

FPGA-based Router Virtualization: A Power Perspective

Thilan Ganegedara, Viktor K. Prasanna
Ming Hsieh Dept. of Electrical Engineering
University of Southern California
Los Angeles, CA90089
Email: {ganegeda, prasanna}@usc.edu

Abstract—Both Internet and semiconductor technology have advanced dramatically over the past decade. These advancements have made great impact on the conventional Internet infrastructure where networking equipment is dedicated on a per network basis. Router virtualization allows a single hardware router to serve packets from multiple networks while ensuring the same throughput and Quality of Service (QoS) guaranteed originally. In this paper, we study the effect of router virtualization, from a power consumption perspective, on the widely used Field Programmable Gate Array (FPGA) platform. An analytical model is proposed to estimate Layer-3 power consumption under different virtual router configurations. The analytical model is verified using post place-and-route results obtained using state-of-the-art FPGA and the models stand accurate with only a $\pm 3\%$ maximum error. Low power FPGA families are explored in this work to highlight the benefits of using such platforms in networking environments. Our experimental results show that by virtualizing, power savings proportional to the number of virtual networks can be achieved compared with non-virtualized routers.

I. INTRODUCTION

Over the past four decades, the Internet has grown from a network of few tens of nodes to a massive web of over 5 billion nodes [12]. This immense and continuous growth is constantly calling for high-bandwidth networking equipment in order to keep up with the increasing demands. With the advancements in semiconductor technology, satisfying these demands has not been a major challenge. However, the proliferation of networking devices (routers, switches, etc.) is posing a challenge in a different dimension- *Power*. The conventional approach has been to assign dedicated networking equipment on a per network basis. This has resulted in a poorly utilized Internet infrastructure where the network equipment operates full time, however the duty-cycle is low (especially at the edge-network level). Recently, *router virtualization* [1] was introduced to mitigate the administration issues of maintaining multiple physical networking equipment, to ease the management tasks. However, there are many more additional benefits in virtualizing a router. In this work, we are highlighting the benefits from a power consumption perspective.

Router virtualization can be described as the consolidation of multiple physical routers to a single shared hardware platform. This process of virtualization must be transparent to the user in the sense that before and after the process, the user should not experience any difference in the service received from the Internet Service Provider (ISP). From the

equipment standpoint, this translates to ensuring the throughput and latency requirements guaranteed originally. This is a challenging task since maintaining multiple routing tables while forwarding packets correctly on a given amount of hardware is not straightforward. Field Programmable Gate Arrays (FPGA) become an ideal candidate for such applications, mainly due to its reconfigurability, high performance and the abundant resources and parallelism provided [19], [14]. Using the memory and logic resources available on FPGA, it is possible to build virtualized routers that can host several virtual networks on a single chip.

In [13] it is shown that when router power taken into account, network layer (Layer-3) operation consumes nearly 62% of the total power. There have been many efforts on reducing Layer-3 power consumption of routers [20], [8], [10]. With virtualization, the power consumption of routers needs to be redefined. Sharing of the router platform allows the static power to be shared among the lookup engines and give more throughput per unit energy spent. Further, modern FPGAs come with various features that help one build architectures for low power applications. These features can be exploited to build virtualized forwarding engines that are power efficient compared with the existing Internet infrastructure.

In order to evaluate these power benefits quantitatively, we provide analytical models to estimate power for different virtual router scenarios and compare the benefits achieved by using each model. We give a comprehensive comparison of virtualized vs. non-virtualized routers from a power standpoint. In addition, we also compare the main two virtualization schemes to show the advantages and disadvantages of using each approach. The benefits of using low power features of FPGA is highlighted from both power and throughput view points. Finally we refine the power models that we propose by identifying the representative values for FPGA based routers.

We summarize our contributions in this work as follows:

- An accurate analytical model to estimate power savings achieved using router virtualization (Section IV)
- Exploration of low power FPGAs to achieve greater power benefits in networking applications (Section VI)
- Detailed power comparison of non-virtualized vs. virtualized schemes on state-of-the-art FPGA (Section VI-A)
- Power efficiency comparison for different virtual router configurations (Section VI-B)

II. BACKGROUND AND RELATED WORK

A. Background

1) *Networking on FPGA*: Various platforms are being used to build router architectures. FPGA, Ternary Content Addressable Memory (TCAM), Application Specific Integrated Circuit (ASIC) and Network Processing Unit (NPU) are the most prominent. As mentioned in Section I, FPGAs are extensively used in networking applications for IP (Internet Protocol) forwarding, firewalls, Network Intrusion Detection Systems (NIDS), etc. [19], [14]. Their reconfigurability and high performance capabilities are the most desirable features for networking applications. Further, the huge amount of logic and memory resources make them suitable for compute and memory intensive applications. Recently, there have been efforts towards low power FPGAs and device families targeted for such applications are available. These platforms have various architectural and algorithmic means by which, the power consumed by the device can be reduced.

