IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 11, November 2017

# Design of low power Precharge-Free CAM design using parity based mechanism

**S.A.Sivakumar[1], A.Swedha[2]**

Assistant Professor, ECE Department, Info Institute of Engineering, Coimbatore, India [1]

PG Scholar, VLSI Design, Info Institute of Engineering, Coimbatore, India [2]

**Abstract:** Content-Addressable Memory (CAM) is the hardware for parallel lookup/search. CAMs are hardware search engines that are much faster than modified algorithmic approaches for search-intensive applications. The parallel search scheme promises a high-speed search operation but at the cost of high power consumption. Parallel NOR-type and NAND-type ML CAMs are suitable for high-search-speed and low-power-consumption applications, respectively. The NOR-type ML CAM requires high power, and therefore, the reduction of its power consumption is the subject of many reported designs. The main challenge in designing this project is to reduce the power consumption without reducing the speed and memory density. Here this memory is used to enhance the performance level while searching data from memory. The focus of this work is on the memory cell design, for match resolution and speed address detection. Recently, a precharge-free ML structure has been proposed but with inadequate search performance. In this project, Parity Bit Based Precharge-free CAM is proposed to reduce the power and increase the search speed. These are suitable for both high-search speed and low power consumption applications. Basically, these are used for packet forwarding in network routers. This design is simulated using Tanner tool.

**Keywords**: NAND-type CAM, NOR-type CAM, Precharge-free CAM, power, search speed , delay

## I. INTRODUCTION

Most of the memory devices store and retrieve data by addressing specific memory locations. This path becomes the limiting factor for those systems that depend on fast Memory access. The time required to find the data stored in memory can be reduced if the data can be identified by its content rather than by its address. A memory used for this purpose is Content Addressable Memory (CAM).CAM is used in applications where search time is very critical and very short. It is well suited for several functions like Ethernet address lookup, data compression, and security or encryption information on a packet-by packet basis for high performance data switches. Since CAM is an extension of RAM first, we have to know the RAM features to understand CAM. In general RAM has two Operations read and write i.e. the data stored in RAM can be read or written but CAM has three operations read, write and compare forwarding of packets [7]. CAM offers a performance advantage over other memory search algorithms, such as binary-based searches, tree-based searches, or look aside tag buffers, because it simultaneously compares the desired information against the entire list of pre-stored entries. Thus, CAM provides an order-of-magnitude reduction in the search time.It is a type of memory that compares the input data with the preloaded contents of the CAM block and generates a given output depending on the kind of CAM [4]. This kind of memory provides a distinct speed advantage over RAM in systems requiring quick address comparison or retrieval. Typical RAM applications utilize a counter to load the addresses into the RAM at a rate of one address per clock cycle. The data from the RAM block would then be compared with the expected data using XOR logic. When a match is found, the address becomes the valid output. The number of clock cycles required for this whole process is the number of addresses. For most applications, this would take more than one clock cycle. Because CAM can take the input (data) and compare it with all of the preloaded contents in the CAM array simultaneously, it can execute the entire lookup in a single clock cycle. This speed is ideally suited for network applications such as address lookups and filtering, packet encryption, and firewalls.

Software-baesd search algorithms are widely used. When used in the applications requiring high-speed search, e.g., IP routing, image processing, data compression, data management, and so on [1]–[4], they are slow and reduce the system speed. Thus, software-based search algorithms are unsuitable for high-speed applications. Contentaddressable memory (CAM) features hardware-based parallel search operation suitable for high-speed applications. Like other memories, CAM stores the data in its data banks. CAM feeds the search data, performs the search, and outputs the match address [5] if any. The parallel search scheme of the CAM activates a large number of circuits during the search operation and therefore needs more power. Designing a low power CAM for a large number of bits of a word without degrading its function (high-speed search) is a challenging task. Every CAM cell in a wordline is connected to a common matchline (ML); thus, as shown in Fig. 1, there are MLs equal to the number of words. Fig. 1 has M = 3 words of N = 4 bits each. The time between successive search operations of a CAM is one precharge cycle, and for every cycle, the ML has to be precharged.

DOI 10.17148/IJARCCE.2017.61123

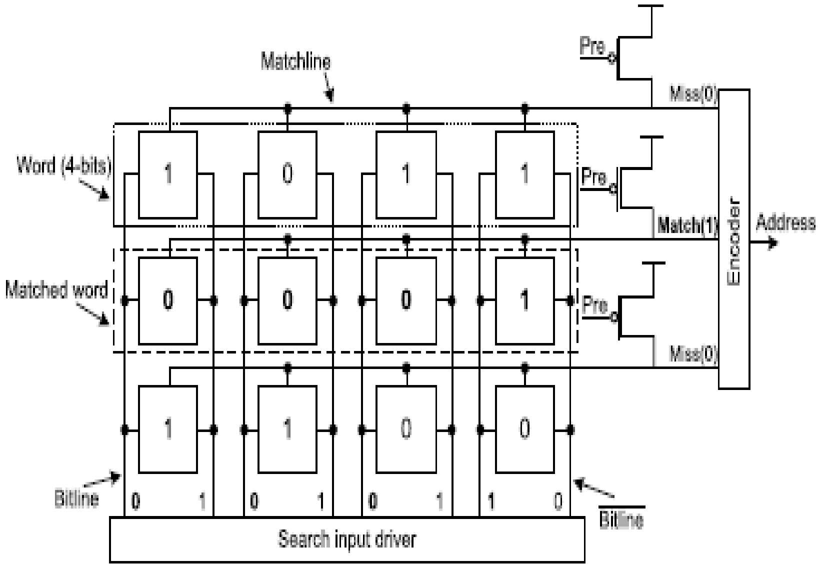


Fig 1. Basic CAM, $M \times N = 3 \times 4$.

