International Journal of Scientific Research & Engineering Trends
Volume 5, Issue 3, May-Jun-2019, ISSN (Online): 2395-566X

Virus-Detection Processors in Mobile Device Using an Adaptively Dividable Dual Port Bi-TCAM

Asst. Prof. S. Mubeena
Electrical & Electronics Engineering
Mohamed Sathak A.J. College of Engineering
Siruseri, Chennai, Tamil Nadu , India
mubee1504@gmail.com

Asst. Prof. K. Vairaperumal
Electrical & Electronics Engineering
Mohamed Sathak A.J. College of Engineering
Siruseri, Chennai, Tamil Nadu, India
vairaperumal.rvs@gmail.com

Abstract- **Network security for mobile devices is in high demand because of the increasing virus count. Since mobile devices have limited CPU power, dedicated hardware is essential to provide sufficient virus detection performance. A TCAM-based virus-detection unit provides high throughput, but also challenges for low power and low cost. In this paper, an adaptively dividable dual-port BiTCAM (unifying binary and ternary CAMs) is used to achieve a high-throughput, low-power, and low-cost virus-detection processor for mobile devices. The dual-port BiTCAM is realized with the dual-port AND-type match-line scheme which is composed of dual-port dynamic AND gates. The dual-port designs reduce power consumption and increase storage efficiency due to shared storage spaces. In addition, the dividable BiTCAM provides high flexibility for regularly updating the virus-database. The BiTCAM achieves a 48% power reduction and a 40% transistor count reduction compared with the design using a conventional single-port TCAM.**

Keywords – **Associative memories, ClamAV, content-addressable memory, dual-port match line, high speed, low power, pattern matching, PF-CDPD, pseudo-footless, Snort, virus detection**

I. INTRODUCTION

Content-Addressable memory (CAM) can be used to simultaneously compare the input datum with all the data stored in the memory. Due to the parallel operations, the CAM is useful for applications that demand high data-search speed, such as data base access, image processing, and IP address lookup. Network security systems require a great amount of pattern matching operations to compare the input network packet with the pre-defined rule set for protecting the system from network attacks such as worms and viruses. Recently, several papers proposed hardware-based pattern matching approaches for wire-line network security systems.

However, these designs are not suitable for mobile devices mainly because of two drawbacks. First, due to their hardware limitation, they are aimed at data matching with only a few thousands of network patterns in SNORT (an open source network intrusion prevention system). They are also not scalable to perform any antivirus scanning, since the number of virus patterns is one order larger than SNORT. The Clam AV open source antivirus software) releases more than 20,000 virus patterns and that number is still increasing. Second, these designs store all the virus patterns in the on-chip memory for achieving high throughput. When dealing with a large number of virus patterns, these designs need a large chip area and significant power due to the enlarged size of the on-chip memory. The scalability of handling more than ten thousands patterns is required for versatile network protection. In addition, the system must be highly flexible to accommodate the rapidly increasing new virus patterns. Intuitively, we can use CAM-based designs to achieve higher search speed. However, traditional CAM designs cannot satisfy the other requirements, including low cost, low power, and high programmability, to realize a good SoC integration.

II. THE VIRUS-DETECTION PROCESSOR

The design considerations for a virus-detection engine in mobile devices are analyzed as follows.

- The system throughput should reach up to 1 Gbps for supporting real-time virus detection in mobile devices adopting 4G wireless systems.
- The scalability of handling more than ten thousands patterns is required for versatile network protection. In addition, the system must be highly flexible to accommodate the rapidly increasing new virus patterns.
- Power consumption is the most important design consideration for mobile devices.

The increasing virus pattern will greatly increase the power consumption and the cost of on-chip CAMs. The memory design is critical for dealing with the increasingly large virus database. The key idea of our proposed virus-detection processor is to condense as much information on-chip as possible such that most of input data can be quickly scanned without further inspection.

The entire virus scanning is split into two phases: fast on-chip filtering by the filtering engine, and the exactly-matching with some off-chip memory accesses, as shown in Fig. 1(a). Only important filtering signatures and skip data are stored on the chip. In order to reduce the on-chip memory, the filtering engine operates only on the fixed amount of the memory, including a TCAM and a SRAM. These filtering data are extracted from the entire virus database by pre-processing the 30K virus patterns released from the Clam AV. The preprocessing tool also generates a suffix pattern tree, which will be stored in the off-chip memory.

