High Speed Packet Classification Using XnorBV

[1]Anil Kumar Reddy Kannapu, [2] Midde Adiseshaiah
[1] M.Tech (VLSI-SD), Dept. Of ECE
[2]M.Tech (Ph.D), Assistant Professor, Dept. Of ECE
[1,2]Chadalawada Ramanamma Engineering College, Tirupati, Andhra Pradesh, India.

***Abstract*—** Packet classification is a crucial technique for secure communication and networking. Security tools and internet services use packet classification technique which involves checking of packets against predefined rules stored in a classifier. The performance of the available software solutions of classification is not desirable and efficient for wire speed processing in high speed networks. Ternary Content Addressable Memory (TCAM), Bit-Vector (BV), field split bit vector (FSBV).In this paper, simple and memory efficient approach for packet classification has been proposed using Xnor gate instead of using lookup tables called XnorBV approach. Packet header fields of Internet protocol (IP) addresses and protocol layer are classified using Xnor gate against predefined ruleset which also support ternary bit pattern of '1', '0' and '*' while port numbers of packet header support range match by comparing port numbers against lower bound and upper bound. The proposed XnorBV architecture is independent of ruleset feature and supports multiple dimension classification.

Index Terms: Xnor, Bit-vector, Field split bit vector, Terrary content addressable memory

I. INTRODUCTION

A sequence of packets coming from the source system to a destination system is popularly label as traffic flow or packet flow and a sequence of packets from particular source to a particular destination is called a flow. A flow can be identified by using technique called packet classification which categorizes the incoming packets into different flow by inspecting values of header fields of packets within a certain time. For identification and arranging packets into different flow, each incoming packet is checked against a set of rule, if an incoming packet is matched with any rule of a rule-set then only it is accepted otherwise rejected. After categorizing incoming packets into different classes, each flow can be processed differently to differentiate the services suggested for the user. Each application and service requested by the user requires packets of same class. Packet classification technique helps to provide respective packets to respective services efficiently using predefined rule-set. Also, various services like firewalls, Virtual private network, network security, policy-based-routing, traffic shaping and quality of services incorporated the packet classification technique to detect threats and to prevent unauthorized access to the network. Due to these manifold advantages of packet classification technique in modern communication, packet classification has become an integrated part of all type of intrusion detection systems, firewalls, internet routers and virtual private networks.

Software solutions are available to perform classification of packets but they are insufficient for high speed network applications. In software tools, classification is generally done by checking only port numbers or IP addresses or protocol layer. Performance of software solutions which support inspection of multiple fields is not desirable for wire speed processing. For wire-speed processing and secure networking, hardware solutions are desirable and classification of packets can be done by checking all fields of packet header. In hardware packet classification solution, multiple fields of an incoming packet are checked against each rule of a rule-set. A size of ruleset may vary from hundred to thousand rules. The challenge and difficulty for hardware implementation of packet classification system is memory requirement to store large number of rules. Generally, rules are stored using on-chip memory resources of field programmable logic array (FPGA) but because of limited on-chip memory resources, storing of a large number of rules is the problem. For packet classification, rules are stored in a decreasing order of their priority in a ruleset and action is taken according to their priority. Figure 1 depicted below shows a standard 5-tuple packet header having destination and source Internet Protocol (IP) address field, destination and source port number field and the protocol field. For different combination of values of the fields require different matches like prefix match for destination and source Internet Protocol address field, range match for destination and source port field and exact match for protocol field.

Source IP address	Destination IP address	Source port	Destination port	Protocol

Figure 1: Standard 5-tuple packet header

II.PACKET CLASSIFICATION

Important issue of packet classification architecture is Power consumption. As throughputs of trillions of bits per second achieved by routers, power consumption becomes an increasingly critical concern. Power efficiency depends on number of rules used to classify incoming packet. This is one of aspect used for evaluation of power efficiency of packet classification system. The power consumed by the router to drive away the extremely large heat created by the router components extensively assist to the operating costs. The power consumption in search engines is becoming an increasingly important evaluation parameter because each port of routers contains packet classification devices and router lookup.

