

Dynamic Power Consumption in Virtex™-II FPGA Family

Li Shang
Princeton University
EE Dept., Princeton, NJ 08540
lshang@ee.princeton.edu

Alireza S Kaviani
Xilinx Inc.
2100 Logic Dr., SanJose, CA95124
alireza.kaviani@xilinx.com

Kusuma Bathala
Xilinx Inc.
2100 Logic Dr., SanJose, CA95124
kusuma.bathala@xilinx.com

ABSTRACT

This paper analyzes the dynamic power consumption in the fabric of Field Programmable Gate Arrays (FPGAs) by taking advantage of both simulation and measurement. Our target device is Xilinx Virtex™-II family, which contains the most recent and largest programmable fabric. We identify important resources in the FPGA architecture and obtain their utilization, using a large set of real designs. Then, using a number of representative case studies we calculate the switching activity corresponding to each resource. Finally, we combine effective capacitance of each resource with its utilization and switching activity to estimate its share of power consumption. According to our results, the power dissipation share of routing, logic and clocking resources are 60%, 16%, and 14%, respectively. Also, we concluded that dynamic power dissipation of a Virtex-II CLB is 5.9 μ W per MHz for typical designs, but it may vary significantly depending on the switching activity.

1. INTRODUCTION

Recent advances in semiconductor process technology has led to rapid scaling of transistor dimensions, allowing a large number of them to be packed on the same chip. Field Programmable Devices (FPDs), which consume higher number of transistors compared to their alternative Application Specific Integrated Circuits (ASICs), have also enjoyed a rapid growth due to these technology advancements. High density of transistors on the same chip has made power consumption one of the major challenges of deep submicron IC design [1]. Traditionally, FPD power consumption has been less of a concern compared to their speed and area efficiency. However, it is likely that large FPDs at the leading edge of CMOS design will soon face tough challenges regarding power consumption.

Large FPDs, which are often called Field Programmable Gate Arrays (FPGAs), consist of a set of logic blocks and a flexible routing structure to connect them together. Using automated CAD tools, designers may program the logic blocks and their corresponding interconnect to implement any desired application

within a reasonable amount of time. Such flexibility and fast time to market, however, comes with the expense of additional transistors and metal resources that are only partially utilized. Therefore, we need to identify utilized logic and routing resources that contribute to a signal for analyzing the dynamic power consumption. Our analysis and results in this paper can be used in 2 ways: 1) Better understanding of where power is consumed in FPGAs will help design of future power-efficient FPGAs. 2) Detailed understanding of power consumption distribution will help expert designers to reduce or control the power characteristics of their design.

The remainder of this section goes over the related work. Section 2 outlines the necessary background including an introduction to Virtex-II, and various types of power consumption to narrow down the focus of this paper. Section 3 explains the role of effective capacitance, followed by Section 4 that explains our methodology for power estimation. In Section 5 we summarize the results and provide an overview of power distribution in FPGA fabric. Finally, in Section 6 we discuss the results to achieve a better understanding and propose future work.

1.1. Related Work

Analysis and estimation for power dissipation of large FPGAs has received little attention compared to that of standard cell ASIC, which has been extensively studied in the literature. Previous works [2,3] have analyzed the power distribution of Xilinx 4000 family, determining the distribution of various resources in the device. Both works have also suggested architectural improvements based on reducing the power supply to make FPGAs power-efficient. A recent work [4] characterizes the power dissipation of the FPGA interconnection using Manhattan distances between logic blocks. This work uses an iterative approach to update the signal values, but the iteration process does not always converge. Work in [4] also considers 4000 family, but it does not capture power distribution of internal resources.

Our work is distinguished from the above in several aspects: unlike previous work we consider state-of-the-art FPGA fabric, which already uses advanced process technology with reduced power supply. Therefore, relevant architectural suggestions in those works need to be revisited based on the new data. In addition, our results are more accurate due to our access to the detailed schematics of the FPGA circuits. Finally, we have based our methodology and results on real large designs as opposed to previous works that used smaller circuits, resulting in somewhat different results.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

FPGA '02, February 24-26, 2002, Monterey, California, USA.
Copyright 2002 ACM 1-58113-452-5/02/0002...\$5.00.

2. BACKGROUND

In this section, we first introduce and justify our choice of target device. Then, we follow by explaining various components of power consumption. At the end, we describe the focus of this paper and introduce our methodology.

