International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 13
September 2016

Algorithm and Architecture for 16*8 parallel pipeline using cam

Sumayya

M.Tech, VLSI

Sahasra College of Engineering for Women, Warangal

Shirisha

Assistant Professor, ECE

Sahasra College of Engineering for Women, Warangal

Abstract—We propose a low-power content-addressablememory (16*8-CAM) employing a new algorithm for associativitybetween the input tag and the corresponding address of theoutput data. The proposed architecture is based on a recentlydeveloped sparse clustered network using binary connections thaton-average eliminates most of the parallel comparisons performedduring a search. Therefore, the dynamic energy consumptionof the proposed design is significantly lower comparedwith that of a conventional low-power 16*8- CAM design. Given aninput tag, the proposed architecture computes a few possibilitiesfor the location of the matched tag and performs the comparisonson them to locate a single valid match.

INTRODUCTION

A CONTENT-addressable memory (CAM) is a type of memory that can be accessed using its contents rather than an explicit address. In order to access a particular entry in such memories, a search data word is compared against previously stored entries in parallel to find a match. Each stored entry is associated with a tag that is used in the comparison process. Once a search data word is applied to the input of a CAM, the matching data word is retrieved within a single clock cycle if it exists. This prominent feature makes CAM a promising candidate for applications where frequent and fast look-up operations are required, such as in translation look-aside buffers (TLBs) [1], [2], network routers [3], [4], database accelerators, image processing, parametric curve extraction [5], Hough transformation [6], Huffman coding/ decoding [7], virus detection [8] Lempel–Ziv compression [9], and image coding [10]. Due to the frequent and parallel search operations, CAMs consume a significant amount of energy. CAM architectures typically use highly capacitive search lines (SLs) causing them not to be energy efficient when scaled. For example, this power inefficiency has constrained TLBs to be limited to no more than 512 entries in current processors. In Hitachi SH-3 and StrongARM embedded processors, the fully associative TLBs consume about 15% and 17% of the total chip power, respectively [11]–[13]. Consequently, the main research objective has been focused on reducing the energy consumption without compromising the throughput. Energy saving opportunities have been discovered by employing either circuit-level techniques [14], [15], architectural-level [16], [17] techniques, or the codesign of the two, [18], some of which have been surveyed in [19]. Although dynamic CMOS circuit techniques can result in low-power and low-cost CAMs, these designs can suffer from low noise margins, charge sharing, and other problems [16]. A new family of associative memories based on sparse clustered networks (SCNs) has been recently introduced [20], [21], and implemented using field-programmable gate arrays (FPGAs) [22]–[24]. Such memories make it possible to store many short messages instead of few long ones as in the conventional Hopfield networks [25] with significantly lower level of computational complexity. Furthermore, a significant improvement is achieved in terms of the number of information bits stored per memory bit (efficiency). In this paper, a variation of this approach and a corresponding architecture are introduced to construct a classifier that can be trained with the association between a small portion of the input tags and the corresponding addresses of the output data. The term CAM refers to binary CAM (BCAM) throughout this paper. Originally included in [26], preliminary results were introduced for an architecture with particular parameters conditioned on uniform distribution of the input patterns. In this paper, an extended version is presented that elaborates the effect of the design's degrees of freedom, and the effect of no uniformity of the input patterns on energy consumption and the performance. The proposed architecture (SCN-CAM) consists of an SCN-based classifier coupled to a CAM-array. The CAM-array is divided into several equally sized sub-blocks, which can beactivated independently. For a previously trained network and given an input tag, the classifier only uses a small portion of the tag and predicts very few sub-blocks of the CAM to be activated. Once

International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 13
September 2016

the sub-blocks are activated, the tag is compared against the few entries in them while keepingthe rest deactivated and thus lowers the dynamic energy dissipation.

