

# Power Consumption of 100GE Packet Processor Depending on a Lookup Table Size

Teodora Komazec, Aleksandra Smiljanić, Milan Bjelica, and Mihailo Vesović

**Abstract**—Power consumption has become a critical performance measure in the design of modern routers. In this paper, we examine power consumption of packet processor implemented on advanced Xilinx UltraScale+ chip, depending on a size of lookup table. We used SDNet to implement IPv4 lookup tables and Vivado power analysis feature to analyze the power consumption of the high-speed router ports.

**Index Terms**—Field Programmable Gate Arrays, Power Consumption, Lookup Tables, Routers.

## I. INTRODUCTION

Internet is made of a vast number of host devices that are connected by routers. The complexity of Internet traffic puts high demands on the design of modern routers. Each Internet router maintains a routing table which is used to determine next-hop IP address for the received packet. Received IP packet contains information about final destination IP address of the packet.

Routing tables store only a portion of the destination IP address, better known as prefix. Prefix contains certain number of most significant bits of IP address. The rest of the bits are considered to be irrelevant for the lookup. The destination IP address of a packet can match more than one routing table entry, and the result of the lookup algorithm is the most specific of the matching entries. The lookup algorithms for IP routing are, for this reason, said to find the longest prefix match (LPM).

There are many reasons why implementation of different lookup algorithms is a very challenging task. Destination IP address of each incoming packet has to be compared to all entries in the routing table, while size of the routing tables can be very large. With the transition from IPv4 to IPv6 addresses, lookup algorithms take only more resources and time. Optimization of existing lookup algorithms is proposed in [1-4].

We examined power consumption [5] of packet processor

Teodora Komazec is with the School of Electrical Engineering, University of Belgrade, 73 Bulevar kralja Aleksandra, 11020 Belgrade, Serbia (e-mail: tea.k94@gmail.com).

Aleksandra Smiljanić is with the School of Electrical Engineering, University of Belgrade, 73 Bulevar kralja Aleksandra, 11020 Belgrade, Serbia (e-mail: aleksandra@etf.rs).

Milan Bjelica is with the School of Electrical Engineering, University of Belgrade, 73 Bulevar kralja Aleksandra, 11020 Belgrade, Serbia (e-mail: milan@etf.rs).

Mihailo Vesović is with the School of Electrical Engineering, University of Belgrade, 73 Bulevar kralja Aleksandra, 11020 Belgrade, Serbia (e-mail: mikives@gmail.com).

implemented on FPGA chip. FPGA chips are proven to be an appropriate solution for data plane programming. The key advantages of FPGA chips are that they are reprogrammable and reusable. This makes them adjustable to the specific needs of the customers, and adaptable over time.

Our main focus was to distinguish power consumption of lookup tables within packet processor, for a range of parameters. Xilinx developed SDNet data plane builder [6] to enable flexible packet processing while achieving high speed. SDNet can be used to implement different lookup algorithms.

Xilinx Vivado Design Suite enables implementation of designs using the UltraScale+ architecture. Xilinx UltraScale+ architecture brings many innovations to the FPGA design, such as next generation routing and enhanced logic blocks to optimize utilization of resources. In this paper, we used UltraScale+ architecture and Vivado power analysis feature to obtain power estimation results for implemented design. This feature is described in detail in [7].

The paper is structured as follows. Section II provides an overview of the Xilinx SDNet data plane builder. In this section we described basic SDNet components and their functionalities. Section III highlights the problems caused by the increased power consumption. In section IV, we analyzed the power consumption of packet processors, and their lookup modules. At the end, main conclusions are given.

## II. SDNET

SDNet is a tool developed by Xilinx for the purpose of generating systems with a variety of different packet processing functions. SDNet takes modular approach of designing systems. SDNet design begins with writing SDNet functional specification. The SDNet functional specification is, then, compiled to obtain a highly efficient hardware implementation. The resulting hardware implementation can achieve processing speed in the range of 10-100 Gb/s.

SDNet employs six types of components, called engines, to create systems with desired functionalities. These engines are: parsing engine, editing engine, tuple engine, lookup engine, user engine and system engine. Engines are triggered when all of their inputs arrive. Inputs that are supported by SDNet engines are packets and tuples. Tuples represent additional information about the packets that is carried by packet headers. Ports corresponding to these inputs need to be defined.

