ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING

National Conference on Advanced Innovation in Engineering and Technology (NCAIET-2015)

Alva's Institute of Engineering and Technology, Moodbidri

Vol. 3,Special Issue 1,April 2015

A low power, high speed parity check parallel data search CAM using complimentary search lines

Mr. Sandesh Kumar[1], Mrs. Prabha Niranjan[2], Mr. Chethan R.[3]

Post Graduate Student, Department of Electronics & Communication, NMAMIT, Nitte, India [1]

Associate Professor, Department of Electronics & Communication, NMAMIT, Nitte, India [2]

Assistant Professor, Department of Electronics & Communication, SMVITM, Bantakal, India [3]

Abstract: In all digital systems, memory acts as brain of the system. Any data such as files, audio, video and images are stored in memory in the form of binary digits such as '0's and '1's. Most memory devices store and retrieve data by addressing specific memory locations. Many methods are implemented to search data from memory. A memory that searches data using parallel method called content-addressable memory or CAM. Content addressable memory (CAM) is a type of solid-state memory that access the input data and performs parallel search operations and gives the address of the data. It starts a compare operation by loading search data into the search data register. This search data is then broadcast into the memory banks through pairs of complementary search-lines(SL and SLbar). The parallel comparison done with every bit of the stored words using comparison circuits. Each stored word has a matched line(ML) that convey the comparison result. Address of the matched word obtained at encoder output.

I. INTRODUCTION

A Content Addressable Memory (CAM) compares input data with a sets of stored data, and gives the address of the matching data. CAM is a type of solid state memory here memory is accessed by its content rather than by its address [1]. A CAM performs three operations called READ, WRITE, and COMPARE, but COMPARE is the main operation as CAM but it rarely performs read or write operations [4]. Figure 1.1 shows a block diagram of conventional CAM which consists of a search data register, comparison circuits, complementary search lines (SLs and SLsbar), match lines (ML), memory banks, match line sense amplifier (MLSA) and an output encoder. An n-bit data to be searched is stored in search data register to perform the COMPARE operation. The input search data is launched into the core memory cells through n-pair of complementary search-lines (SLs and SLsbar) and comparison will be done with every bit using comparison circuits. A match line (ML) is associated with each store data is shared between each bit to convey the comparison result. Matched data address is available at encoder output. In the case of multiple matched data location, then priority encoder can be deployed such that lower address data will get a highest priority. During the pre-charge MLs line is charged to V_{DD} voltage. During evolution stage search data present in the search data register will transmit through complementary search lines SLs and SLsbar. If match occur between search data and stored data present in memory cells or CAM cells (for example at the first cell of the row D ="1" ; Dbar ="0" and SL = "1";SLbar = "0"), transistors M1 and M4 will turn OFF and transistors M2 and M3 will turn ON, ML is held at higher potential (V_{DD}). In case of mismatches (D="1"; Dbar = "0" and SL="0"; SLbar = "1") in the CAM cell, both transistors in the CAM cell M3 or M4 is turns ON, then voltage of the ML is discharges to ground potential. Finally, if ML is equal to V_{DD} potential indicate for matched bits and ML is at ground potential indicates mismatched bits. A match line sense amplifier (MLSA) is used to detect voltage change on the ML and amplifies it to a full CMOS output voltage.

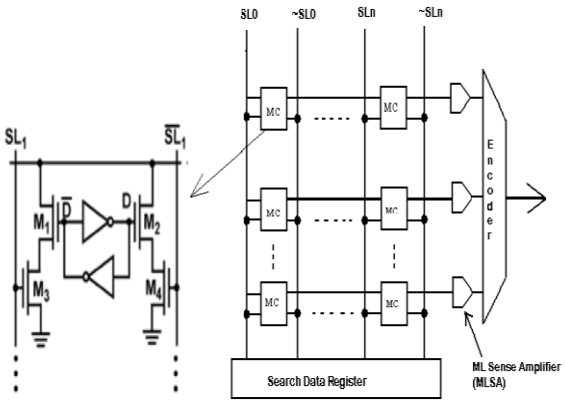


Figure 1.1 Functional block diagram of conventional CAM.[1]

Since all available bits present in search data register is compared parallel with the CAM memory bits. Hence, CAMs are faster than other serial bit comparison systems [1]. They are therefore preferred in high-throughput applications such as network routers and data compressors. In this proposed work, searching speed of the CAM is increased by using parity check bit.