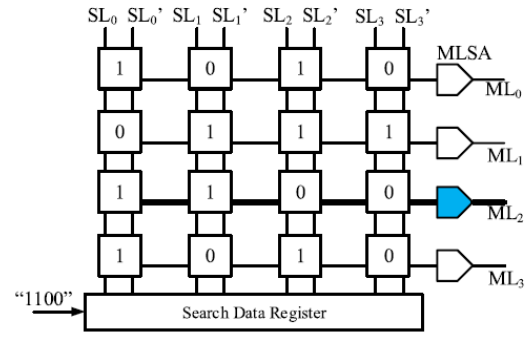

Fig. 1. Simple example of a 4×4 CAM array consisting of the CAM cells, MLs, sense amplifiers, and differential SLs.

We extend this in to 16*8 cam as shown in below fig 2

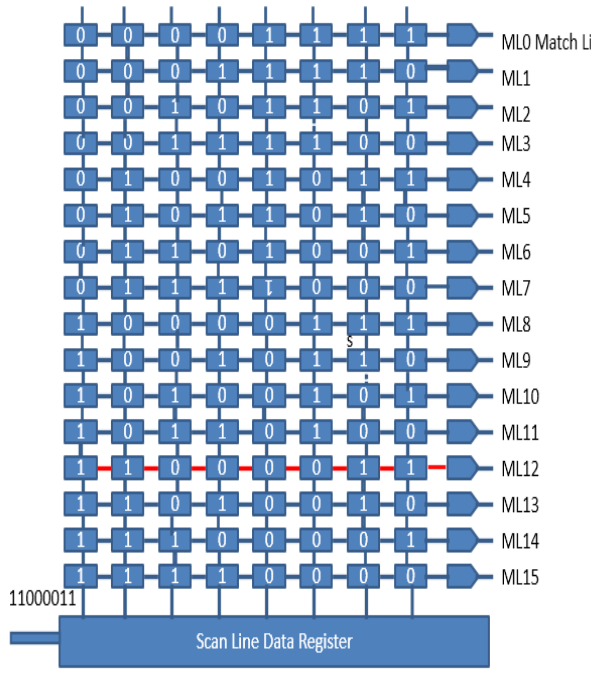

Fig. 2. Simple example of a 16*8 CAM

The 16*8 CAM contains 0s and 1s binary data it stores the address in each cam cell. SL(scan line) It take 8 bits of input data and search the data in each cam cell if data is match with cam address the data is indicate by the ML(match line). The input data is matched with addressable cam data at ML12 line. The process is done by parallel pipelining.This

below tabular form gives the 16*8 cam T.T.

0	0	0	0	1	1	1	1
0	0	0	1	1	1	1	0
0	0	1	0	1	1	0	1
0	0	1	1	1	1	0	0
0	1	0	0	1	0	1	1
0	1	0	1	1	0	1	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	0	0
1	0	0	0	0	1	1	1
1	0	0	1	0	1	1	0
1	0	1	0	0	1	0	1
1	0	1	1	0	1	0	0
1	1	0	0	0	0	1	1
1	1	0	1	0	0	1	0
1	1	1	0	0	0	0	1
1	1	1	1	0	0	0	0

Table 16*8 cam using binary numbers

The proposed architecture consists of a neural-networkbased classifier coupled to a CAM array. The CAM array is divided into several equally-sized sub-blocks which can be activated independently. For a previously trained network and given an input tag, the classifier only uses a small portion of it and will predict, on average, only two out of several sub-blocks of the CAM to be activated. If the number of sub blocks is equal to the number of entries in the CAM, only two CAM entries should be compared to find the match with the cost of higher hardware complexity. Once the sub-blocks are activated, the tag is compared against the few entries in them while keeping the rest deactivated. The total number of sub-blocks can be designed depending on the silicon area availability since each sub-block will slightly increase the silicon area. If the input data word is not uniformly distributed, more sub-blocks will be activated during a search and the accuracy of the final output is not affected. However, since the full-length of the tag is not used in the proposed architecture, it is possible to select the reduced-length tag bits depending on the application and according to a pattern to reduce the tag correlation.