The search input driver feeds in new search data for every precharge cycle through bitline and bitline. ML needs to be precharged for every cycle, and it holds the charge only if the word connected to it matches the search word. Usually, only one ML holds the charge in the case of unique words stored in the CAM's data bank. During search, as shown in Fig. 1, all but one MLs discharge. bitline and bitline toggle if the new search bit is different from its previous value. Thus, the frequent discharging and precharging of the MLs majorly contribute to the power consumption of the CAM. Here, we briefly review some of the power reduction techniques reported in the literature. Dual feedback positive sense amplifiers were used to improve performance as well as save energy by the early termination of discharging ML [6]. This was achieved by stopping the discharge of all the mismatching MLs as soon as the match was detected on any one line. In [7], networking mechanisms such as sparse clustered network schemes were incorporated in the CAM architecture, thereby eliminating many parallel searches and significantly reducing dynamic power consumption. Miyatake et al. [8] reported power optimization with the configurable macros of CAM for application-specific integrated circuit, designed with the flexibility of number of words and word length. In addition, search rush current was reduced by using pMOS ML drivers. A precomputation stage was used in [9].The parameters of each word in a CAM were computed and were stored in the precomputation stage. Before applying the search word into the CAM, the parameter of the search word was computed, and only those MLs in the CAM whose parameters matched were activated for further search. This eliminated the unnecessary precharging of all the MLs [9]. The precomputation block comprises a logic either to count the frequency of 0/1 as in [10] or to extract the parity bit of the word as in [11]. Adding parity bit increased the robustness of the CAM. In [12], ML segmentation and hierarchical matching were reported. Successive segments were precharged only if all the preceding segments matched. In [13], differential ML with a self-disabling sensing technique was designed to choke down the ML draining current. Employing differential ML, instead of a single-ended ML, helped boost the search speed without the overhead of power consumption.

The main challenge in designing the CAM is to reduce the power consumption without reducing the speed and memory density. Here the CAM memory is used to enhance the performance level while searching data from memory. The focus of this project is on the CAM memory cell design, for match resolution and speed address detection in CAM. Parity bit based Precharge-free CAM is proposed to reduce the power and increase the search speed of the CAM. In this project a comprehensive analysis about the power consumption and the delay for various precharge-free CAMs has been presented. By this, the search required address for single clock period will obtain quickly. With the help of parity based CAM architecture, the data fetching speed will be increased. CAM has a single clock cycle throughput making them faster than other hardware and Software based search systems. Basically, these CAMs are used for packet forwarding in network routers.

This paper is structured as follows: Section I provides the introduction to Content-Addressable Memory. Survey of CAM operation is explained in section II. Section III explains the design of Precharge-Free CAM. Section IV explains the design of Parity-Bit based Precharge-Free CAM. Results and performance for newly proposed designs are compared in terms of average power and delay in Section V. Paper ends with the conclusion in Section VI.

## II. CONTENT ADDRESSABLE MEMORY

CAM comprises memory element, usually built with 6T SRAM cell and a circuit to compare search bit against the stored bit. NOR- and NAND-type MLs are the two basic comparison circuits presented as follows.

### A. Overview of CAM and RA*M*

The RAM has two major operations: Write and Read. (i.e.) user supplies the memory address and the RAM returns the data stored in that address. A CAM has three major operations: Write, Search and Read. (i.e.) user supplies the data word and CAM search its entire memory to see the supplied data word. If the data word is found, the CAM returns one or more storage address where the word was found. The are two types of CAM .They are Binary CAM (BCAM)and Ternary CAM(TCAM) where BCAM stores 0’s and 1’s and TCAM stores 0’s, 1’s and don’t care. CAM is used in various application like IP Packets in network routers, Broad ATM communication system and Image coding.

### B. Basic operation of CAM cell

The basic CAM cell is based on the static memory cell. Data is stored in two cross coupled inverters. The two NMOS transistors controlled by the word line allow the CAM to be written. The four additional transistors used for matching. The bit storage portion is a standard 6T static RAM (SRAM) cell. Hence, this cell performs READ and WRITE operations similar to an SRAM cell. Third operation is MATCH operation. For matching a data bit with stored bit. First leave the wordline low then Precharge match line. Place key on bitline. Match line evaluated.

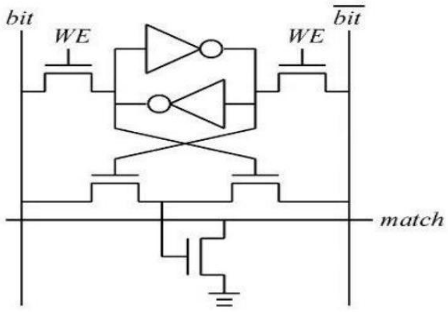


Fig 2. Block diagram of CAM cell

### C. NAND-type CAM cell

The NAND cell implements the comparison between the stored bit, D, and corresponding search data on the corresponding search lines, (SL,~SL ), using the three comparison transistors $M_1$,$M_D$, and $M_{DB}$ , which are all typically minimum-size to maintain high cell density. Consider the case of a match when SL =1 and D=1 Pass transistor $M_D$ is 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. The logic 1 on node B turns ON transistor $M_1$. Note that $M_1$ is also turned ON in the other match case when SL =0 and D=0. In this case, the transistor $M_{DB}$ passes a logic high to raise node B. The remaining cases, where SL not equal to D result in a miss condition, and accordingly node B is logic is 0 and the transistor $M_1$ is OFF. Node B is a pass-transistor implementation of the XNOR function SL xnor D. The NAND nature of this cell becomes clear when multiple NAND cells are serially connected. In this case, the $ML_n$ and $ML_{n+1}$ and nodes are joined to form a word. A serial NMOS chain of all the transistors resembles the pull down path of a CMOS NAND logic gate.