2.1. Virtex-II FPGA

The largest FPGA device, which is recently introduced to the market, is in Virtex-II family. Virtex-II uses 0.15-micron process with eight layers of metal at 1.5 V power supply. In addition to advancements in its process technology, Virtex-II is the first FPGA with fully buffered interconnect, which may be considered as a turning point in routing architecture. Figure 1 shows 2v40, which is the smallest member of Virtex-II family. As shown in the figure, Virtex-II includes a number of hard cores, including memory blocks, IO blocks, digital clock managers, encryption circuitry, and custom multipliers. However, most of the silicon area in the largest members of the family is consumed by what is referred to as programmable *fabric*. While we admit to the importance of power consumption of hard cores on the FPGA, we focus on the fabric for the following reason. Power and performance of FPGAs are often compared to their standard cell ASIC counterparts that use less silicon area for realizing the same functionality. The power inefficiency of FPGAs is often contributed to its programmable fabric that trades additional silicon area for its flexibility. The hard cores in FPGAs are expected to perform as good as their equivalent in ASIC, or in some cases even better due to their custom design. Since we would like to identify the source of power inefficiency in FPGAs, we only consider the programmable fabric of FPGAs.

Virtex-II fabric consists of Configurable Logic Blocks (CLBs), which are connected using a rich set of routing resources. Each Virtex-II CLB contains four slices, where each slice consists of two 4-input Lookup Tables (LUTs), two Flip-flops (FFs), and a variety of dedicated circuitry to accommodate more efficient implementation of some specific logic. Virtex-II uses a *segmented* routing structure to minimize the number of transistors and wires that a signal needs to traverse to reach its destination.

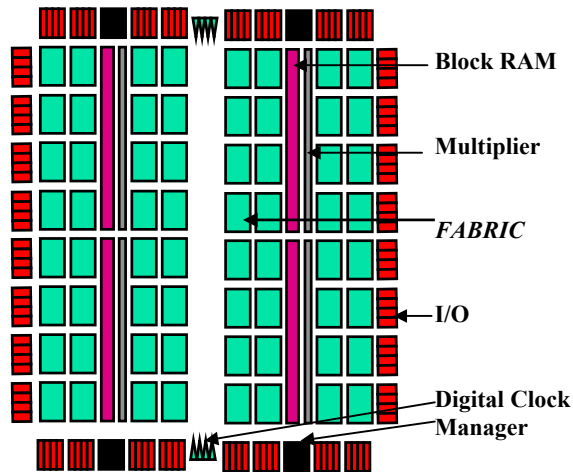


Figure 1. Virtex-II platform FPGA.

The segmented routing architecture includes wires that travel two CLBs (called Doubles), six CLBs (called Hexes), and the length of the chip (called Longs), in both vertical and horizontal dimensions. There are also pass transistors and buffers associated with each set of wires. For example, when we refer to a Hex switch for its power consumption we are considering both wire and its supporting transistors. There are also two sets of switches to connect the wire segments to the inputs and outputs of each CLB; we refer to these sets as Input Crossbar (IXbar) and Output Crossbar (OXbar). The CLB slices are also referred as logic and the above five sets of switches comprise interconnect. In addition to logic and interconnect, we will consider global resources and switches that accommodate the clocking for the circuits. More detailed information regarding Virtex-II architecture can be found in [5].

2.2. Power Consumption

There are two types of power consumption in FPGAs: static and dynamic. In CMOS logic, which includes SRAM-based FPGAs, leakage current is the only source of static power dissipation. There are two major sources of leakage current: 1) reverse biased PN-junction current, 2) subthreshold channel conduction. Both these components have similar characteristics such as high dependency to temperature, process variation, and logic states of the circuit. Leakage current has often been ignored in the past due to its negligible amount; but this is likely to change with scaling of transistor dimensions. Scaling often comes with a reduction in power supply voltage (V_{dd}), and lower V_{dd} reduces the speed. To maintain or increase the speed we need to reduce the threshold voltage (V_{th}) of the transistor along with the scaling. However, the subthreshold channel current of a transistor exponentially increases with any V_{th} decrease, leading to a rapid increase in static power consumption. We believe in the importance of static power dissipation in future FPGAs, and have analyzed it elsewhere. In this paper, however, we focus on the dynamic components of power dissipation. According to our empirical results, the static power is between 5-20% of total power dissipation in Virtex-II, depending on the temperature, device, running frequency, and the design.

