

STATISTICAL POWER ESTIMATION FOR FPGA'S

*Elias Todorovich and Eduardo Boemo**

School of Engineering
Universidad Autónoma de Madrid, Spain
Ctra. Colmenar Km 15, 28049
email: etodorov@uam.es,
eduardo.boemo@uam.es

Fabian Angarita and Javier Valls

School of Engineering, Gandía
Universidad Politécnica de Valencia, Spain
Ctra. Nazaret-Oliva, 46730
email: faanpre@doctor.upv.es,
jvalls@eln.upv.es

ABSTRACT

This article presents a power estimation tool integrated with an FPGA design flow. It is able to estimate total and individual-node average power consumption for combinational blocks. The tool is based on the statistical approach, allowing the user to specify the tolerated error and confidence level of the power estimation. An important feature of this software is the short pulse filtration that leads, in other case, to overestimation. Power maps generation is implemented to help both to detect hot-spots, and perform a power optimization. These maps show the power at every physical position in the die. Several circuits have been tested in order to demonstrate the tool features and usability. The estimated values of dynamic power have been compared with physical measurements for Virtex and Virtex-E devices.

1. INTRODUCTION

Several techniques have been developed to estimate the power consumption for digital circuits. Basically, power estimation methods can be based on statistics or probabilities propagation. Comprehensive surveys about power estimation are presented in [1] and [2].

In FPGA technology, just tools based on spreadsheets, like [3] and [4], were the only possibilities available until recently. An implementation based on probabilities propagation was presented in [5]. In [7], the authors analyze the dynamic power consumption in Virtex-II devices. These steps to estimate the power consumption have not been integrated within an EDA tool.

Since 5.1i version, in the Xilinx ISE, it is available a power estimation tool called XPower[6]. The designer must provide a set of meaningful input vectors for simulation. As power consumption depends on these vectors, the tool itself cannot guarantee that simulated activity will converge to the average value for a given scenario. For this reason, XPower is considered here a power computation tool.

*This work is supported by Project 658005 of the *Fundacion General de la Universidad Autonoma de Madrid*.

The main problem is the activity measurement. It is hard to estimate because it depends on the inputs values. This problem is known as the *pattern-dependence* problem. Both probabilistic and statistic approaches propose solutions to this problem. Activity estimation could be yet harder due to the glitches. Furthermore, simulators can produce very short glitches that lead to activity overestimation [8-10]. These short pulses do not produce rail-to-rail transitions and consume less or no power.

This paper tries to contribute to the previous research lines by the development of a new FPGA-oriented power estimation tool, based on the statistical approach. The current version is available for combinational blocks.

Xilinx's FPGAs are used as technological framework, but the presented tool can be ported to other FPGA development environments.

2. STATISTICAL POWER ESTIMATION

The statistical approach for power estimation is based on the Monte Carlo simulation technique. It minimizes the pattern-dependence problem: randomly generated input patterns are applied to the circuit inputs, while the activity per time interval T is monitored by a simulator. The process continues until a stopping criterion is reached.

The first work applying a Monte Carlo technique for total average power estimation was [11]. In [12], the technique was extended, providing both the total and individual-gate power values. Other works ([13], [14]) make use of the statistical approach on sequential circuits.

In order to estimate the power consumption for individual nodes, in [14] it is proposed to partition them in two sets. Let \bar{n} be the simulated average activity over a period T , and s is its standard deviation. The user defines an activity threshold n_{min} that classifies the nodes into regular and low-density nodes.

$$N \geq \left(\frac{z_{\alpha/2} s}{\bar{n} \epsilon_1} \right)^2 \quad (a) \quad N \geq \left(\frac{z_{\alpha/2} s}{n_{min} \epsilon} \right)^2 \quad (b) \quad (1)$$

(1a) and (1b) are used as stopping criterion for the regular nodes ($\bar{n} > n_{min}$) and low-density nodes ($\bar{n} < n_{min}$)

Table 1. Test circuits.

