IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 12, Issue 4, Ver. II (Jul.-Aug. 2017), PP 48-52
www.iosrjournals.org

Implementation of SCN Based Content Addressable Memory

*Shruthi H Shetty[1], Ashwath Rao[1], Megha N[1]
[1](Dept. Electronics & Communication Engineering, Sahyadri College of Engineering& Management, India)
Corresponding Author: Shruthi H Shetty

Abstract: *Most memory devices store and retrieve data by addressing specific memory locations. As a result, this path often becomes the limiting factor for systems that rely on fast memory accesses. CAM is introduced employing a new mechanism for associativity between the input tags and the corresponding address of the output data. The proposed CAM is based on a recently developed sparse clustered network (SCN) that on-average will eliminate most of the parallel comparisons performed during a search. SCN-CAM is suitable for low-power applications, where frequent and parallel look-up operations are required. In this paper the SCN-CAM is implemented in HDL*

Keywords: *Content Addressable Memory, CAM cell, Sparse Clustered Networks*

Date of Submission: 15-07-2017 Date of acceptance: 26-07-2017

I. Introduction

A CAM is a new technology memory device that accelerates any application requiring fast searches. It is a special type of memory structure that can be accessed using its contents rather than an explicit address. Such memory structures are used in diverse applications ranging from branch prediction in a processor to complex pattern recognition; it is particularly well suited to perform any kind of search operations. CAM is also known as associative memory, associative storage or associative array. Unlike standard computer memory (RAM) in which user supplies a memory address and the RAM returns the data word stored at the address, a CAM searches its entire memory to see if that data word is stored anywhere in it. If the data is found, the CAM returns a list of one or more storage addresses where the word was found. In order to access a particular entry in 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), network routers, database accelerators, image processing, parametric curve extraction, Hough transformation, Huffman coding/decoding and image coding. Due to the frequent and parallel search operations, CAMs consume a significant amount of energy. The main research objective has been focused on reducing the energy consumption. A new family of associative memories based on sparse clustered networks (SCNs), reducing the unnecessary parallel computation in CAM array. Such memories make it possible to store many short messages instead of few long ones as in the conventional Hopfield networks with significant lower level of computational complexity. The CAM is divided into binary and Ternary CAM. The CAM throughout in this paper refers to Binary CAM.

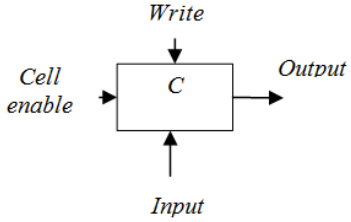

Fig1. Simple CAM Cell

The individual CAM cell is as shown in Fig1. The CAM cell operation is divided as bit storage (usually a RAM memory that uses SRAM cell) and bit comparison is unique operation CAM (that uses XNOR logic). The simple block diagram of CAM cell is as shown in the Fig 2. It consists of a search data register that stores the search data (input data), Match Line Search Amplifiers (MLSAs) and an Encoder. CAM performs three modes of operation Read (R), Write (W) and Search (S) operation. Hence the cell is fed with R, W, S three enable signals respectively. The CAM Read/ Write operation is same as the traditional memory. The Search

operation i.e. it performs bit comparison, it compares the each bit of search data with the data input that is present in the CAM.

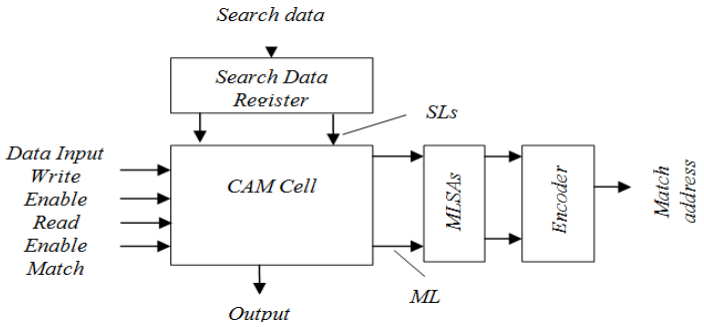

Fig2. Simple Block Diagram of CAM

II. SCN-Cam Overview

2.1 CAM Review

The Fig. 3 shows CAM array consisting of 4 words. There is a match-line corresponding to each word (ML_0, ML_1, etc.) feeding into match line sense amplifiers (MLSAs), and there is a differential search line pair corresponding to each bit of the search word (SL_0, $\overline{SL0}$, SL_1, $\overline{SL1}$, etc.). CAM search operation begins with loading the search-data word into the search-data registers followed by precharging all match lines high, putting them all temporarily in the match state. Next, the search line drivers broadcast the search word onto the differential search lines, and each CAM core cell compares its stored bit against the bit on its corresponding search lines. Match lines on which all bits match remain in the precharged-high state. The MLSA then detects whether its match line has a matching condition or miss condition. Finally, the encoder maps the match line of the matching location to its encoded address.