2.2.1. Dynamic power consumption

Dynamic power dissipation is caused by signal transitions in the circuit. A higher operating frequency leads to more frequent signal transitions and results in increased power dissipation. The most significant source of dynamic power consumption in CMOS circuits is the charging and discharging of capacitance. This can be modeled as $P = \sum_i C_i V_i^2 f_i$ (1), where C_i , V_i , and f_i are

the capacitance, voltage swing, and operating frequency of resource i , respectively [6].

Another component of dynamic power dissipation, also caused by signal switching, is *short-circuit* power. According to our simulations, short-circuit current in FPGA interconnect is less than 10% of the total. This is consistent with the literature [6] because interconnect short-circuit power dissipation is mostly caused by switching of inverters in the buffers. However, short-circuit power in the logic inside the CLB slice is a higher percentage of its total power. For the sake of simplicity, we emulate short-circuit power with an additional capacitance.

To calculate total power dissipation we consider three factors: First, we define *effective capacitance* as the sum of parasitic

effects due to interconnection wires and transistors, and the emulated capacitance due to short-circuit current. We will obtain this capacitance for each resource as it is explained in the next section. The second important factor is the *resource utilization*. In typical FPGA designs, the majority of the resources are not used after the configuration and thus they will not consume any dynamic power. Since the resource utilization varies with design, we consider a large set of real circuits to obtain statistically valid results. The third factor in determining power dissipation is the *switching activity*, which is defined as the number of signal transitions in a clock period. For example, a clock signal has a switching activity of two. The switching activity for each resource also requires a statistical representation, because it depends not only on the type of design, but also the input stimuli. We explain our methodology to obtain statistical representations for resource utilization and switching activity in Section 4, and present their corresponding results in Section 5.

3. EFFECTIVE CAPACITANCE

We obtain the effective capacitance of each resource using two methods: measurement and spice simulation. Using two sources of data helped us verify our results to improve their accuracy.

3.1. Measurement

In order to measure the effective capacitance of each resource, we first implement a simple reference circuit in the FPGA and measure its power. Then the target resource is added to the reference design and the power is measured again. The difference between first and the second power measurement determines the power dissipation of the target resource. To improve the accuracy, both reference design and the target resource are replicated to fill up the device in both measurements. We used 2V1000, with 5120 slices (10240 LUTs), for all our measurements. For further verification we have used several frequencies for each resource; the linear change of our power measurements with respect to frequency insures the correctness of our results. In addition, the power supplies of the FPGA core, FPGA I/O, and the testing environment are isolated to minimize the possibility of errors. The important FPGA resources that are used for measurement are described in subsection 2.1.

3.2. Transistor-level Simulation

Since we have access to the schematic of internal circuits in Virtex-II, transistor-level simulation is also used to identify the effective capacitance of resources. The primary reason for using simulation is that there are some resources whose capacitance can not be isolated for measurement. In addition, using simulation will enable us to investigate future architectures for the purpose of power efficiency. Needless to say that simulation helps us to improve the accuracy of our results. In many cases, we repeated either our measurements or simulation until the results were consistent.

We isolate the circuits corresponding to each resource, generate the netlist using Cadence tool, and use Hspice to simulate the netlist. Although our netlist are extracted before the layout, we examined the layout in each case to add the correct metal and transistor loading to our circuits. This further enables us to reuse our circuits for investigation of other architectures, as opposed to using a post-layout netlist.

3.3. C_{eff} Results

Table 1 summarizes the effective capacitance for the major resources in the Virtex-II device. The capacitance data in this table is characterized based on device 2v1000FG256-5. Our choice of target was solely due to its availability. The capacitance of all resources except Long and Clocking are the same for other members of the Virtex-II family. The capacitance of long lines and global clock tree vary according to the width and height of the device. In our estimations for Long and Clocking, we linearly extend the wire capacitance based on the device.