Circuit	# Slices	Slice FF	Min. Period (ns)
C1: QDDFS-CORDIC (portable RTL code)	484 (15%)	773 (12%)	8.591
C2: QDDFS-CORDIC (Area-restricted)	484 (15%)	773 (12%)	9.220
C3: DA FIR Single Rate Digit 1 (serial)	159 (3%)	307 (3%)	5.781
C4: DA FIR Single Rate Digit 2	303 (6%)	597 (6%)	7.305
C5: DA FIR Single Rate Digit 3	456 (9%)	897 (9%)	6.276
C6: DA FIR Single Rate Digit 4	595 (12%)	1177 (12%)	6.484
C7: DA FIR Single Rate Digit 8	1163 (24%)	2305 (24%)	5.903
C8: FFT A	3424 (36%)	6364 (33%)	12.803
C9: FFT B	3384 (35%)	6364 (33%)	11.767
C10: FFT C	3424 (36%)	6364 (33%)	11.731
C11: FFT D	3424 (36%)	6364 (33%)	10.457

respectively. They solve the slow convergence: when \bar{n} decreases, the sample size increases. In both cases, the stopping criterion is tested after $N > 30$. $(1 - \alpha) \times 100\%$ is the confidence level that error in the estimation is less than a specified value. ε_l is an upper bound of the percentage error, ε is the user specified error tolerance ($\varepsilon = \varepsilon_l / (1 - \varepsilon_l)$).

3. IMPLEMENTATION DETAILS

The implementation details of the activity estimation subsystem can be found in [15]. Basically, in its inner loop, input vectors are generated for the simulator according to the user specifications, and it reports the design activity using VCD format files. This is done iteratively and, at the end, the average activity of the individual nodes is stored in a database. The same input vectors generated for the simulator, are written for the pattern generator in order to physically measure the designs.

Even though the dual stopping criterion can drastically reduce the sample size, it is still unnecessarily large. For example, in one of the test circuits presented below, halfway through the estimation process, 99% of the power has been gauged. At this point, the user required accuracy is exceeded. To solve this problem, the estimation process can be interrupted earlier, when a specified percentage of the nodes have reached the stopping criterion.

Capacitances must also be obtained to compute the power. They could come from datasheets or detailed schematics of the circuit. Nevertheless, this information is not easily available for FPGA end-users. In this work, capacitance values are determined by running XPower on the placed and routed circuits. The capacitances are extracted from a text file (pwa) that reports them in femtofarads (fF).

Another problem that must be faced is the node names coherence. Identifiers for the nets used in the simulation files (vhdl, sdf, and particularly vcd), and in the vendor files

(xml, xdl, pwa, pwr) are related but different. The key to associate these identifiers is the parameter `-aka`, (also known as) when the post PAR VHDL model is generated with the netgen command. The alternative identifiers are written within VHDL comments beside the corresponding component instantiation or signal declaration.

The position of the pin that drives every net (CLB row, column, and slice) is extracted and related with the node name from the xdl file (Xilinx Design File). This file can be obtained from the routed design file (.ncd) executing the xdl program that is part of the ISE distribution. With this information, the tool can build power, capacitance and activity maps. It can generate maps for different CLB resolutions, starting from 1x1 CLB, but it is also possible to zoom in on an individual slice, or even on the resources within the slices.

Several works [8-10] recognized that current simulators are reporting short glitches that physically do not produce rail-to-rail transitions and should be filtered. This is a very important source of error. One proposed approach is filtering the pulses shorter than the logic delay of the net's driving gate. In this paper, the pulse is not considered when it is shorter than a user specified time, T_G . This enables a fine tuning that can lead to accurate results.

4. EVALUATION AND RESULTS

In order to test the power estimation tool, some designs are measured and analyzed. Its main characteristics are listed in Table 1. The designs are implemented using the Xilinx ISE 6.1 to 6.3i. Tight timing constraints are specified in order to fulfill practical requirements. This enabled the routing tool to select the fastest (and then, the lowest capacitance) resources.

Several target devices are selected to test the tool. C1 and C2 are mapped into a XCV300E-pq240-8 device, C3-C7 into a XCV400E-pq240-8, both Virtex-E. C8-C11 are

Table 2. QDDFS dynamic power consumption, in mW/MHz for several output frequencies.

Out. freq.	C1				C2			
	Meas.	Estimated	Adjusted	XPower	Meas.	Estimated	Adjusted	XPower
1 MHz	4.61	7.15(+55)	4.39(-5)	6.10(+32)	4.36	7.49(+72)	4.42(+2)	6.33 (+45)
10 MHz	6.17	10.21(+66)	6.03(-2)	9.01(+46)	5.87	10.78(+84)	6.43(+9)	9.32(+59)
20 MHz	7.00	11.27(+61)	6.66(-5)	9.82(+40)	6.53	11.84(+81)	6.91(+6)	10.12(+55)
30 MHz	7.36	11.56(+57)	6.75(-8)	10.20(+39)	6.88	12.14(+77)	6.98(+2)	10.51(+53)

Table 3. Dynamic power consumption in mW/MHz.