Because most of the input data is safe, experimental results show that more than 80% of the data can be quickly processed by the filtering engine. Fig. 1(b) shows the operation principle of the virus-detection processor.

The filtering engine screens impossible matches by consulting two TCAM lookup tables (named no-plane and yes-plane), which are used to perform two steps of the on-chip data-scanning in Fig. 1(c) and (d). Wu-Manber has modified to obtain a fast shift table, which indicates the impossible matching patterns (so-called no-plane). By comparing the input datum with the no-plane TCAM from the least significant bit (LSB), the engine first looks up the shift table to perform a quick shift of impossible bytes until locating a possible match, as shown in Fig. 1(c).

If the input datum is matched with an entry of no-plane, the input string will be skipped according to the shift count stored in the shift SRAM. Then we further look up another signature table (called the yes-plane) to eliminate any false positives by ensuring that the prefix has the same signature. The yes-plane TCAM performs a modified bloom filter algorithm. In the earlier processor, the on-chip memory takes up over 75% of the overall power consumption and shares about 80% of the transistor count of the whole chip.

Hence, compressing filtering data into the on-chip memory critically determines the performance and power efficiency for virus detection. In order to further reduce the power consumption and silicon cost of the on-chip memory, an adaptively dividable dual-port BiTCAM is substitute the two separate TCAM tables, so that it not only used to enhance the storage utilization, but also used to reduce the power consumption of the TCAM.

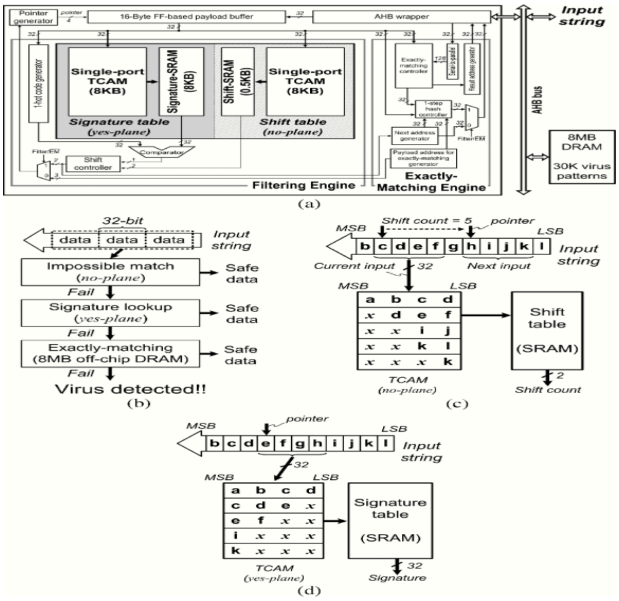

Fig.1. Architecture of the proposed virus-detection processor. (a) System architecture, (b) input data matching flow, (c) no-plane TCAM, and (d) yes-plane TCAM

III. THE DUAL-PORT BITCAM-BASED MEMORY BLOCK

1. The Evolution

Fig. 2(a) shows the evolution from a conventional single-port TCAM to a dual-port TCAM. The left part of Fig. 2(a) illustrates a design example for realizing the memory part of the virus-detection processor. It includes two single –port TCAMs and two SRAMs. One TCAM serves as the no-plane look-up table (or the shift table) and the other as the yes-plane look-up table (or the signature table). Since tail "don't care" (represented as "x" hereafter) bits are unnecessary in reality, the prefix of the yes-plane will be aligned to the left side of the TCAM array, and the TCAM entries in the yes-plane are sorted to form a triangle.