Table 1. Effective capacitance summary

Type	Resource	Capacitance (pF)
Interconnect per CLB	IXbar	9.44
	OXbar	5.12
	Double	13.20
	Hex	18.40
	Long	26.10
Logic per CLB	LUT inputs	26.40
	FF inputs	2.88
	Carry	2.68
Clocking	Global wiring	300
	Local	0.72

Each of the interconnect crossbars consist of a number of switches, where the switch capacitance can be further divided into that of pass transistor network, buffer, and the metal wire. For IXbar and OXbar the buffer capacitance is the dominant part, but for Double and Hex the wire capacitance exceeds the rest. Long switches drive a wire crossing the whole chip; therefore, the metal wire in Long dominates the capacitance. Note that the capacitance of switches in the same set varies slightly, due to the layout. While we consider some of those variations in our final results, presenting their details is beyond the capacity of this paper. For the same reason, we have presented the total capacitance of all the LUT inputs in the CLB, as a lump value. The LUT inputs can be classified into two groups of fast and slow based on which their individual capacitance varies. The resources in Table 1 are not only partially utilized, but also contribute to the final capacitance based on various rates of occurrences depending on the design. Therefore, it is imperative to consider the resource utilization for being able to obtain meaningful results. We will do this in the next two sections.

4. ESTIMATION METHODOLOGY

In this section we explain how we augmented Xilinx standard design flow to estimate the power share for each resource. The first step in the standard flow is synthesis, which creates a structural EDIF netlist. Xilinx CAD tool then reads the EDIF netlist to map, place, and route the design in several steps. Finally, the resulting bitstream can be generated and configured into the FPGA. The routed design, which is available in an internal format (called NCD), can be used for power estimation as follows.

4.1. Resource Utilization Flow

While the routed NCD file contains all the routing information including the utilized resources, it can not be read directly. Therefore, we require another step in addition to Xilinx standard flow, as shown in the left side of Figure 2. We use Xilinx Design Language (XDL) utility in Xilinx tools to convert the binary NCD file to text format. Then using a number of Perl programs we obtain the utilization for each resource in the routed designs. Our design set includes more than sixty real circuits of all sizes. We present the resource utilization of the largest ten circuits in Section 5.

4.2. Switching Activity Flow

The right side of Figure 2 shows the switching activity flow, which starts with the routed design similar to utilization flow. We back annotate the routed NCD to generate a structural VHDL file that contains all the resources used in the design and their corresponding delays. Then, Modelsim is used to feed the design with input stimuli and perform a real-delay timing simulation. Finally, the result of simulation is read into our Perl script along with the routed design in XDL format to obtain the statistical representation of the switching activity for each resource. The switching activity depends on both the design and its input test vector. While we had access to a large set of real designs, the realistic test vectors were not available. Therefore in some cases, we applied random inputs to the design. However, this was not possible in most cases, and thus our design set for calculation of switching activity was smaller than that of resource utilization. Nonetheless, we believe our results are conclusive as discussed in the next section.

Calculation of switching activity requires considering two types of elements in the design: nets and logic. Nets often have one source and multiple destinations, and signals maintain their switching activity going through the nets. However, the logic manipulation occurs in the LUTs and occasionally in the other parts of the slice, and may change the switching activity. The switching activity of a LUT output depends on both the activity of its inputs and the configuration of the LUT, which determines its logic. It is possible to obtain the switching activity of LUT inputs and output from the simulation results, but we need to consider the LUT configuration to be able to identify the power dissipation at intermediate nodes inside the LUT. In order to model the switching activity of LUTs, we apply a statistical estimation approach similar to that of [7]. First, we calculate partial switching activity of the output due to its i th input as

$$S_{pi}(OUT) = P\left(\frac{\partial OUT}{\partial INP_i}\right) \cdot S(INP_i), \text{ where } S(INP_i) \text{ is the}$$

switching activity of input signal, and $P\left(\frac{\partial OUT}{\partial INP_i}\right)$ is the

probability of the Boolean difference equation, which is determined by LUT configuration. The value of $S_{pi}(OUT)$ estimates the transitions of intermediate output signal caused by the toggling of input signal INP_i . The total switching activity of the output intermediate signal is then calculated as $S(OUT) = \sum_i S_{pi}(OUT) \cdot S(INP_i)$ over all its corresponding

inputs. Note that in this approach we have assumed that the switching activities of input signals are not correlated. This assumption will introduce some error, but the analysis of correlation of input signals is too complicated.

5. RESULTS

There are three factors that determine the overall results of dynamic power dissipation: effective capacitance, utilization, and switching activity. A summary of capacitance measurements and simulation is given in 3.3. This section presents the results of the second and third factor to obtain the overall results of power dissipation.

