

# Design of a Low Latency and High Throughput Packet Classification Module on FPGA Platform



Anita P, Manju Devi

**Abstract:** The Packet classification method plays a significant role in most of the Network systems. These systems categories the incoming packets in various flows and takes suitable action based on the requirements. If the size of the network is vast and complexity will arise to perform the different operations, which affects the network performance and other constraints also. So there is the demand for high-speed packet classifiers to reduce the network complexity and improve the network performance. In this article, The Bit vector Packet classifier (BV-PC) Module is designed to improve the network system performance and overcome the existing limitation of Packet classification approaches on FPGA. The BV-PC Module contains Packet generation Unit (PGU) to receive the valid incoming packets, Memory Unit (MU) to store valid packets, Header Extractor Unit (HEU) extracts the IP Header address information from the Valid packets, The BV-Based Source and Destination Address (BV-SA, BV-DA) unit receives the IP packet header Information and Process with BV based rule set and aggregates the BV-SA and BV-DA outputs, Priority Encoder encodes the Highest priority BV Rule for the generation of Classified output. The BV-PC utilizes <2% Chip area (slices), works at 509.38MHz, and consumed Less 0.103 W of total Power on Artix-7 FPGA. The BV-PC operates with a latency of 5 clock cycles and works at 815.03Mpps throughput. The BV-PC is compared with existing approaches and provides Better improvements in Hardware constraints.

**Keywords:** Bit vector (BV), Packet classifier, Ruleset, FPGA, Throughput, Packet generation Unit, Source Address, Destination Address, Latency.

## I. INTRODUCTION

The demand for Network constraints like Security, Traffic analysis, Load Balancing, and Quality of Service (QoS) is increasing exponentially by increasing the Network System speed. To link up these, the Packet classification approach is necessary and much needed. The packet classification is the Process of matching the packet header fields with a suitable rule set. In general, there are 5 different classification fields used in Process, Namely, Source and Destination Address (SA and DA), Source and Destination Port (SP and DP), and Protocol.

Revised Manuscript Received on April 30, 2020.

\* Correspondence Author

Anita P\*, Research scholar, Department of Electronics & Communication Engineering, TOCE, Bengaluru, India. Email: anitasp2002@gmail.com

Dr. Manju Devi, Professor & Head, Department of Electronics & Communication Engineering, TOCE, Bengaluru, India

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](https://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC-BY-NC-ND license

<http://creativecommons.org/licenses/by-nc-nd/4.0/>

The different approaches use only 5- fields or 15 fields for classification to match the rule set [1-4]. The challenges of Packet classification are Classification Speed, Scalability, Modularity, Power consumption, fast updation, implementation flexibility, and Storage Space. Many approaches are available to design the Packet classification, which includes clustering-based, Tree-based, Tuple space-based, geometrical approaches, and Bit Vector-based approaches. Similarly, many Hardware-based Packet classification approaches are available, which includes, Ternary Content Addressable Memory (TCAM) based, RAM based, FPGA based, and Multi-core Processor approaches [5-8]. The packet classification is used many applications to improve network security [9] and Network-on-Chip (NoC) [10] Performance on FPGA Chip.

In this article, The Bit vector Packet classifier (BV-PC) is designed and implemented on Artix-7 FPGA, Which provides low latency and High throughput performance on FPGA, which also consumes less amount memory and supports more massive Rule sets. Section 1.1 provides the background of the current research works of different packet classifiers, followed by findings. The proposed Bit vector Packet classifier (BV-PC) hardware architecture and its sub modules are discussed in section 2. Section 3 elaborates on the results and discussion by concerning the hardware constraints of BV-PC and comparison of the BV-PC with existing approaches with improvements. The overall work with improvements and future scope is highlighted in Section 4

## A. The Background

This section describes the existing approaches of different Packet classifier approaches and applications. Li et al. [11] discuss the high-speed classification using the Binary tree approach, which provides quick rule updation and better memory performance. The binary tree approach has binary tree searching Nodes with rule set followed by Linear matching for rules. The binary tree searching module contains mainly leaf node with rule's base address followed by a middle node with child node address and also has Main root node contains first level, second-level, and third-level nodes with rule address. The linear searching Module contains storage rules with the upper boundary and lower memory field address. This linear search engine compares the lower and upper boundary values to provide the output matching results. Ganegedara et al. [12] explain the high-performance packet classification module with scalable and modular architectures. The modular BV for high-speed PC on FPGA also introduces the Stride BV architectures, which improves the optimization goal in Modular BV and provides the scalability features,

which is quite better than Conventional approaches. The range search module is incorporated in Modular BV to eliminate the rule set enlargement. The modular BV supports 100Gbps speed and supports broad rule set up to 28K on on-chip Memory.

