Computers, Materials & Continua CMC, vol.62, no.3, pp.1097-1107, 2020

High Speed Network Intrusion Detection System (NIDS) Using Low Power Precomputation Based Content Addressable Memory

R. Mythili[1, *] and P. Kalpana[2]

Abstract: NIDS (Network Intrusion Detection Systems) plays a vital role in security threats to computers and networks. With the onset of gigabit networks, hardware-based Intrusion Detection System gains popularity because of its high performance when compared to the software-based NIDS. The software-based system limits parallel execution, which in turn confines the performance of a modern network. This paper presents a signature-based lookup technique using reconfigurable hardware. Content Addressable Memory (CAM) is used as a lookup table architecture to improve the speed instead of search algorithms. To minimize the power and to increase the speed, pre-computation based CAM (PBCAM) can be used, as this technique avoids repeated search comparisons. PBCAM employs the two-stage comparison with a parameter memory in the first stage and data memory in the second stage. Only the matched data in the parameter memory are compared in the data memory. This reduces the number of comparisons, thereby increasing the speed of the system. In this work dual-port RAM-based PBCAM (DP-PBCAM) is used to design a signature-based intrusion detection system. A low power parameter extractor is used with a minimum number of gates for precomputation. The hardware implementation is done using Xilinx Spartan 3E FPGA. The proposed DP-PBCAM lookups support a gigabit-speed of 7.42 Gbps.

Keywords: NIDS, FPGA, dual port RAM, CAM, PBCAM, DP-PBCAM.

1 Introduction

An Intrusion Detection System (IDS) is a vigilant tool designed to find malevolent behavior in an information system. It can be a hardware or software design which automatically alerts the officials during security policy violations. It monitors the system function by checking the integrity of files, investigating the incoming patterns against the known attacks and examining vulnerabilities in the system.

Network Intrusion Detection Systems (NIDS) protect network communication from hackers. NIDS warn the network administrator whenever hacking occurs. Thus, hackers are prevented from obtaining the information. The threat to the organization's network is

[1] KIT-Kalaignarkarunanidhi Institute of Technology, Coimbatore-641402, Tamilnadu, India.

[2] PSG College of Technology, Coimbatore-641004, Tamilnadu, India.

[*] Corresponding Author: R. Mythili. Email: mythili@kitcbe.com.

identified and examined using NIDS. Classification of IDS includes Network Intrusion Detection Systems (NIDS) and Host-based Intrusion Detection Systems (HIDS). HIDS examine specific host-based actions, such as what applications are being used, what files are being accessed and what information resides in the kernel logs. NIDS analyze the flow of information between computers, i.e., network traffic.

Based on detection approach, IDS is classified as signature-based IDS (SIDS) and anomaly-based IDS (AIDS). SIDS works on the principle of matching; an alert is issued when the incoming data is matched with the known attacks. It consists of a statistical model of normal network traffic, which consists of the bandwidth used, the protocols defined for the traffic, the ports and devices which are part of the network. It regularly monitors the network traffic and compares it with the statistical model. In case of any anomaly or discrepancy, the administrator is alerted. Benefits of SIDS include accuracy, low alarm rates and increased speed whereas AIDS does exhaustive monitoring.

When the number of signatures is more in number, analyzing the incoming packets for telltale signatures is time-consuming. In gigabit networks, keyword matching can be used as a part of the intrusion detection problem. Hardware implementation of the keyword match problem is designed using Content Addressable Memory.

Software algorithms are used to perform the search and match operations in IP routing table lookups, intrusion detection, and authentication, directory lookups [Aho (1975); Boyer (1977); Fisk (2001); Anagnostakis (2003)]. Hardware systems are used to increase the search speed [Wade (1989); Motomura (1990); McAuley (1993); Panigrahy (2003)]. CAMs are used in hardware implementations.

Bu et al. [Bu and Chandy (2004)] introduce software-based NIDS using FPGA based CAM. Bu et al. [Bu and Chandy (2004)] demonstrate a keyword match processor using CAM. Shruthi [Shruthi (2014)] proposes FPGA based deep packet inspection engine in NIDS for speeding up the process. Sheela et al. [Sheela, Srinath and Murtaza (2015)] gives an overview of IDS and methodologies used by IDS. Soni et al. [Soni, Kukade and Lawhale (2015)] implemented the CAM based on dual-port RAM. This method improves speed when compared to previous techniques. Komal et al. [Komal and Rao (2016)] designed RAM based TCAM suitable to use in FPGAs. Akshay et al. [Akshay and Gireeshkumar (2016)] implemented a hardware-based IDS using an extended Bloom filter. Razan et al. [Razan, Faezipour and Khaled (2016)] presented the reviews and compare the hardware-based techniques that are commonly used in IDS. Pre-computation is the most efficient strategy used in CAM to increase the speed of comparison.