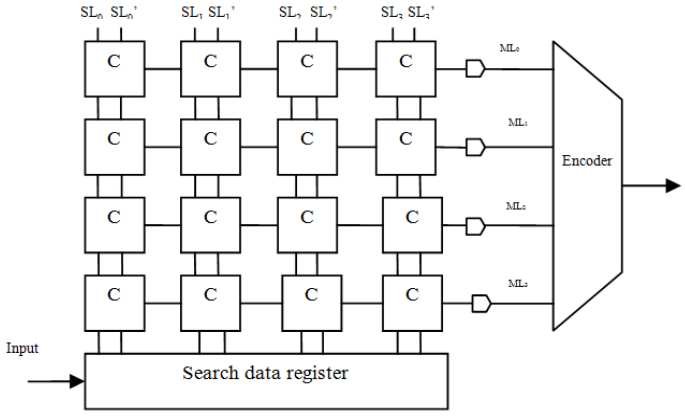

Fig3. Conceptual representation of CAM array

2.2 SCN-CAM

A new family of associative memory based on sparse clustered network is introduced such that these memories make it possible to store many short messages instead of few long ones, with significant lower level of computational complexity. 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 be activated independently. For a 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 the sub-blocks are activated, the tag is compared against the few entries in them while keeping the rest deactivated and thus lowers the dynamic energy dissipation. This reduces the number of comparisons to only one in average, SCN-CAM uses only a portion of the actual tag to create or recover the association with the corresponding output. SCN-based classifier is for either of the two situations: training or decoding. The values are set during the training process, and are stored in a memory module such that they can later be used to retrieve the address of the target data. When an update is requested in SCN-CAM, retraining the entire SCN-based classifier with the entries is not required. The new entry can therefore be added by adding new connections while keeping the previous connections. Once an input tag is presented to the SCN-based classifier, it predicts which CAM sub-block(s) need to be compare-enabled and thus saves the dynamic power by disabling the rest.

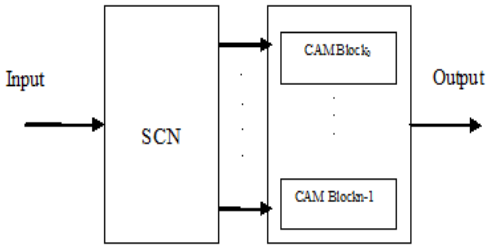

Fig4. Block diagram of SCN-CAM

III. Simulation Results

In this paper the SCN-CAM is implemented using HDL in Xilinx ISE. SCN-CAM employs an SCN-based classifier, which is connected to several independently compare-enabled CAM sub-blocks, some of which are enabled once a tag is presented to the SCN-based classifier. Fig 5 shows the Design flow of system. Once the input is given to SCN classifier, it predicts which CAM block has to be selected from the array. The classifier generates the compare- enable signals to array such that it avoids unnecessary search operations thus reducing the dynamic power.

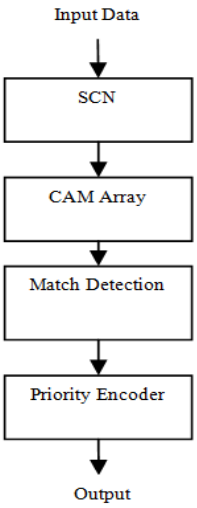

Fig5. Design Flow of SCN-CAM

The content addressable memory compare input search data against stored data and return address of matched data indicating match signal bit high as shown in Fig7. The power and timing delay is also analyzed.

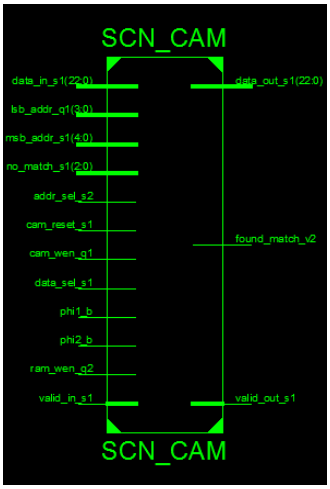

Fig 6. RTL of SCN-CAM

Table 1 Timing summary of SCN-CAM

Timing summary	
Maximum frequency	1290.073MHz
Clock period	0.775ns
Maximum combinational path delay	0.930ns
Delay	0.775ns

Table 2 Power Analysis of SCN-CAM

Device		**On-Chip**	**Power(W)**
Family	Vertex6	Clocks	0.011
Part	Xc6vcx75t	Logic	0.003
Package	ff484	Signal	0.002
Temp Grade	Commercial	IO	0.004
Process	Typical	Leakage	1.291
Speed Grade	-2	Total	1.312