Yun Qu et al. [13] discuss the dynamically Updatable PC engine, which provides high performance and high Throughput on FPGA. The 2-dimensional array of modular Processing Elements (PEs) are used, which provides exact memory allocation and prefix match effectively. The Dynamic updateable PC also supports optimization features like striding, power-gating, dual-port Memory, and clustering. The Modular PE's are self-reconfigurable, which updates the ruleset dynamically. The dynamically Updatable PC engine sustains 650Mpps Throughput on FPGA and better than the Existing TCAM approach.

Qu et al. [14] explain the many field PC on FPGA, Graphical Processing Unit (GPU), and General propose Processor (GPP) with multi-core support. The Many filed PC is optimized using Module PE and is designed and concatenated using systolic array and also divides the generic ranges into multiple parts. The multi filed PC works at 500Mpps, 14.7 Mpps, and 30.5 Mpps throughput for FPGA, GPP, and GPU platforms. The multi filed PC works also support 1.5K Rule sets, 32Krule sets, 32K rule sets for FPGA, GPP, and GPU platforms. Zhou et al. [15] discuss the large scale PC on the FPGA Platform, which uses a decomposition approach, which has searching and merging phases. The searching phases use BV algorithms or Rule Identifier Set (RIDS), and the merge phase provides the intermediate results, both the results concatenated each other for the generation of final results. The Large scale PC works at 147 Mpps and supports 256k Rules on the FPGA platform. Chang et al. [16] describe the Range Enhanced PC on FPGA, which is a hybrid combination of Stride BV features and a sub-range comparison method. The Range Search PC uses 12 tuple header fields to support multi-field PC and also used to store pre-computed values in Memory. The field lengths are defined as per open flow 1.0 in Range Search PC. The Range Search PC contains Range BV Encoding (RBVE) method has lower and Upper Boundary, which is split into Strides to improve the optimization in PC. Khan et al. [17] present high-performance Module design for PC, which contains essential XNOR gate operation for matching input packets against Rule sets and generates the BV of same size and AND'ing all the bits in BV to generates the final Matched Results. The design supports 4bits per Rules with low latency. Yu et al./ [18] present the static-RAM (SRAM) based PC works in one memory access, also exhibit the behavior of pseudo-TCAM. The SRAM-based PC provides header fields that are encoded using the Prefix Inclusion Coding (PIC) technique. The bit selection method is used to map the encoded rules with SRAM –based Match units. The SRAM-based PC works at 426 Mpps and supports 10K to 100K Rules.

The existing Packet classification approaches have many limitations and challenges to address, which includes multiple fields matched against the large rule sets and the drastic increment in network traffic data rate, which causes complexity and performance metrics in PC. There is a need for PC which maintain high performance, high Throughput, and Low latency for Networks.

## II. PACKET CLASSIFICATION ARCHITECTURE

The Packet classification plays a vital role in many network systems like the Intrusion detection system, Routing systems, Firewalls, Traffic control systems, and many. These network systems need data packets that are divided based on the various design flows to address the different application requirements. This functionality much suited and provided by the Packet classifier (PC), which is defined by a set of rules. This section contains the methodology used in the research for BV-Packet classification, and detailed architecture is discussed in detail.

**a. Research Methodology:** The proposed methodology of the Packet classification is represented in Figure 1. The Proposed BV-Packet Classifier Module contains Packet Generation Unit (PGU), Memory Unit (MU), Header Extractor Unit (HEU), Packet classification module using Bit Vector (BV), Packet Aggregator and Priority Encoder. The Packet generation Unit (PGU) mainly used to receive valid Incoming from any external sources or by the user. It receives only valid and proper data packets for Futureclassification. The Memory Unit is used to store the incoming packets based on write enable signal. The Header extractor unit (HEU) is a central part of the packet classification process. The HEU generates different header addresses, which includes TCP header, IP header, and Ethernet header along with primary 32-bit packet data for future usage in packet classification.