The main task of parsing engines is to read and decode packet headers. They acquire important information from the packet headers without modifying them. Editor engines are

used to modify packet data. They can add or remove packet headers. Main task of SDNet tuple engines is to process tuple data.

Lookup engines perform lookup based on the given key. SDNet supports different types of lookup algorithm: EM (Exact Match), LPM (Longest Prefix Match), TCAM (Ternary Content Addressable Memory) and direct address lookup.

User engines enable custom IP cores to be included in the SDNet design. System engines are used to interconnect other engines.

In the SDNet functional specification, lookup tables are defined using the following parameters: type of lookup algorithm, number of entries in lookup table, key width, width of returned value, tuple format of the response, and indication whether the lookup logic is external. Each of the lookup methods has limitations in terms of the number of entries, number of bits that the key contains, and width of the returned value. In this paper, we explored LPM and direct lookup.

LPM is the algorithm which performs lookup by comparing the most significant bits of the given key to a table of prefixes. The result of LPM lookup is the longest matching prefix from the table. As the result, we get response which contains associated prefix and indication if matching indeed occurred.

Direct address match utilizes the key as the direct address to access the stored value at that location. This algorithm is the easiest to implement. However, downside of direct address match lookup is that it does not scale well for larger tables.

### III. TESTING ENVIRONMENT

Power consumption has become one of the main constraints on the design of modern chips. Researchers and designers put extreme efforts to overcome problems caused by power dissipation. Shrinking the chips reduced the required power supply, which in turn reduced dynamic power. However, smaller geometries increased static power of the chips.

Device static power is the power that originates from transistor leakage. In ideal case, transistors are not drawing any current when they are in the off state. However, this is not a reality. There is a small amount of current that transistors draw even in the off state. Chips consists of millions or even billions transistors; so, the accumulated consumption can become significant. Device static power is a function of process, voltage and temperature [5]. It can be estimated by programing a blank bitstream into the device.

Dynamic power is the power that occurs as design runs. It varies with each clock cycle. Dynamic power is directly proportional to the square of the supply voltage.

Total on-chip power is calculated as a sum of device static and dynamic power. It represents the power that is consumed internally within the chip.

Characteristics of semiconductor materials are highly dependent on temperature. Ambient temperature is the air temperature of the device. Junction temperature is defined as the temperature range in which Xilinx guarantees that the device will operate according to the specification [7].

Main component of our design is the packet processor that was presented in [8]. Fig. 1 shows block diagram of the implemented component. It contains parsing, editing, lookup and tuple engines. Parsing engine extracts IP destination address from packet header, and forwards this IP address to the first lookup engine. The first lookup engine then performs LPM in the IPv4 lookup table, and provides output port number as a response. This is the port through which packet should be forwarded. The second lookup table uses direct address match to determine new source and destination MAC addresses, which will be forwarded to editing engine. At the end, editing engine performs adequate packet modifications.

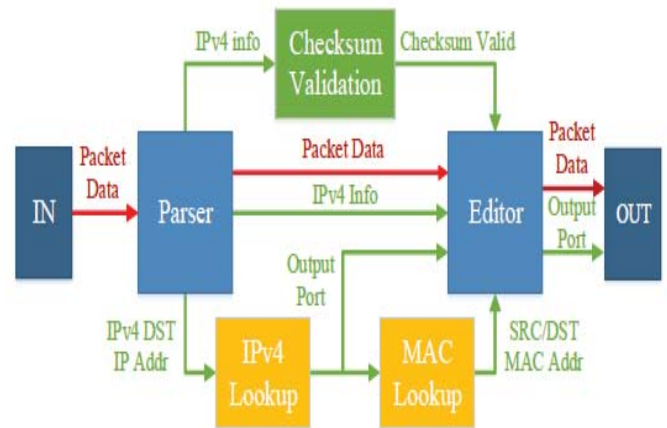


Fig. 1 SDNet packet processor – forwarding engine diagram [8]

### IV. RESULTS

We used Vivado power analysis feature to generate power reports for the design of packet processor. This feature can be used to obtain power reports through all stages of the flow: post-synthesis and post-implementation. For the purpose of this paper, we performed power analysis in both post-synthesis and post-implementation stage. The results are more accurate in post-implementation stage because unused signals and blocks are not considered in that stage.