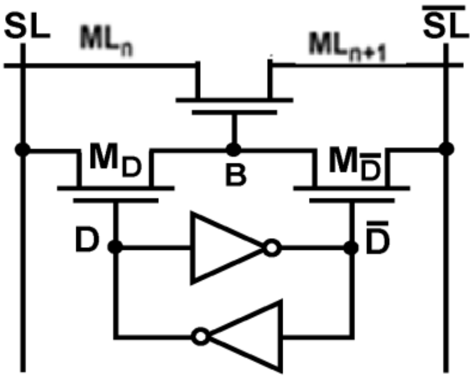


Fig 3. NAND-type CAM cell

DOI 10.17148/IJARCCE.2017.61123

*D. NOR-type CAM cell*

The NOR cell implements the comparison between the complementary stored bit, D (and ~D), and the complementary search data on the complementary search line, SL (and ~SL), using four comparison transistors, $M_1$ through $M_4$, which are all typically minimum-size to maintain high cell density. These transistors implement the pull down path of a dynamic XNOR logic gate with inputs SL and D. Each pair of transistors, $M_1/M_3$ and $M_2/M_4$, forms a pull down path from the match line, ML, such that a mismatch of SL and D activates least one of the pull down paths, connecting ML to ground.

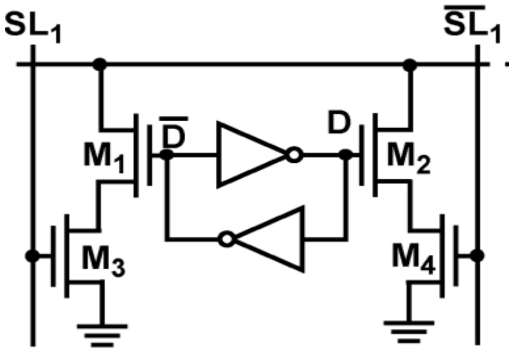


Fig 4. NOR type CAM cell

A match of SL and D disables both pull down paths, disconnecting ML from ground. The NOR nature of this cell becomes clear when multiple cells are connected in parallel to form a CAM word by shorting the ML of each cell to the ML of adjacent cells. The pull down paths connect in parallel resembling the pulldown path of a CMOS NOR logic gate. There is a match condition given ML only if every individual cell in the word has a match.

In typical use, a CAM has only one or a small number of matches and most words mismatch. Since mismatches dominate, most matchlines transition both during precharge and during evaluation. This leads to a high power consumption on the matchlines. Further, the search-lines, which broadcast the data to the CAM cells are highly capacitive. The search-lines are another large source of power dissipation in CAM. Because of these large sources of power dissipation, recent research in CAM design focuses on circuit techniques for reducing power consumption.

## III. PRECHARGE-FREE CAM

Designing a high-speed CAM for larger word lengths is a challenging task. Recently PF-CAM architectures have been presented, but these lack the search performance at higher word lengths. A self-controlled PF-CAM based on early prediction has been reported in this paper to avoid the dependence among CAM cells and to improve the frequency of operation.

A. Precharge-Free CAM

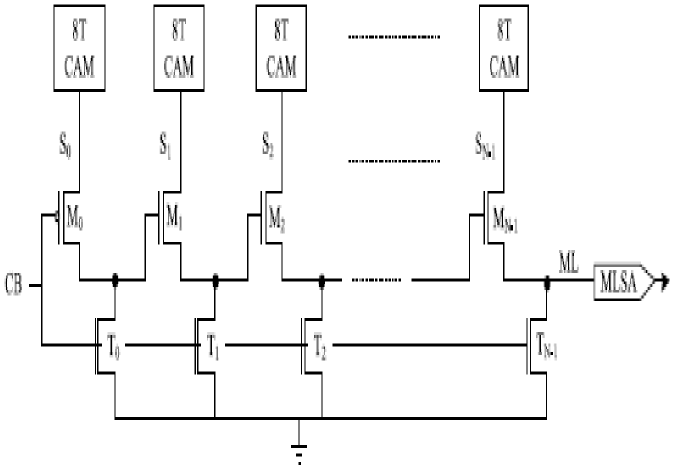


Fig 5. Precharge-Free CAM

DOI 10.17148/IJARCCE.2017.61123

The development of a CAM structure is carried out with a PRE signal, which ends up at a lower speed of search operation. In this drawback has been eliminated by removal of the PRE phase, as shown in Figure 3.1. Instead, they used control bits (CBs), which reduced the overall search time by one level. In a CAM, the first operation is write, followed by precharge, and then search. However, in the PF-CAM architecture, the write is followed directly by the search phases. The operation of the PF-CAM is as follows.

1) While storing the data, CB is set to a high value (logic 1). This setting of CB turns M0 OFF and, simultaneously, $M_1$,………$M_{N-1}$ will be turned OFF as $T_0$,$T_1$,………$T_{N-1}$ are ON, which will provide GND directly to the gate of M1,………$M_{N-1}$, as shown in Fig. 3.1.

2) Once the data are stored, CB is reset (0) and this will turn ON M0 as M0 is a pMOS, since the source of $M_0$ is connected as a control to M1, thus S0 value is passed to the gate of $M_1$, which is an nMOS. If CAM cell-1 is matched, M0 will pass logic high to $M_1$, which will result in turning ON of $M_1$. If CAM cell-2 is also matched, then in a similar fashion logic high will be passed to M3 from CAM cell-2, and likewise a cascaded chain of CBs will be passed from one cell to another. If there is a mismatch at any cell, then the forthcoming cells will be turned OFF. The PF-CAM structure gives advantage by reduction in the number of SC paths, which in turn results in overall reduction of power; however, due to cascading of CAM cells overall, the speed of search operation is significantly reduced. To avoid this problem, the SCPF-CAM architecture is proposed.