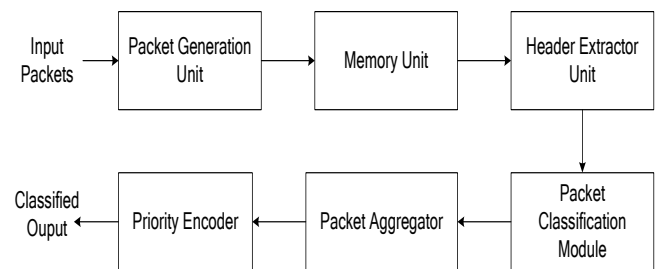


Figure 1 Methodology used in BV- Packet Classifier Module

The Packet Classification Module is designed Using the Bit Vector (BV) approach. The HEU provides the Internet protocol (IP) address used as Packet header information in the BV method. The BV source and destination address modules are matched with the corresponding memory location, which is set by rules. The BV Source and Destination address output information is aggregated using Packet Aggregator. The priority encoder encodes the highest priority rule that matches the incoming packet for the generation of the classified packet. The detailed internal Architecture of BV-PC is explained in the below section.

**b. Bit Vector Packet Classifier (BV-PC) Design:** The Bit Vector Packet Classifier (BV-PC) design is used to receive the incoming packet and Process with BV and generates the classified output. The detailed internal architecture of Bit Vector Packet Classifier (BV-PC) design is represented in figure 2.

**Packet Generation Unit (PGU):** The packet generation Unit mainly used to receives valid Incoming from any external sources or by a user. It receives only valid and proper data packets for Future communications.

In this design, 76-bytes are receiving for the formation of 608-bit Valid Packets. The design incorporates the Main control signals like sop (Start of the packet) to initiates the PGU process, the valid signal indicates receives only valid packets, eop (End of the packet) indicates the last packet of the PGU.

These control signals are analyzing and validating the incoming Packets for the formation PGU. The PGU receives 76-bytes of incoming packets and generates the 608-bit outgoing packets. The 7-bit Packet counter is used to count the number of packets until it reaches 76. The Error output signal will be activated if the control signals and the error signal violates the conditions.

**Memory Unit (MU):**The Memory Unit is used to store the Incoming generated packets based on write enable signal. The MU receives 608-bits of packets, and once write enable signal is activated. The MU has 16-memory locations, and each memory location can hold 608-bit packets. Based on the user address, stored packets can read and send to the next Process.

When the write enables is low, memory location will be read and stores the results in 608-bit Memory out.

**Header Extractor Unit (HEU):**The Header Extractor Unit is a central part of the packet classification process. The HEU is working based on Internet Version protocol 4 (IPv4). It generates different header address which includes TCP header, IP header, and Ethernet header along with primary 32-bit packet data for future communication in packet classification. The Header extractor unit extracts the different header addresses like 208-bits Ethernet header, 160-bits for IP header, and 160-bits TCP header. These TCP header, IP header, and Ethernet headers are acts as a source and destination address in detailed packet classification process like Bit vector (BV) and other processes. Out of 160-bit IP header values, The BV Process uses the first 32-bits [31:0] for a Destination address (DA) and next 32-bit [63:32] for Source Address (SA).