Similarly, the no-plane TCAM forms another symmetrical triangle. The no-plane is matched with the bits starting from the LSB and that the yes-plane is matched with the MSBs. The ternary cells storing "x" not only waste hardware cost but also waste energy. Here the idea is to merge these two single-port TCAMs into a single rectangular dual-port TCAM and concurrently match with the whole prefix. To achieve this goal we need a dual-port TCAM and two SRAMs as shown in the right part of Fig. 2(a),

The ternary cells storing "x" not only waste hardware cost but also waste energy. Here the idea is to merge these two single-port TCAMs into a single rectangular dual-port TCAM and concurrently match with the whole prefix. To achieve this goal we need a dual-port TCAM and two SRAMs as shown in the right part of Fig. 2(a), with a division line inserted in the dual-port TCAM array to separate the no-plane entries and the yes-plane entries. With the proposed dual-port TCAM, the ternary cells storing "x" terms can be minimized, and consequently both the total memory capacity and the power consumption are reduced. However, the partition of two CAM entries should not be fixed because the virus-detection application demands the flex flexibility of updating the contents of the look-up tables. To satisfy this design requirement, the division line should be adaptively adjusted according to different rule sets. Fig. 2(b) shows another design example.

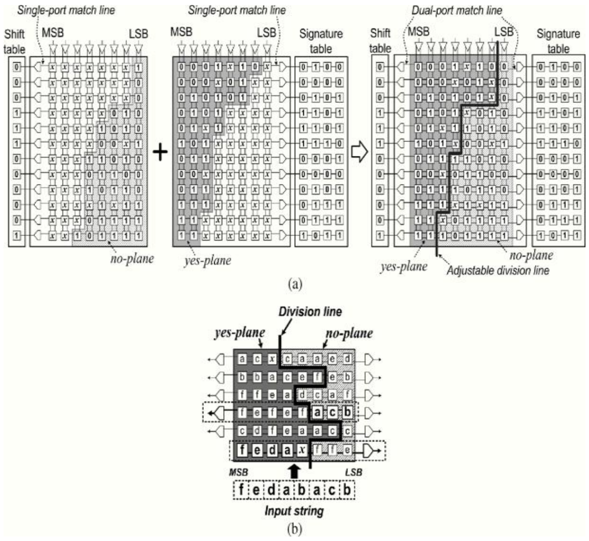

Fig. 2 (a) The design concept of the proposed dual-port TCAM, and (b) another design example.

This analysis shows that a few leading bytes in the yes-plane contain no "x" bits. Therefore, the ternary cells for storing those leading bytes can be replaced by binary cells for further area and power reduction, since usually a binary cell is simpler than a ternary cell. Thus, this new design is named as a BiTCAM, since it combines binary and ternary cells. According to the characteristics of the virus patterns in Clam AV, here are some rules for constructing the CAM's contents.

- The data width of the no-plane is at most 24 bits.
- The data width of the yes-plane is between 8 and 32 bits, and the bit width of the binary data in the yes-plane is set to 8.
- The number of entries in the no-plane is less than 2048. Therefore, the total bit width of those entries not used by the no-plane can be fully used to store data of the yes-plane.
- If the summation of the bit width of the yes-plane and that of the no-plane is larger than 32 for a particular entry to be inserted into the CAM array, either the data for the yes-plane or that for the no-plane will be cut short, depending on which solution cause less sacrifice in the filtering rate.

2. The Complete Memory Block

The complete memory block is shown in Fig. 3. The memory block is divided into 4 memory banks. Each bank contains one 512×32b BiTCAM, one 512×2b shift SRAM, and one 512×32b signature SRAM. The yes-plane and no-plane triangles are located on the left and the right part of the BiTCAM bank, respectively. The two SRAMs and the BiTCAM share one address decoder for reducing the hardware cost. Owing to the circuit structure, the search operation of each match-line of the yes-plane goes from the left-most CAM cell to the right, and the outputs from the right port of the BiTCAM should be used as the word-lines for the signature-SRAM. The match result of each match-line is ORed with the corresponding output of the address decoder to become the word-line of the signature-SRAM.

Similarly, the search operation of the no-plane begins from the right-most CAM cells, and the outputs from the left port of the BiTCAM should be used as the word-lines for the shift-SRAM. The match results are also ORed with the corresponding output of the address decoder to become the word-lines of the shift- SRAM on the left. Hit- detection circuits are added to determine any un-match cases. The symbol "SC" in Fig. 3 stands for the conventional 6T SRAM cell, while "B" and "T" represent 4-bit dual-port dynamic *AND* (DP-AND) gates for binary and ternary CAM cells, respectively. Each 32b match-line circuit is constructed with eight DP-AND gates.