Memory requirement is another important issue of packet classification. Nowadays, researchers aim to find out solutions for large ruleset. Method of classification and number of rules stored in classifier is related to amount of memory required. Due to limited resources available on FPGA, memory has become very important issue of hardware solution to support large number of rules.
Speed and pliability in specifications are the issues in packet classification devices. In packet classification process, packets are categorized based on a set of predefined rules also called as packet filters. Rules or filters define patterns that are to be matched against incoming packets for arranging packets for different flows. Packet filters or rules specify possible values for each field of a standard 5–tuple packet header. The address fields of a packet header are often used prefixes to define the addresses, although in address fields arbitrary bit masks are acceptable in a classifier or ruleset and this feature is widely used in real filter sets. Rules or Filters specify a range value for port -fields of packet header for matching incoming packets. Protocols can be in two ways either exact value or as a wildcard. Values specified by bit masks are allowed in some system for protocol field of incoming packet, even if it's not clear how convenient that feature is.

III.RELATED WORK

Methods which are efficient and desirable for hardware implementation can be broadly classified into two approaches decision-tree based approach and decomposition based approach. In decomposition based approach, classification of packets is done in two phase: In first phase, independent searches are performed on each field of packets, while in second phase: results from the first phase are combined. Decomposition based algorithms are suitable for hardware implementation can sustain high throughput at low latency. Bit Vector (BV), Aggregated Bit Vector (ABV), Bit Vector- Ternary Content Addressable Memory (BV-TCAM), Field-Split Bit Vector (FSBV), Crossproducting, Recursive Flow Classification (RFC) and StrideBV are the some example of decomposition based approach. Bit Vector- Ternary Content Addressable Memory (BV-TCAM) algorithm and StrideBV algorithm support all matches and are scalable to large number of rule in a rule-set.
Ternary Content Addressable Memory (TCAM) is the desirable hardware solution because of its simple management and speed. To check all fields at a time and at high speed Ternary Content Addressable Memory (TCAM) based search engine is used. Extension of Ternary Content Addressable Memory (Ternary Content Addressable Memory (TCAM) approach is Bit Vector-Ternary Content Addressable Memory (BV-TCAM) uses Ternary Content Addressable Memory (TCAM) approach and Bit-Vector approach to support prefix, range and exact match. Bit Vector- Ternary Content Addressable Memory (BV-TCAM) approach is used to increase throughput and to compress data representation. This approach is generally used in network intrusion detection systems where report of multi matches at gigabits link rate is necessary. In packet classification, from multi match only single match of highest priority is reported for further processing due to routing problems. In Bit Vector- Ternary Content Addressable Memory (BV-TCAM) approach, IP addresses and protocol layer of header are matched using Bit-Vector approach and port numbers are matched using TCAM approach in parallel and results are ANDing to get final output. This approach supports multi match without use of range to prefix conversion.
Bit vector algorithm is desirable and widely used algorithm for hardware implementation of packet classification.The bit vector algorithm where bit value '1' indicates matching of incoming packet against a set of rule while bit value '0' indicates the mismatch of incoming packet against a predefined ruleset. In Bit-Vector (BV) algorithm, rules are arranged in a ruleset based on their priority. Generally to avoid complexity in assigning a priority to each rule, rules are arranged in decreasing order to their priority. Bit-vector is simple and has low computational complexity on hardware. For multi field packet classification, each field generates bit vector and then the bit vector of each fields are ANDing together to get final bit vector indicating the status of an incoming packet against a ruleset as shown in Figure 2.

IV.XNORBV ALGORITHM

In this work, classification of each field or tuple of incoming packet is done using XnorBV method instead of using look-up tables. In XnorBV, an Xnor gate is used as a basic comparator for comparing incoming packet with rule of a ruleset. Use of Xnor gate makes the architecture simple and efficient for wide variety of communication network involves packet filtering or packet classification. Using XnorBV algorithm, the proposed design achieves good results on same operating frequency of 300MHz. in XnorBV algorithm, each field of a packet header generates a bit vector which will be ANDing with bit vector generated by others' field to get final output bit-vector. A final bit-vector is given to priority encoder module to fetch higher priority matched rule. In the proposed method, checking of each bit of a field against each bit of a rule stored in a ruleset is done using XNORing operation. Using behavioral modeling of VERILOG, designed system supports ternary bit format of '1', '0' and '*' (wildcard entry). The proposed XnorBV method of packet classification is illustrated in figure 2, with the same ruleset and field value=1101 as that of Field split bit vector (FSBV) and StrideBV method of packet classification. After XNORing operation, each bit of obtained output after XNORing is ANDing to get one bit which indicates the status of a rule for incoming packet field [5]. A 5-tuple standard packet header having five fields which are source Internet Protocol (IP) address, destination Internet Protocol (IP) address, source port number, destination port number and protocol layer. In this paper, the classification of IP address fields and protocol field are performed using XnorBV method. Proposed XnorBV module supports prefix and exact match for Internet Protocol (IP) addresses and protocol layer respectively.