In networking applications, the throughput, resource (logic and/or memory) consumption, latency of operation and the power consumed are the critical driving parameters. Using FPGA, it is possible to build architectures that meet most, if not all, these demands. Pipelining improves the performance while reducing the latency. Various algorithmic techniques can be employed in order to build resource efficient forwarding solutions and they can be easily mapped on to the reconfigurable FPGA fabric. Low power network equipment has gained much attraction recently with the interest in *green networks*. FPGAs provide various architectural and algorithmic features with which, significant power benefits can be achieved (by slightly sacrificing the throughput).

2) *Router Virtualization*: Router virtualization is an emerging research area that has attracted interest in both industrial and academic community. The main advantage of router virtualization is that all the networking equipment can be brought into a single administrative domain which makes the management tasks much easier. In addition to this, several other benefits such as, reduction in equipment cost and space are also prominent. In the context of this work, we consider virtualization of the data plane of the router. Control plane virtualization can be achieved simply by adopting the existing Operating System (OS) virtualization techniques. However, data plane virtualization requires more careful consideration since multiple factors such as throughput, resource limitation, power, etc. come into picture.

From an industry standpoint, deployment of router virtualization can be seen in the Juniper J series [9] routers and Cisco's Catalyst-6500 router [2], [3]. In the research community, two main categories of virtualization techniques can be found: 1) Separate and 2) Merged. As the names suggest, in the separate case [18], each virtual network gets its own lookup engine whereas in the merged case [5], [6], [11], [4], [17], all the virtual networks share a single lookup engine using some merging technique. The merging process exploits the structural similarity of tries to reduce the addition of new nodes by

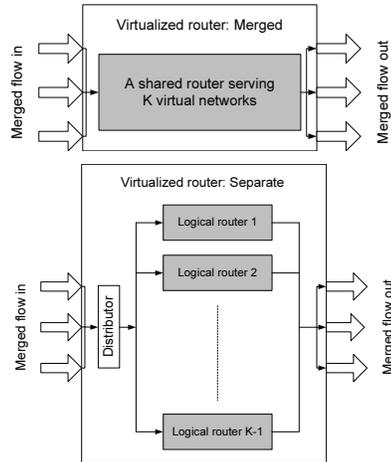


Fig. 1. Merged (top) and separate (bottom) router virtualization approaches

increasing node sharing. This leads to resource efficiency. The two router virtualization approaches are depicted in Figure 1.

B. Related Work

As mentioned in Section II-A, there are several techniques that implement router virtualization. However, for this work, we do not focus on a particular technique but evaluate two different classes of virtualization techniques, merged and separate. We generalize these two techniques in such a way that any given technique can be modeled by adjusting the parameters of the proposed model. The focus of most of the router virtualization techniques proposed in the literature are focused on reducing the memory requirement to store the routing tables [5], [6], [11], [4], [17]. The power benefits of virtualization is not studied nor quantified.

Improving the power efficiency of networking equipment is critical. With the constantly growing size and demands of the Internet, cooling of equipment has become a major issue. Several solutions have been proposed to improve the power efficiency of networking equipment. Both TCAM and algorithmic solutions exist in literature. In [7], [8], various trie partitioning methods are used to reduce the pipeline depth as well as per stage memory requirement, to reduce the power consumed per lookup. Memory balancing is integrated with these solutions to further enhance the power efficiency. TCAMs are known to be power hungry due to its massively parallel search. However, by properly organizing the TCAMs (with the aid of some algorithmic techniques), the power consumption of TCAM can also be reduced. In [20], the authors propose a load balancing scheme for multi-chip TCAM based IP lookup. By controlling the TCAM entries triggered by a lookup, power efficiency is achieved. IPStash [10] is an alternative solution for TCAM and is based on a memory architecture that is similar to set associative memory. By appropriately mapping the routing table to the set associative memory and using controlled prefix expansion the authors achieve 35% power savings compared to state-of-the-art TCAM solutions.

III. NOTATIONS AND ASSUMPTIONS

To denote the various schemes, we use a set of abbreviations and they are as follows:

- NV: Non-virtualized (conventional)
- VS: Virtualized separate approach
- VM: Virtualized merged approach

TABLE I
NOTATIONS AND SYMBOLS

Description	Symbol
No. of virtual networks	K
Virtual router i	VR_i
Lookup pipeline i	P_i
No. of stages per pipeline	N
Memory size of stage j of pipeline i	$M_{i,j}$
Logic slices in stage j of pipeline i	$L_{i,j}$
Power consumed	$P(\cdot)$
Device (FPGA/ASIC/etc.)	D
Leakage power	P_L
Utilization of virtual router i	μ_i
Merging efficiency	α

In this study, we make several assumptions to simplify the model we propose. However, it should be noted that these assumptions may be altered depending on the considered application.

Assumption 1: Network traffic is uniformly distributed among the K virtual routers. In other words, $\mu_i = 1/K$ for $i = 0, 1, \dots, K-1$

Assumption 2: All routing tables are of same size. An upper bound value is assumed considering real life edge-level routing table (10000 prefixes) to simulate a worst case scenario. This translates to $M_{i,j} = M_{k,j}$ for $i, k = 0, 1, \dots, K-1$ and $j = 0, 1, \dots, N$.