Implemented design has two packet processors. Each of the packet processors is independent and contains two lookup tables as illustrated in Fig. 1. One of the lookup tables is used for IP forwarding and it utilizes the LPM algorithm, while the other is used to change the MAC addresses. The MAC table employs direct address match algorithm. MAC table, also known as FIB (Forwarding Information Base) determines output interface to which input interface should forward a packet.

First, we observed changes in power consumption while changing the depth of the LPM lookup table. We used lookup tables with 7, 8192, 32767 and 65535 entries. In this way, we wanted to cover both minimal and maximal possible numbers of entries when calculating the power consumption of the LPM lookup table. The maximal number of lookup entries supported by SDNet is appropriate for network edge routers. Since, core routers are larger than edge routers, as they

perform centralized routing for subnetworks, they require larger lookup tables.

Fig. 2. illustrates total on-chip power consumption.

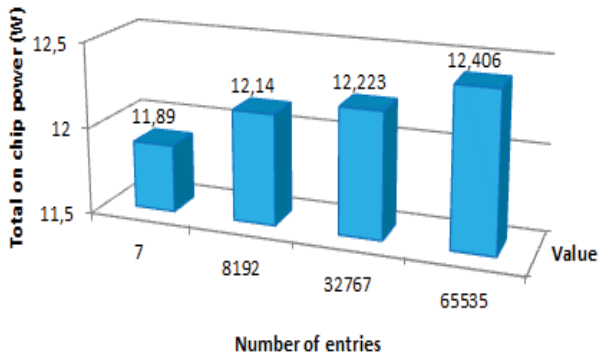


Fig. 2. Total on-chip power for longest prefix match lookup

The effect that the increase in the number of entries has on the increase of dynamic power consumption is shown in Fig. 3. Fig. 3. shows four columns. Each column has two components stacked. The bottom component represents dynamic power consumption while the top component illustrates static power consumption. We can see that the static power consumption did not change significantly. This is mainly because static power consumption primarily depends on the manufacturing process which is the same for all observed cases [9]. However, when comparing lookup table with minimal and maximal number of entries, there is an increase in dynamic power consumption for 5.5 %.

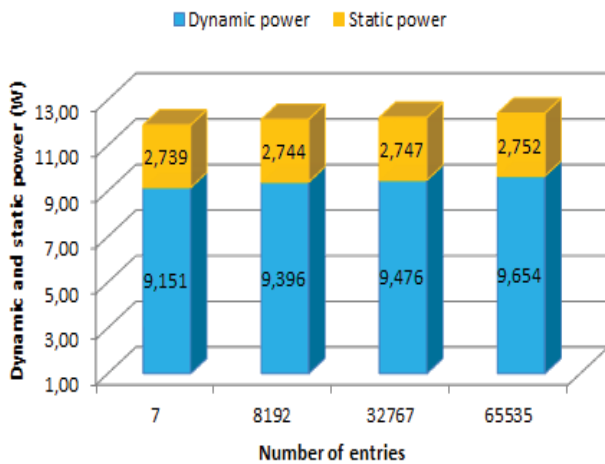


Fig. 3. Dynamic and static power for LPM lookup with respect to the number of entries

Generated power reports have a section about utilization of on-chip components. This section provides information about number of available and used components, as well as the information about power consumed by those components.

One of the on-chip components that shows significant

increase in percentage of utilized available units is Block RAM. Block RAM (Random Access Memory) components are used to store large amounts of data on FPGA chips. The FPGA has a limited number of Block RAM components, depending on its application. Fig. 4. shows how the Block RAM utilization increases with the increase of the number of entries in lookup table. Consequently, the power consumed by Block RAM components is getting higher, which is shown in Fig. 5. We can observe that there is an increase of 46.48 % in the power consumed by Block RAM components when number of entries in lookup table is maximal.

The amount of power that is consumed by Block RAM components is directly proportional to the amount of time during which they are enabled [7]. One of the ways to utilize resources more efficiently is to drive Block RAM enable low during clock cycles when Block RAM components are not used in the design.