The precomputation-based CAM (PB-CAM) architecture(also known as one's count) was introduced in [16].PB-CAM divides the

comparison process and the circuitryinto two stages. First, it counts the number of ones in aninput and then compares the result with that of the entriesusing an additional CAM circuit that has the number of ones
in the CAM-data previously stored. This activates a few MLs(Match line) and deactivates the others. In the second stage, a modifiedCAM hierarchy is used, which has reduced complexity, andhas only one pull-down path instead of two compared with theconventional design. The modified architecture only considers0 mismatches instead of full comparison since the 1s havealready been compared. The number of comparisons can bereduced to $M \times \_\log(N+2)\_+(M \times N)/(N+1)$ bits, where$M$ is the number of entries in the CAM and N is the number ofbits per entry. In the proposed design, we demonstrate how it ispossible to reduce the number of comparisons to only N bits.Furthermore, in PB-CAM, the increase of the tag length affectsthe energy consumption, the delay, and also complicates theprecomputation stage.

As shown in Fig. 2 the proposed architecture consists of a clustered-neural-network (CNN) connected to a modified CAM array. The CNN is at first trained with the association between a reduced-length tag and the address of the data to be later retrieved. The CAM array is based on a conventional architecture but is divided into several sub-blocks that can be compare-enabled independently. Once an input tag is applied to the CNN, it will predict which CAM sub-block(s) need to be compare-enabled and thus saves power by disabling the rest. If the full length of the tag is used, the classifier will be able to always point to a single sub-block. However, training the network with the full length of the tags will affect the hardware complexity of the CNN. If the reduced-length tags are uniformly distributed, on average, only two possibilities are found with the right number of bits of the reduced-length tag. On the other hand, in some cases, this truncation may cause ambiguities in finding the valid match causing more than one possible CAM sub-block to be activated. This effect will not affect the accuracy of the final result but will cost more power.

As shown in Fig. 2, the network consists of two parts: PI and PII . PI corresponds to the input tag and consists of neurons that are grouped into c equally-sized clusters with l neurons each. Each neural value is binary, i.e. it is either activated or not. The processing of an input message can be within either of the two situations: training or decoding. In this paper, either for training or decoding purposes, the input tag is reduced in length to q bits, and then divided into c equally-sized partitions of length κ bits each. Each partition is then mapped into a neuron in its corresponding cluster using a direct binary-to-integer mapping from the tag portion to the index of the neuron to be activated. Thus $l = 2\kappa$. If l is a given parameter, the number of clusters is calculated to be $c = q/\log_2(l)$. It is important to note that there are no connections within the neurons and clusters inside PI .

In order to exploit the prominent feature of the CN-basedassociative memory in the classification of the search data, aconventional CAM array is divided into sufficient number ofcompare-enabled sub-blocks such that: 1) the number of subblocksare not too many to expand the layout and to complicatethe interconnections and 2) the number of sub-blocks shouldnot be too few to be able to exploit to energy-saving opportunitywith the SCN-based classifier. Consequently, the neuronsin PII are grouped and OR as shown in Fig. 7 to constructthe compare-enable signal(s) for the CAM array. Even theconventional CAM arrays need to be divided into multiplesub-blocks since long bit lines and SLs can slow down theread, write, and search operations due to the presence of drain,gate, and wire capacitances.

CONCLUSION

In this paper, a low-power Content Addressable Memory (16*8 CAM) is introduced. The proposed architecture employs a novel associativity mechanism based on a recently developed family of neural-network-based associative memories. This architecture is suitable for low-power applications where frequent and parallel look-up operations are required. The proposed architecture employs a clustered-neural-network module which is connected to several independently-compare-enabled CAM sub-blocks. With optimized lengths of the reducedlength tags, the network will eliminate most of the comparisons given a uniformly random distribution of the reducedlength inputs. Non-uniformity will cost power but will not affect accuracy. Conventional NAND and NOR-type architectures were also implemented for comparison purposes.