Assumption 3: In the case of non-virtualized and virtual-separate (described in Section II-B), packets belonging to different virtual networks are assumed to be properly distributed among the virtual router instances and the packet distributor energy is considered negligible.

Assumption 4: Merging of routing tables is generalized to make our analysis generic. *Merging efficiency* is defined as the amount of node overlap in a given level or equivalently:

$$\alpha = \frac{\text{number of common nodes}}{\text{total number of nodes}}$$

IV. ROUTER MODELS FOR POWER ESTIMATION

The power modeling done in this work is for Layer-3 lookup operation of a router. We consider three main types of routers: 1) Non-virtualized, 2) Virtualized-Separate and 3) Virtualized-Merged. We provide comprehensive models to estimate the power consumption of a router in these three different scenarios. For this work, we consider linear pipelined lookup architectures only. Hence, we consider tree/trie structures for IP lookup which are mapped onto the stages of a pipeline.

We consider three main components that contribute to the power consumption of a router's data plane operation. Leakage

power represents the static power dissipation while power consumed by logic and memory accounts for dynamic power. The static power is proportional to the area of the device used, while dynamic power highly depends on the clock frequency, the type (logic, memory, etc.) and amount of resources used. Hence, we first show the resource consumption for each setup and translate that to the power consumption on a per resource type basis.

When the router is not serving any packets, the logic or memory resources can be sent to an idle mode. Hence, during the off period of the duty cycle, the dynamic power can be assumed to be zero, but the static power is dissipated constantly since the device has to be operating despite the duty cycle. Turning off the logic and memory resources can be effectively done using flags (boolean values indicating whether service is required or not) and clock gating, respectively.

A. Non-virtualized

A non-virtualized router is the conventional approach in networking. Network equipment is dedicated for individual networks and the utilization of each equipment is fairly low due to the behavior of the edge-network users. The resource consumption is expressed in Eq. 1. The device D here refers to the chip on which the lookup engine is implemented. Since we have multiple equipment, multiple devices are required, hence, the static power consumed increases proportional to the number of networks. For dynamic power, we introduce the utilization for a fair comparison. As stated in Assumption 1, we assume a uniform distribution of packets across the virtual networks. If required, more complex distributions can be modeled by appropriately changing the μ_i values. Power consumed in the non-virtualized case is expressed in Eq. 2.

$$R_{NV} = \sum_{i=0}^{K-1} (D + \sum_{j=0}^{N-1} (L_{i,j} + M_{i,j})) \quad (1)$$

$$P_{NV} = \sum_{i=0}^{K-1} (P_L + \mu_i \sum_{j=0}^{N-1} (P(L_{i,j}) + P(M_{i,j}))) \quad (2)$$

B. Virtualized-separate

The virtualized-separate is very similar to the non-virtualized case, except for the fact that now we have a single shared platform hosting all the virtual routers. Hence, the static power dissipation is nearly brought down by a factor of K . However, the dynamic power consumption remains the same with its correlation to the utilization. The resource utilization and the power models are expressed in Eq. 3 and Eq. 4 respectively.

$$R_{VS} = D + \sum_{i=0}^{K-1} \sum_{j=0}^{N-1} (L_{i,j} + M_{i,j}) \quad (3)$$

$$P_{VS} = P_L + \sum_{i=0}^{K-1} \mu_i \left(\sum_{j=0}^{N-1} (P(L_{i,j}) + P(M_{i,j})) \right) \quad (4)$$

It should be noted that the separate approach has its disadvantages just as in the non-virtualized case. The number of separate lookup instances that can be implemented on a given device is limited by the available resources. Hence, the scalability of the separate virtualization approach dictated by the platform used. However, from a power perspective, we have fine grained control over the resources and temporarily turn off the resources that are not being used, while using a single device.

C. Virtualized-merged

This approach is radically different from the previous two cases. Here, the multiple virtual routing tables are merged (using some table merging technique), to produce a single lookup tree. The incoming packet stream, consisting of packets from different virtual networks, is sent through the lookup engine and based on the virtual network identifier (VNID), the router loads the corresponding routing table data and forwards the packet. Hence, the router hardware is time-shared among the virtual networks (in the case of separate router virtualization, the hardware was space-shared). The resource utilization and power model is given in equations Eq. 5 and Eq. 6 respectively. Since there is only one lookup pipeline, we use the index 0 for the pipeline instead of using i .

$$R_{VM} = D + \sum_{i=0}^{N-1} (L_{0,j} + \alpha \sum_{j=0}^{K-1} M_{i,j}) \quad (5)$$

$$P_{VM} = P_L + \sum_{i=0}^{N-1} (P(L_{0,j}) + P(\alpha \sum_{j=0}^{K-1} M_{i,j})) \quad (6)$$

In the case of merged, the scalability limitation has two aspects. First one is the resource limitation. The purpose of merging is to reduce the overall memory requirement. However, as we merge multiple routing tables, the total size of memory required to store the merged lookup tree may exceed the memory available on the device. This is one aspect. The second aspect is that when we merge two routing tables, the lookup engine has to be able to sustain the required throughputs of the two virtual networks, even in the worst case. When multiple such routing tables are merged, the throughput is shared among the virtual networks, hence at some point, the lookup engine may fail to sustain the required throughput. These two are the major limitations in the merged approach. However, the merged approach is more scalable than the separate approach considering the resource consumption.