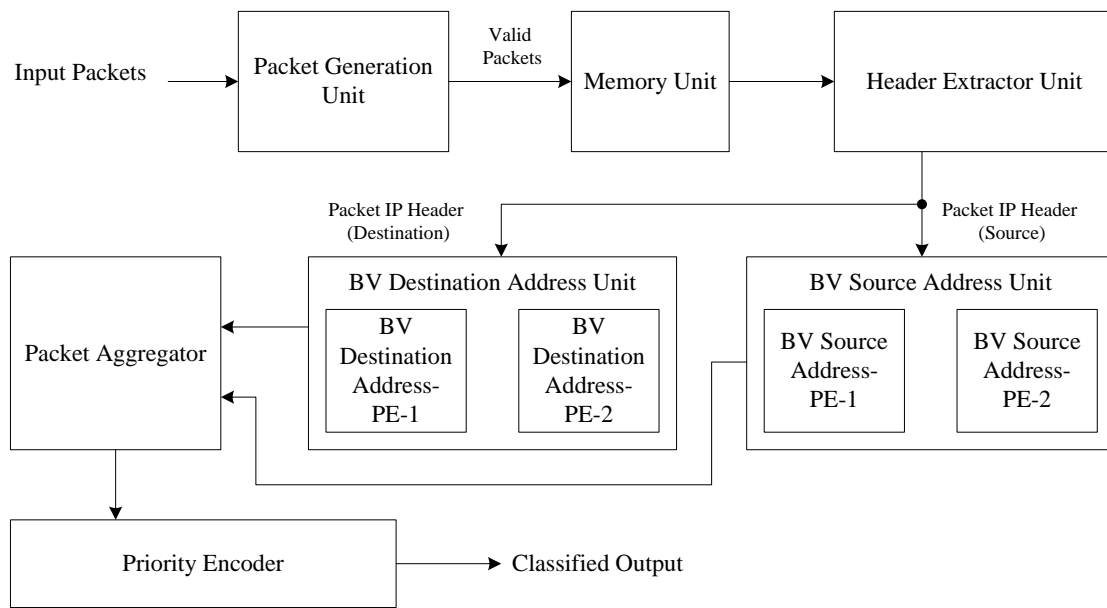


Figure 2 Internal Architecture of BV- Packet Classifier

**Bit Vector Architecture:**

The Bit Vector (BV) process is also considered as Field-Split BV (FSBV) method [13], and its associates split each field into several sub-fields. The Ruleset is mapped onto each sub-filed based on {0, 1, \*} ternary string. The Memory operations (Lookup table) is performed in all the sub-fields in a pipelined manner. BV of corresponding bits defines the temporary results of each processing element (PE). The Final Results are obtained by merging all the extracted BV bits using logical AND operations on FPGA. BV architecture is represented in figure 3.

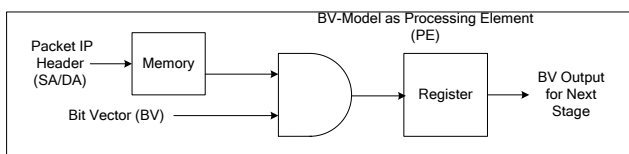


Figure 3 Bit Vector Architecture

The HEU generates the Packet IP address, which is divided into 32-bit Source and Destination IP address. Each Source and destination IP Address acts as packet header address to BV-SA and BV-DA process. Each BV-SA and BV-DA unit contains Two Processing elements, which perform the BV process. The Packet IP Header of

SA PE-1, acts as header address in Memory, and finds the corresponding field Value and Perform Logical AND operation with User BV data and results are stored in temporary Register. The Register Output is input to the next BV-SA of PE-2 and perform the same BV process and generates the Final Results of BV-SA output. A similar process is applied for BV-SA of PE-1 and PE-2 and generates the BV-DA Output. The Memory Unit contains 16 locations, and each location has 32-bit information.

**Packet Aggregator:** The Aggregator is used to Perform Logical AND operation of BV-SA output and BV-DA output and generates 32-bit Aggregator Output, which is input to priority encoder.

**Priority Encoder:** The encoder extracts the highest priority rule that should match with an incoming packet. The 32-bit encoder input performs bit-wise rule checking with the highest priority for the generation of classified output.

The example of the BV process is illustrated in figure 4. The BV process is applied to match the 4-bit header address field, which is against with rule of 4-bits width. The input header field is either SA or DA address is set to 1100. The 3 rulesets are defined with corresponding field (F) values. Splitting the 4-bit field into 1-bits, to get Bit vectors of each sub-fields: F [3], F [2], F[1], and F[0].

# Design of a Low Latency and High Throughput Packet Classification Module on FPGA Platform

The subfield F[0] and others are either '0' or '1' field value. For bit 3 or (F[3]) of the ruleset is "101". Based on field values, The Ruleset generates BV values. So The R1 has 1111, R2 has 0100, and R3 has 1111. The extracted BV values R1, R2, and R3, are performed

logical AND (&) operation. The matched results are obtained by using R1& R2& R3. So for this example, the Matched Results is 101.