The pre-computation technique uses parameter memory as a first part and data memory as a second part of CAM memory. Only the matched data in the first part will be compared against the stored data in the second part. This reduces the number of comparisons, thereby increases the speed of operation. The pre-computation technique in CAM reduces the processing time. Arun et al. [Arun and Krishnan (2012)] introduces XOR-based CAM to increase throughput. The proposed work utilizes DP- PBCAM to improve the performance of the system and to prevent hackers from obtaining important information.

The rest of the paper is organized as follows, Section 2 describes the Content Addressable memory. Section 3 explains about the proposed work. Section 4 shows the

hardware implementation of the proposed work. Section 5 deals with the Simulation results and finally section 6 concludes the work.

2 Content addressable memory

Content addressable memories (CAMs) are widely used for search-intensive applications. Compared to algorithmic approaches CAMs are faster. CAM has two units, one for storage, built using SRAM and other is a comparison unit. Fig. 1 shows the schematic of a 4×3 CAM with CAM cells (C), Search lines (SL), Match lines (ML) and Matchline sense amplifiers. CAM compares the search data with the stored data.

The read and write operations performed by the memory devices are in reference to its selected address, whereas in CAM, the read (compare) operation is based on its content rather than its address. Thus, a considerable time saving is achieved during the read operation.

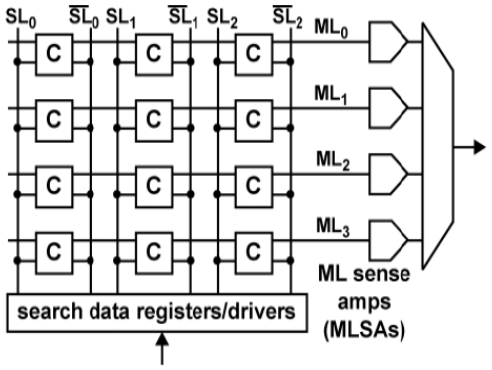

Figure 1: Simple schematic of a CAM

The comparison process in parallel increases the search speed of the system. Since search operation is the power-hungry phase in CAM, Several power reduction techniques are used to reduce the power consumption of the CAM [Pagiamtzis and Sheikholeslami (2004); Yang and Kim (2005); Pagiamtzis and Sheikholeslami (2006); Chang and Liao (2008); Chang and Wu (2014)]. CAM is most commonly used for high-speed memory search. It is also used in networking devices, pattern recognition, fully associative and processor-specific cache memories.

3 Proposed system

The main goal is to design an FPGA based hardware search engine without increasing the detection time and system complexities. Single port and dual port RAM-based PBCAM is used for the efficient matching process. Parallel architecture is used to achieve fast detection and accelerating the search performance.

In the proposed system, the server consists of a set of preloaded good and bad IP addresses in its memory which is the proposed CAM lookup. The server is connected to the computer systems. The outsider sends the message to the server or the host system directly. If at all the outsider sends via the server, the message will reach the host network without any deny. But if the outsider tries to send the message indirectly, that is direct to

the host without passing it to the server, the host network computer forwards the message to the server without reading the message. So that the server detects the IP address as the bad one and discards the message. In that way, the host network of computers will be safe.

In the proposed system pre-computation based CAM is used to enhance the search speed. There are different pre-computation techniques available such as One's count approach, Block-XOR approach, Gate block selection technique and parity bit [Lin, Chang and Liu (2003); Hsieh and Ruan (2009); Do, Chen, Kong et al. (2013)] each technique employs different parameter extractors for pre-computation. The proposed parameter extractor extracts parameter for the given input data utilizing AND gates. Bitwise AND remainder function is shown in Fig. 2, where AND gates are used to perform fast modulo operation. This structure provides LSB's of the given input. This LSB's act as a parameter and gets a hold on within the parameter memory. Throughout data searching, comparison of data bits has been carried out. First stage comparison checks whether the search data's LSB matches with stored data's LSB. If no match is returned in the first stage comparison, it implies that there is a data mismatch with the data stored, else the input search data ought to be compared within the second stage comparison section. Moreover, comparisons made in the first stage have been already filtered out the unequaled data, the second stage solely needs a comparison of the information that matches from the first stage as in standard PBCAM.

