International Journal of Advanced Research Methodology in Engineering & Technology,
Special Issue, March 2017, ISBN 978-1-63535-324-2

An Energy Efficient, High Speed Content Addressable Memory Using MSML Scheme

N.Ragavi
Student, Department of ECE,
AAA College of Engineering and Technology,
Sivakasi.

B.Vijayalakshmi
Student, Department of ECE,
AAA College of Engineering and Technology,
Sivakasi.

R.Renita Rexy
Assistant Professor, Department of ECE,
AAA College of Engineering and Technology,
Sivakasi.

G.Mani Sankar
Assistant Professor, Department of ECE,
Renganayagi Varadharaj College of Engineering,
Sivakasi.

***Abstract*— Content Addressable Memory is a storage unit built on hardware. It is majorly used in internet routers to access fast look up tables. This paper implemented the content addressable memory using Master-Slave Match Line architecture. In the Master–Slave configuration, the CAM word is divided into number of segments. Each segment is provided with an additional slave match lines along with one Master Match Line. Using MSML, the power consumption across the Match Line is reduced. Finally, 128x8 CAM memory is implemented. The CAM array is tested for different possible cases. When the data is given to the CAM array, it is searched simultaneously in the entire memory. The simulation results show that the MSML design with the best configuration can reduce the ML energy consumption by range 7%–57%, which increases with the word size. In addition, we further propose a modified CAM cell to facilitate the MSML match performance, i.e., MSML hp design, which can even result in 28% and 69% energy-delay product improvement compared with the original MSML and traditional CAM designs in the 128-bit word size case.**

Keywords— Content-addressable memory (CAM), Low-power, Master–slave architecture, Match line (ML).

I. INTRODUCTION

Content addressable memory (CAM) is a storage that is addressed by the content (or data) rather than the memory address. It is widely used in many applications that require fast table lookup. Due to the frequent lookup and the parallel comparison feature where a large amount of transistors and wires are active on each lookup, the power consumption of CAM is usually considerable. In the CAM memory, the match lines (MLs) and search lines (SLs) are the major power consumers. The ML is long wire with large capacitance and every search will cause a large amount of ML switching activities. Thus, the ML power consumption is very large.

Traditionally, there are two ML architectures, i.e., NOR-type ML and NAND-type ML. The NOR-type ML provides the best search performance, but it costs a large ML power consumption. In contrast, the NAND-type ML trades the search performance for low-power feature. From the related work, the ML power consumption can be reduced by several methods, including the ML segmentation, pipelining search scheme, reducing the ML voltage swing and so on. In this paper, we propose a new ML architecture, called master–slave ML (MSML). The key concept of the MSML design is to combine the master–slave architecture and charge sharing technique to reduce the CAM power dissipated in the ML switching.

The features of the MSML design are as follows:

Unlike the conventional design, where only one single ML is used, the MSML design uses one master-ML (MML) and several slave-MLs (SMLs) to perform the search operation.

Instead of discharging the entire MML to 0, only the mismatched SMLs would draw the charge from the MML, and then be discharged. The charge loss is minimized.

Because we only refill the MML by the charge distributed to the mismatched SMLs, which is much less than the entire ML charge refill in the conventional design, the ML power consumption can be reduced effectively.

Theoretically, the MSML can reduce ML power by 50% in the worst case. In other words, 50% power saving is guaranteed, which is independent of the match case. Of course, this optimal value occurs in the CAM memory with large word size due to the MSML overhead.

In the high performance version, i.e., $MSML_{hp}$, we further modify the CAM cell to facilitate the MSML design to speed up the charge sharing process for better performance and energy-delay product (EDP). Compared with the conventional CAM design, the original MSML design can effectively reduce the ML power consumption.