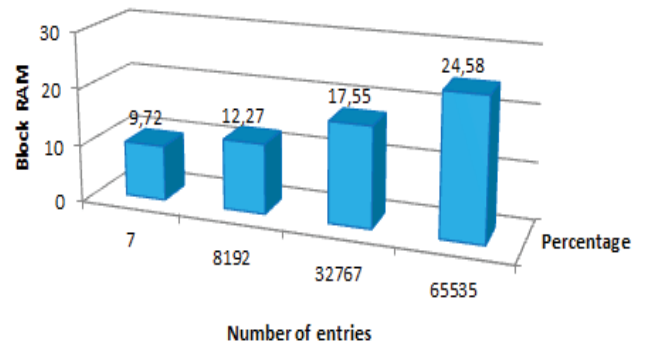


Fig. 4. The Block RAM utilization for LPM lookup with respect to the number of entries

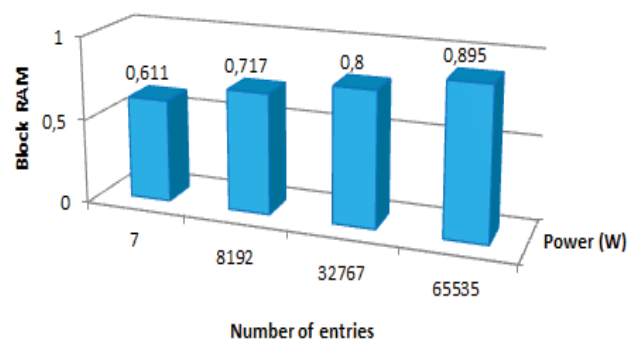


Fig. 5. Power consumed by block RAM units for LPM lookup with respect to the number of entries in lookup tables

Vivado power analysis feature gives an overview of hierarchical power consumption of a system. Fig. 6. shows increase in power consumption per packet processor with the

number of entries. It was shown that as the number of entries in lookup table increases, power consumed by both packet processors varies. The power consumed per packet processor when the number of entries of lookup table is maximal is almost 1,5 times higher than when the number of entries is minimal.

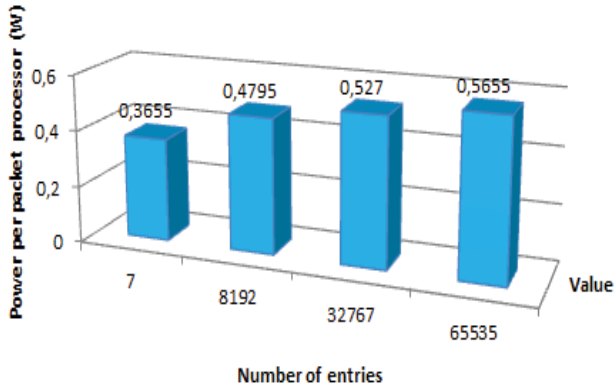


Fig. 6. Power per packet processor with respect to the number of entries in lookup tables

Once we have determined the output port through which packet should be forwarded, new source and destination MAC address have to be added to the header of the packet.

Direct address match algorithm is used for MAC lookup. MAC lookup tables usually have smaller depth. We used the following number of entries for lookup table: 128, 2048 and 4096. Fig. 7. shows increase in the total on-chip power with the number of entries in the lookup table. As in the case of LPM lookup, device static power did not change significantly with increase of the number of lookup table entries.

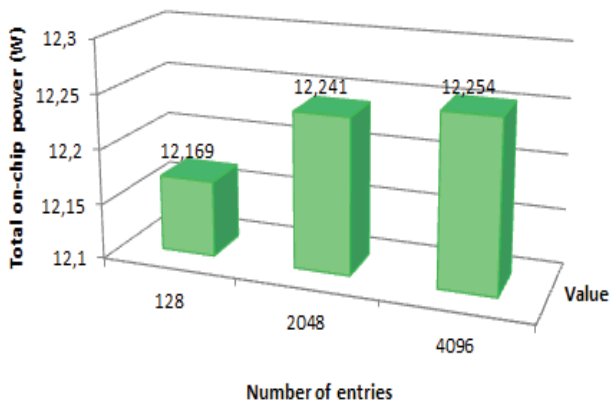


Fig. 7. Total on chip power for direct address match with respect to the number of entries

Increase in dynamic power with the change of the number of entries for direct address match lookup table is illustrated in Fig. 8. There is an increase of 0,9 % when

number of entries is 4096.