V. VIRTUAL ROUTERS ON FPGA

In the previous section, we discussed how to model the power consumption of a virtual router on FPGA. We now focus on implementing these different architectures on state-of-the-art FPGA. For these experiments, we consider a Xilinx Virtex 6 platform (XC6VLX760) under two speed grade scenarios: 1) speed grade -2 for high performance and 2) speed grade -1L for lower power. This device was chosen considering its onboard resources, listed in Table II. In order to support

multiple virtual networks, having abundant on-chip resources, mainly Block RAM (Random Access Memory), distributed RAM and I/O (Input/Output) pins, is critical.

TABLE II
VIRTEX 6 XC6VLX760 DEVICE SPECS

Resource	Amount
Logic Cells	758K
Max. distributed RAM	8 Mb
Block RAM	26 Mb
Max. I/O pins	1200

In the proposed model, we consider three main contributors for power: static, logic and memory. We initially identify the representative values and/or functions for these two components (P_L , $P(L_{i,j})$ and $P(M_{i,j})$) on the aforementioned two platforms. For all our power calculations, we use the Xilinx XPower Analyzer (XPA) and XPower Estimator (XPE) tools. These tools provide a means by which a given design can be evaluated from a power standpoint at resource type level and at different operational frequencies.

A. Static power

The static power is the minimum power required to keep the device “powered up” with no switching. Even though static power does not depend on the frequency at which the device operates, it is proportional to the area of the device, process technology, and the operating temperature (which affects the leakage current). Various circuit optimization techniques can be adopted to reduce this component and we see such deployments in the low power FPGA devices. The main distinction in a high-performance and low power variants is the supply current, which is significantly lower (2000 mA difference) in the low power FPGAs.

In our case, we examined the static power dissipation of the device under the two speed grades and the results are as follows:

- Speed grade -2: $4.5 \pm 5\%$ W
- Speed grade -1L: $3.1 \pm 5\%$ W

The variation is based on the amount of resources used (or equivalently area covered by the used resources). We observed a maximum of $\pm 5\%$ deviation in our application and the value may vary depending on the resource consumption.

B. Power consumed by memory

Two types of memories exist on FPGA. Distributed RAM and Block RAM (BRAM). Even though both types of memories may be used in our applications, for simplicity, we assume only BRAM is used. On the device we are considering, 26 Mb of BRAM is available. However, BRAM (on Xilinx devices) is organized into 36 Kb blocks (contains two independent 18 Kb blocks). Hence, despite how small the amount of memory required, a BRAM block has to be assigned to serve the purpose. Therefore, BRAM power is determined by the number of blocks used rather than the total size of memory.

The other determining factors are 1) operating frequency, 2) duty cycle, 3) write rate, and 4) bit width of read out data. We assumed a write rate of 1% (low update rate) and 18 bit wide data for the comparison. The effect of bit width was negligible compared with the effect of other parameters.

We conducted experiments using the XPE tool to analyze the behavior of BRAM based on the size and the frequency. The observation was that BRAM power monotonically increased with both size and frequency. However, it should be noted that the behavior of 18 Kb and 36 Kb modules were different. The increase with respect to size was predictable as each BRAM block is an independent component. Figure 2 illustrate the power variation for a single BRAM.

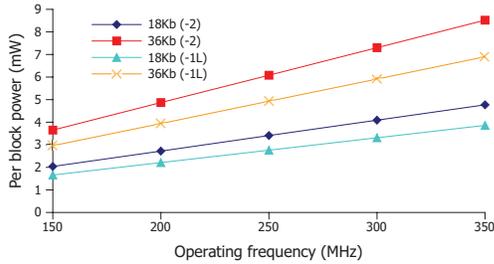


Fig. 2. BRAM power variation with operating frequency (Note: The number within parenthesis denotes the speed grade)

Using these details, we generate a power model for BRAM under different scenarios. The model is summarized in Table III. The notations used in the table are M - memory requirement in bits, f - operating frequency in MHz. These results can be used to predict the $P(M_{i,j})$ values in the models proposed in Section IV.

TABLE III
BRAM POWER MODEL

Setup	Power (μ W)
18Kb (-2)	$\lceil M/18K \rceil \times 13.65 \times f$
36Kb (-2)	$\lceil M/36K \rceil \times 24.60 \times f$
18Kb (-1L)	$\lceil M/18K \rceil \times 11.00 \times f$
36Kb (-1L)	$\lceil M/36K \rceil \times 19.70 \times f$

C. Power consumed by logic

In most studies related to networking, the power consumed by logic is considered negligible compared to that of memory. However, in our study, we identified that logic power (including signaling power) can become comparably significant. Logic power is distributed among Look-Up Tables (LUT), shift registers, distributed RAM and flip-flops. Signal power includes the power dissipated when communicating among the aforementioned logic resources as well as memory components. In order to avoid the clutter, we treat both logic and signal power as a whole and present the results.