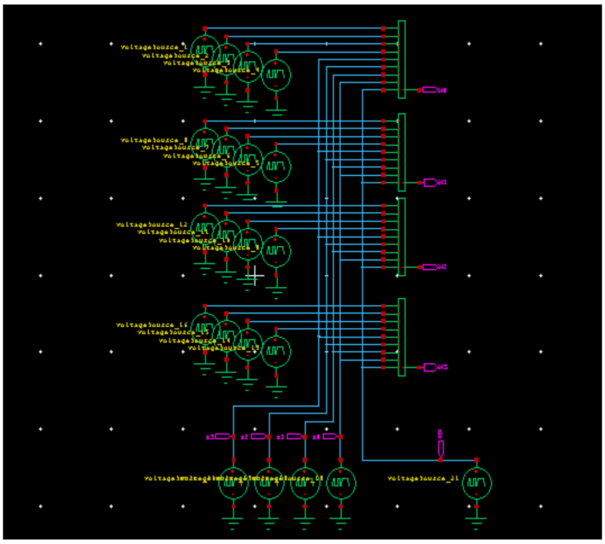


Fig 6. Precharge-Free CAM circuit diagram

### B. Self-Controlled Precharge-Free CAM

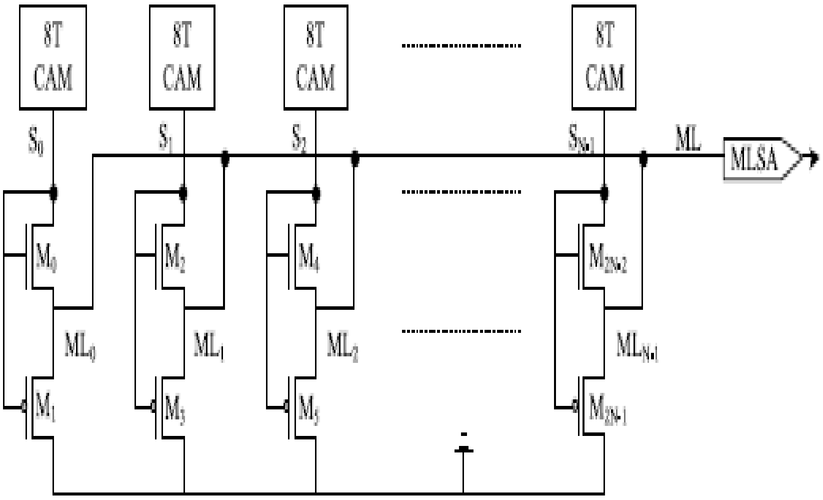


Fig 7. Self-Controlled Precharge-Free CAM

DOI 10.17148/IJARCCE.2017.61123

Precharged MOS connected to all MLs of CAM architectures are available in the literature. The CAM precharges and evaluates all the MLs for every PRE cycle during the low and high level of the PRE signal, respectively. In butterfly and hierarchical manner word structures are designed based on splitting the complete word structure into fragments. Precomputation-based precharging has been used by selectively precharging all MLs. PRE-based CAMs mainly suffer from the following.

1) The speed of the search operation is restricted by the precharge phase.
2) Dependence among CAM cells restricts the speed of search operation and this affects the overall performance.
3) NAND-type ML suffers from the charge sharing problem. In SCPF-CAM, the advantage of the PF-CAM structure is exploited; moreover, drawbacks of the precharge-based earlier reported circuits (which was cascading) are also taken care of by removing dependence among different CAM cells.

The advantage of this architecture is the design of larger word lengths with higher performance at a higher frequency of operation. Owing to the larger delay metric in PF-CAM, it is not useful for forming longer word lengths and cannot operate at a higher frequency of operation. Figure 3.2. illustrates the SCPF-CAM structure, which solves the deficiency of all the mentioned problems present in the precharge-based CAMs and improves the speed of operation compared to the PF-CAM architecture. An 8T CAM cell has been used as the basic block to design a word.

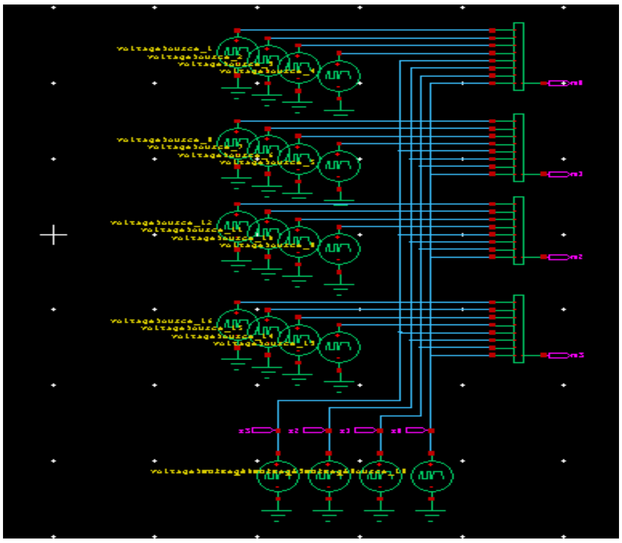


Fig 8. Self-Controlled Precharge-Free CAM circuit diagram

Two major contributions are made: 1) self-control operation, where the charge stored at the node *S* controls the ML transistors, thereby avoiding the dependence on previous ML value; and 2) the scheme eliminates the precharge phase to provide a higher search frequency. The SCPF-CAM is self-controlled; the node (S) value of the 8T CAM cell controls the evaluation logic and produces the output. If the search content matches the prestored data, then it passes a high value through M9; otherwise it passes a low value through M10 to the ML. The minimum operating voltage is limited to $V$THP +2$V$THN+$VM$. The voltage $VM$ is mostly MLSA dependent and the most dominating among the three voltages. Transistor M9 is chosen to have a low threshold to push the supply voltage scaling limit.