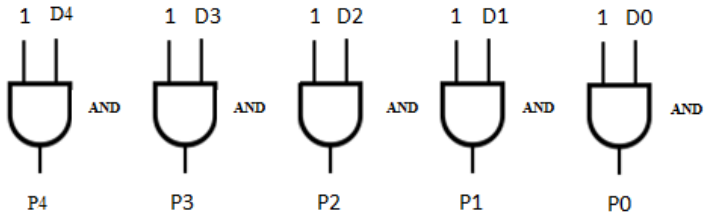

Figure 2: Structure of Bitwise AND remainder function

The number of bits to be compared in the first stage is important, because it determines the number of comparison operations needed in the second stage. It is supposed by the bits of remainder function. The remainder function obtains the modulus (MOD) value of the input data. The divisor of the MOD ought to be chosen in such a way that comparison in the second stage and also the number of bits compared in the first stage ought to be minimized. The fast modulo operation will be performed utilizing bitwise AND operation. The data are bitwise ANDed with (divisor-1) for extracting its parameter.

For example, for 32 bit data word 111111100001111011 11100001111011 (4263442555)

Modulus=numerator && (divisor-1)

4263442555MOD32=4263442555 && (32-1)=27

4263442555 : 111111100001111011 11100001111011 &&

31 : 000000000011111

Answer: 11011 (27) is the last 5 LSBs of input data.

MOD must be chosen in the way that the number of data associated with an equivalent parameter and also the number of bits needed to represent the parameter should be minimum. The parameter extractor of the remainder function PBCAM is better than that of the present PBCAM in terms of power, area and speed. The results are given in Tab.1.

Since Dual-port RAM can read and write different memory cells simultaneously at different addresses, they serve the need for faster devices [Rajeshwari, Geetanjali and Shirakola (2013)]. DP-PBCAM is used to improve the speed of the intrusion detection system.

4 Hardware implementation

The proposed work is implemented using Xilinx Spartan 3E FPGA. The code is written using Verilog. It contains the search algorithm which uses CAM as a lookup table. The size of the lookup is 128×32. The block diagram of the work implemented is shown in Fig. 3.

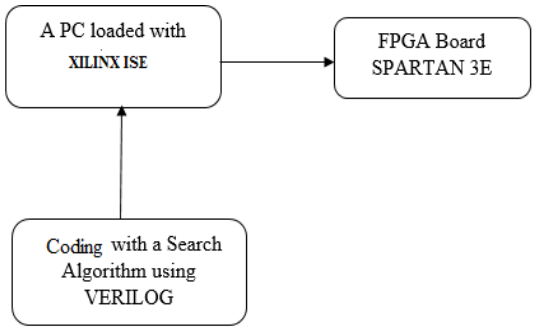

Figure 3: Block diagram of work implemented

The Spartan-3 family is an alternative to mask programmed ASICs. It offers a high-performance logic solution for high-volume consumer-oriented applications. FPGAs avoids the high initial cost, lengthy development cycles, and inherent inflexibility of conventional ASICs. FPGA also programmability permits design upgrades in the field with no hardware replacement necessary, an impossibility with ASICs [Xilinx (2008)].

The Flow chart of work implementation is shown in Fig. 4. It depicts the way; the host network of computers will be safe for applications, including broadband access, home networking, and digital television equipment.

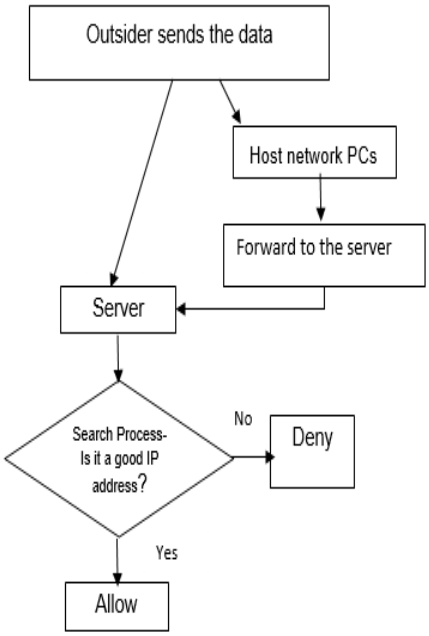

Figure 4: Flow chart of work implementation

CAM performs the search process. It compares the search data with the stored data. The output of CAM leads to two cases, one is to allow case during a match and the other is to deny case during mismatch.