In order to evaluate logic power, we stay at the granularity of a single processing element (PE) of a pipeline stage. This

includes the stage registers and any type of logic resources that are required to perform the memory access and computations required at each stage. In the case of our uni-bit trie, the logic resource consumption was as follows:

- Slice registers as flip-flops: 1689
- Slice LUTs as logic: 336
- Slice LUTs as memory: 126
- Slice LUTs as routing: 376

The power consumed depends on the frequency of operation and the amount of resources used. The observation was that logic power linearly increases with the number of pipeline stages. The variation with frequency is illustrated in Figure 3. Further, for a trie based IP lookup implementation, per stage logic power dissipation as a function of operating frequency, in MHz, can be expressed as:

- Speed grade -2: $5.180 \times f \mu$ W
- Speed grade -1L: $3.937 \times f \mu$ W

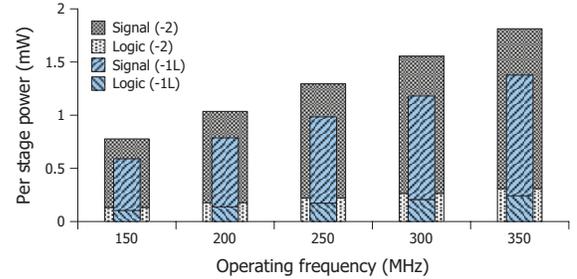


Fig. 3. Per stage logic and signal power consumption (Note: The value inside parenthesis denotes the speed grade)

D. Pipelined IP lookup

Algorithmic (i.e. trie/tree based) IP lookup have become popular over TCAM based IP lookup due to their flexibility and scalability. Mapping such trie/tree based solutions to FPGA platforms can be done efficiently. Most router virtualization solutions are trie based [5], [6], [4], [17]. Hence we use trie as the representative example. Each trie level is mapped onto a pipeline stage and each stage is associated with an independently accessible memory [7], [11], [8]. When a lookup request is received, the packet traverses the pipeline similar to the trie traversal and at the end of the pipeline, outputs the appropriate next-hop port information (NHI). Generally, the NHI information is stored at the leaf nodes of the trie (nodes that do not possess any children nodes) using techniques such as leaf pushing [16], in order to reduce the memory consumption for the trie storage. In the case of virtualization, a leaf node is simply a vector that has routing information corresponding to all the considered virtual networks. And the vector is indexed using the VNID to extract the forwarding information [5], [4].

The three cases considered here (non-virtualized, virtualized-separate and virtualized-merged) have the same architecture with the following distinctions:

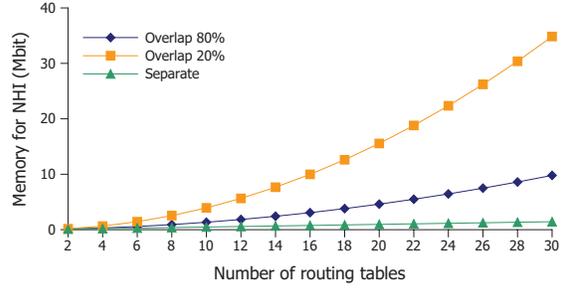
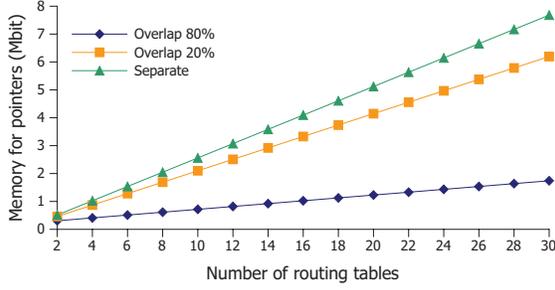


Fig. 4. Pointer and NHI memory requirements for merged ($\alpha = 80\%$ and $\alpha = 20\%$) and separate approaches

- Non-virtualized implements a single lookup engine on a single device and all the devices are dedicated on a per network basis. For a K virtual network scenario, K devices are required.
- Virtualized-separate implements multiple lookup engines on a single device and between two lookup engines, there is no resource sharing except for the FPGA fabric itself.
- Virtualized-merged implements a single shared lookup engine and all the virtual networks share the same memory and logic in the lookup engine. The amount of logic used, remains almost the same as in the two other cases, however, the amount of memory required may significantly increase depending on the merging efficiency, α .

E. Routing tables

Router virtualization is most effective at edge level of the network since the problem of underutilization is most prevalent at the edge network level. In order to demonstrate the results for a more realistic scenario, we use routing tables from real networks obtained from [15]. To simplify the implementation, we assume all the routing tables to be of same size and we use the largest routing table we obtained from [15] to report the results for the worst case scenario. This particular routing table consisted of 3725 prefixes and when the corresponding trie had 9726 nodes without leaf pushing and 16127 nodes with leaf pushing. Figure 4 demonstrates the effect of virtualization on memory under different scenarios and illustrates the amount of memory used for pointers (non-leaf nodes) and for NHI/forwarding information (leaf nodes).