The minimum amount of time required in the conventional CAM operation is

$$T_{total} = T_{write} + T_{precharge} + T_{search.} \qquad (1)$$

The minimum amount of time required in the Self-controlled Precharge-Free CAM is

$$T_{total} = T_{write} + T_{search.} \qquad (2)$$

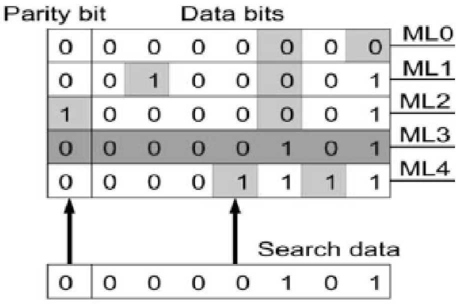
## IV. PROPOSED PARITY BASED PRECHARGE-FREE CAM

Content Addressable Memory (CAM) is a type of solid-state memory in which data are accessed by their contents rather than physical locations. It receives input search data, i.e., a search word, and returns the address of a similar word that is stored in its data bank. CAM offers high-speed search function in a single clock cycle. In this project, we introduce a parity bit that leads to delay reduction and high speed.

DOI 10.17148/IJARCCE.2017.61123

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 11, November 2017

The condition of even or odd for the value one for the number of strings can be denoted by a parity bit. Parity bits are used in error detection whose variants are even parity and odd parity bit. If it is even parity, parity bit is set to 1, if the count of ones in a given set of bits (excluding the parity bit) is odd. If the count of ones is even, it is set to 0. For odd parity, the parity bit is set to 1 if the count of ones in a given set of bits (excluding the parity bit) is even. The odd parity bit is set to 0 if the count of set bits is odd. A unique case of a CRC is even parity, Polynomialx+1 generates one bit cyclic redundancy check. Parity bit can be space parity if the parity bit is 1, which is present and not used. Parity bit may be mark parity if parity bit is always 1. An XOR can be used for parity bit calculation, yielding 0 for even parity and 1 for odd parity. ). If there are two mismatches in the data segment or the parity bits are the same and overall we have two mismatches. With more mismatches, we can ignore these cases as they are not crucial cases. The sense amplifier now only has to identify between the 2-mismatch cases and the matched cases.



| Parity bit | Data bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ML0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | ML1 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | ML2 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | ML3 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | ML4 |

Search data

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

Fig 9. Parity bit based CAM structure

A. Parity

Parity refers to the oddness or evenness of a number. Parity for a binary number can be indicated by the least significant bit alone. However in computing, parity means the evenness or oddness of bits with value one within a given set of bits, determined by the value of all the bits. An XOR can be used for parity bit calculation, yielding 0 for even parity and 1 for odd parity. In this project, multiplexer functions as a XOR. This property of being dependent upon all the bits and changing value according to any one bit change allow for its use in error detection schemes. In this we consider the parity as index value for searching and fetch the data for given data.

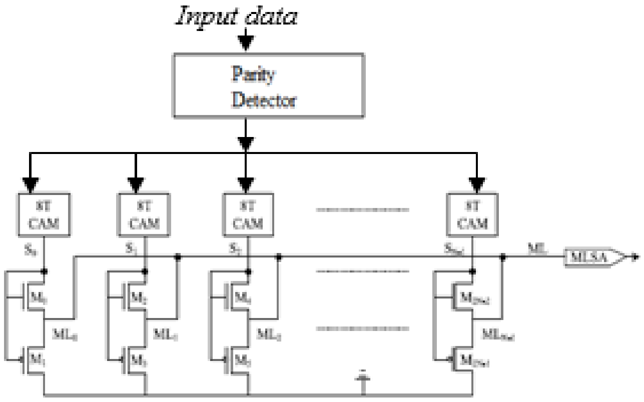


Fig 10. Parity bit based Precharge-Free CAM Architecture

B. High speed using Parity bit

We introduce a versatile auxiliary bit to boost the search speed of the CAM at the cost of less than 1% area overhead and power consumption extra bit. During the search operation, there is only one single stage assign conventional CAM. Hence, the use of these parity bits does not improve the power performance. However, this additional parity bit, in theory, reduces the sensing delay and boosts the driving strength of the 1-mismatch case (which is the worst case) by half, as discussed below. In the case of a matched in the data segment the parity bits of the search and the stored word is the same, thus the overall word returns a match. When mismatching of 1 occurs in the data segment, numbers of "1"s in the stored and search word must be different by 1. As a result, the corresponding parity bits are different. Therefore now we have two mismatches (one from the parity bit anode from the data bits). If there are two mismatches in the data segment or the parity bits are the same and overall we have two mismatches. With more mismatches, we can ignore these cases as they are not crucial cases. The sense amplifier now only has to identify between the 2-mismatch cases

DOI 10.17148/IJARCCE.2017.61123

and the matched cases. Since the driving capability of the 2-mismatch word is twice as strong as that of the 1-mismatch word, the proposed design greatly improves the search speed.

Parity bit based Prechrge-Free CAM will have the following functions:

- Speed of the processor will be increased
- Low-delay,
- Low area, low power consumption

C. Flowchart design

The searched word is entered first the parity bit for the corresponding entered searched word is extracted and parity bit value of the stored word is calculated, this is compared with the parity bit value of the searched word. After performing this compare operation match is calculated. If any stored word match with the searched word means match condition will occur if it is not matched means miss condition will occur. After that the word is forwarded to the corresponding match line. Due to this process only the speed of the process will be increased and processor performance is improved. The flow chart design for parity bit based CAM is shown below.