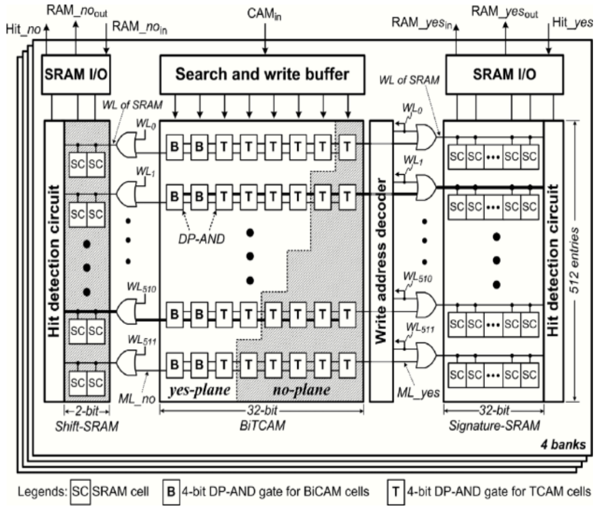

Fig.3 Memory organization in the virus-detection processor.

III.THE DUAL-PORT DYNAMIC GATE AND THE MATCH-LINE SCHEME

The dual-port dynamic AND (DP-AND) gate (Fig. 4(a)) is derived from the single-port pseudo-footless clock-and-data pre-charged dynamic (PF-CDPD) AND gate (Fig. (b)) , which is used to construct a PF-CDPD AND-type match circuit, a technique which leads to high speed and low power BiCAM or TCAM designs. Shown as the dotted and solid lines in Fig. 4(a),

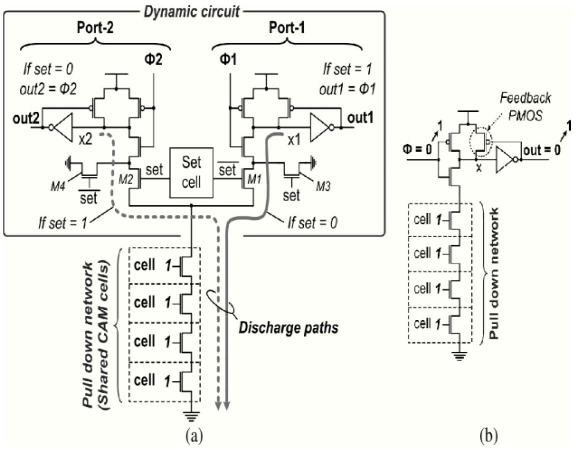

Fig. 4 The schematic diagrams of (a) the DP-AND gate and (b) the single-port PF-CDPD AND gate.

Fig. 5(a) illustrates one row of circuitry in the memory block, and depicts the connection arrangement of two DP-AND gates located at both sides of the division line. ML_no and ML_yes are match-lines in each row, and they require only one clock signal. However, for ease of explanation, in Fig. 5(a), the clock signals for both match-lines are designated differently.

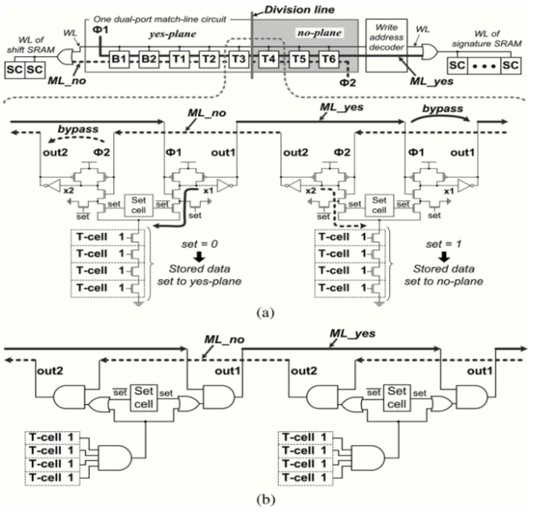

Fig.5 (a) The PF-CDPD dual-port match-line circuits (b) the static dual-port match-line circuit.