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING

National Conference on Advanced Innovation in Engineering and Technology (NCAIET-2015)

Alva's Institute of Engineering and Technology, Moodbidri

Vol. 3,Special Issue 1,April 2015

II. PROPOSED ALGORITHM

2.1 Working of proposed parity check parallel data search CAM and Conventional precomputation CAM

The newly proposed versatile parity check bit CAM is similar to the existing Pre-computation CAM scheme but it has a different operating principle. First briefly discuss the precomputation CAM scheme before presenting proposed parity bit scheme.

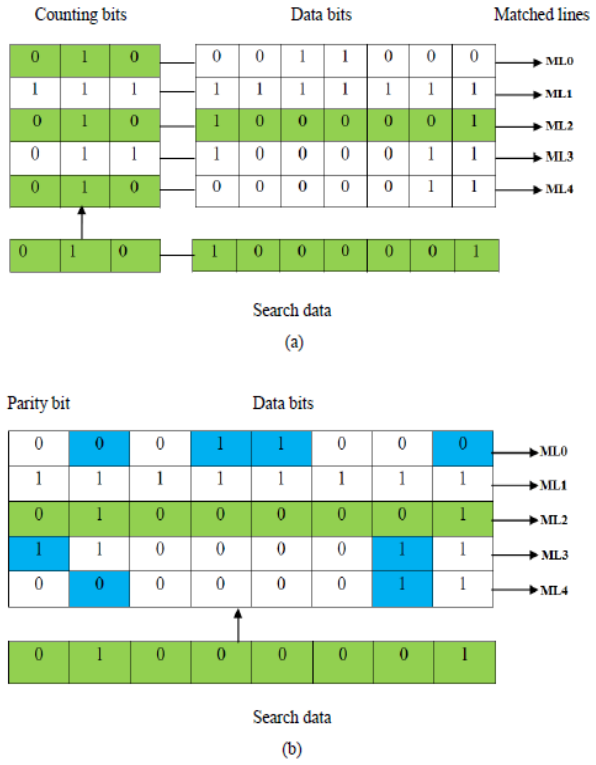


Figure 2.1 Conceptual view of (a) conventional pre-computation CAM and (b) proposed parity check based CAM.[1]

2.1.1 Pre-Computation CAM

The pre-computation CAM has 2 sets of memory banks (segments). In the First sets of memory segments has counting bits, shows the total number of "1" bits present in data called counting bits. And second sets of memory segments has actual data. Each data has counting bits, separately stored along the data shown in figure 2.1 (a). The additional counting bits used to filter some mismatched CAM data before the actual comparison. These extra bits called counting bits are derived from the data bits and are used as the first comparison stage. For example, in figure 2.1(a) number of "1" in the data segments are counted and stored in the counting segments.

When a search operation starts, number of "1"s in the search data is counted and stored into the counting segment on the left shown in figure 2.1 (a). The actual data is stored in separate data segment and launched into CAM through complementary search lines (SL and SLbar). During first clock cycle these counting bits are compared first and only those that have the same number of "1"s (e.g., the first, third and the fifth) are turned ON in the second sensing stage for further comparison. This scheme reduces a significant amount of power required for data comparison, statistically. At the second clock cycle actual data is compared only those with matched counting bits. Finally only one match line (ML3) will be high, the case when both counting bits and data bits are matched. Totally precomputaion CAM has two stages of comparisons.