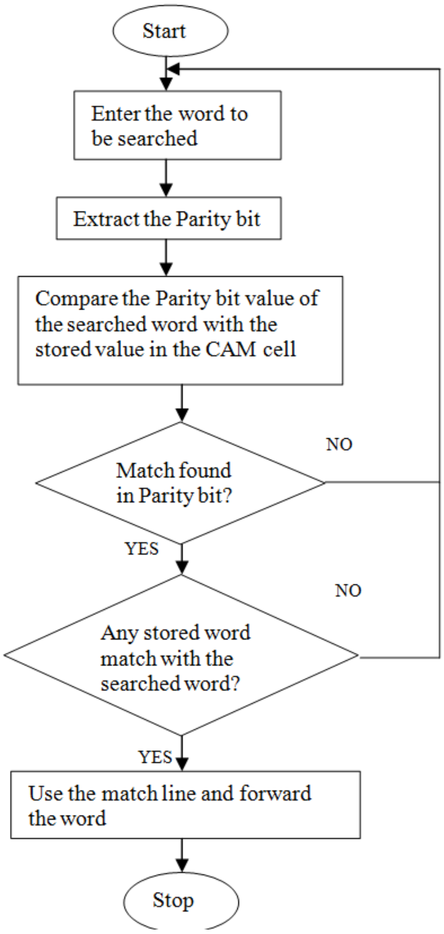

Start
Enter the word to be searched
Extract the Parity bit
Compare the Parity bit value of the searched word with the stored value in the CAM cell
Match found in Parity bit?
NO
YES
Any stored word match with the searched word?
NO
YES
Use the match line and forward the word
Stop

Fig 11. Flow chart of parity based CAM

D. Proposed system operation

The parity bit based CAM design is shown in Fig. 2(b) consisting of the original data segment and an extra one-bit segment, derived from the actual data bits. We only obtain the parity bit, i.e., odd or even number of “1”s. The obtained parity bit is placed directly to the corresponding word. Thus the new architecture has the same interface as the conventional CAM with one extra bit. During the search operation, there is only one single stage as in conventional CAM. Hence, the use of this parity bits does not improve the power performance. However, this additional parity bit, in theory, reduces the sensing delay and boosts the driving strength of the 1-mismatch case (which is the worst case) by half. In the case of a matched in the data segment, the parity bits of the search and the stored word is the same, thus the overall word returns a match. When 1 mismatch occurs in the data segment, numbers of “1”s in the stored and search word must be different by 1. As a result, the corresponding parity bits are different. Therefore now we have two mismatches (one from the parity bit and one from the data bits). If there are two mismatches in the data segment, the parity bits are the same and overall we have two mismatches. With more mismatches, we can ignore these cases as they

DOI 10.17148/IJARCCE.2017.61123

are not crucial cases. The sense amplifier now only have to identify between the 2-mismatch cases and the matched cases. Since the driving capability of the 2-mismatch word is twice as strong as that of the 1-mismatch word, the proposed design greatly improves the search speed.

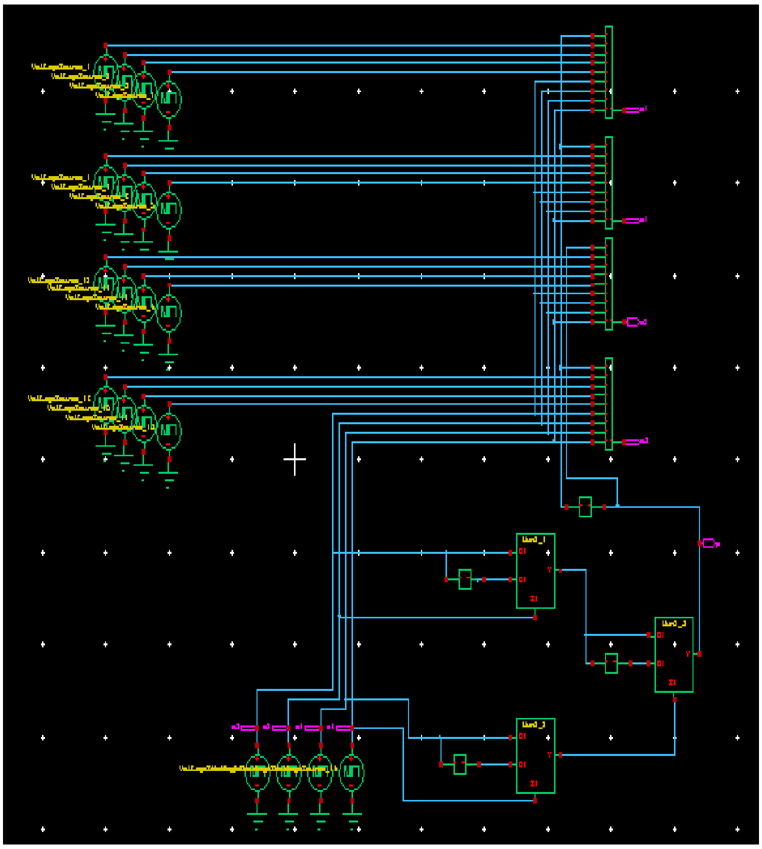


Fig 12. Parity bit based Precharge-Free CAM Circuit

## V. SIMULATION AND PERFORMANCE COMPARISON

The simulation and power analysis is obtained with the help of TANNER tool, using IBM013 CMOS Technology. All simulations are carried out using W-edit simulation tool. The simulated waveform of the proposed modified pipe logic flip flop is shown in Fig.12