II. CONTENT ADDRESSABLE MEMORY

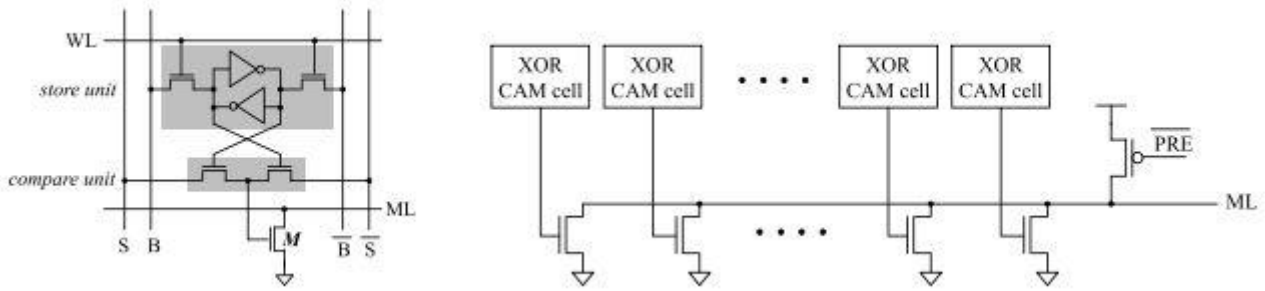

Fig.1 (a) Typical XOR CAM cell. (b) Conventional NOR-type ML.

The CAM consists mainly of the CAM cells. Fig.1 (a) shows a typical XOR CAM cell that consists of two parts: 1) one for storing data called store unit; and 2) the other for comparing data referred to as compare unit. The store unit is usually implemented as the traditional 6T SRAM cell that contains a cross coupled inverter pair. The compare unit is a pass-transistor logic (PTL) for comparing the stored with search data. Depending on the different applications, the NOR compare unit can be modified as XNOR logic. Besides the store and compare units, a pull-down transistor *M*, which is gate-controlled by the comparison result, is necessary to connect/disconnect the ML to/from the ground.

Conventional NOR-Type CAM

The conventional NOR-type CAM design is shown in Fig. 1(a) , in which the CAM cell is XOR-type, and the pull-down transistors of each CAM cell are arranged in NOR type. There are two phases in a search operation, i.e., precharge phase and evaluation phase. During the precharge phase, precharge the ML to high PRE=1. Then, PRE is pulled down to 0 to start the evaluation phase. For a CAM word, if one or more cells are mismatched, the ML would be discharged to 0.

Fig. 1(b) shows the conventional NOR-type CAM design, in which the CAM cell is XOR-type, and the pull-down transistors of each CAM cell are arranged in NOR type. There are two phases in a search operation, i.e., precharge phase and evaluation phase. During the precharge phase, PRE = 1 will precharge the ML to high. Then, PRE is pulled down to 0 to start the evaluation phase. For a CAM word, if one or more cells are mismatched, the ML would be discharged to 0.

Only when all cells are matched, i.e., the search data is identical to the stored data, the ML can retain logic high as in the precharge phase. Because the pull-down path is very short, in case of a mismatch the ML is discharged to 0 quickly. Thus, the NOR-type CAM provides the best search performance. Note that the pull-down transistors arranged in NOR type is beneficial for search performance, but they contribute a lot of drain capacitances to the ML. Because in many applications most of the CAM words are mismatched, a large number of ML switching would consume a huge dynamic power. For example, in the CAM tag used in the translation look-aside buffer or cache memory, at most one word is matched on each lookup, which implies that almost all the MLs would be discharged to 0, and then be charged to high before the next search. Consequently, the NOR-type CAM is power inefficient, although it can provide the best performance.

In contrast to the NOR-type CAM, the NAND-type CAM aims to reduce the power dissipated in search operation, where the pull-down transistors of each CAM cell in the same word are arranged in NOR type. The ML is initially precharged to high, and discharged to 0 only when all CAM cells are matched. Because the load capacitance of ML is small and only a few MLs are discharged to 0 during a search, the power consumption is minimal. However, the pull-down path is too long, such that the ML discharge is very slow in case of a match. Thus, the NOR-type CAM trades the poor performance for a large power saving.

III. MSML DESIGN

A. Overview

The key idea behind our design is to combine the master–slave architecture with the charge refill minimization technique to reduce the ML switching power. Fig. 4 shows a MSML design example, MS_2, which consists of one MML and two SMLs. Unlike the conventional CAM design which uses a single ML, our design uses both MML and SML to perform the search operation. By sharing the charge between the MML and the SML, we can reduce the MML refill swing effectively, such that the search power dissipated in the MMLs can be largely reduced. From Fig. 4, besides the MML and SML, an additional final-ML (FML) is used to indicate the

match result. Note that the parasitic capacitance of the FML is generally smaller than that of the MML.