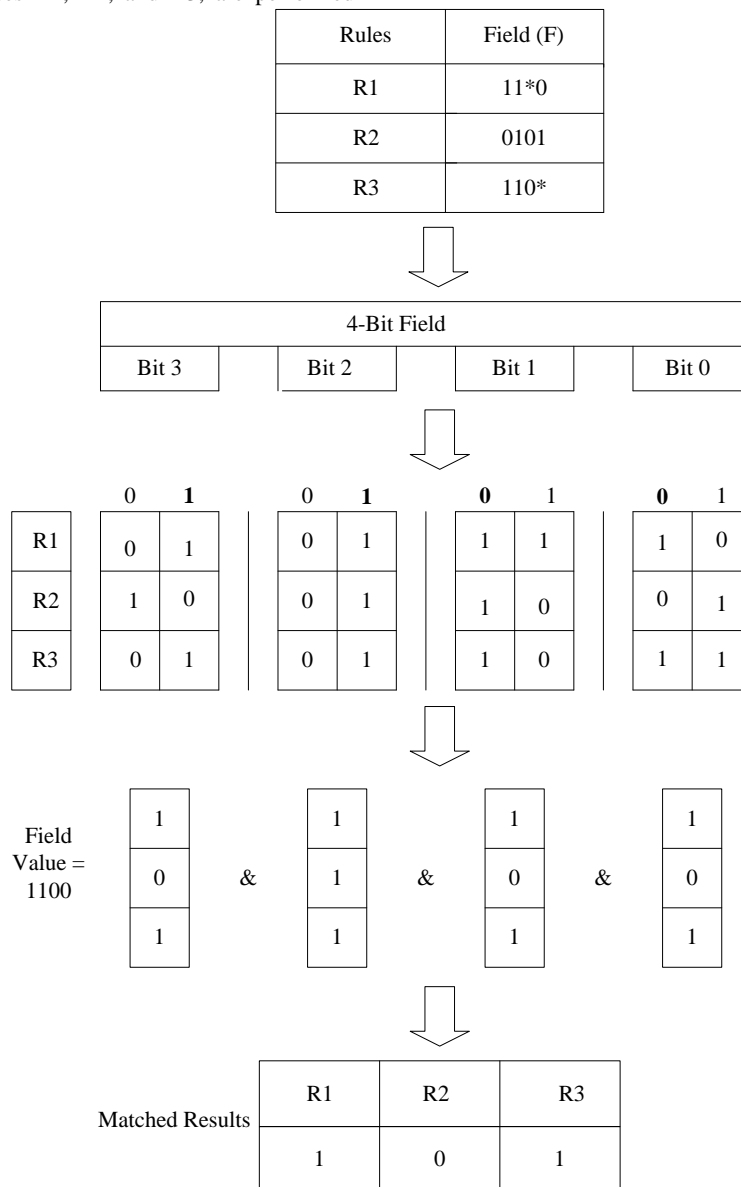


Figure 4 Example of BV Process

Throughput are tabulated in table 1 for the BV-PC module. The BV-PC utilizes the 3636 Slice registers, 2641 LUT's, and 2453 LUT-FF pairs. The BV-PC is operating at 509.398 MHZ on Artix-7 FPGA with a minimum period of 1.963ns and a combinational delay of 1.061ns. The present Classifier consumes a total power of 0.103W with the inclusion of static power 0.082W and Dynamic power of 0.021W.

### III. RESULTS AND ANALYSIS

An efficient Bit-Vector Packet Classifier (BV-PC) module is designed and implemented on Artix-7 FPGA. The BV-PC module is simulated on the Modelsim simulator and used for the calculation of latency. The Synthesized results are generated on the Xilinx14.7 ISE environment. The resource constraints like Chip area, Frequency, power, latency, and

Table 1 Resource Utilization of BV-packet Classifier on Artix-7

Resources	BV_Packet Classifier
<b>Area Utilization</b>	
Slice Registers	3636
Slice LUT's	2641
LUT-FF pairs	2453
<b>Timing Analysis</b>	



Minimum Period (ns)	1.963
Max. Frequency (MHz)	509.398
Combinational Delay (ns)	1.061
<b>Power Utilization</b>	
Dynamic Power (W)	0.021
Total Power (W)	0.103
<b>Latency and Throughput</b>	
Latency (Clock cycles)	5
Throughput (Gbps)	61.94

The BV-PC module receives 76 bytes of input packets for classifying the packet at high speed based on the BV algorithm and Ruleset. The BV-PC module is simulated on ModelSim Simulator for latency calculation, and the Classifier takes 5 clock cycles (50ns) to complete the classification for 76 bytes of packets. The Throughput is calculated using  $((\text{Number of bits} * \text{Frequency}) / \text{latency})$ . The 76 bytes packets contain 608-bits, and the maximum operating Frequency of BV-PC is 509.398 MHz with a

latency of 5 clock cycles. So BV-PC is working with a throughput (speed) of 61.94Gbps. The Throughput is also represented in terms of Millions Packet per second (MPPS), so the BV-PC works at 815.03 MPPS.