2.1.2 Parity check Based CAM

The parity check based CAM design is shown in figure. 2.1(b) consisting of the original data segment along with an extra one-bit called parity check bit. The parity bit indicates odd or even number of "1"s in a corresponding data segments. This parity bit is placed directly with the data or word at left side. If the number of "1"s present in a data has odd in number then the parity bit is set else reset. This new architecture has the same interface as the conventional CAM with one extra parity check bit. During the search operation, there is only one single stage comparison operation. Hence, the use of this parity check bit improves the power performance. However, this additional parity check bit 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 condition between the search data and data segment (e.g., ML2), the parity bits of the search data and the stored data in the data segment is the same, thus the overall data returns a match. As a result the corresponding match line will be charge to high potential (V_{DD}). When one mismatch occurs in the data segment (e.g., ML3), numbers of "1"s in the search data and stored data must be different by 1. As a result, the corresponding parity check bits are different. Thus totally we have two mismatches one from the parity check bit and one from the data bits. Thus the corresponding match lines will be discharged to ground potential. Suppose if there are two or more mismatches between search data and stored data (e.g., ML0, and ML4), the parity bits are the same but the overall we have more than one mismatches in the data segments. With more mismatches, we can ignore these cases as they are not crucial cases. The sense amplifier now only have to identify between the mismatch case and the matched cases. The sense amplifier only senses high potential voltage (V_{DD}) from match line and it amplifies it to a full CMOS voltage output. So overall matched result obtained in only one clock cycle. And total comparisons are also less. So compared to precomputation CAM the power consumed by the parity check CAM is less and also speed is improved almost two times.

III. CAM DESIGN

Basically CAM cell performs two operations: Bit storage (as in RAM), bit comparison (as in CAM). Figure 3.1(a) shows a NOR-type CAM cell and the figure 3.1(b) NAND type CAM cell. The bit storage in both cases is an SRAM

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING

National Conference on Advanced Innovation in Engineering and Technology (NCAIET-2015)

Alva's Institute of Engineering and Technology, Moodbidri

Vol. 3,Special Issue 1,April 2015

cell where cross coupled inverters realized the bit-storage nodes D and Dbar. The nMOS transistors and bit lines are used to perform read and write operation in the SRAM storage. In the figure 3.1 (a) and (b) read and write transistor concept is not shown. Although some CAM cell implementations use lower area DRAM cells typically, CAM cells use SRAM storage. The bit comparison, which is logically equivalent to an EX-OR of the stored bit and the search bit is implemented in a somewhat different fashion in the NOR and the NAND cells.

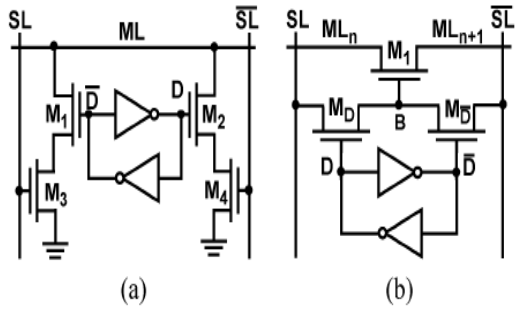


Figure 3.1 CAM core cells for (a) 10-T NOR-type CAM and (b) 9-T NAND-type CAM.[8]

3.1 NOR Cell

Figure 3.1(a) shows the 10-T NOR type CAM cell. In the NOR cell circuit the comparison done between the complementary stored bit, D (and Dbar), and the complementary search data is applied through the complementary search line, SL and SLbar. Using four comparison transistors M1, M2 , M3 and M4, the search data bit is compared with stored data bit D (and Dbar). All four transistors are typically minimum-size to reduce high cell density. These transistors implement the pulldown path of a dynamic EX-NOR logic gate with inputs SL and D. Each pair of transistors, M1/M3 and M2/M4, forms a pulldown paths from the match line (ML) to ground. In the case of mismatch data, that is SL and D activates least one of the pulldown path, connecting ML to ground. A match of data SL and D disables both pulldown paths and disconnecting ML from ground. The NOR nature of such cells are connected in parallel to form a CAM word by shorting the ML of each cell to the ML of adjacent cells. There is a match data word condition on a given ML only if every individual cell in the word has a match, so ML is disconnected from ground. In case of mismatch data, one or more one or more pulldown paths are activated and connects ML to ground.

3.2 NAND Cell

Figure 3.1(b) shows the 9-T NOR type CAM cell. The NAND cell implements the comparison between the stored bit 'D', and input search data bit on the corresponding search lines, SL and SLbar. The comparison done using three comparison transistors M_1, M_D and M_{Dbar}. All three transistors are typically minimum-size to minimize high cell density. The bit-comparison operation of a NAND cell is explained below.