Circuit	Measured		Estimated		Adjusted		XPower	
C3	0.82		0.91 (+11)		0.82 (0)		0.87(+6)	
C4	1.80		1.94 (+8)		1.74 (-4)		1.74 (-4)	
C5	2.63		2.90 (+11)		2.63 (0)		2.55(-3)	
C6	3.74		3.99 (+7)		3.56 (-5)		3.44(-8)	
C7	5.33		6.46 (+21)		5.72 (+7)		5.76(+8)	
C8	27.75	28.00	33.44(+21)	33.35(+19)	27.41(-1)	27.56(-2)	26.8(-3)	26.75(-4)
C9	27.75	27.13	32.54(+17)	32.49(+20)	26.25(-5)	26.12(-4)	26.45(-5)	26.45(-3)
C10	27.63	27.75	32.79(+19)	32.72(+18)	26.71(-3)	26.59(-4)	26.65(-4)	26.55(-4)
C11	24.88	24.88	33.67(+35)	33.59(+35)	27.70(+11)	27.65(+11)	26.55(+7)	26.85(+8)

mapped into a Virtex XCV800-hq240-4. All these FPGAs are mounted on different Xilinx development boards where the power supply jacks are separated for the core and I/O. An ammeter is used to measure average currents maintaining the core voltage, V_{ccint} , at the nominal values. Input vectors are generated with a Tektronix pattern generator.

QDDFS circuits synthesize sine and cosine digital waveforms [16]. QDDFS outputs have been generated with 14-bit word length in all cases. Clock frequency is 100 MHz while several outputs are shown in Table 2. QDDFS inputs are clock enable, reset, and a 17-bit frequency control word. The only difference between C1 and C2 is that the second has been compiled adding an area restriction. None of these inputs is random, so the statistical technique can not be tested with this experiment, just the tool accuracy with respect to the physical measurements.

C3-C7 are different implementations of a 64-tap FIR filter using distributed arithmetic. They use 6-bit coefficients, 8-bit input and output words, 12.5 MHz fixed sampling frequency, and a $2/3$ cut-off frequency. The difference among these implementations is the internal digit size. The clock frequency ranges from 100 (C3) to 12.5 MHz (C7).

C8-C11 are 64-point pipelined FFT implementations that fulfill the Hiperslan/2 and IEEE 802.11a-g standards.

Table 2 shows the results for C1-C2 circuits. Column 2 shows the measured average power; column 3, the estimated power without any glitch filtering; column 4 shows the estimated power after the short pulses were

filtered. For each device a single minimum pulse width value was calculated in order to minimize the overall error. Column 5 shows the XPower results. Relative errors are presented in parenthesis. Note that, for the selected cases, the higher the output frequency, the higher the power consumption. For the highest frequencies there are fewer points per output cycle, and the discrete steps must be larger, generating more activity in the MSBs [16].

Power consumptions depicted in the Table 3 correspond to measurements and estimated values for the rest of the studied circuits. The tolerated error was specified as 10% with 90% confidence. For C3-C7, the 8-bit input was defined as independent random patterns, whilst for C8-C11 input data are modulated QAM and QPSK at the right and left column respectively.

A noticeable difference is observed between the estimations without short pulse filtering and the physical measurements. As reported in [8-10], if all the transitions generated by a gate level simulator are counted, an overestimation is obtained. Nevertheless, after the short pulses filtering, accurate estimations are observed.

It is interesting to present XPower results for comparison. However, they must be carefully considered. They were obtained running a long simulation with the same vectors generated by the proposed tool. In a more real situation, without using this tool, another input set must be obtained, maybe, intuitively by the designer. In this second case the differences could be important. In the case the estimations were as accurate as the obtained here, it could be calculated by generating huge simulation result files.

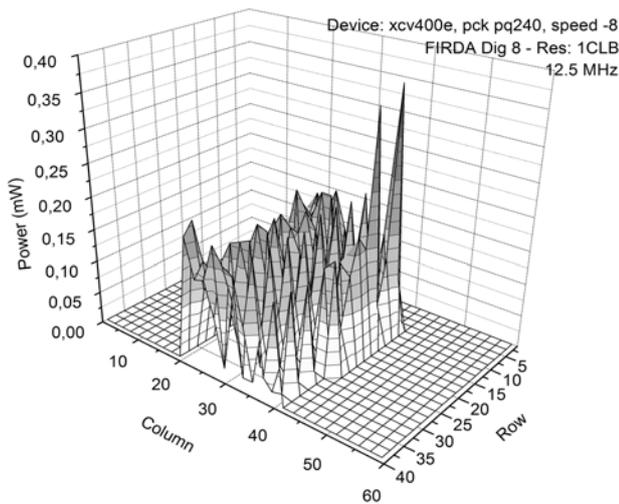


Fig. 1. Power Map for C7.