5.1. Utilization

Resource utilization strongly depends on the design. Although we have access to a large set of benchmark sets, we observed that the resource utilization is dominated by a few large designs. Therefore, we intentionally excluded the small designs to bias our results for future devices. Our benchmark designs fit in 2v3000 to 2v6000 Virtex-II devices, and contain more than a quarter of million LUTs. To insure statistically valid results, we made sure all the circuits occupy most of their corresponding device. This is important because if a circuit were too small for the device, some of the global resources such as long wires would have lower than usual utilization, which would have led to inaccuracy. Figure 3 presents the overall utilization of various interconnect resources. The results of the figure depend on the quality of the tools as well as timing and physical constraints of the design. It is recommended that the timing constraints for circuits to be as tight as possible even if the requested speed is not required. Tighter timing requirement directs the place and route tools to choose the resources with lower capacitance where

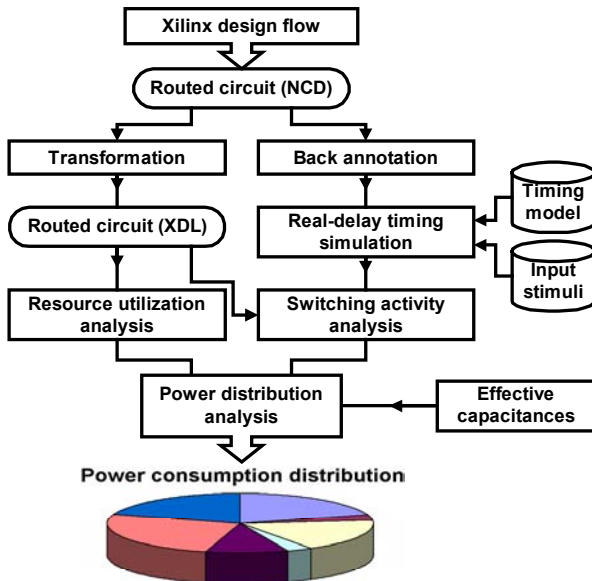


Figure 2. Power estimation flow.

possible, resulting in lower power consumption for the same frequency. The detailed resource utilization is combined with capacitance measures to calculate the overall results in 5.3.

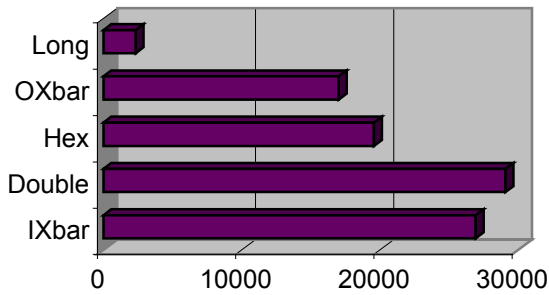


Figure 3. Interconnect resource utilization.

5.2. Switching Activity

Perhaps the most complicated part of accurate power estimation is to calculate the average number of signal transitions corresponding to each resource. This is because the signal transitions or switching activity are determined by the input patterns that are applied to the design. It will be a tedious task to obtain the test vectors that fully emulate the real behavior of a design. Due to flexibility of FPGAs, a large number of designers tend to drop their design into the device and test it. Therefore, they only partially simulate the design, which does not require a complete test vector. Nonetheless, we managed to obtain a complete set of test vectors for one of our large benchmark circuits.

Our benchmark circuit occupies more than 90% of a 2V3000 device (with 14336 slices), and its number of flip-flops is as high as 85% of its number of LUTs. The designers have supplied the input stimuli, which is the next best thing to having real inputs, in our opinion. Figure 4 shows switching activity results for all five types of resources in our test case. The horizontal axis in the figure represents the switching activity, as defined in Subsection 2.2, with the accuracy of ± 0.02 . The vertical axis, which is logarithmic for the sake of clarity, represents the number of occurrences for switches in each set of resource.

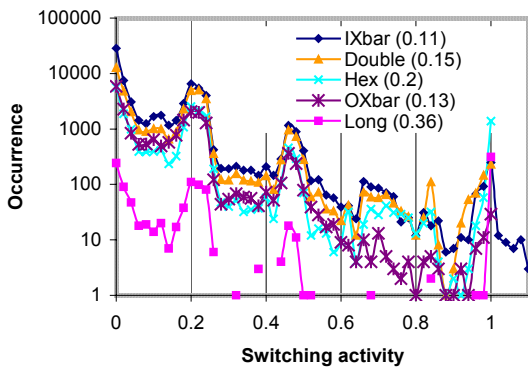


Figure 4. Switching activity with real input stimuli.