Consider the case of a match when SL=1 and D=1(SLbar=0 and Dbar=0). The Pass transistor M_D is ON and passes the input logic "1" on the SL to node B. Node B is the bit-match node, if there is a match in the cell this node is logic "1". The logic "1" on node B turns ON transistor M_1. The transistor M_1 is also turned ON in the other match case when SL=0 and D=0. In this case, the transistor M_{Dbar} passes logic high to node B. The remaining cases, where SL ≠ D, result in a miss condition, and accordingly node B is logic "0" and the transistor M_1 is OFF. Node B is a pass-transistor implementation of the EX-NOR function of SL and D. The NAND nature of such cells are serially connected form a word. In this case, the ML_n and ML_{n+1} nodes are joined to form a word. A serial nMOS chain of all the M_i transistors forms the pulldown path of a CMOS NAND logic gate. A match condition for the entire word occurs only if every cell in a word is in the match condition.

An important property of the NOR cell is that it provides a full rail voltage at the gates of all comparison transistors. On the other hand, a deficiency of the NAND cell is that it provides only a reduced logic "1" voltage at node B, which can reach only $V\text{-}V_{tn}$ when the search lines are driven to V (where V is the supply voltage and V_{tn} is the nMOS threshold voltage).

3.3 Match-line Structure

In this section implemented CAM match-line structure using NOR cells and NAND cells called NOR match-line and NAND match-line respectively.

3.3.1 NOR Match-line

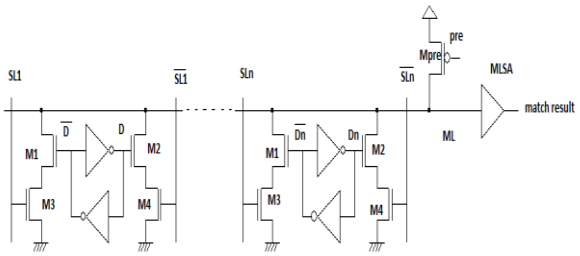


Figure 3.2 Structure of a NOR match-line with n cells.[8]

In figure 3.2 the circuit shows, how NOR cells are connected in parallel to form a NOR match-line. A typical NOR search cycle operates in three phases: search line precharge, match-line precharge, and match-line evaluation. First, the search lines are precharged low to disconnect the match lines from ground by disabling the pulldown paths in each CAM cell. Second, with the pulldown paths disconnected, the transistor M_{pre} precharges the match-lines high. Finally, through search lines (SL and SLbar) the search word (data) is applied into

CAM cells, triggering the match-line evaluation phase. In the case of a match, the ML voltage, V_{ML}, stays high as

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING

National Conference on Advanced Innovation in Engineering and Technology (NCAIET-2015)

Alva's Institute of Engineering and Technology, Moodbidri

Vol. 3,Special Issue 1,April 2015

there is no discharge path to ground. In the case of a miss, at least one pulldown path activated and ML is connected to ground that discharges the match-line. The match-line sense amplifier (MLSA) senses the voltage on ML, and generates a corresponding full-rail output match result.

The main feature of the NOR match-line is its high speed of operation. In the slowest case of a one-bit miss in a word, the critical evaluation path is through the two series transistors in the cell that form the pulldown path. Even in this worst case, NOR-cell evaluation is faster than the NAND case.

3.3.2 NAND Match-line

Figure 3.3 shows the structure of the NAND match-line. A number of cells, n, are connected in series to form the match-line structure.

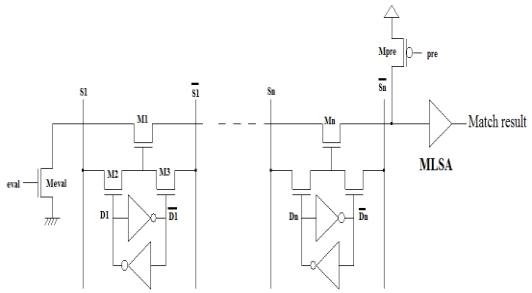


Figure.3.3 NAND match-line structure with precharge and evaluate transistors.[8]

On the right of the figure, the precharge pMOS transistor, M_{pre}, sets the initial voltage of the match-line (ML) to the supply voltage, V. Next, the evaluation nMOS transistor, M_{eval}, turns ON. In the case of a match, all series nMOS transistors M_1 through M_n are ON, that creating a path to ground from the ML node, hence discharging ML to ground. In the case of a mismatch, at least one of the series nMOS transistors, M_1 through Mn, is OFF, so disconnecting the ML from ground, leaving the ML voltage high. A sense amplifier, MLSA, detects the difference between the match (low) voltage and the mismatch (high) voltage.

