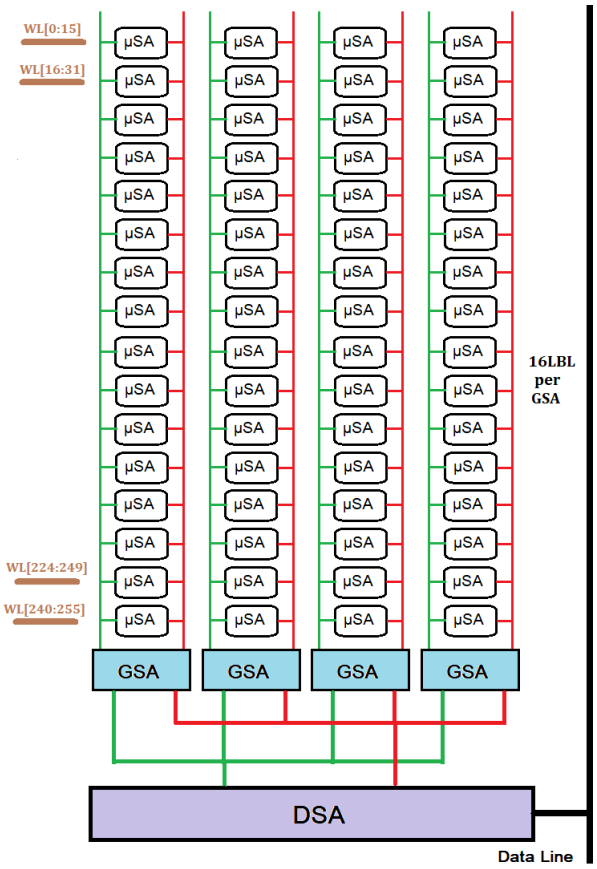


Figure 5.4: Proposed 3T Micro Sense Amplifier Hierarchy

During one cycle of a search operation, one data cell from each LBL is activated. To improve parallelism, 16 LBL per GSA is proposed. Data cells in a LBL are turned on serially. To reduce the number of cycles taken to search the entire array, 16 Wordlines or 8 data cells per LBL are proposed . 2 wordlines are required per data cell. So, one DSA can store a word length of 128 bits. To store 1024 bits, we should have 8 DSAs in the vertical direction.

Chapter 6

Analysis on Refresh Time

6.1 Write-back during a Search Operation

To obtain parallelism, multiple Local Bitlines should be present. These local bitlines are connected to a global Bitline through mux transistors which will be a part of the sense amplifier. The search operation is implemented by loading the search bits in the wordlines and performing a regular read. DRAM's read operation inherently has a write-back property because, during a read of a '1', the bit capacitor charge shares with the bitline, and the voltage reduces. Hence a write-back of '1' is usually performed to restore this voltage. During a search operation, the bitline charge shares with the cell only during a mismatch case. As multiple local bitlines are connected to a common global bitlines, if a match and a mismatch case occur in two local bitlines connected to a common global bitline, write-back should be disabled to prevent the wrong value from getting written back.

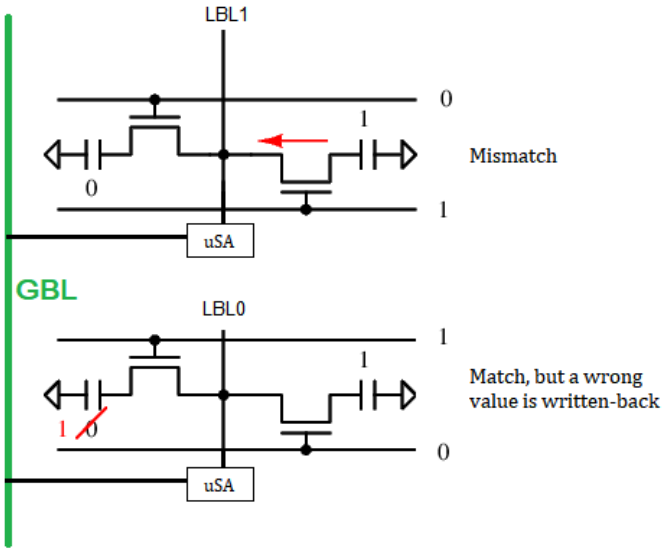


Figure 6.1: Wrong write-back during a search operation

In a normal read or a write operation, only one wordline is turned on. Hence only one local bitline is connected to the global bitline. But in the case of a search operation, some modifications should be done to the sense amplifier such that a write-back is disabled. The Sense Amplifier should still be able to perform all the functions of a regular DRAM.