ML_yes is triggered by clock signal from the left-most DP-AND gate. This DP-AND gate belongs to the yes-plane, and its out1 will be evaluated according to the matching result of CAM cells. If matching, out1 goes “high,” and it will serve as the clock signal for the next DP-AND gate; if un-matching, out1 stays “low,” then all the right-side DP-AND gates will be kept quiet. Accordingly, signal of T3 comes from the output of its left stage, and the output of T3 becomes signal of T4 belonging to the no-plane.

Afterwards, the output of each DP-AND gate functions as the output of T3 until the final matching result is generated. Similarly, the operation of ML_no is triggered by clock signal. The ML_no only collects the matching results of the dual-port AND gates belonging to the no- plane while skipping the matching results of the dual-port AND gates belonging to the yes-plane. The dual-port match-line circuit can also be implemented as a static circuit as shown in Fig. 5(b). Comparing the layouts in Fig. 6, it has been found that although the dynamic block of the dual-port design is larger than that of the single-port design, the total layout area of the former design is 33% smaller than that of the latter design because the latter one uses double TCAM cells and dynamic blocks.

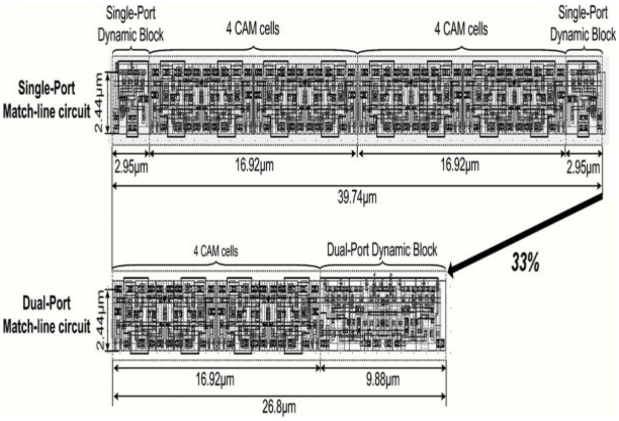

Fig.6 Comparison of layout design of single-port and dual-port match-line circuits.

V. EXPERIMENTAL RESULTS

This processor was implemented in the 0.13 m 1.2 V CMOS process. Post-layout simulation waveforms are shown in Fig.7, wherein the clock cycle time is set to be 2.6 ns for a clock frequency of 380 MHz. At the typical operating corner, the search time of the BiTCAM is designed to be 760 ps, and the access time of SRAMs is 970 ps. Therefore, the total delay time of the memory block is only 1.73 ns. The generated pulse width of word-lines is about 500 ps, which leads to about 200 mV bit-line voltage separation for read operations. The processor can operate at a maximum frequency of 380

MHz with a power consumption of 131.22 mW. Accordingly, the proposed processor achieves a maximum throughput of 3.04 Gbps with an energy cost of 0.44 fJ/ pattern-byte/ scan at peak throughput. Cost and power comparisons of on-chip memories are illustrated in Fig.8. Design 1 uses two single-port TCAMs and two SRAMs with four their own address decoders. Design 2 has one dual-port TCAM and two SRAMs the simplified address decoder. Fig. 8(a) shows the transistor count comparison.

The BiTCAM design has a 40% transistor reduction compared to two single-port TCAMs. Fig. 8(b) shows the power reduction of the CAM. One dual-port TCAM achieves 23% power reduction compared to two single-port TCAMs. When assuming a 25% occupation of binary cells, the power of a dual-port BiTCAM is further reduced to 38%.

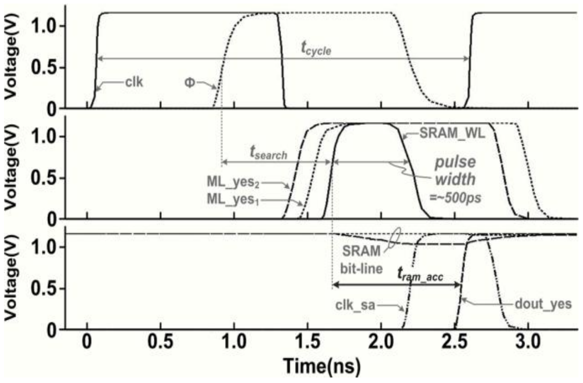

Fig.7 Detail timing diagrams of the memory block.