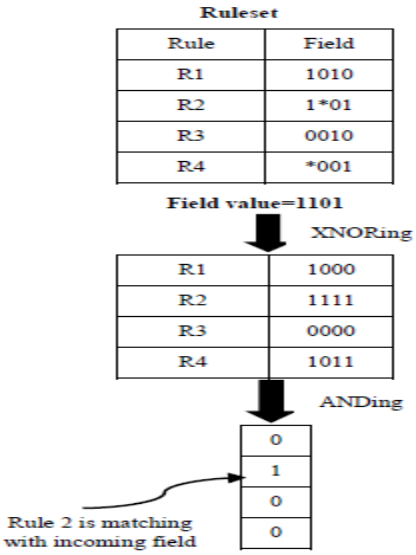


Figure 2 : Proposed XnorBV algorithm

Figure 3 shows the circuit diagram of proposed XnorBV method of generating bit vectors. A field of 5-tuple incoming packet is checked against N rules of a ruleset. To understand the generation of bit vector using XnorBV method with the help of circuit diagram, let the length of rule and a field of an incoming packet be k bits. Let the first rule of a ruleset is given by R1=Wk-1Wk-2….. W0 and a field of an incoming packet is given by F1= Tk-1Tk-2….. T0. Each bit of a rule and a field is XNORing and after completion of XNORing operation, result of k-bits is ANDing to get single bit indicating the matching or mismatching of field with a rule. Same operation is performed for each and every rules of a ruleset of size N to get N-bit vector for the particular field of a packetThe detailed algorithm of generating bit vector and performing packet classification is given below.

Algorithm 1: Bit Vector Generation for each field of a packet using XnorBV method

Require: N rules each of which is represented as a K-bit ternary string of a field of packet: Rn=Wn k-1 Wn k-2 Wn k-3……. Wn 0,

F=Tk-1 Tk-2 Tk-3 …….. T0, where= 1 …….. N

1: for n 1 to N do {Process Rn}

2: for k k-1 to 0 do

3: S[n] [k] = Wn k-1 Xnor Tk-1

4: end for

5: for b 0 to k-1 do

6: let Y=1,

7: Y = Sn b AND Y

8: end for

Algorithm 2: Packet Classification using XnorBV

Require: let the B be bit vector after comparing the incoming packet with a set of rules.

Require: let the B1, B2, B3, B4 and B5 be the bit vector of 5-tuple packet

1: for n 1 to N do {bit-wise AND}

2: V= B1 n AND B2 n AND B3 n AND B4 n AND B5 n

3: end for

4: V be the final bit-vector indicating the match of mismatch of packet with against rule of ruleset

4: V is the input to priority encoder to get highest priority matched rule

5: Vm V { Vm Output of Priority Encoder}

To support range match for port numbers, comparison of a field value against the lower bound and upper bound of a rule is done. Figure 7 shows the range module to perform range match for port numbers of a packet. To make designed architecture for supporting range match lower bound and upper bound has to be defined for each rule of a ruleset and method of performing range match is illustrated in Figure 7. A ruleset with lower and upper bound is depicted in Figure 7 with field value = 1000. A field value is to compare against lower bound, if field value is greater or equal to lower bound then it gives '1' otherwise '0' similarly if a field value is lower than or equal to upper bound then it gives '1' otherwise '0'. Bit values obtained after comparing field value against lower bound and upper bound are ANDing to get one bit which indicating that field value is in between lower bound and upper bound. Range search module is used for source port number and destination port number each of 16 bits for various applications. The proposed architecture supports prefix match for IP addresses, range match for port numbers and exact match for protocol field. Also, it is independent of ruleset feature and supports multiple dimension classification.