B. Search Operation

Similar to the conventional CAM, in our design there are two phases during a search. They are precharge and match evaluation phases, respectively. In the precharge phase, the MML and FML are first precharged to high, and then in the match evaluation phase only the mismatch case will change the logic level of the FML from high to low.

1) Precharge Phase: In this phase, the control signal PRE is high. Thus, the MML and FML are precharged to high, and all SMLs, i.e., SML_1 and SML_2 shown in Fig. 4, are discharged to 0. Because the search data are not available in this phase, all BLs (bit and ~bit) are reset to 0, such that the XOR result is 0 to turn OFF the share transistor ($M_{\mathbf{X}}$). Thus, the charge sharing paths between MML and SML, i.e., **S1** and **S2**, are all disconnected in this phase.

2) Match Evaluation Phase: After the precharge phase, the control signal PRE is pulled down to 0 and the search data have to be loaded on the search lines to start the matching process. This phase is called match evaluation phase. Because we divide a CAM word into two SML segments, i.e., SML_1 and SML_2 as shown in Fig. 2, depending on the match results of each SML there are four possible cases in the match evaluation phase. It is a real match only when both the SML_1 and SML_2 are matched.

Case 1 (Both SML_1 and SML_2 are Match): This is only the match case. In this case, both the charge sharing paths S1 and S2 do not conduct. All ML logics are the same as in the precharge phase, i.e., MML is 1, FML is 1, and both SML_1 and SML_2 are 0.

Case 2 (Either SML_1 or SML_2 is Mismatch): We first assume that SML_1 is mismatch, and SML_2 is match. In the SML_1 segment, because at least one share transistor is turned ON to conduct the charge sharing path **S1**, the MML charge will be distributed to the SML_1. This results in a rise of the SML_1 voltage, while the MML voltage level goes down. After the complete charge sharing, both the MML and SML_1 will finally saturate to the same voltage, i.e., final balance voltage.

Case 3 (Both SML_1 and SML_2 are Mismatch): In this case, both SML_1 and SML_2 segments are mismatch. Because the charge sharing path s1 and s2 are conducted, the MML charge will be distributed to the SML_1 and SML_2. For MS_2 configuration, this is the worst case.

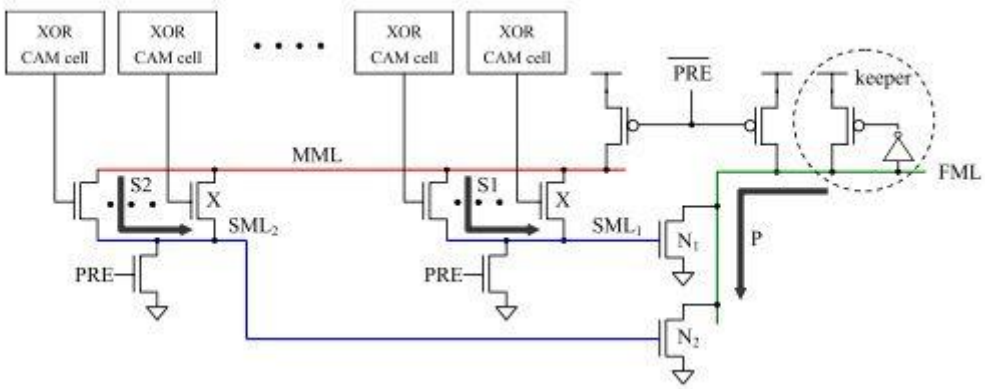

Fig.2. MSML design configured with two SMLs.

Table 1
Key Node Voltage and Path connection/disconnection for each case in the MSML Design.

	SML_1	SML_2	Path			Key Node Voltage				Result
			S1	S2	P	MML	SML_1	SML_2	FML	
Case 1	match	match	X	X	X	V_{DD}	0	0	V_{DD}	match
Case 2	mismatch	match	O	X	O	$\frac{2}{3}V_{DD}$	$\frac{2}{3}V_{DD}$	0	0	mismatch
	match	mismatch	X	O	O	$\frac{2}{3}V_{DD}$	0	$\frac{2}{3}V_{DD}$	0	mismatch
Case 3	mismatch	mismatch	O	O	O	$\frac{1}{2}V_{DD}$	$\frac{1}{2}V_{DD}$	$\frac{1}{2}V_{DD}$	0	mismatch