6.2 Refresh & Retention

It is possible to have a common enable signal which only turns OFF the power supply used to charge the Bitline. This power supply is named V_{dd_cell}. Disabling V_{dd_cell} prevents the write-back of '1'. Due to this reason, it is necessary to perform a refresh before and after a search operation to restore the voltages to strong ones.

A refresh operation is nothing but a read which inherently has a write-back property, thus restoring the bit's voltages. In case of a refresh, all cells are read one after the other row-wise. Turning on one wordline refreshes all the bits that are connected to it, and hence the entire array is refreshed by turning on all the wordlines serially. It is possible to restore the voltage to a strong '1' when the bit was initially storing a weak '1' by keeping the wordline ON for a longer period of time. But to complete the refresh of the entire array in an optimal period of time without having any dependency on the initial voltage stored in the cell, the wordlines are kept turned ON for a constant time.

Retention time (T_{ret}) is the time taken for a bit corruption to occur due to leakage. The retention time of the DRAM cell should be greater than the time taken to complete a search and a refresh operation.

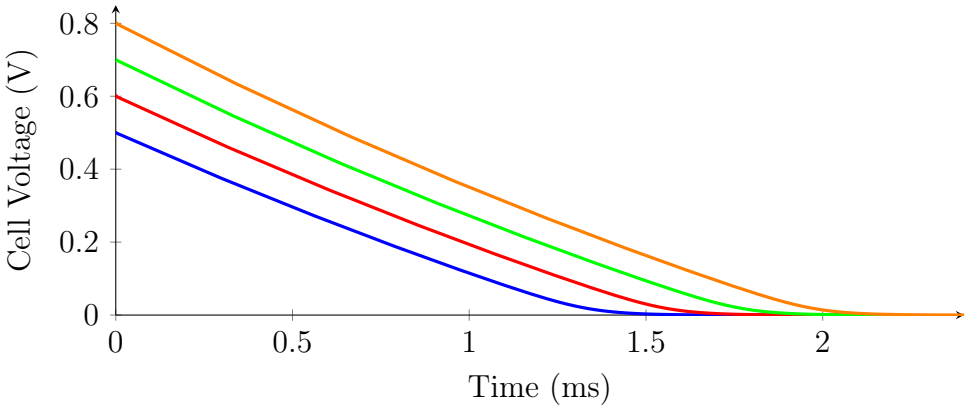


Figure 6.2: Cell Voltage due to leaking

$$T_{leak} = C * \frac{\Delta V_{leak}}{I_{leak}} \tag{6.1}$$

$$I_{leak} = 100pA,\ V_i = 720mV$$

$$TR = \frac{10fF}{10fF + 0.2fF * 16} = 0.75$$

$$V_{search} = V_i * TR = 540mV$$

$$V_{leak\_min} = \frac{V_{Tn}}{TR} = \frac{300}{0.75} = 400mV$$

$$T_{ret} = 10fF * \frac{[540 - 400]mV}{100pA}$$

$$T_{ret} = 14\mu s$$

If a search operation is performed on a cell that's storing an initial voltage V_i, it'll go to V_{BL_i} after charge sharing. This voltage further leaks to V_{leak}, which settles to V_{BL_leak} after charge sharing. This V_{BL_leak} should still be recognised as a '1' and get restored back to a strong '1', that is, a voltage greater than 720mV.