4.1 Single and dual port RAM based CAM-allow case

The Network Intrusion System allows the good IP address, which is matched with an address in the memory. Allow case is illustrated in Fig. 5.

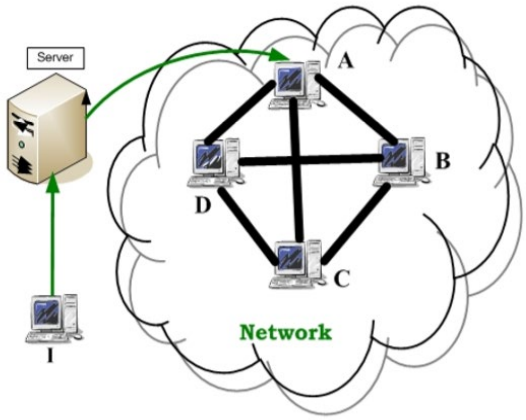

Figure 5: Illustration of the allow case [Martuza, Rima, Mojammel et al. (2009)]

Hardware result of the allow case is shown in Fig. 6. The green light glows showing that there is no intrusion in the system.

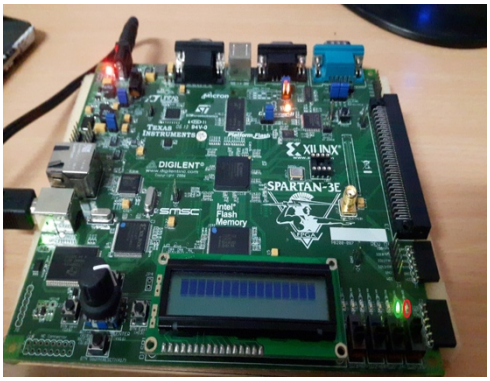

Figure 6: Hardware result of allow case

4.2 Single and dual port RAM based CAM-deny case

The Network Intrusion System denies the Bad IP address, which is mismatched with the address in the memory. Deny case is explained in Fig. 7.

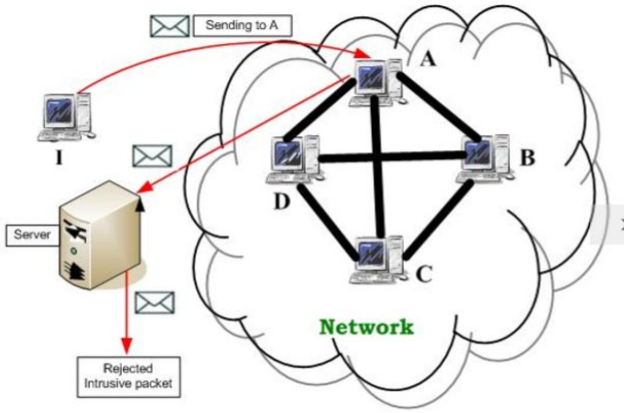

Figure 7: Illustration of the deny case [Martuza, Rima, Mojammel et al. (2009)]

Hardware result of the deny case is shown in Fig. 8. The green light glows next to the LED configured to allow case that there is an intrusion in the system.