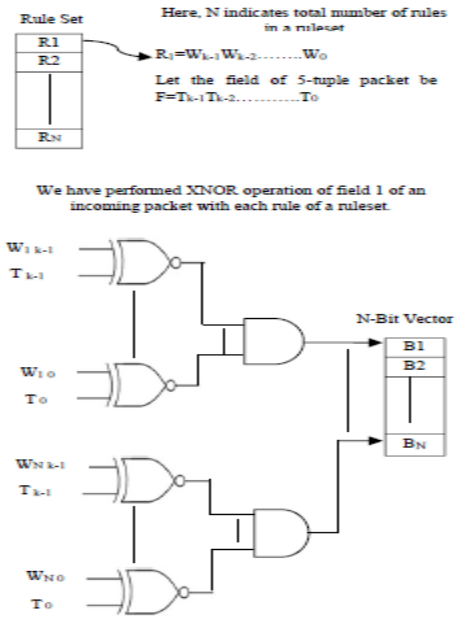


Figure 3: circuit diagram of proposed XnorBV

The complete architecture for packet classification supporting prefix, range and exact match is depicted in Figure 8. Rules are arranged in ruleset in a decreasing order of their priority. The architecture shown in Figure 8 performs the classification of complete packet header of 104 bits with multi-match packet classification feature. The storage of rules for each tuple is done separately and checked each respective tuple or field of an incoming packet against a respective rule-set. A five tuple packet header gives five N-bit vectors; each N-bit vector indicates the status of that tuple against predefined rules in a ruleset. After getting partial results from the classification process of each tuple of the packet, the results of five tuple are undergo ANDing operation to get final bit vector indicating match or mismatch of the packet against the rules of a ruleset. For IP addresses and protocol layer, XnorBV module is used to perform prefix as well as exact match. An XnorBV module can support ternary bit format '0', '1' and '*' (wildcard entry). For port numbers of a packet, range module is used to generate bit-vector. In this way, the proposed architecture can perform prefix, range, and exact match. Priority encoder is used to decide the highest priority rule from the final bit vector and select the rule for further operation.

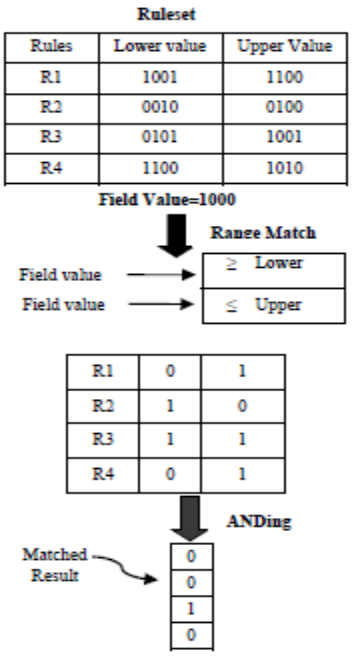


Fig 4: Range search Mobule for Range march

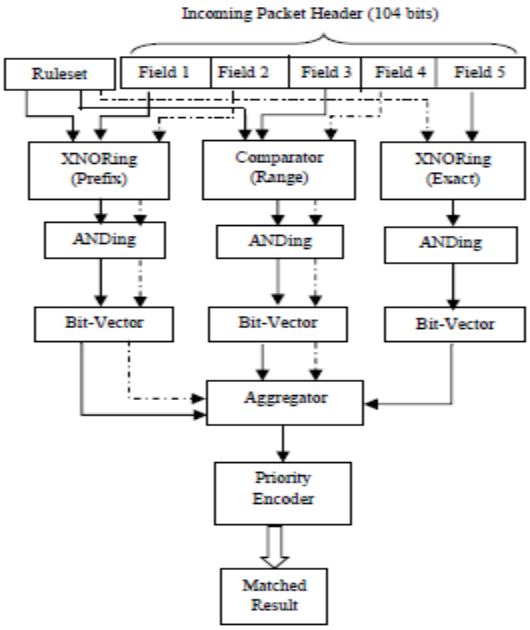


Figure 8: Proposed Architecture for Packet Classification

Fig 5: Proposed Architecture for packet

V.Results:

The Verilog HDL is used to design the architecture on Xilinx ISE design 12.1 suite. Design utilization summary of the architecture for the proposed XNOR BV Algorithm on SPARTAN3E FPGA trainer kit is shown below.