The BV-PC module consists of sub-modules of the Packet generation Unit (PGU), Memory unit (MU), Header Extraction Unit (HEU), BV-Source address, and destination Address Unit, and finally Priority Encoder. The Area utilization and operating Frequency of submodules in the BV-PC module is tabulated in table 2.

**Table 2 BV-PC sub-modules Area utilization and operating Frequency**

BV-PC Sub Modules	Slice Registers	Slice LUT's	Max. Frequency (MHz)
Packet Generation Unit	1237	28	584.62
Memory Unit	608	610	870.09
Header Extractor Unit	1744	2281	1023.227
Bit Vector SA	14	46	NA
Bit Vector DA	15	46	NA
Priority Encoder	1	31	NA

The PGU, MU, and HEU consume more slices registers and LUTs. The PGU, MU, and HEU operate Frequency of 584.62MHz, 870.09MHz, and of 1023.22MHz, respectively. The HEU is used for the extraction of Internet protocol (IP) and TCP header address information. The HEU boosts the BV-PC performance and reduces the complexity by providing headers information. Each BV-Source Address (SA) and Destination Address (DA) contains two processing Elements (PEs). Each PE is designed using Packet header and BV inputs.

classifiers like Decision Tree [20], Large Scale [21], Ultra scale [22], and HiCuts [23] are implemented on different FPGA devices. The Performance metrics like Chip area, Frequency, and Throughput (Speed) are analyzed with Proposed BV-PC. The proposed efficient high throughput Bit Vector-Based Packet classifier (BV-PC) is designed using the Bit Vector approach with pipelined architecture. The BV-PC is implemented on Low-cost Artix-7 FPGA. The BV-PC utilizes only 3636 slices, operated at Frequency of 509.398 MHz on Artix-7 with a high throughput of 815.03 Mpps, and it supports more massive rule sets

The performance comparison of BV-PC with other existing Packet classifier is tabulated in table 3. The existing Packet

**Table 3 Performance Comparison of Classifiers on Different FPGA's platform**

Packet Classifier Approach	FPGA Device	Slices	LUT's	FF's	Block RAMs	Frequency (MHz)	Speed (Mpps)
Decision Tree PC [20]	Spartan 3E	NA	6442	5336	22	100	NA
Large Scale PC [21]	Virtex-5	10307	NA	NA	407	125.4	250
Ultra-scale PC [22]	Stratix-III	40 070	NA	NA	NA	219	433
HiCuts [23]	Stratix-IV	15936	NA	NA	NA	150	100
Proposed BV-PC	Artix-7	3636	2641	2453	1	509.39	815.03

The Decision Tree Packet classifier [20] uses 8-parallel instances of 4-stages with pipelined architecture for packet classification, which utilizes 6442 LUT's, 5336 Flip-flops (FFs) and Block RAM of 22. The Decision Tree PC works at 100 MHz using Spartan -3E FPGA Board. The Large Packet wire-speed (LPWS) classifier [21] is designed using decision

Tree-based, Multi-field, 2dual pipelined architecture. The Large Packet wire-speed classifier is implemented on Virtex-5 FPGA. The LPWSclassifier utilizes 10307 Slices, operated at 125.4 MHz on Virtex-5 with a throughput of 250 Mpps,

and it supports 12 pipeline stage rules. The Ultra-scale Packet classifier [22] is designed using Cutting Scheme with the Decision Tree algorithm. The Ultra-scale Packet classifier is implemented on Stratix-IV and Cyclone III FPGA's. The Ultra-scale Packet classifier utilizes 40070 Logical elements (Slices), a peak power of 9.03W, operated at 219 MHz on Stratix-IV with a throughput of 433Mpps and it supports large rule sets. The HiCuts Based Packet classifier [23] is designed with high degree pipelined and parallel architecture using decision Tree and rule memory approach. The HiCuts Based Packet classifier is implemented on Stratix-IV FPGA. The HiCuts Based Packet classifier utilizes 15936 Logical elements (Slices); operated at 150 MHz on Stratix-IV with a throughput of 100Mpps and it supports 500 rule sets. The proposed BV-PC provides better resource utilization and Throughput than existing Packet classifier approaches.