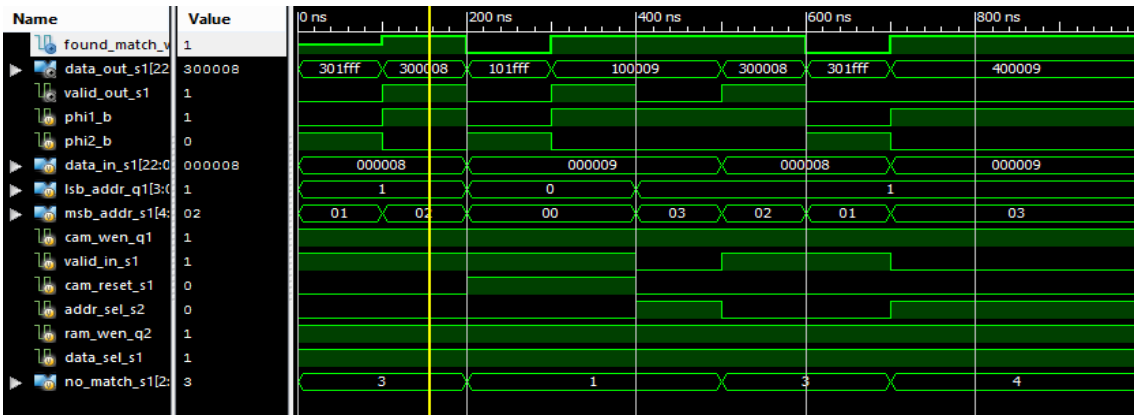

Fig7. SCN-CAM Waveform

IV. Conclusion

For any application that requires a fast memory search, CAM can provide a solution. CAM can be used to speed up any applications ranging from local-area networks, database management, file-storage management, pattern recognition, artificial intelligence, fully associative and processor-specific cache memories, and disk cache memories. Although CAM has many applications, it is particularly well suited to perform any kind of search operations. SCN-CAM employs an SCN-based classifier, which is connected to several independently compare-enabled CAM sub-blocks, some of which are enabled once a tag is presented to the SCN-based classifier. SCN-CAM eliminates most of the comparison operations, thus SCN-CAM is suitable for low-power applications, where frequent and parallel look-up operations are required.

References

[1] Hooman Jarollahi , Vincent Gripon, Naoya Onizawa and Warren J. Gross, "Algorithm and Architecture for a Low-Power Content-Addressable Memory Based on Sparse Clustered Networks" IEEE Transactions On Very Large Scale Integration (VLSI) Systems, Vol. 23, No. 4, April 2015

[2] Dejan Georgiev, "Low Power Concept For Content Addressable Memory (Cam) Chip Design", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering Vol. 2, Issue 7, July 2013

[3] Kostas Pagiamtzis and Ali Sheikholeslami, "Content-Addressable Memory (CAM) Circuits and Architectures: A Tutorial and Survey", IEEE Journal Of Solid-state Circuits, Vol. 41, No. 3, March 2006

[4] Shanq-Jang Ruan, Chi-Yu Wu, and Jui-Yuan Hsieh, "Low Power Design of Precomputation-Based Content-Addressable Memory", IEEE Transactions On Very Large Scale Integration (VLSI) Systems, Vol. 16, No. 3, March 2008

[5] A. Agarwal et al., "A 128×128 b high-speed wide-and match-line content addressable memory in 32 nm CMOS," in Proc. ESSCIRC, pp. 83–86, Sep. 2011.

[6] Y.-J. Chang and M.-F. Lan, "Two new techniques integrated for energy efficient TLB design," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 15, no. 1, pp. 13–23, Jan. 2007.

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

[8] S. Panchanathan and M. Goldberg, "A content-addressable memory architecture for image coding using vector quantization," IEEE Trans. Signal Process., vol. 39, no. 9, pp. 2066–2078, Sep. 1991.
[9] Y.-J. Chang and Y.-H. Liao, "Hybrid-type CAM design for both power and performance efficiency," IEEETrans. Very Large Scale Integr. (VLSI) Syst., vol. 16, no. 8, pp. 965–974, Aug. 2008.
[10] H. Jarollahi, V. Gripon, N. Onizawa, and W. J. Gross, "A low-power content-addressable memory based on clustered-sparse networks," in Proc. 24th IEEE Int. Conf. ASAP, pp. 305–308, Jun. 2013.

IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) is UGC approved Journal with Sl. No. 5016, Journal no. 49082.

Shruthi H Shetty. " Implementation of SCN Based Content Addresable Memory." IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) 12.4 (2017): 48-52.