Device utilization summary:

Selected Device: XC3s500efg320-5

Number of Slices: 139 out of 4656 2%

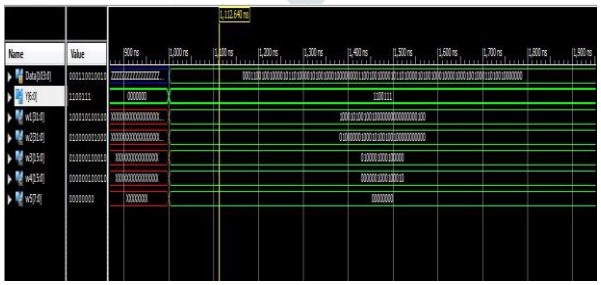
Number of 4 input LUTs: 242 out of 9312 2%

Number of IOs : 111

Number of bonded IOBs: 111 out of 232 47%

The Latency of a system is the time required to get output after applying input. In packet classification, the latency is defined as time required for completing one classification process. In XnorBV method, the classification process is performed in three stages. In first step, there is separation each field of an incoming packet to classify against ruleset to generate bit vectors. In second stage, bit-vector of each field generated in first stage i.e. partial results are combined to get final bit vector which indicates the status of rules against incoming packet. Final result obtained in second stage is forward to priority encoder to get single match result from multi match result for further process. Extraction of highest priority matched rule using priority encoder is done in third stage. In this way, the proposed XnorBV method requires three clock cycle to perform classification of one incoming packet. So, the latency of proposed architecture is 3 clock cycles which is also desirable for low latency application.

The below fig depicts the simulation result of proposed algorithm.



VI.Conclusion:

Proposed method XnorBV architecture using Xilinx ISE 12.1 suite selecting XC3s500efg320-5,SPARTAN3E FPGA as target device is memory efficient requires 15 byte/rule less than any other existing technique of packet classification. Architecture supports prefix, exact and range match without use of range to prefix conversion and is independent of ruleset feature. Design of High Performance Packet Classification Architecture for Communication Networks.Power efficiency is also improved with power increment in addition of one rule. The proposed architecture can sustain high throughput at low latency which is desirable for low latency applications.

REFERENCES

[1] Andrea Sanny, Thilan Ganegedara, Viktor K. Prasanna; "A Comparison of Ruleset Feature Independent Packet Classification Engines on FPGA," *in 27th International Symposium on Parallel & Distributed Processing Workshops and PhD Forum*, 978-0-7695- 4979-8/13 $26.00 © 2013 IEEE

[2] T. Ganegedara and V. Prasanna, ''StrideBV: 400G+ Single Chip Packet Classification,'' *in Proc. IEEE Conf. HPSR*, 2012, pp. 1-6.

[3] Mahmood Ahmadi, S. Arash Ostadzadeh, and Stephan Wong; "An Analysis of Rule-Set Databases in Packet Classification," in *18th Annual Workshop on Circuits, Systems and Signal Processing (ProRISC 2007)*, 29-30 November 2007, Veldhoven, The Netherlands.

[4] Nekoo Rafiei Karkvandi, Hassan Asgharian, Amir Kusedghi, Ahmad Akbari, "Hardware Network packet Classifier for High Speed Intrusion Systems," in *International Journal of Engineering and Technology*; Volume 4 No.3, March, 2014.

[5] Ausaf Umar Khan, Yogesh Suryawanshi, Dr. Manish Chawhan, Sandeep Kakde, "Design and Implementation of High performance Architecture for Packet Classification," in *International Conference on Advances in Computer Engineering and Applications,* IMS Engineering College, Ghaziabad, India, page 598-602, IEEE.

[6] Aladdin Abdulhassan and Mahmood Ahmadi, "Parallel Many Fields Packet Classification Technique using R-Tree ," in *Annual Conference on New Trends in Information & Communications Technology Applications-(NTICT'2017),* 7-9 March 2017.

[7] Safaa O.Al-Mamory and Wesam S.Bhaya; "Taxonomy of Packet Classification algorithms," in *Journal of Babylon University/Pure and Applied. s*

[8] Balasaheb S. Agarkar and Uday V. Kulkarni, Ph.D., "A Novel Technique for Fast Parallel Packet Classification," in *International Journal of Computer Applications (0975 – 8887)* Volume 76–No.4, August_2013.

[9] Andreas Fiessler, Sven Hager and Björn Scheuermann, "Flexible Line Speed Network Packet Classification Using Hybrid On-chip Matching Circuits," in *IEEE 18th International Conference on High Performance Switching and Routing (HPSR),* 18-21 June 2017.

[10] Pankaj Gupta and Nick Mckneown; "Algorithms for packet classification," in IEEE magazine, March/April 2001 pp. 24-32