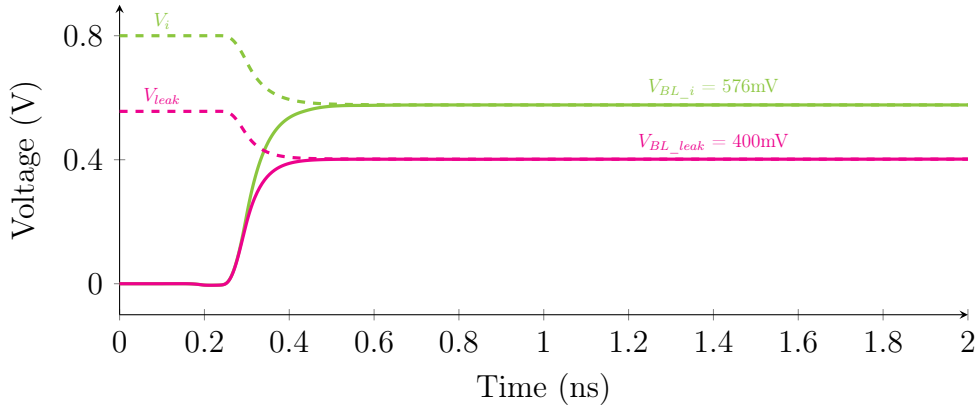


Figure 6.3: Cell voltage after charge sharing, before and after leakage

As V_{leak} decreases the time taken to restore it back to 720mV increases. The refresh time of the entire array should be less than $2\mu s$. In $2\mu s$ the cell Leaks by $\Delta V = 0.02V$. In the following simulations, a 3T Sense Amplifier was used for the simulations. The wordline voltages used are 1.6V (V_{pp}) and -0.3V (V_{ss}) in the ON and OFF conditions, respectively. The peripherals are operating in the voltage domain 0-0.8V. The voltage written to the cell is

dependent on V_{dd_cell}. It charges the bit voltage to a voltage either between 0-0.8V (section 6.2.1) or 0-0.9V (section 6.2.2).

6.2.1 Refresh Analysis in 0-0.8V domain

In the following analysis, the peripherals and the bit voltages are operating in the 0-0.8 Voltage domain. V_i is the initial Voltage before a Search operation, which goes to V_{search} after charge sharing during a search. This is assumed to leak for $2\mu s$ by $\Delta V = 0.02V$ to V_{leak}. The time taken to restore V_{leak} back to a strong '1' is given in the table below.

V_i (mV)	V_{search} (mV)	V_{leak} (mV)	T_{res} to 720mV (ns)
760	546.97	526.97	3.27
720	518.36	498.36	4.02
680	489.72	469.72	5.13
640	461.07	441.07	6.76
600	432.02	412.02	9.24

Table 6.1: T_{res} for different V_i when V_{dd_cell} is 0.8

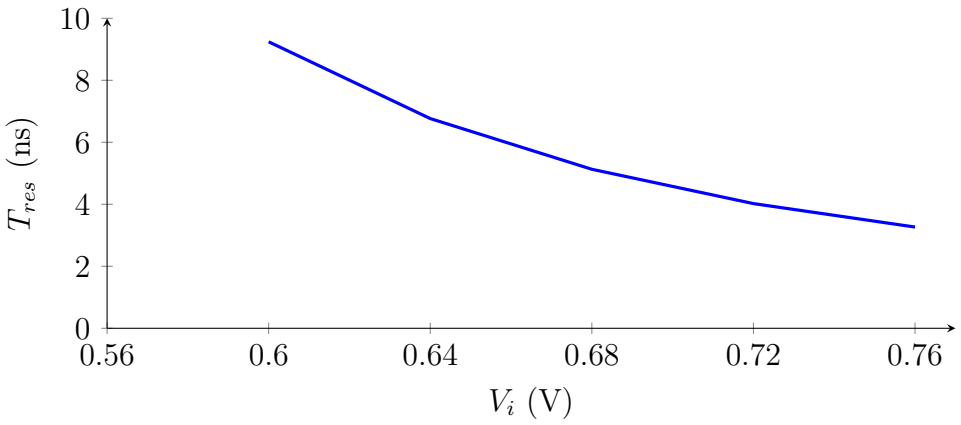


Figure 6.4: T_{res} vs V_i when V_{dd_cell} is 0.8

From the above plot and table it can be seen that as V_i decreases T_{res} increases. Having a high T_{res} takes a longer time to refresh the entire array. So, T_{on} of WL during refresh is chosen to be 6.8ns. V_i should not go below 80% of V_{dd} before a search operation. The refresh period of refreshing one WL takes around 7.3ns. This can be further reduced by increasing V_{dd_cell}.