### IV. CONCLUSION AND FUTURE WORK

The High-performance Bit Vector Packet classifier is designed and implemented on Artix-7 FPGA. The BV-PC works at Low latency and high Throughput on FPGA. The BV-PC is used to receive the Incoming packets and Process through the BV module and generates the classified output. This BV-PC is used in Most of the Network modules for Packet classification and also improves the system performance. The BV –PC main contains PGU, Memory Unit, HEU, BV-SA, and DA Unit and Packet aggregator, also Priority encoder. The BV-PC results are analyzed and discussed using hardware constraints like Chip area, Frequency, and Power. The BV-PC Utilizes < 2% slices resources and works at 509.39 MHz and also consumed 0.103W of total power on Artix-7 FPGA. The BV-PC is simulated on Modelsim Simulator for Latency calculation, and BV-PC uses 5 clock cycles for 76 Bytes packets process. The BV-PC works at 815.03Mpps Throughput Supports a more extensive rule set and utilizes less amount of Memory on FPGA. The BV-PC is compared with existing Packet classification approaches by concerning performance parameters with improvements. In the future, The BV-PC is optimized using striding, clustering, and other approaches with the help of a Modified BV approach to improving the Network performance further.

### REFERENCES

1. Kumar, V. Anand Prem, Vidya Thiyagarajan, and N. Ramasubramanian. "A Survey of Packet Classification Tools and Techniques." In 2015 International Conference on Computing Communication Control and Automation, pp. 103-107. IEEE, 2015.
2. Nagpal, Bharti, Nanhay Singh, Naresh Chauhan, and Radhika Murari. "A survey and taxonomy of various packet classification algorithms." In 2015 international conference on advances in computer engineering and applications, pp. 8-13. IEEE, 2015.
3. Srinivasan, T., N. Dhanasekar, M. Nivedita, R. Dhivyakrishnan, and A. A. Azeezunnisa. "Scalable and parallel aggregated bit vector packet classification using the prefix computation model." In International Symposium on Parallel Computing in Electrical Engineering (PARELEC'06), pp. 139-144. IEEE, 2006.
4. Choorat, Thapana, and Akharin Khunkitti. "A packet classification algorithm." In 2010 The 2nd International Conference on Computer and Automation Engineering (ICCAE), vol. 1, pp. 145-148. IEEE, 2010.
5. Song, Haoyu, and John W. Lockwood. "Efficient packet classification for network intrusion detection using FPGA." In Proceedings of the 2005 ACM/SIGDA 13th international symposium on Field-programmable gate arrays, pp. 238-245. 2005.