It can be seen that the memory saving achieved by the merged schemes is highly dependent on the node overlap percentage or merging efficiency α . Also it is clear that pointer saving becomes less and less effective as the number of virtual routers increase and α decreases. Since we cannot assume any particular structure for the considered routing tables, merging efficiency cannot be determined in advance and leads to indeterministic memory requirements whereas in separate (even non-virtualized) approach, the memory requirement is deterministic. Also, it should be noted that merging schemes are appropriate (from a memory standpoint) when the number of virtual routers is small.

VI. VIRTUALIZED ROUTER: POWER PERFORMANCE

In the previous section, we observed how each component behaves for the two scenarios and derived relationships in terms of operating frequency and amount of logic resources required. However, the standard metric used for power measurements in routers is Watts per Gbps (Giga bits per second). This describes how much energy is spent to provide a unit throughput. In order to evaluate the two virtualized routers against the non-virtualized router, we analyzed the post place-and-route behavior of these architectures on the device we considered earlier (Virtex 6 XC6VLX760) under the two speed grades (-2 and -1L). Without loss of generality, for all pipelines we assume a length of 28 stages.

A. Total power dissipation: Experimental vs. estimation

Here, we validate the models proposed in Section V against the experimental results we obtain. To clearly illustrate the performance of difference schemes, we first compare the total power utilized by all the schemes (non-virtualized, virtualized-separate and virtualized-merged ($\alpha = 20\%$ and $\alpha = 80\%$)) and then we show the comparison of all the virtualized schemes. These results are shown in Figure 5 and Figure 6, respectively. It can be observed clearly that the non-virtualized router consumes power proportional to the number of (virtual) networks. In contrast, virtualized routers consume very small amount of power since the static power consumed by the lookup engine is shared among the considered virtual networks.

Another interesting observation is that in Figure 6, the total power dissipation decreases with the increasing number of virtual networks. According to the model (Eq. 4), the power consumption must remain the same since only one lookup engine is active at a given time (Assumption 1). However, the experimental value decreases due to various hardware optimizations applied when implementing multiple parallel architectures.

We limited the maximum number of virtual networks to 15 since in the case of virtualized-separate, the I/O pin requirement exceeded when the number of virtual networks was increased. In a complete router implementation (parsing, lookup, editing, scheduling, etc.), this number may become even less when other inputs and outputs are considered. The goal of this work is to analyze the power behavior of the lookup portion of a router. Therefore, the above implementation stand as an accurate prototype for the considered purpose.

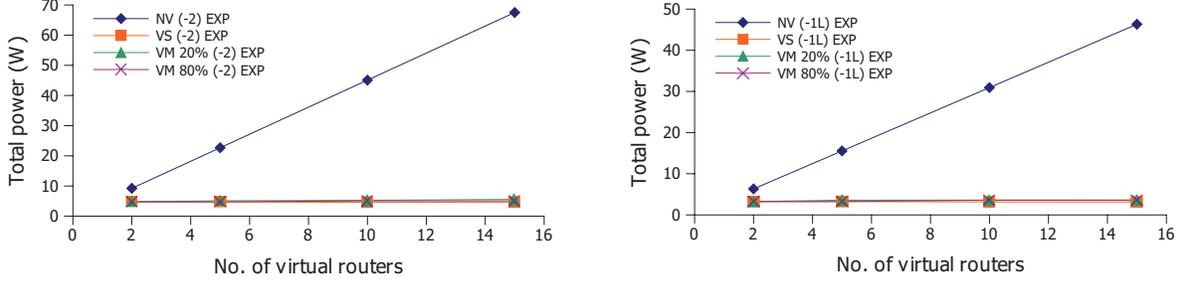


Fig. 5. Comparison of total power consumption in virtualized and non-virtualized schemes for speed grades -2 (left) and -1L (right)

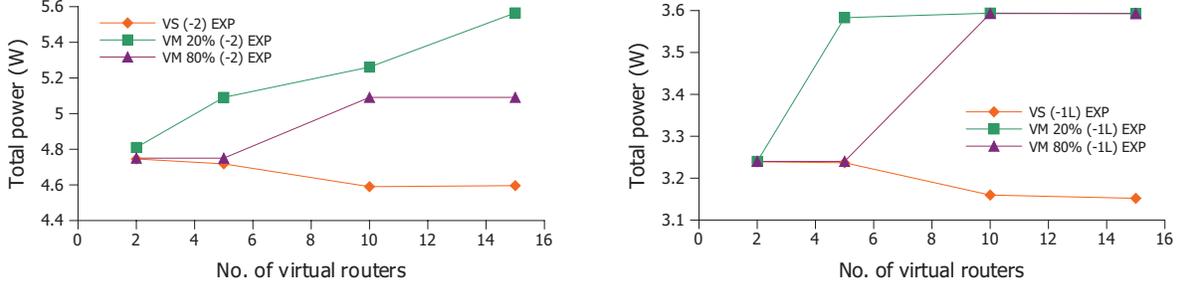


Fig. 6. Comparison of total power consumption in different virtualized schemes for speed grades -2 (left) and -1L (right)