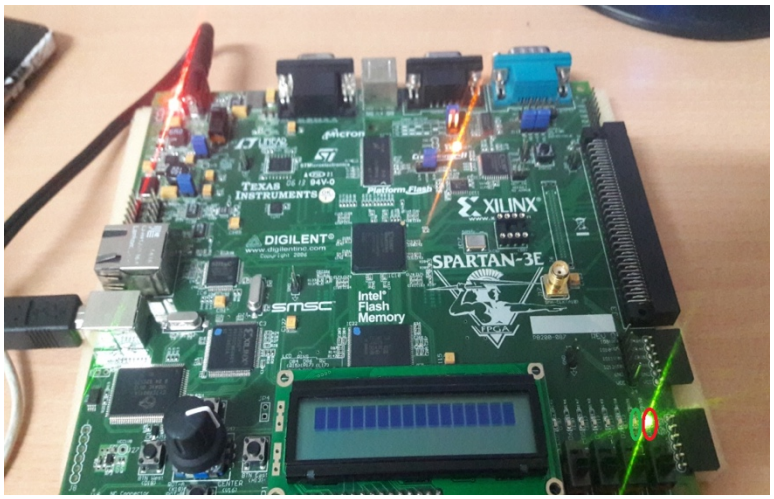

Figure 8: Hardware result of deny case

5 Simulation results

The critical path, area, average power consumption of different types of parameter extractors are compared in Tab. 1. The proposed Remainder Function extractor requires only a minimum number of gates for a 32-bit input data. This concept can be implemented directly without gates.

Table 1: Comparison of parameter extractors

Parameter Extractor	One's Count [Hsieh and Ruan (2009)]	Block-XOR [Hsieh and Ruan (2009)]	Remainder Function
Critical path	9 FA+1 OR	3 XOR	1 AND
Area	42 FA+2 OR	27 XOR	5 AND
Average Power	4.52 mW	0.92 mW	91.1pW

Implementation of pre-computation using the remainder function as a parameter extractor speeds up the search process by minimizing the critical path and power.

The proposed work is implemented using Xilinx. The code is written using Verilog. It contains the precomputation algorithm which uses CAM as a lookup table. Speed of Single Port RAM-based CAM and Dual Port RAM-based PBCAM is given in Tab. 2.

Table 2: Results of single port and dual port RAM based CAM

Type of CAM	Size of Signature (bit)	Clock period (ns)	Throughput (Gbps)
Single Port RAM based CAM	32	6.88	4.65
Dual Port RAM based PBCAM	32	4.31	7.42

From Tab. 2, the throughput of dual port RAM-based CAM is higher than the single port RAM-based CAM. The Tab. 3 shows the comparison of implementation results. Block-XOR (XPCAM) and One's Count (OCCAM) precomputation based CAM is compared against the proposed technique. The proposed DP-PBCAM increases throughput.

Table 3: Implementation results

Design	Size of Signature(bit)	Period (ns)	Throughput (Gbps)
XPCAM [Arun and Krishnan (2012)]	32	5.10	6.27
OCCAM [Arun and Krishnan (2012)]	32	5.80	5.52
Proposed DP-PBCAM	32	4.31	7.42

6 Conclusion

Thus, the proposed work provides an efficient way of detecting the intrusion caused by hackers. The Remainder function pre-computation technique reduces the power and area when compared with the other existing techniques. The simulation results of Single port RAM-based CAM and Dual port RAM-based PBCAM proves that the dual port RAM-based CAM using pre-computation offers better performance. This is suitable to handle intrusion detection on current gigabit networks.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Aho, A. V.; Corasick, M. (1975): Efficient string matching: an aid to bibliographic search. *Communications of the ACM*, vol. 18, no. 6, pp. 333-343.

Akshay, E. J.; Gireeshkumar, T. (2016): Gigabit network intrusion detection system using extended bloom filters in reconfigurable hardware. *Proceedings of the Second International Conference on Computer and Communication Technologies*, pp. 11-19.

Anagnostakis, K.; Antonatos, S.; Markatos, E. P.; Polychronakis, M. (2003): E^2xB: a domain-specific string matching algorithm for intrusion detection. *Proceedings of IFIP International Information Security Conference*, vol. 122, pp. 217-228.

Arun, M.; Krishnan, M. (2012): Functional verification of signature detection architectures for high speed network applications. *International Journal of Automation and Computing*, vol. 9, no. 4, pp. 395-402.

Boyer, R. S.; Moore, J. S. (1977): A fast string searching algorithm. *Communications of the ACM*, vol. 20, no. 10, pp. 762-772.

Bu, L.; Chandy, J. A. (2004): FPGA based network intrusion detection using content addressable memories. *Published in 12th Annual IEEE Symposium on Field-Programmable Custom Computing Machines*.

Bu, L.; Chandy, J. A. (2004): A keyword match processor architecture using content addressable memory. *ACM GLSVLSI'04 Proceedings of the 14th ACM Great Lakes symposium on VLSI*, pp. 372-376.

Chang, Y. J.; Liao, Y. H. (2008): Hybrid-type CAM design for both power and performance efficiency. *IEEE Transactions on Very Large Scale Integration Systems*, vol. 16, no. 8, pp. 965-974.