6. Taylor, David E., and Jonathan S. Turner. "Scalable packet classification using distributed cross producing of field labels." In Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies. vol. 1, pp. 269-280. IEEE, 2005.
7. Linan, Chen, Lin Zhaowen, Ma Yan, Huang Xiaohong, and Li Chunqiang. "Multidimensional packet classification with improved cutting." In 2014 4th IEEE International Conference on Network Infrastructure and Digital Content, pp. 409-413. IEEE, 2014.
8. Li, Wei, and Xiufen Yu. "An online flow-level packet classification method on multi-core network processor." In 2015 11th International Conference on Computational Intelligence and Security (CIS), pp. 407-411. IEEE, 2015.
9. Pak, Wooguil, and Young-June Choi. "High performance and high scalable packet classification algorithm for network security systems." IEEE Transactions on Dependable and Secure Computing 14, no. 1 (2015): 37-49.
10. Guruprasad, S. P., and B. S. Chandrasekar. "An Efficient Bridge Architecture for NoC Based Systems on FPGA Platform." In International Conference on Intelligent and Interactive Systems and Applications, pp. 377-383. Springer, Cham, 2019.
11. Li, Jingjiao, Yong Chen, Cholman Ho, and Zhenlin Lu. "Binary-tree-based high speed packet classification system on FPGA." In The International Conference on Information Networking 2013 (ICOIN), pp. 517-522. IEEE, 2013.
12. Ganegedara, Thilan, Weirong Jiang, and Viktor K. Prasanna. "A scalable and modular architecture for high-performance packet classification." IEEE Transactions on Parallel and Distributed Systems 25, no. 5 (2013): 1135-1144.
13. Qu, Yun R., and Viktor K. Prasanna. "High-performance and dynamically updatable packet classification engine on FPGA." IEEE Transactions on Parallel and Distributed Systems 27, no. 1 (2015): 197-209.
14. Qu, Yun R., Hao H. Zhang, Shijie Zhou, and Viktor K. Prasanna. "Optimizing many-field packet classification on FPGA, multi-core general purpose processor, and GPU." In 2015 ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS), pp. 87-98. IEEE, 2015.
15. Zhou, Shijie, Yun R. Qu, and Viktor K. Prasanna. "Large-scale packet classification on FPGA." In 2015 IEEE 26th International Conference on Application-specific Systems, Architectures and Processors (ASAP), pp. 226-233. IEEE, 2015.
16. Chang, Yeim-Kuan, and Chun-Sheng Hsueh. "Range-enhanced packet classification design on FPGA." IEEE Transactions on Emerging Topics in Computing 4, no. 2 (2015): 214-224.
17. Khan, Ausaf Umar, Yogesh Suryawanshi, Manish Chawhan, and Sandeep Kakde. "Design and implementation of high performance architecture for packet classification." In 2015 International Conference on Advances in Computer Engineering and Applications, pp. 598-602. IEEE, 2015.
18. Yu, Weiwien, Srinivas Sivakumar, and Derek Pao. "Pseudo-TCAM: SRAM-based architecture for packet classification in one memory access." IEEE Networking Letters 1, no. 2 (2019): 89-92.
19. Huang, Jiamin, Yueming Lu, and Kun Guo. "A Hybrid Packet Classification Algorithm Based on Hash Table and Geometric Space Partition." In 2019 IEEE Fourth International Conference on Data Science in Cyberspace (DSC), pp. 587-592. IEEE, 2019.
20. Saqib, Fareena, Aindrik Dutta, Jim Plusquellic, Philip Ortiz, and Marios S. Pattichis. "Pipelined decision tree classification accelerator implementation in FPGA (DT-CAIF)." IEEE Transactions on Computers 64, no. 1 (2013): 280-285.
21. Jiang, Weirong, and Viktor K. Prasanna. "Large-scale wire-speed packet classification on FPGAs." In Proceedings of the ACM/SIGDA international symposium on Field programmable gate arrays, pp. 219-228. 2009.
22. Kennedy, Alan, and Xiaojun Wang. "Ultra-high throughput low-power packet classification." IEEE Transactions on Very Large Scale Integration (VLSI) Systems 22, no. 2 (2013): 286-299.
23. Tao, Zhang, Wang Yonggang, Zhang Lijun, and Yang Yang. "High throughput architecture for packet classification using FPGA." In Proceedings of the 5th ACM/IEEE Symposium on Architectures for Networking and Communications Systems, pp. 62-63. 2009.

## AUTHORS PROFILE



**Anita P** is a research scholar in the department of ECE at The Oxford College of Engineering Bangalore. She had worked as assistant professor at CMRIT, Bangalore. She obtained her B.E (ECE) degree in 2002 from (GVIT) Bangalore University, M.Tech degree in VLSI Design and Embedded system from CMRIT. She has almost nine years of academic teaching experience and worked for both NBA and NAAC. She has almost 4 publications in international journals. Her areas of interest are VLSI design, Analog and Digital Electronics.



**Dr. Manju Devi** is working as Professor and head in the department of ECE at The Oxford College of Engineering Bangalore. She has worked as Vice-Principal and professor at BTLIT, Bangalore. She obtained her B.E (ECE) degree in 1996 from Anna University, M.Tech degree in Applied Electronics from BMSCE, and Ph.D from Visvesvaraya Technological University (VTU), Karnataka. She has almost twenty two years of academic teaching experience and worked for both NBA and NAAC. She has almost 75 publications in international conference and journals. She is guiding eight students from Visvesvaraya Technological University (VTU), Karnataka. Her areas of interest are VLSI design, Analog and Mixed mode VLSI design and Digital Electronics.