6.2.2 Refresh Analysis in 0-0.9V domain

Having a separate power supply V_{dd_cell} for charging the bitline alone will allow it to charge faster. Other peripherals are still operating at a lower voltage domain. This doesn't allow the pull-up path through FB in a 3T Micro Sense Amplifier to be turned OFF completely.

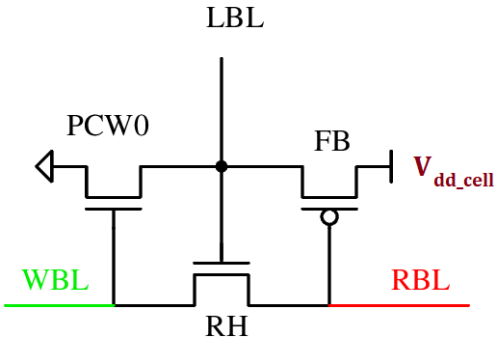


Figure 6.5: 3T Micro Sense Amplifier

So to turn OFF the pull-up path, a 4T SA is used. The signal Search Enable swings from 0-0.9V.

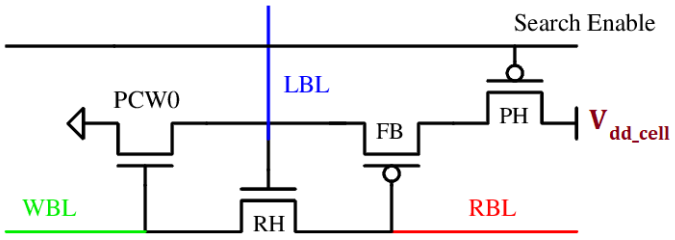


Figure 6.6: 4T Micro Sense Amplifier

In this subsection, the voltage source V_{dd_cell} is assumed to be at 0.9V.

V_i (mV)	V_{search} (mV)	V_{leak} (mV)	T_{res} to 720mV (ns)
760	546.97	526.97	2.18
720	518.36	498.36	2.61
680	489.72	469.72	3.24
640	461.07	441.07	4.15
600	432.02	412.02	5.51

Table 6.2: T_{res} for different V_i when V_{dd_cell} is 0.9V

In this case, T_{on} of WL during refresh can be as low as 4.15ns. The refresh

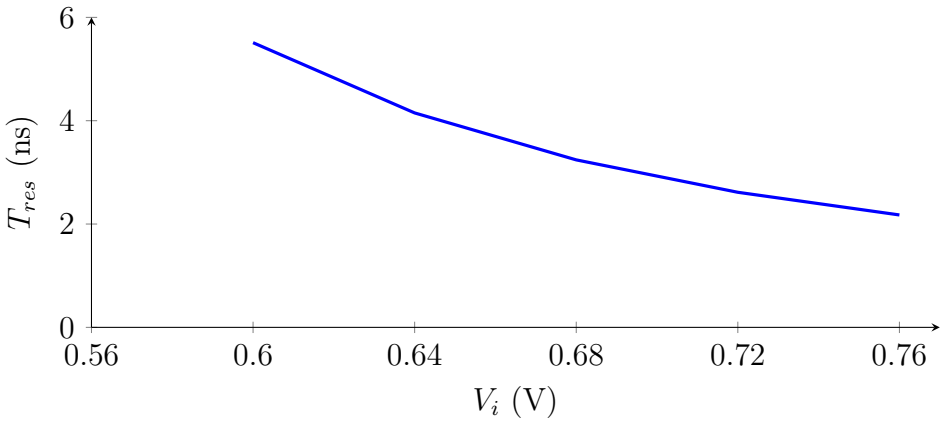


Figure 6.7: Voltage during a Refresh operation (0-0.9V Domain)

period reduces to 4.7ns. The only disadvantage of this method is that we need a separate power grid for V_{dd_cell}.

Chapter 7

Simulation Results

7.1 Write

All the simulations were done using a 3T uSA.