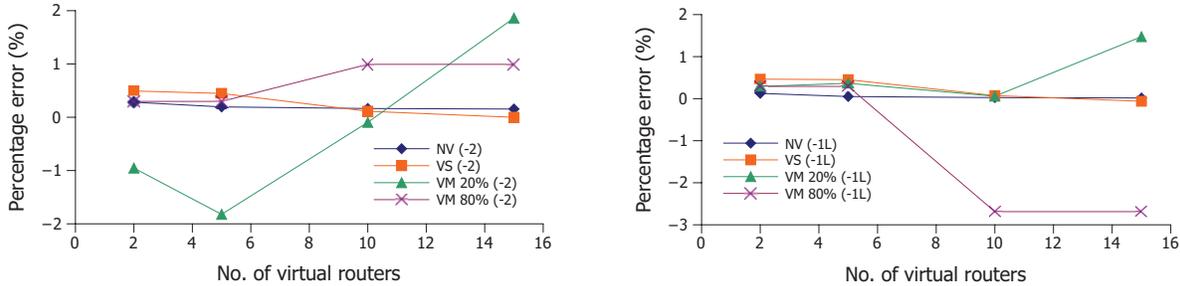


Fig. 7. Percentage error of the model estimation compared with the experimental results for speed grades -2 (left) and -1L (right)

Figure 7 shows the percentage error of the models we proposed in Section V. The percentage error is calculated as follows:

$$\text{Percentage error} = \frac{P_{\text{Model}} - P_{\text{Experimental}}}{P_{\text{Experimental}}} \times 100\%$$

It can be seen that the model estimation is highly reliable with a maximum error of $\pm 3\%$. The cause for this error is the various hardware optimizations that are performed, by the synthesis tool, when the amount of resources used, increases. As shown in the figure, for non-virtualized and virtualized-separate, the error is much less compared to that of virtualized-merged. In the merged approach, we use more BRAM per pipeline stage, to accommodate the increasing number of virtual routers. The synthesis tool performs various routing and placement optimizations to improve the performance of the design which causes our predictions to deviate slightly from the exact measurement. Nevertheless, the model we proposed here provide an accurate means by which the power consumption of virtualized routers can be estimated.

B. Power efficiency

In the context of lookup engines, one important metric is the packet handling rate. In this work, we use Giga bits per second as the metric to measure packet handling rate and to compute this, we use minimum packet size as 40 bytes. A router may use more and more power to support higher throughput. In order to compare such architectures with power efficient architectures, we use the power dissipated per unit throughput as the metric for our comparisons. Figure 8 illustrates the comparison of the three approaches with respect to the considered metric.

The lower the mW/Gbps number is, the better the architecture. Therefore, by analyzing the results in Figure 8, the virtualized separate approach yields the best power efficiency. The conventional router is the second best while merged approach shows the worst performance. The main reason behind the poor performance of the merged approach is the reduction in operating frequency (hence, throughput) with the increasing number of virtual routers. Due to the higher resource consumption, the operating frequency decreases significantly. The

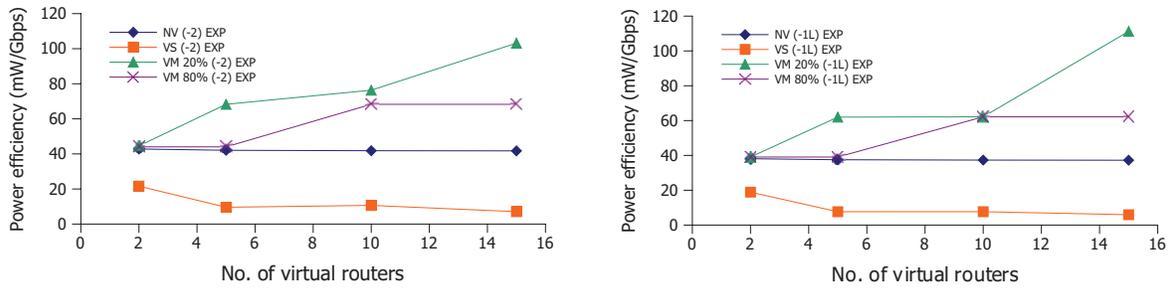


Fig. 8. Power dissipated per unit throughput for speed grades -2 (left) and -1L (right)

power consumed by the resources increase, but the throughput drops. As a result, the power per unit throughput increases. Also, the performance difference between the two cases, $\alpha = 20\%$ and $\alpha = 80\%$ is also intuitive. When the merging efficiency is much less, the amount of resources consumed by the router increases, while the throughput decreases.

In both total power consumed and power per unit throughput, we have presented the results for the two speed grades (-2 and -1L). We observed a 30% less power consumption when speed grade -1L was chosen compared to speed grade -2. However, the power saving comes at the expense of throughput. This fact becomes clear when comparing the two speed grades with respect to mW/Gbps in Figure 8. The two speed grades perform almost the same way with same variation and performance numbers. Hence, low power FPGAs are suitable in environments where throughput is not the major concern.