Chang, Y. J.; Wu, T. C. (2014): Master-slave matchline design for low power content addressable memory. *IEEE Transactions on Very Large Scale Integration Systems*, vol. 23, no. 9, pp. 1740-1749.

Do, A. T.; Chen, S.; Kong, Z. H.; Yeo, K. S. (2013): A high speed, low power CAM with a parity bit and power-gated ML sensing. *IEEE Transactions on Very Large Scale Integration Systems*, vol. 21, no. 1, pp. 51-56.

Fisk, M.; Varghese, G. (2001): Fast content-based packet handling for intrusion detection. (*Technical Report). Department of Computer Science, University of California, San Diego*.

Hsieh, J. Y.; Ruan, S. J. (2009): Synthesis and design of parameter extractors for low power precomputation based content addressable memory. *IEICE Transactions on Electronics*, vol. E92. C, no. 10, pp. 1249-1257.

Komal, R. C.; Rao, M. V. (2016): Low power RAM based hierarchical CAM on FPGA. *International Journal of Engineering and Technology*, vol. 8, no. 2, pp. 863-869.

Lin, C. S.; Chang, J. C.; Liu, B. D. (2003): A low power precomputation based fully parallel content addressable memory. *IEEE Journal of Solid State Circuits*, vol. 38, no. 4, pp. 654-662.

Martuza, A.; Rima, P.; Mojammel, H.; Naser, A.; Bikas, M. et al. (2009): NIDS: a network based approach to intrusion detection and prevention. *International Association of Computer Science and Information Technology-Spring Conference.*

McAuley, A. J.; Francis, P. (1993): Fast routing table lookup using CAMs. *Proceedings of IEEE INFOCOM '93-The Conference on Computer Communications*, vol. 3, pp. 1382-1391.

Motomura, M.; Toyoura, J.; Hirata, K.; Ooka, H.; Yamada, H. et al. (1990): A 1.2-million transistor, 33 MHz, 20-b dictionary search processor (DISP) ULSI with a 160-kb CAM. *IEEE Journal of Solid State Circuits*, vol. 25, no. 5, pp. 1158-1164.

Pagiamtzis, K.; Sheikholeslami, A. (2004): A low power content addressable memory (CAM) using pipelined hierarchical search scheme. *IEEE Journal of Solid State Circuits*, vol. 39, no. 9, pp. 1512-1520.

Pagiamtzis, K.; Sheikholeslami, A. (2006): Content addressable memory circuits and architectures: a tutorial and survey. *IEEE Journal of Solid State Circuits*, vol. 41, no. 3, pp. 712-727.

Panigrahy, R.; Sharma, S. (2003): Sorting and searching using ternary CAMs. *IEEE Computer Society*, vol. 23, no. 1, pp. 44-53.

Rajeshwari, M.; Geetanjali, K.; Shirakola, S. K. (2013): Dual port SRAM. *International Journal of Current Engineering and Technology*, *Proceedings of National Conference on Women in Science & Engineering, SDMCET Dharwad*, pp. 348-352.

Razan, A.; Faezipour, M.; Khaled, M. (2016): Network intrusion detection using hardware techniques: a review. *IEEE Long Island Systems, Applications and Technology Conference*.

Sheela Evangelin Prasad, S. N.; Srinath, M. V.; Murtaza, S. (2015): Intrusion detection systems, tools and techniques-an overview. *Indian Journal of Science and Technology*, vol. 8, no. 35, pp. 1-7.

Shruthi, S. (2014): Design and implementation of a high speed network intrusion detection system. *International Journal of Engineering Research and Technology*, vol. 3, no. 5, pp. 992-994.

Soni, M.; Kukade, S.; Lawhale, P. (2015): FPGA implementation of content addressable memory based information detection system. *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 4, no. 2, pp. 360-362.

Wade, J. P.; Sodini, C. G. (1989): A ternary content addressable search engine. *IEEE Journal of Solid State Circuits*, vol. 24, no. 4, pp. 1003-1013.

Xilinx. Inc. (2008): Spartan-3 generation FPGA user guide, vol. 3.

Yang, B. D.; Kim, L. S. (2005): A Low power CAM using pulsed NAND-NOR matchline and charge recycling search-line driver. *IEEE Journal of Solid State Circuits*, vol. 40, no. 8, pp. 1736-1744.