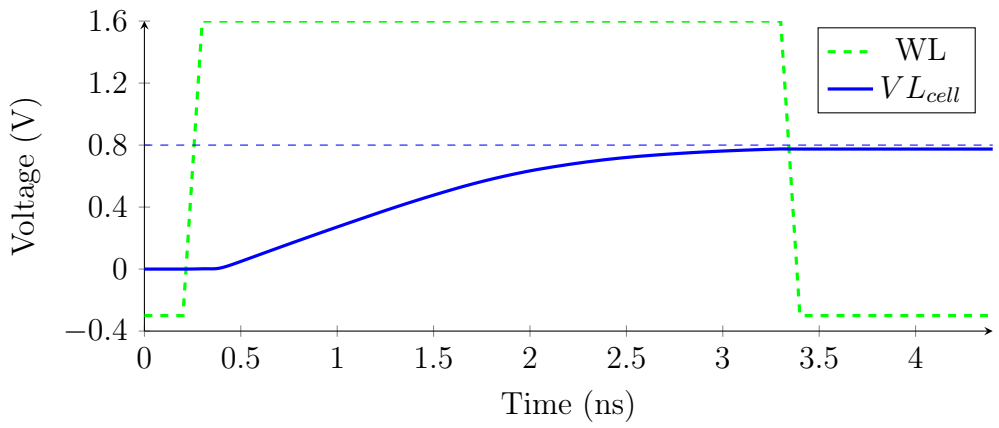


Figure 7.1: Cell Voltage during a write operation of '1'

The voltage written into the cell is 774mV.

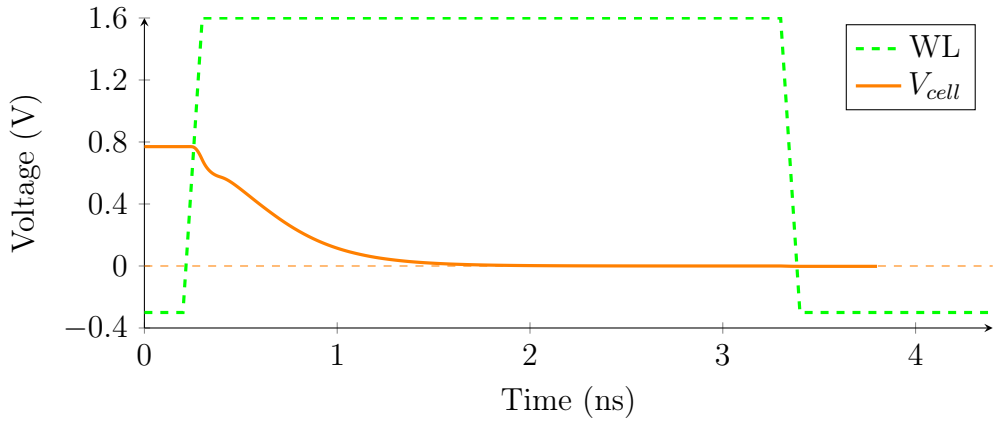


Figure 7.2: Cell Voltage during a write operation of '1'

7.2 Read

The following plots show the cell voltage during a Read. Write-back is enabled here.
Initial voltage in the cell before reading '1' is assumed to be 770mV.

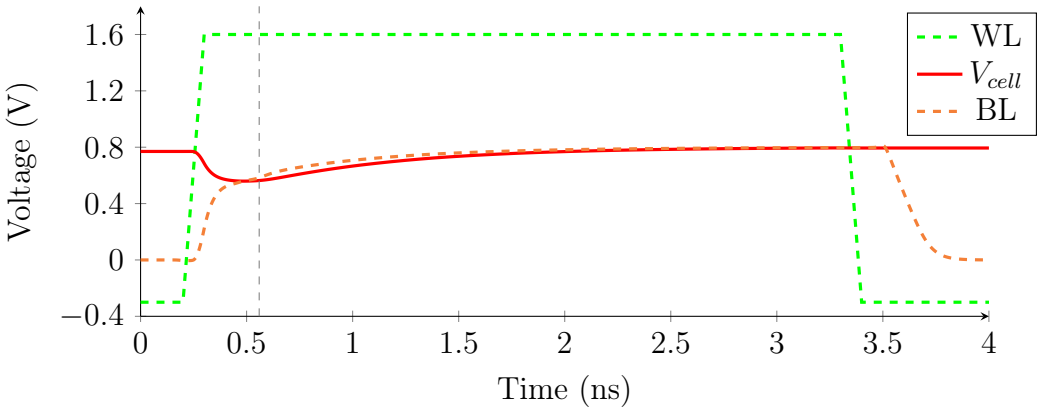


Figure 7.3: Read of a bit '1'

Writeback of '1' happens through a Feedback mechanism.