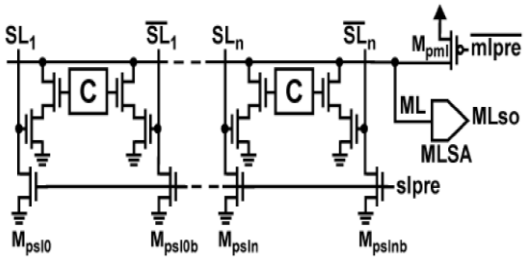
3.4 Matchline Sensing Schemes

This section reviews matchline sensing schemes that generate the match and miss-match result and also signal timing diagram and its signal transitions.

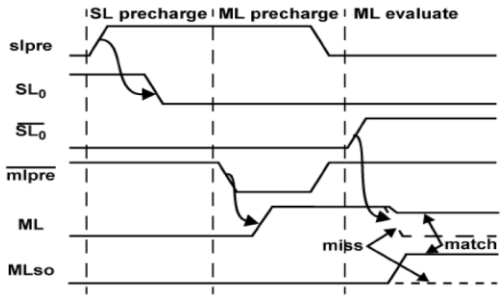
3.4.1 Basic operation

In NOR matchline structure the data search operation done in three phases: search line precharge, match-line precharge, and match-line evaluation. The basic scheme for sensing the state of the NOR matchline is first to precharge high the matchline and then evaluate the pulldown NOR cells by applying the search data through complementary search lines (SLand SLbar) shown in figure 3.4 (a). The matchline (ML) is pulldown to ground potential in the case of a miss, or at high potential (V_{DD}) in the case of a match. Figure 3.4(a) shows, in schematic form, an implementation of this matchline-sensing scheme. Figure 3.4(b) shows the signal timing which is divided into three phases: SL precharge, ML precharge, and ML evaluation.

The operation begins by asserting slpre to precharge the searchlines low, disconnecting all the pull down paths in the NOR cells. With the pull down paths disconnected, the operation continues by (mlpre) asserting to precharge the matchline high. Once the matchline is high, both slpre and (mlpre) are de-asserted. The ML evaluate phase begins by placing the search word on the searchlines.



(a)



(b)

Figure 3.4 (a) The schematic with precharge circuitry for matchline sensing using the precharge-high scheme, and (b) the corresponding timing diagram and signal transitions.[8]

If there is at least one single-bit miss on the matchline, a path (or multiple paths) to ground will discharge the matchline (ML), indicating a miss for the entire word, which is output on the MLSA sense-output node, called MLso. If the stored data and search data is matched then the matchline will remain high indicating a match for the entire bits shown in signal timing diagram figure 3.4 (b).

3.4.2 Matchline Delay

Using the simple matchline model, we can find the time required to precharge and evaluate the matchline. The time to precharge the matchline (which we define as the 10% to 90% rise time of ML) through the precharge device M_{pml} of Figure 3.5(a) is given by,

$t_{MLpre} = 2.2\tau_{MLpre}$

$= 2.2R_{eqpre}C_{ML}$

Where, R_{eqpre} is the equivalent resistance of the precharge transistor.

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING

National Conference on Advanced Innovation in Engineering and Technology (NCAIET-2015)

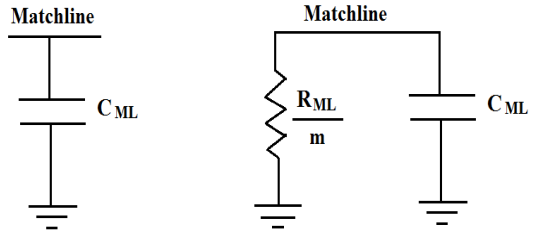
Alva's Institute of Engineering and Technology, Moodbidri

Vol. 3,Special Issue 1,April 2015

The time to evaluate the matchline depends on the matchline capacitance and the matchline pulldown resistance. The worst case matchline pulldown resistance occurs when only a single bit misses, activating only a single pulldown path. Referring to figure 3.5(b), for a single miss, m=1, and the time for the evaluation, which we define as the time for the matchline to fall to 50% of the precharge voltage, is given by