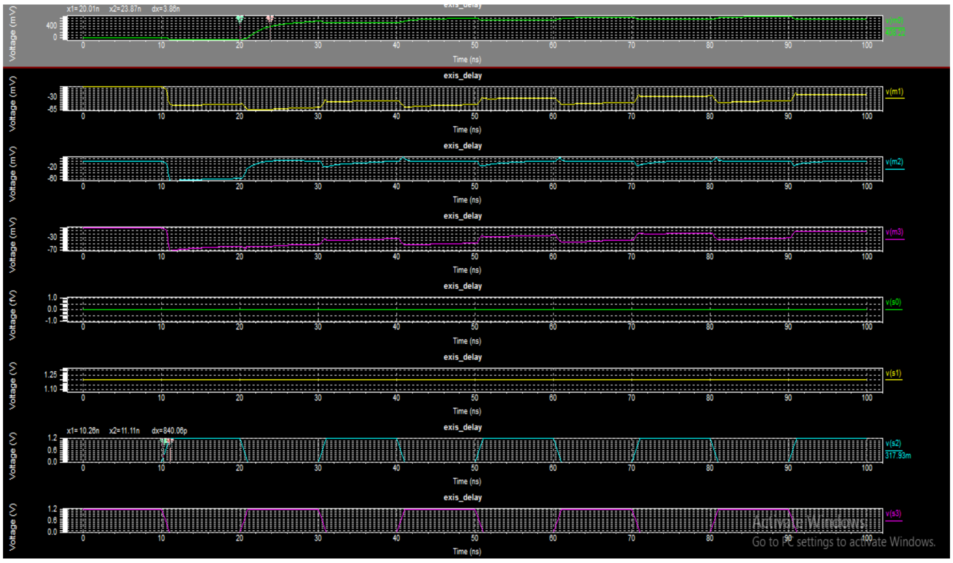


Fig 13. Waveform of Precharge-Free CAM

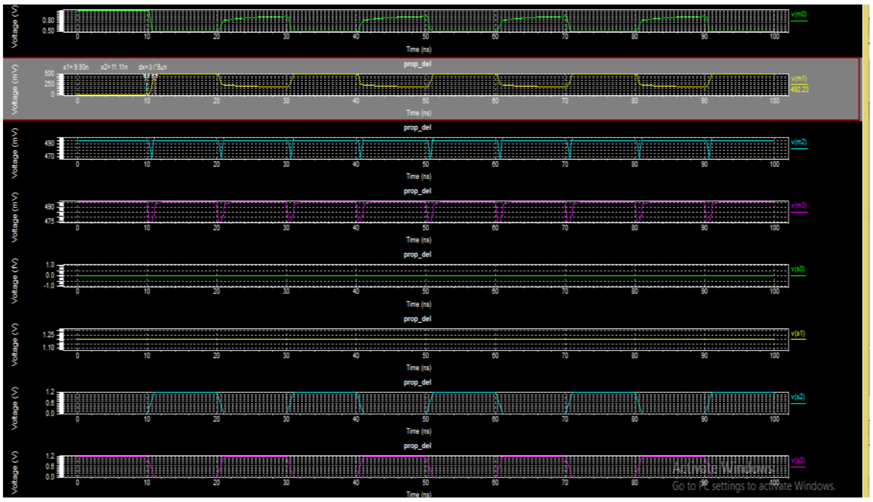


Fig 14. Waveform of Self-Controlled Precharge-Free CAM

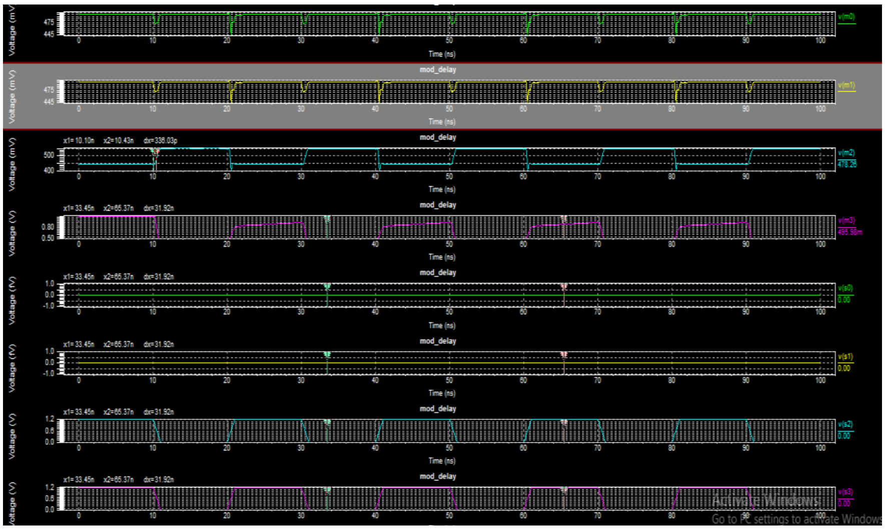


Fig 15. Waveform of Parity bit based Precharge-free CAM

TABLE I PERFORMANCE OF PARITY-BIT BASED PRECHARGE-FREE CAM

| VARIOUS CAMDESIGNS | PF-CAM | SCPF-CAM | PARITY BIT BASED PF-CAM |
|---|---|---|---|
| AVERAGE POWER CONSUMPTION | 1.394μW | 1.338μW | 1.298μW |
| DELAY | 9.223NS | 4.265NS | 2.232NS |
| SPEED | 108.42MHZ | 234.46MHZ | 448.02MHZ |

The comparison showing power, delay and speed of PF-CAM, SCPF-CAM and proposed parity bit based PF-CAM is shown in fig.16