The first observation based on Figure 4 is that all resources follow the same statistical behavior. This is intuitive because these resources are driving each other and routing resources do not change the switching activity. The numbers in parentheses next to a resource name in the figure are the switching activity averages. The average switching activity of the circuit for IXbar, Double, Hex, OXbar, and Long are 0.11, 0.15, 0.2, 0.13, and 0.36, respectively. In the next subsection, we will use these averages as a representative for the switching activity to calculate the overall power dissipation.

There are two dominant local peaks in the curves of Figure 4 at or around switching activity 0 and 1. Zero activity for some signals is expected, and switching activity one corresponds to signals that change with the clock edge. Also, the 50% switching activity, which is equivalent to the activity of a flip-flop whose output is fed back to its input, happens frequently. Keep in mind that clock signal has a switching activity of two, but it often uses dedicated resources instead of the shown routing resources. We account for the clock power consumption separately in our Perl program.

To investigate switching activity further, we also applied random input patterns to a set of designs to observe their behavior. Unfortunately, this restricts our choice of benchmark circuits, because the majority of our real circuits will not work with random input patterns. Our example circuits include FIR and FFT filters, DES encryption, and a circuit with a number of multipliers. These designs utilize 7790 slices (with 13276 LUTs) and 2483 flip-flops. Figure 5 summarizes the switching activity results when a new random input is supplied every five clock cycles. Since random inputs change with the probability of 50%, the input patterns represent a switching activity of 10% (or 0.1). The local peaks around 0.1 switching activity in Figure 5 confirm the dependency of the circuit switching activity on the activity of input patterns. One noticeable difference with previous results is that the average switching activity for various resources are close. The average of Long and Hex switching activity in Figure 4 are higher than that of other resources. Closer examination shows that the high average is caused by high occurrence of the switching activity of one, which is associated to a clock-enable signal. Therefore, a slight design modification is likely to reduce the average Long and Hex switching activity in our real circuit.

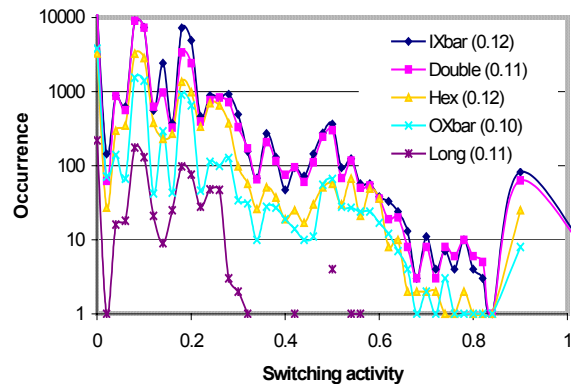


Figure 5. Switching activity with random test vectors.

A final experiment shows the switching activity for the same four benchmark circuits, when random inputs are supplied every clock cycle. This corresponds to 50% switching activity of the input patterns. Figure 6 shows the new curves and averages when high activity is enforced by random inputs. The dominant peak moves to 0.5 from the 0.1 in Figure 5. The average activities, however, have not increased linearly, due to the existence of a large number of nets with zero activity in both cases.

An interesting observation from Figure 6 is that the switching activity for some signals is higher than one. In synchronous circuits the switching activity higher than one may be contributed to spurious transitions. These unwanted transitions, which are often called *glitches*, will dissipate additional dynamic power. Glitching occurs when the inputs to LUTs arrive at different time and therefore the LUT output will have multiple transitions in a single clock cycle before settling down to the correct logic level. Glitching of static CMOS circuits in ASIC is previously studied, concluding that it may contribute to 20% to 70% of power dissipation [8]. At this point we have no reason to believe that FPGAs will be different than their alternative ASIC with respect to glitches. However, glitches will increase with the depth of combinational logic, and flip-flops at the right place will significantly reduce the glitching. Since there is an abundance of flip-flops in FPGAs, techniques such as pipelining or retiming can be used to reduce glitches without a significant area or power penalty.

Figure 6 also shows the average of the switching activity for each resource in parentheses. The highest activity for the examined circuits is 4.26, but is not shown in the figure for clarity sake. If we consider the switching activity higher than one as glitching, the power dissipation due to glitches in IXbar, Double, Hex, and OXbar is 10%, 11%, 9%, 18% of total power dissipation of that resource, respectively. Glitching power in the real circuit (Figure 4) is less than 1%, and thus negligible. We believe the glitching in Figure 6 is higher than that of real circuit for two main reasons. First, the number of flip-flops in our real circuit is higher than our examples, and second, the typical circuit input stimuli does not cause as many glitches as random inputs.