IV. EXPERIMENTAL RESULTS

A. MSML Power Consumption

Table II shows the final balance voltage for 128-bit MSML design with various configurations under room temperature 27 °C and TT process corner. For a given configuration, from the power saving aspect, the best case is that only one SML is mismatch, i.e., 1-miss, since its balance voltage is highest (or the CRS is smallest). On the other hand, the worst case occurs when all SML segments are mismatch. As discussed previously, the theoretical balance voltage of worst case is 0.5 V, but the real balance voltage is lower than 0.5 V. Due to the increase of circuit overhead, the real balance voltage will decrease as the SML number increases. Fig. 3 shows the ML power consumption for every case. As it is expected that the ML power consumption will increase with the mismatched SML number.

Table 2
Final Balance Viltage.

	1-miss	2-miss	3-miss	4-miss	5-miss	6-miss	7-miss	8-miss
MS_1	0.47V							
MS_2	0.63V	0.47V						
MS_4	0.78V	0.62V	0.53V	0.46V				
MS_8	0.85V	0.75V	0.67V	0.61V	0.56V	0.52V	0.48V	0.45V

B. MSML Performance

In this paper, the metric used to evaluate the CAM performance is the match delay (MD), which is defined as the elapsed time from PRE=0 to the FML discharged to 0 in case of a mismatch. Of course, the load time of search data is included in the MD. The MD will decrease as the number of mismatched SML increases.

The worst case is that only one SML is mismatch, where only one pull-down transistor (N_x) is turned ON to discharge the FML. On the other hand, the best case is that all SML segments are mismatch, since the FML is discharged through all pull-down paths.

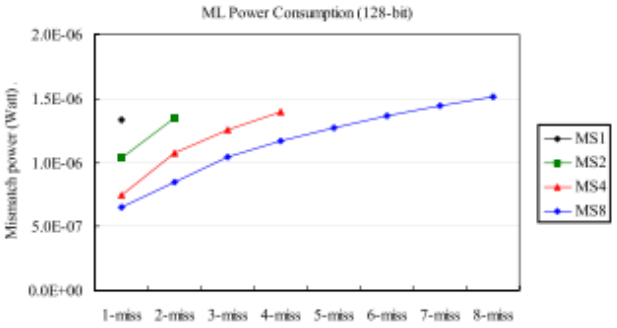

Fig.3. ML Power Consumption.

Because the charge sharing between the large MML and small SML is fast, the small SML size is beneficial to the performance. Therefore, the MSML performance can be improved by increasing the number of SML segment.