Initial voltage in the cell before reading '0' is assumed to be 300mV.

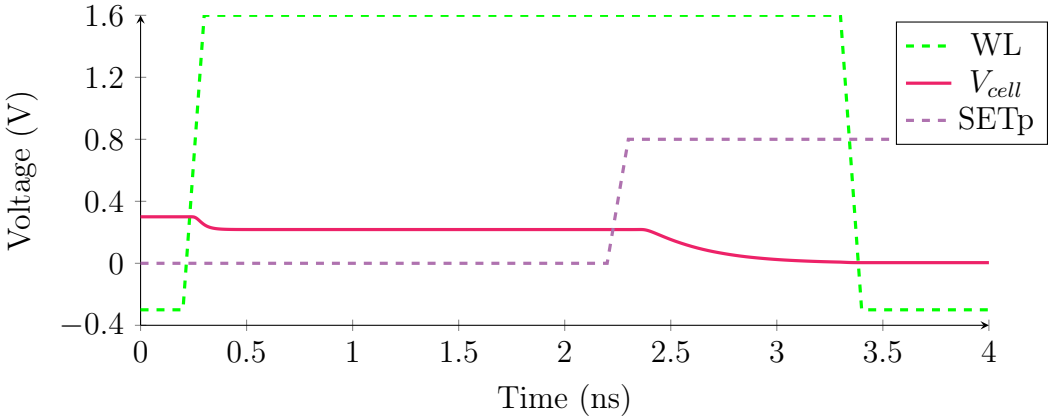


Figure 7.4: Read of a bit '0'0

Writeback of '0' is initiated through a timed signal 'SETp'.

7.3 Search

During a Search operation the write-back is disabled. The bit voltage discharges only in case of a mismatch. The wordline is kept turned ON for 1.5ns.

Initial voltage in the cell is assumed to be 770mV.

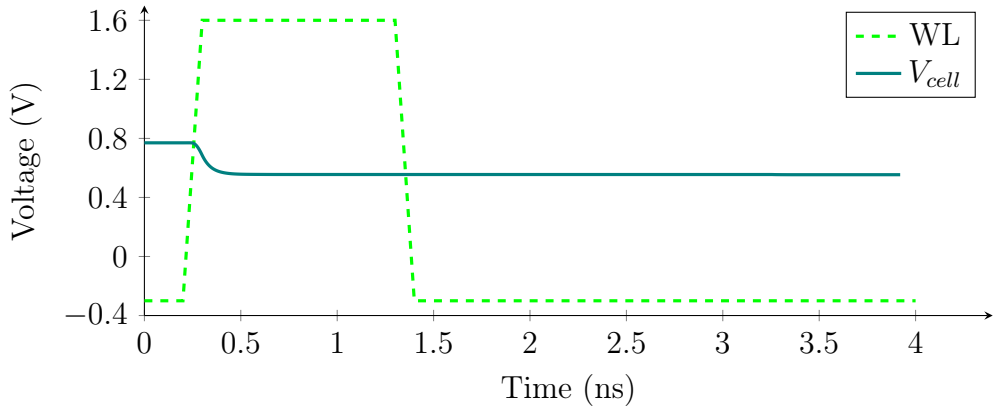


Figure 7.5: Cell Voltage during search operation

The final cell Voltage after the operation is 554mV.

7.4 Refresh

During a refresh, V_{dd_cell} is 0.9V to reduce the time taken to restore the bit to a strong '1'.

The voltage after before a refresh is assumed to be 450mV.