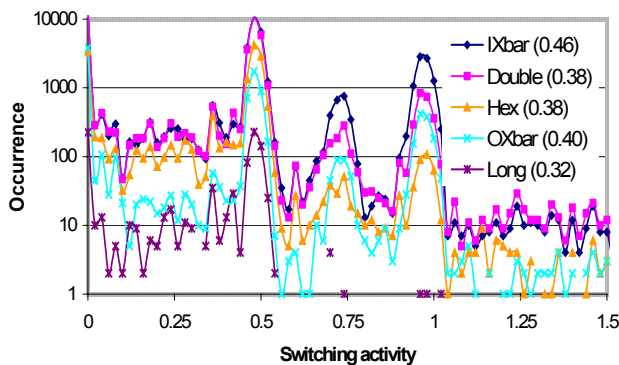


Figure 6. High switching activity (random inputs).

5.3. Overall Results

So far, we have discussed all three important factors that are required for dynamic power dissipation. Equation (1) in Subsection 2.2.1 can be rewritten as $P = V^2 f \sum_i C_i U_i S_i$,

where V is the supply voltage, f is the operating frequency, and C_i , U_i , S_i are the effective capacitance, utilization and switching activity of each resource, respectively. Using this new equation, we first obtain the power distribution for our real circuit whose switching activity is shown in Figure 4. Overall results for this benchmark circuit are presented in Figure 7. The switching activity for logic is calculated as explained in Subsection 4.2, and the switching activity for clocking resources is two. This design uses flip-flops heavily and as a result the power consumption of the clocking resources is as high as 22%.

One single benchmark circuit is not representative, and we have already obtained the statistically valid utilization results for a number of circuits. Therefore, we present three sets of results in Figure 8, using all our real circuits, but with the switching activity averages obtained from Figures 4, 5, and 6. Although our focus in this paper is the FPGA fabric, we would also like to insure that the power dissipation of Input-Output Blocks (IOBs) is not the dominant part.

Virtex-II FPGAs can be configured for several IOB standards, and their detailed investigation is beyond the scope of this paper. We only consider LVTTTL standard with fast slew rate and drive strength of 12 mA, and measure the power dissipation of each IOB running at a known frequency. We add 10 pf for the package and board dependent capacitance associated with each output buffer. The IOB power supplies are both 1.5 V and 3.3 V depending on the part and type of the block. We also consider the toggle rate of IOBs as the activity average of all routing resources, which is reflected in the IXbar switching activity.

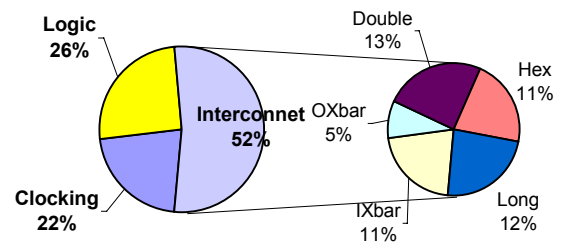
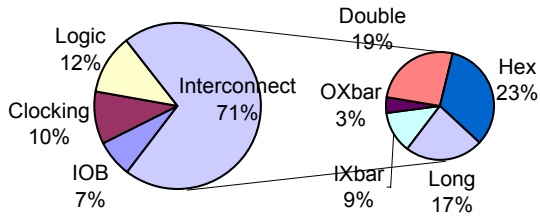
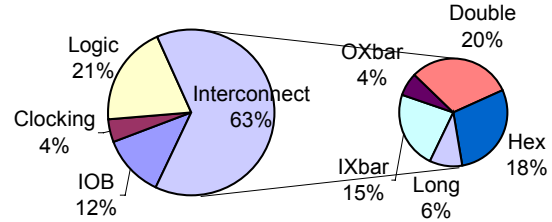


Figure 7. Power distribution for a real circuit.