Fig. 8(c) shows the power reduction of the whole memory block. This work achieves 48% power reduction compared to design 1.

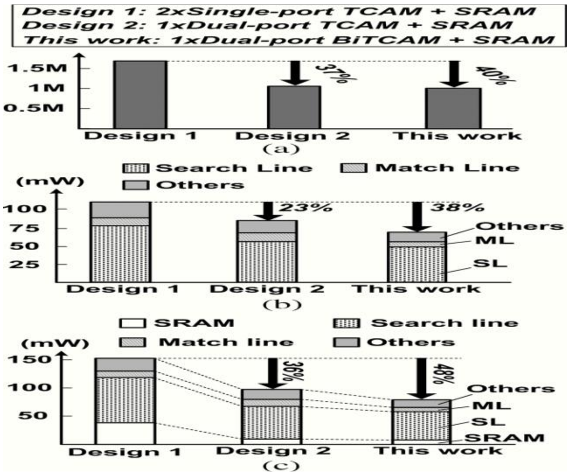

Fig.8 Comparison of (a) transistor count, (b) power consumption of the CAM, and (c) power consumption of the on-chip memory.

VI. CONCLUSION

This paper proposed an adaptive dividable dual-port 0.13 m BiTCAM for a high- speed, low-power and low-cost virus-detection processor in mobile devices. The proposed dual-port match-line scheme reduces the transistor count by 40% and provides 48% savings in power consumption compared to the single-port match-line scheme. The design of the adjustable division line provides high flexibility for updating virus databases. The virus-detection processor with the embedded BiTCAM performs up to 3 Gbps virus detection for 30K ClamAV rules at an energy cost of only 0.44 fJ/pattern-byte/scan at peak throughput**.**

REFERENCES

[1] K. J. Lin and C. W. Wu, “A low-power CAM design for LZ data compression,” IEEE Trans. Comput., vol. 49, no. 10, pp. 1139–1145, 2000.

[2] T. Ikenaga and T. Ogura, “A fully parallel 1-MbCAMLSI for real-timepixel-parallel image processing,”IEEE J. Solid-State Circuits, vol. 35, no. 4, pp. 536–544, 2000.

[3] N. F. Huang, W. E. Chen, J. Y. Luo, and J. M. Chen, “Design of multifield IPv6 packet classifiers using ternary CAMs,” in Proc. IEEE Int. Conf. Global Telecommunications, 2001, vol. 3, pp. 1877–1881.

[4] Y. H. Cho and W. H. Mangione-Smith, “A pattern matching coprocessor for network security,” in Proc. IEEE 2005 Int. Conf. Design Automation, pp. 234–239.

[5] L. Tan and T. Sherwood, “A high throughput string matching architecture for intrusion detection and prevention,” in Proc. IEEE Int. Symp. Computer Architecture, 2005, pp. 112–122.

[6] M. Yadav, A. Venkatachaliah, and P. D. Franzon, “Hardware architecture of a parallel pattern matching engine,” in Proc. IEEE Int. Symp. Circuits and Systems, 2007, pp. 1369–1372.

[7] Snort Users Manual 2.8.1. [Online]. Available: http://www.snort.org/ docs/snortht manuals/htmanual 281/

[8] About ClamAV. 2008 [Online]. Available: http://www.clamav.org/

[9] C. C. Wang, C. J. Cheng, T. F. Chen, and J. S. Wang, “An adaptively dividable dual-ported BiTCAM for virus detection processors in mobile devices,” in IEEE Int. Solid-State Circuits Conf. Dig., 2008, pp. 390–391.

[10] S. Wu and U. Manber, “A fast algorithm for multi-pattern searching,” Univ. Arizona, Report TR-94-17, 1994.

[11] B. H. Bloom, “Space/time trade-offs in hash coding with allowable errors,” Commun. ACM, vol. 13, no. 7, pp. 422–426, 1970. IEEE J. Solid-State Circuits, vol. 43, no. 2, pp. 530–540, Feb. 2008.