REFERENCES

[1] A. Agarwal, S. Hsu, S. Mathew, M. Anders, H. Kaul, F. Sheikh, and R. Krishnamurthy, "A 128x128b high-speed wide-and match-line content addressable memory in 32nm CMOS," in ESSCIRC (ESSCIRC), 2011 Proceedings of the, Sep. 2011, pp. 83–86.
[2] N.-F. Huang, W.-E. Chen, J.-Y. Luo, and J.-M. Chen, "Design of multi-field IPv6 packet classifiers using ternary CAMs," in Global Telecommunications Conference, 2001. GLOBECOM '01. IEEE, vol. 3, 2001, pp. 1877–1881.
[3] N. Onizawa, S. Matsunaga, V. C. Gaudet, and T. Hanyu, "Highthroughput low-energy content-addressable memory based on self-timed overlapped search mechanism," in Proc. International Symposium on Asynchronous Circuits and Systems (ASYNC), May 2012, pp. 41–48.
[4] C.-S. Lin, J.-C. Chang, and B.-D. Liu, "A low-power precomputationbased fully parallel content-addressable memory," IEEE Journal of Solid-State Circuits, vol. 38, no. 4, pp. 654 – 662, Apr. 2003.
[5] S.-J. Ruan, C.-Y. Wu, and J.-Y. Hsieh, "Low power design of precomputation-based content-addressable memory," Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, vol. 16, no. 3, pp. 331–335, Mar. 2008.
[6] P.-T. Huang and W. Hwang, "A 65 nm 0.165 fJ/Bit/Search 256x144 TCAM macro design for IPv6 lookup tables," Solid-State Circuits, IEEE Journal of, vol. 46, no. 2, pp. 507–519, Feb. 2011.
[7] K. Pagiamtzis and A. Sheikholeslami, "Content-addressable memory (CAM) circuits and architectures: a tutorial and survey," Solid-State Circuits, IEEE Journal of, vol. 41, no. 3, pp. 712–727, march 2006.
[8] V. Gripon and C. Berrou, "Sparse neural networks with large learning diversity," Neural Networks, IEEE Transactions on, vol. 22, no. 7, pp. 1087–1096, Jul. 2011.
[9] B. Wei, R. Tarver, J.-S. Kim, and K. Ng, "A single chip Lempel–Zivdata compressor," in *Proc. IEEE ISCAS*, May 1993, pp. 1953–1955.[10] S. Panchanathan and M. Goldberg, "A content-addressable memoryarchitecture for image coding using vector quantization," *IEEE Trans.Signal Process.*, vol. 39, no. 9, pp. 2066–2078, Sep. 1991.
[11] T. Juan, T. Lang, and J. Navarro, "Reducing TLB power requirements,"in *Proc. Int. Symp. Low Power Electron. Des.*, Aug. 1997, pp. 196–201.
[12] Y.-J. Chang and Y.-H. Liao, "Hybrid-type CAM design for both powerand performance efficiency," *IEEE Trans. Very Large Scale Integr. (VLSI)Syst.*, vol. 16, no. 8, pp. 965–974, Aug. 2008.
[13] Z. Lei, H. Xu, D. Ikebuchi, H. Amano, T. Sunata, and M. Namiki,"Reducing instruction TLB's leakage power consumption for embedded processors," in *Proc. Int. Green Comput. Conf.*, Aug. 2010, pp. 477–484.
[14] S.-H. Yang, Y.-J. Huang, and J.-F. Li, "A low-power ternary contentaddressable memory with Pai-Sigma matchlines," *IEEE Trans. VeryLarge Scale Integr. (VLSI) Syst.*, vol. 20, no. 10, pp. 1909–1913,Oct. 2012.
[15] N. Onizawa, S. Matsunaga, V. C. Gaudet, and T. Hanyu, "Highthroughputlow-energy content-addressable memory based on self-timedoverlapped search mechanism," in *Proc. Int. Symp. Asynchron. CircuitsSyst.*, May 2012, pp. 41–48.
[16] C.-S. Lin, J.-C. Chang, and B.-D. Liu, "A low-power precomputationbasedfully parallel content-addressable memory," *IEEE J. Solid-State Circuits*, vol. 38, no. 4, pp. 654–662, Apr. 2003.
[17] S.-J. Ruan, C.-Y. Wu, and J.-Y. Hsieh, "Low power design ofprecomputation-based content-addressable memory," *IEEE Trans. VeryLarge Scale Integr. (VLSI) Syst.*, vol. 16, no. 3, pp. 331–335, Mar. 2008

Authors:

Sumayya pursing M.Tech in VLSI from **Sahasra College of Engineering for Women, Warangal**

Shirisha working as Asst. Professor, Department of ECE in **Sahasra College of Engineering for Women, Warangal**