VII. CONCLUSION

Due to the proliferation of networking devices, power dissipated in the network is drastically increasing. Router virtualization was proposed to overcome the complications and disadvantages of the conventional networks. In order to achieve these benefits, the underlying hardware must be capable of supporting virtualization. With its reconfigurability and abundant resources, Field Programmable Gate Arrays (FPGAs) become an attractive platform. In this work, we demonstrated how various virtualized router architectures perform on state-of-the-art FPGA platforms and highlighted the benefits of using low power FPGA families for networking applications. The experimental results revealed that for maximum power efficiency, the virtualized separate approach gives the best results while the results of the merged approach varied depending on the merging efficiency. However, it should be noted that the separate approach suffers from scalability issues with increasing number of virtual routers because of resource exhaustion. We demonstrated the benefits of using low power families of FPGA and discovered that they give same power efficiency as the high-speed platforms while consuming low power and yielding lower throughput.

REFERENCES

[1] J. Carapinha and J. Jiménez. Network virtualization: a view from the bottom. In *Proceedings of the 1st ACM workshop on Virtualized infrastructure systems and architectures*, VISA '09, pages 73–80, New York, NY, USA, 2009. ACM.

[2] Cisco. Cisco catalyst 6500 virtual switching system 1440. <http://www.cisco.com/en/US/products/ps9336/index.html>.

[3] Cisco. Hardware and software virtualized routers. http://www.cisco.com/en/US/solutions/collateral/ns341/ns524/ns562/ns573/white_paper_c11-512753_ns573_Networking_Solutions_White_Paper.html.

[4] J. Fu and J. Rexford. Efficient ip-address lookup with a shared forwarding table for multiple virtual routers. In *Proceedings of the 2008 ACM CoNEXT Conference*, CoNEXT '08, pages 21:1–21:12, New York, NY, USA, 2008. ACM.

[5] T. Ganegedara, W. Jiang, and V. Prasanna. Multiroot: Towards memory-efficient router virtualization. In *Communications (ICC), 2011 International Conference on*, 2011.

[6] T. Ganegedara, H. Le, and V. Prasanna. Towards on-the-fly incremental updates for virtualized routers on fpga - unpublished article. In *Field Programmable Logic and Applications (FPL), 2011 International Conference on*, 2011.

[7] W. Jiang and V. Prasanna. Multi-way pipelining for power-efficient ip lookup. In *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*, pages 1–5, 30 2008-dec. 4 2008.

[8] W. Jiang and V. Prasanna. Towards green routers: Depth-bounded multi-pipeline architecture for power-efficient ip lookup. In *Performance, Computing and Communications Conference, 2008. IPCCC 2008. IEEE International*, pages 185–192, dec. 2008.

[9] Juniper. Jcs1200 control system. <http://www.juniper.net/us/en/local/pdf/whitepapers/2000261-en.pdf>.

[10] S. Kaxiras and G. Keramidas. Ipstash: a power-efficient memory architecture for ip-lookup. In *Microarchitecture, 2003. MICRO-36. Proceedings. 36th Annual IEEE/ACM International Symposium on*, pages 361–372, dec. 2003.

[11] H. Le, T. Ganegedara, and V. Prasanna. Memory-efficient and scalable virtual routers using fpga. In *Field Programmable Gate Arrays (FPGA), 2011 International Symposium on*, 2011.

[12] A. M. Lyons, D. T. Neilson, , and T. R. Salamon. Energy efficient strategies for high density telecom applications. http://imsresearch.com/press-release/Internet_Connected_Devices_About_to_Pass_the_5_Billion_Milestone&from=all_pr.

[13] A. M. Lyons, D. T. Neilson, , and T. R. Salamon. Energy efficient strategies for high density telecom applications. Princeton University, Supelec, Ecole Centrale Paris and Alcatel-Lucent Bell Labs Workshop on Information, Energy and Environment, June 2008.

[14] NetFPGA. Netfpga boards. <http://netfpga.org/>.

[15] Potaroo. Bgp analysis reports. <http://bgp.potaroo.net/>.

[16] M. Ruiz-Sanchez, E. Biersack, and W. Dabbous. Survey and taxonomy of ip address lookup algorithms. *Network, IEEE*, 15(2):8–23, 2001.

[17] H. Song, M. Kodialam, F. Hao, and T. Lakshman. Building scalable virtual routers with trie braiding. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9, 2010.

[18] D. Unnikrishnan, R. Vadlamani, Y. Liao, A. Dwaraki, J. Crenne, L. Gao, and R. Tessier. Scalable network virtualization using fpgas. In *Proceedings of the 18th annual ACM/SIGDA international symposium on Field programmable gate arrays, FPGA '10*, pages 219–228, New York, NY, USA, 2010. ACM.

[19] Xilinx. Xilinx xcell journal. <http://www.xilinx.com/publications/xcellonline/>.

[20] K. Zheng, C. Hu, H. Liu, and B. Liu. An ultra high throughput and power efficient tcam-based ip lookup engine. In *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 3, pages 1984–1994 vol.3, march 2004.