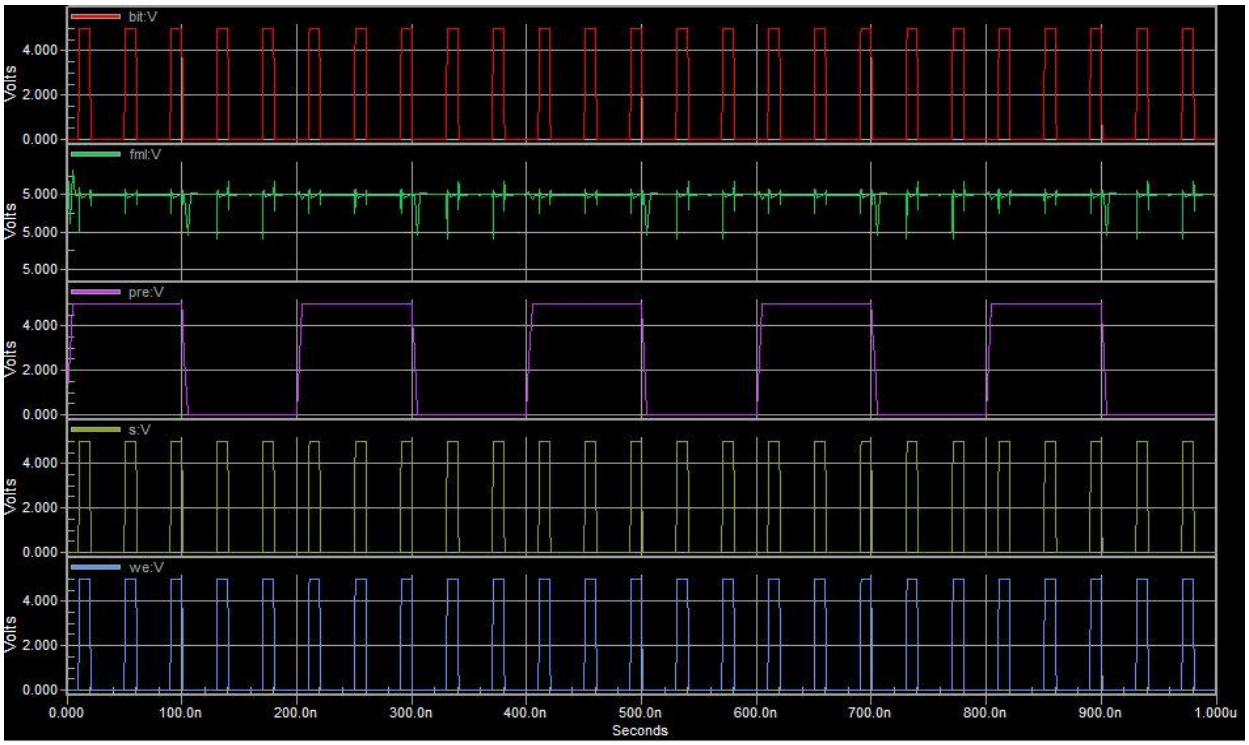

Fig.4. Simulation Results of MSML Design.

V. CONCLUSION

This paper introduces a low-power ML design, called MSML design, in which we combine the master–slave architecture with the charge refill minimization technique to reduce the CAM ML power consumption. The TANNER simulation results shows that the proposed MSML design is suitable to the cases with large word size (64-bit or 128-bit) rather than the cases with small word size (32-bit). By minimizing the MML charge loss, the MSML design can largely reduce the ML energy consumption. Unlike the most related work, where the power saving depends on the occurrence of best case, in the MSML design at least 50% ML power saving is guaranteed theoretically. This feature makes the MSML design more attractive than other related work.

REPERENCES

[1] G. Kasai, Y. Takarabe, K. Furumi, and M. Yoneda, "200 MHz/200 MSPS 3.2 W at 1.5 V Vdd, 9.4 Mbits ternary CAM with new charge injection match detect circuits and bank selection scheme," in *Proc. IEEE Custom Integr. Circuits Conf.*, Sep. 2003, pp. 387–390.

[2] I. Arsovski, T. Chandler, and A. Sheikholeslami, "A ternary content- addressable memory (TCAM) based on 4T static storage and including a current-race sensing scheme," *IEEE J. Solid-State Circuits*, vol. 38, no. 1, pp. 155–158, Jan. 2003.

[3] 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.

[4] 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.

[5] N. Mohan and M. Sachdev, "Low-capacitance and charge-shared match lines for low-energy high-performance TCAMs," *IEEE J. Solid-State Circuits*, vol. 42, no. 9, pp. 2054–2060, Sep. 2007.

[6] J. Zhang, Y. Ye, and B. Liu, "A new mismatch-dependent low power technique with shadow match-line voltage-detecting scheme for CAMs," in *Proc. ACM/IEEE Int. Symp. Low Power Electron. Design*, Oct. 2006, pp. 135–138.

[7] J. Zhang, Y. Ye, and B. Liu, "A current-recycling technique for shadow-match-line sensing in content-addressable memories," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 16, no. 6, pp. 677–682, Jun. 2008.

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

[9] J.-S. Wang, C.-C. Wang, and C. Yeh, "TCAM for IP-address lookupusing tree-style AND-type match lines and segmented search lines," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 2006, pp. 577–586.

[10] P.-T. Huang, S.-W. Chang, W.-Y. Liu, and W. Hwang, "'Green' micro-architecture and circuit co-design for ternary content addressable memory," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2008, pp. 3322–3325.

[11] 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.

[12] P.-T. Huang, S.-W. Chang, W.-Y. Liu, and W. Hwang, "A 256×128 energy-efficient TCAM with novel low power schemes," in *Proc. IEEE Int. Symp. VLSI Design, Autom., Test*, Apr. 2007, pp. 28–31.

[13] J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits: A Design Perspective*, 2nd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 2003.

[14] P. Drennan, K. Breen, J. Dyck, and A. Gupta, *Variation-Aware Design of Custom Integrated Circuits: A Hands-on Field Guide*. New York, NY, USA: Springer-Verlag, 2012.