Part (a): switching activity from Figure 4



Part (c): switching activity from Figure 6



Part (b): switching activity from Figure 5

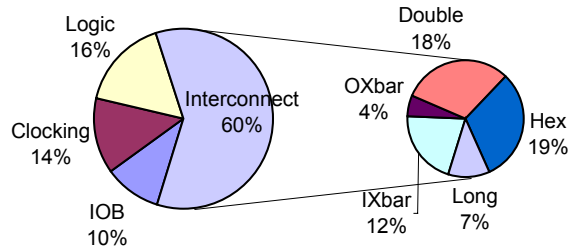


Figure 8. Power dissipation distribution

According to Figure 8, most of the power dissipation in FPGA fabric occurs in the interconnect resources. Part (b) of the figure also helps us to observe the effects of utilization and effective capacitance if the switching activity of the resources would have been roughly the same. Keep in mind that the switching activity of the clock is two in all parts of the figure. The high wire capacitance in Long and Hex contributes to the most of their power dissipation, while high utilization of Doubles is the main cause of their high share in the pie charts. It is clear from our results that the designs should take advantage of their locality as much as possible to reduce the power consumption for the same functionality. Both careful designing and improving the CAD tools can achieve this goal. The above results are independent of supply voltage and the operating frequency. We also estimated the total power dissipation of all the circuits at 100 MHz and supply voltage of 1.5V to obtain the average consumption of a Virtex-II CLB. According to our results, one CLB approximately consumes 5.9 μW per MHz. This number is a good measure for CLB power dissipation in typical designs, but high switching activity can significantly raise the CLB power dissipation. For example, a switching activity of 50% (one new random input in every clock cycle) would cause CLB power consumption as high as 23 μW per MHz. On the other hand, a switching activity of 5% would reduce the CLB power

dissipation to 3.1 μW per MHz. Uncertainty management and better understanding of the switching activity is an important potential future work.

6. CONCLUDING REMARKS

The power dissipation of semiconductor devices is rapidly growing to be a major concern as the device sizes increase. FPGA devices contain the largest number of transistors on the same chip, but most of those transistors do not dissipate dynamic power. However, rapid growth of FPGAs will soon leave them on the hot plate of power hungry devices. In this paper we thoroughly analyzed the dynamic power dissipation in Virtex-II, which is the most recent and the largest FPGA product. We pinpointed three important factors that contribute to total power dissipation as effective capacitance, resource utilization, and switching activity. We investigated these three factors in detail and concluded with presenting the distribution of total power dissipation for a number of real circuits. Our results are somewhat different than previous work, which is mostly due to using larger circuits and the state-of-the-art FPGAs.

The results of our work can be extended to investigate various techniques to reduce the power consumption. There are three possible avenues that can help reduce power dissipation: design changes, architectural modifications, and CAD tool

improvements. All these three approaches require the results of our work as the first step. We intend to continue this work toward the goal of reducing power using any of these three techniques. Finally, we plan to obtain other realistic test vectors to expand our understanding of switching activity.

7. ACKNOWLEDGEMENTS

We would like to thank Nabeel Shirazi and Suresh Sivasubramaniam from Xilinx for their help. Also, comments from other Xilinx employees improved the quality of this work. The opinions expressed by authors are theirs alone and do not represent the opinions of Xilinx and are not an indication of any future policy on FPGA software or hardware held by Xilinx.

8. REFERENCES

[1] D. Sylvester, H. Kaul, "Future Performance Challenges in Nanometer Design," Design Automation Conference," pp. 3-8, June 2001.
[2] E. A. Kusse, and J. Rabaey, "Low-energy embedded FPGA structures," Int. Symp. On Low Power Electronics & Design, pp. 155-160, Aug. 1998.

[3] A. Gracia, "Power consumption and optimization in field programmable gate arrays," Ph.D. thesis, Département Communications et Électronique, Ecole Nationale Supérieure des Télécommunications, 2000.

[4] T. Osmulski, et. al., "A probabilistic power prediction tool for the Xilinx 4000-series FPGA," in Proc. 5th Int. Wksp. Embedded/Distributed HPC Systems and Applications," pp. 776-783, May 2000.

[5] Xilinx Inc., "Virtex-II Platform FPGA Handbook," 2000.

[6] Gary Yeap, "Practical Low Power Digital VLSI Design," Kluwer Academic Publishers, 1998.

[7] S. Gupta and F. N. Najm, "Analytical models for RTL power estimation of combinational and sequential circuits," *IEEE Trans. on Computer-Aided Design*, vol. 19, no. 7, pp. 808-814, July 2000.

[8] A. Shen, et. al, "On average power dissipation and random pattern testability of CMOS Combinational Logic Networks," IEEE ICCAD, pp. 402-407, 1992.