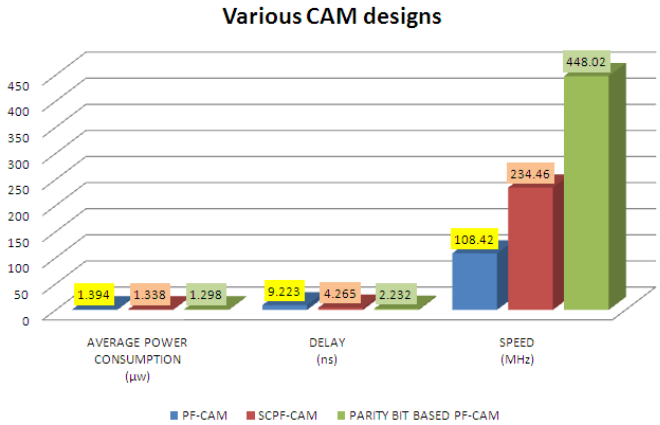


Fig 16. Comparison showing various CAM designs

DOI 10.17148/IJARCCE.2017.61123

## VI. CONCLUSION

The architecture of CAM design is changed to speed up the process of searching data. This proposed parity bit based Precharge-Free CAM proved to be most efficient when compared to the existing architecture. CAM is ideally fitted to many functions, as well as LAN address search, knowledge compression, pattern-recognition, cache tags, high bandwidth address filtering, and quick search of routing, user privilege, security or coding data on a packet-by packet basis for superior knowledge switches, firewalls, bridges and routers. CAM may be used to accelerate any applications starting from local-area networks, management, file storage management, pattern recognition, computer science, totally associative and processor-specific cache recollections, and cache recollections. Though CAM has several applications, it's notably used to perform any quite search operations. Parity-bit based architecture offer several major advantages, namely average power consumption and boosted search speed. Experimental result shows better performance result than traditional CAM architecture.

## ACKNOWLEDGMENT

We sincerely thank the Management of Info institute of Engineering-Coimbatore for their constant support and encouragement towards this research paper and rendering us the research lab for our purpose.

## REFERENCES

[1] S. K. Maurya and L. T. Clark, “A dynamic longest prefix matching content addressable memory for IP routing,” *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 19, no. 6, pp. 963–972, Jun. 2011.

[2] A. Bremler-Barr and D. Hendler, “Space-efficient TCAM-based classification using gray coding,” *IEEE Trans. Comput.*, vol. 61, no. 1, pp. 18–30, Jan. 2012.

[3] Y.-C. Shin, R. Sridhar, V. Demjanenko, P. W. Palumbo, and S. N. Srihari, “A special-purpose content addressable memory chip for real-time image processing,” *IEEE J. Solid-State Circuits*, vol. 27, no. 5, pp. 737–744, May 1992.

[4] S. M. Jalaleddine, “Associative memories and processors: The exact match paradigm,” *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 11, pp. 45–67, Mar. 2013. [Online]. Available: http://www.sciencedirect.com/ science/article/pii/S1319157899800032

[5] K. Pagiamtzis and A. Sheikholeslami, “Content-addressable memory (CAM) circuits and architectures: A tutorial and survey,” *IEEE J. Solid-State Circuits*, vol. 41, no. 3, pp. 712–727, Mar. 2006.

[6] S. I. Ali and M. S. Islam, “A match-line dynamic energy reduction technique for high-speed ternary CAM using dual feedback sense amplifier,” *Microelectron. J.*, vol. 45, no. 1, pp. 95–101, Jan. 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0026269213002462

[7] H. Jarollahi, V. Gripon, N. Onizawa, and W. J. Gross, “Algorithm and architecture for a low-power content-addressable memory based on sparse clustered networks,” *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 23, no. 4, pp. 642–653, Apr. 2015.

[8] H. Miyatake, M. Tanaka, and Y. Mori, “A design for high-speed lowpower CMOS fully parallel content-addressable memory macros,” *IEEE J. Solid-State Circuits*, vol. 36, no. 6, pp. 956–968, Jun. 2001.

[9] S.-J. Ruan, C.-Y. Wu, and J.-Y. Hsieh, “Low power design of precomputation-based content-addressable memory,” *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 16, no. 3, pp. 331–335, Mar. 2008.

[10] H.-Y. Li, C.-C. Chen, J.-S. Wang, and C. Yeh, “An AND-type matchline scheme for high-performance energy-efficient content addressable memories,” *IEEE J. Solid-State Circuits*, vol. 41, no. 5, pp. 1108–1119, May 2006.

[11] A.-T. Do, S. Chen, Z.-H. Kong, and K. S. Yeo, “A high speed low power CAM with a parity bit and power-gated ML sensing,” *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 21, no. 1, pp. 151–156, Jan. 2013.

[12] V. M. Zackriya and H. M. Kittur, “Selective match-line energizer content addressable memory (SMLE-CAM),” *Int. J. Appl. Eng. Res.*, vol. 8, no. 19, 2013.

[13] C.-C. Wang, C.-H. Hsu, C.-C. Huang, and J.-H. Wu, “A self-disabled sensing technique for content-addressable memories,” *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 57, no. 1, pp. 31–35, Jan. 2010.

[14] P.-T. Huang and W. Hwang, “A 65 nm 0.165 fJ/bit/search 256 × 144 TCAM macro design for IPv6 lookup tables,” *IEEE J. Solid-State Circuits*, vol. 46, no. 2, pp. 507–519, Feb. 2011.

[15] C.-S. Lin, J.-C. Chang, and B.-D. Liu, “A low-power precomputationbased fully parallel content-addressable memory,” *IEEE J. Solid-State Circuits*, vol. 38, no. 4, pp. 654–662, Apr. 2003.

## BIOGRAPHIES

**S. A. Siva Kumar** has a B.E., degree in ECE and M.E., degree in VLSI Design and currently working as Assistant Professor at Info Institute of Engineering-Coimbatore-India. He is pursuing his Research in Anna University-Chennai. His areas of interests are VLSI design and embedded systems.

**A. Swedha** has a B.E., degree in ECE and currently pursuing M.E VLSI Design at Info Institute of Engineering-Coimbatore-India. Her areas of interests are low power VLSI design.

DOI 10.17148/IJARCCE.2017.61123