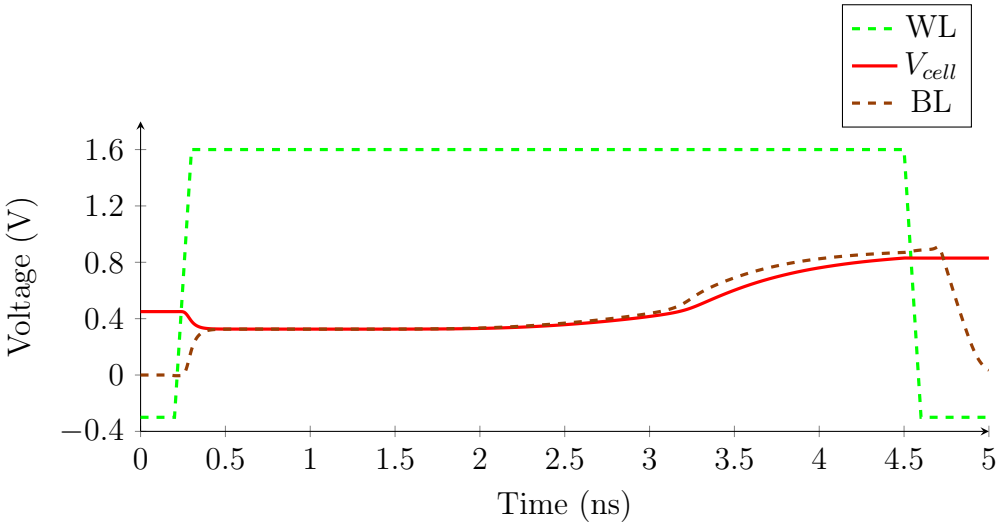


Figure 7.6: Refresh after a Search

The Wordline is turned ON for 4.2ns to restore the cell voltage to 720mV (90% V_{dd}).

Chapter 8

Conclusion

A new architecture for DRAM TCAM has been proposed which will conserve area and will give the output of a search operation in 8 cycles. The challenges faced in this architecture has been discussed in detail. To improve parallelism it is proposed to have multiple Local Bitlines. Atmost only one wordline per local Bitline can be turned ON. To preserve the cell voltage during a search operation, the write-back operation should be disabled. To ensure that the cell voltage after a search operation is recovered, it is necessary to do a refresh of the whole array. To increase the cell retention time, it is advised to increase the power supply used to write to the cell.

Future works include, better management of Refresh period to reduce the delay between to consecutive search operations and intelligently accessing the cells often to eliminate the need to refresh after every search operation.

Bibliography

[1] K. Pagiamtzis and A. Sheikholeslami, "Content-addressable memory (cam) circuits and architectures: a tutorial and survey," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 3, pp. 712–727, 2006.

[2] J. Delgado-Frias, A. Yu, and J. Nyathi, "A dynamic content addressable memory using a 4-transistor cell," in *Proceedings of the Third International Workshop on Design of Mixed-Mode Integrated Circuits and Applications (Cat. No.99EX303)*, 1999, pp. 110–113.

[3] V. Lines, A. Ahmed, P. Ma, S. Ma, R. McKenzie, H.-S. Kim, and C. Mar, "66 mhz 2.3 m ternary dynamic content addressable memory," in *Records of the IEEE International Workshop on Memory Technology, Design and Testing*, 2000, pp. 101–105.

[4] G. Fredeman, D. W. Plass, A. Mathews, J. Viraraghavan, K. Reyer, T. J. Knips, T. Miller, E. L. Gerhard, D. Kannambadi, C. Paone, D. Lee, D. J. Rainey, M. Sperling, M. Whalen, S. Burns, R. R. Tummuru, H. Ho, A. Cestero, N. Arnold, B. A. Khan, T. Kirihata, and S. S. Iyer, "A 14 nm 1.1 mb embedded dram macro with 1 ns access," *IEEE Journal of Solid-State Circuits*, vol. 51, no. 1, pp. 230–239, 2016.

[5] J. Barth, W. R. Reohr, P. Parries, G. Fredeman, J. Golz, S. E. Schuster, R. E. Matick, H. Hunter, C. C. Tanner, J. Harig, H. Kim, B. A. Khan, J. Griesemer, R. P. Havreluk, K. Yanagisawa, T. Kirihata, and S. S. Iyer, "A 500 mhz random cycle, 1.5 ns latency, soi embedded dram macro featuring a three-transistor micro sense amplifier," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 1, pp. 86–95, 2008.