$t_{MLeval} = 0.69\tau_{ML}$

$= 0.69R_{ML}\, C_{ML}$



(a) (b)

Figure 3.5 Matchline circuit model for (a) the match state and (b) the miss state.[2]

IV. RESULT AND PERFORMANCE ANALYSIS

In this section, performance of the proposed parity check CAM design will be evaluated with the conventional CAM circuit. The performance analysis is done in backend design. The power consumption is limited by the amount of charge injected to the ML at the beginning of the search. The proposed parity check CAM using complementary search lines circuit consumes lower power consumption than conventional CAM circuit and the delay is reduced almost half compared to precomputation CAM.

4.1 CAM schematic and Simulation result in backend design using DSCH2 and Microwind tool

Using backend analysis, calculated both delay, area and power consumed by both types of CAM cells. Backend design and result analysis done for data has single bit and its corresponding counting bit and parity check bit combination shown in tables 4.1 and 4.2 below.

Table 4.1 Precomputation CAM bit combinations and its result.

CAM type	Stored data bit	Stored counting bit	Search data bit	Search counting bit	Result
Precomputation	1	1	1	1	match
	1	1	0	0	Miss-match

Table 4.2 Parity check CAM bit combinations and its result

CAM type	Stored data bit	Stored counting bit	Search data bit	Search parity check bit	Result
Parity check	1	1	1	1	match
	1	1	0	0	Miss-match

4.1.1 Simulation result of Precomputation CAM

The figure 4.1 shows the simulation result of precomputation CAM in CMOS 0.12µm technology for matched data bit and its counting bit.

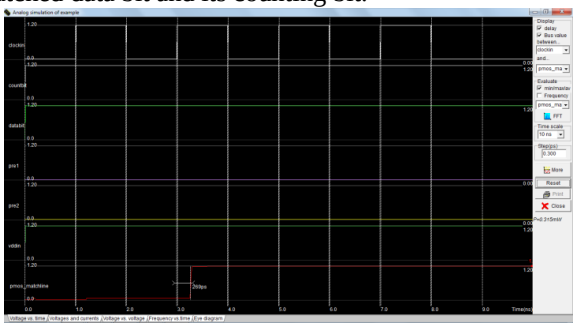


Figure 4.1Simulation result of precomputation CAM in CMOS 0.12µm technology for matched data.

The figure 4.2 shows the simulation result of precomputation CAM in CMOS 0.12µm technology for miss-matched data bit and its counting bit.

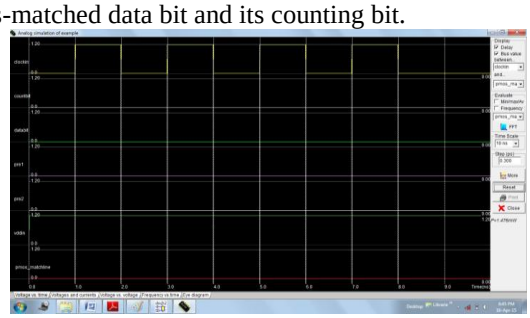


Figure 4.2 Simulation result of precomputation CAM in CMOS 0.12µm technology for miss-matched data.

4.1.2 Simulation result of parity check CAM

The figure 4.3 shows the simulation result of parity check CAM in CMOS 0.12µm. technology for matched data bit and its parity check bit.

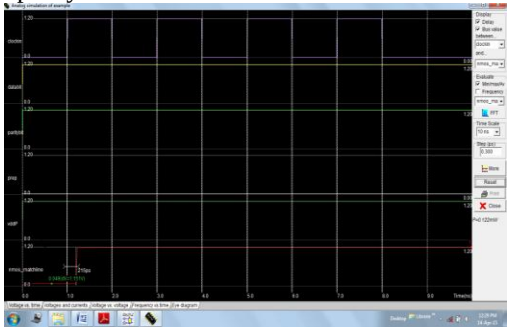


Figure 4.3 Simulation result of parity check CAM in CMOS 0.12µm technology for matched data.