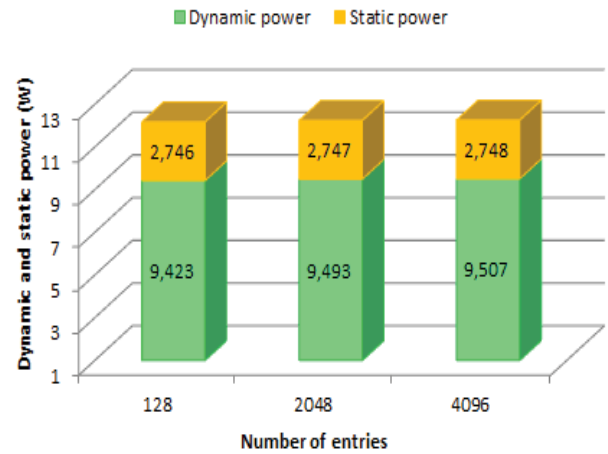


Fig. 8. Dynamic and static power for direct address match with respect to the number of entries

The percentage of available block RAM units that are utilized also records increase. This is shown in Fig. 9.

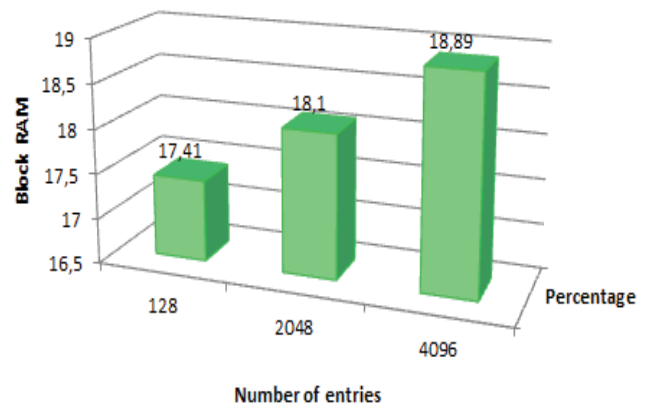


Fig. 9. Percentage of utilized block RAM units for direct address match with respect to the number of entries

## V. CONCLUSION

In this paper, we examined power consumption of LPM lookup tables and direct address match lookup tables. With the increase of the number of lookup table entries, there is also an increase of the total on chip power, dynamic power of a chip, percentage of utilized RAM blocks and number of used signals.

Changes in dynamic power consumption go to 5.5 % for LPM and 0.9 % for direct address match. The estimated static power consumption stays the same because it mostly depends on a manufacturing process. Increase in the number of used signals with the number of entries in lookup tables was expected and confirmed.

## REFERENCES

- [1] Levy, Gil; Kfir, Aviv, "Longest prefix match using a binary search tree with compressed hash tables," U.S. Patent Application No 16/039, 372, 2020.
- [2] Ruiz-Sanchez, Miguel A.; Biersack, Ernst W.; Dabbous, Walid, "Survey and Taxonomy of IP Address Lookup Algorithms," IEEE Network, vol. 15, no. 3, pp. 8-23, 2001.
- [3] Brown, David A, "Method and apparatus for storing sparse and dense subtrees in a longest prefix match lookup table," U.S. Patent No 6,539,369, 2003.
- [4] Huynh, Jeffrey, et al. "Chained lookups and counting in a network switch", U.S. Patent Application No 16/054, 797, 2020.
- [5] Kim, Nam Sung, et al. "Leakage current: Moore's Law meets static power computer," vol. 36, no. 12, pp. 68-75, 2003.
- [6] "SDNet Packet Processor User Guide", *Xilinx.com*, 2018. [Online]. Available: [https://www.xilinx.com/support/documentation/sw\\_manuals/xilinx2018\\_2/ug1012-sdnet-packet-processor.pdf](https://www.xilinx.com/support/documentation/sw_manuals/xilinx2018_2/ug1012-sdnet-packet-processor.pdf). [Accessed: 08-March-2020].
- [7] "Vivado Design Suite User Guide Power Analysis and Optimization", Xilinx.com, 2018. [Online]. Available: [https://www.xilinx.com/support/documentation/sw\\_manuals/xilinx2019\\_1/ug907-vivado-power-analysis-optimization.pdf](https://www.xilinx.com/support/documentation/sw_manuals/xilinx2019_1/ug907-vivado-power-analysis-optimization.pdf). [Accessed: 08-March-2020].
- [8] Vesović, Mihailo, Smiljanić Aleksandra, Radošević Andreja, "Evaluation of SDNet Packet Processors on Xilinx Chips", Proceedings of 5<sup>th</sup> International Conference on Electrical, Electronics and Computing Engineering, Palić, Serbia, 2018.
- [9] <https://www.sciencedirect.com/topics/computer-science/static-power> [Accessed: 08-March-2020].