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING

National Conference on Advanced Innovation in Engineering and Technology (NCAIET-2015)

Alva's Institute of Engineering and Technology, Moodbidri

Vol. 3,Special Issue 1,April 2015

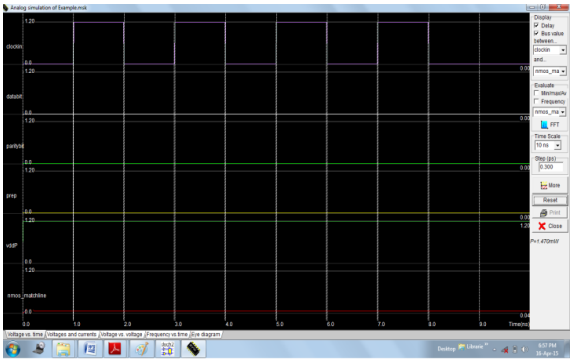


Figure 4.4 Simulation result of parity check CAM in CMOS 0.12µm technology for miss-matched data.

Table 4.3 shows the power and delay analysis of parity check CAM and proposed precomputation CAM in CMOS 0.12µm technology.

CAM type	Technology	Power	Delay
Precomputation	CMOS 0.12µm	0.315mW	3.269 ns
Parity check	CMOS 0.12µm	0.122mW	1.215 ns

V. APPLICATIONS

Content Addressable Memories (CAMs) play an important role in many modem digital systems. CAMs are widely used wherever fast parallel search operations are required. Some examples of CAMs found on modem processors are translation-look aside buffers (TLBs), branch prediction buffers, branch target buffers and cache tags. CAMs have also been used in such applications as database accelerators and network routers. They can also be used in a variety of applications requiring pattern matching operations on bits, such as virtual memory, data compression, and caching and table lookup applications.

VI. CONCLUSION

The parity check parallel data search CAM offer several major advantages over other memory systems. It performs parallel compare operations between input data and stored data segments and match result obtained in only one clock cycle. The parity bit scheme boosts the driving strength of the 1-mismatch case and reduces the sensing delay of the CAM.

As we know in the precomputaion CAM, the overall compare operations done in two clock pulses and the match result obtained at second clock pulse. Compare to precomputation CAM, the parity check CAM gives the match result at first clock pulse so it reduces average power consumption around 50% and increases data search speed almost two times more that reduces the dealy round 50%.

In the parity check CAM, the parity bit corresponding to each data segment represented by single bit. But in precomputation CAM, if the data size increases the corresponding counting bits also increases. So conclude that design of parity check CAM required less silicon area compared to other conventional pre-computation CAM architecture.

REFERENCES

[1]. K. Pagiamtzis and A. Sheikholeslami, "Content addressable memory (CAM) and architectures: A tutorial and survey," IEEE J. Solid- State Circuits, vol. 41, no. 3, pp. 12–727, Mar. 2006.

[2]. A. T. Do, S. S. Chen, Z. H. Kong, and K. S. Yeo, "A low-power CAM with efficient power and delay trade-off," in Proc. IEEE Int. Symp. Circuits Syst.(ISCAS),2011, pp. 2573–2576.

[3]. I. Arsovski and A. Sheikholeslami, "A mismatch-dependent power allocation technique for matchline sensing in content-addressable memories," IEEE J. Solid-State Circuits, vol. 38, no. 11, pp. 1958–1966, Nov. 2003.

[4]. N. Mohan and M. Sachdev, "Low-leakage storage cells for ternary content addressable memories," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 17, no. 5, pp. 604–612, May 2009.

[5]. O. Tyshchenko and A. Sheikholeslami, "Match sensing using matchline stability in content addressable memories (CAM)," IEEE J. Solid- State Circuits, vol. 43, no. 9, pp. 1972–1981, Sep. 2008.

[6]. N. Mohan, W. Fung, D. Wright, and M. Sachdev, "A low-power ternary CAM with positive-feedback match-line sense amplifiers," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 56, no. 3, pp. 566–573, Mar. 2009.

[7]. S. Baeg, "Low-power ternary content-addressable memory design using a segmented match line," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 55, no. 6, pp. 1485–1494, Jul. 2008.

[8]. K. Pagiamtzis and A. Sheikholeslami, "A low-power content-addressable memory (CAM) using pipelined hierarchical search scheme," IEEE J. Solid-State Circuits, vol. 39, no. 9, pp. 1512–1519, Sep. 2004.