5. CONCLUSIONS

A statistical-based power estimation tool oriented to FPGA devices has been presented. Its main characteristics are:

- Integration with a commercial FPGA design flow.
- Use of standard formats widely accepted in the industry like the Standard Delay Format (sdf), Value Change Dump (vcd), and eXtended Markup Language (xml). This enables the tool to be integrated with several simulators and ported to other FPGA IDEs.
- Design Automation: once the user specifies the necessary parameters, the power estimation and analysis is done without any other user interaction. Input vectors for the simulator are automatically generated; simulation results are analyzed, etc. A Tcl script glues the different programs developed for each specific task.
- The geometric information, in addition to the individual power estimations, enables the generation of power maps, giving a view of the power distribution inside the device. Fig. 1 represents the power consumption of C7. Also capacitance and activity maps can be drawn.
- Accuracy: Power is obtained according to a statistical technique with user defined error and confidence level. The required number of samples monotonically increases as the required accuracy increases. Vendor's tools do not provide any program in order to generate statistically valid power vector and then there is no guarantee about the precision of the results in a given scenario.
- Speed: specifying a moderate accuracy (10% error and 90% confidence), with current PCs, the execution time for all the process took from 2 to 8 minutes for the test cases presented in this work.

6. REFERENCES

- [1] F. Najm, "Estimating Power Dissipation in VLSI Circuits", *IEEE Circuits and Devices*, Vol 10, No 4, pp. 11-19, 1994.
- [2] M. Pedram, "Design technologies for Low Power VLSI", In *Encyclopedia of Computer Science and Technology*, Vol. 36, pp. 73-96, Marcel Dekker, Inc., 1997.
- [3] J. Tan, "Virtex Power Estimator User Guide", *XAPP 152*, 1999.
- [4] Xilinx Inc., "XC4000XL Power Calculation", *XCELL*, No 27, pp. 29, 2000.
- [5] T. A. Osmulski, "Implementation and Evaluation of a Power Prediction Model for a Field Programmable Gate Array", Master's Thesis, Department of Computer Science, Texas Tech University, Lubbock, May 1998.
- [6] Xilinx Inc., "Chapter 11: XPower". In *Development System Reference Guide*, available at <http://www.xilinx.com>.
- [7] Li Shang, Kaviani, A.S., and Bathala, K., "Dynamic Power Consumption in Virtex-II FPGA Family", *Proc. of Int. Symp on Field Programmable Gate Arrays*, 2002, pp. 157-164.
- [8] C. Baena, J. Juan-Chico, M. Bellido, P. Ruíz, C. Jiménez, M. Valencia, "Measurement of the switching activity of CMOS digital circuits at the gate level", *Lecture Notes in Computer Science*, No. 2451, pp. 353-362, 2002, Springer-Verlag, Berlin.
- [9] Fei Li, Deming Chen, Lei He, Jason Cong: "Architecture evaluation for power-efficient FPGAs", *Proc. of Int. Symp on Field Programmable Gate Arrays*, 2003, pp. 175-184.
- [10] J. A. Anderson, F. N. Najm, "Power Estimation Techniques for FPGAs", *IEEE Trans. on VLSI Systems*, vol. 12, no. 10, pp. 1015-1027, 2004.
- [11] R. Burch, Najm, F. N., Yang, P., Trick, T., "A Monte Carlo approach for power estimation", *IEEE Transactions on VLSI Systems*, 1(1), pp 63-71, 1993.
- [12] F. N. Najm, Xakellis, M. G., "Statistical estimation of the switching activity in VLSI circuits", *VLSI Design*, vol. 7, no. 3, pp. 243-254, 1998.
- [13] T. Chou, Roy, K., "Accurate Power Estimation of CMOS Sequential Circuits", *IEEE Trans. on VLSI*, Vol.4, n°3, pp. 369-380, 1996.
- [14] J. Kozhaya, Najm, F. N., "Accurate power estimation for large sequential circuits", *IEEE/ACM International Conference on Computer-Aided Design*, 1997, pp. 488-493.
- [15] E. Todorovich, M. Gilabert, G. Sutter, S. Lopez-Buedo, and E. Boemo, "A Tool for Activity Estimation in FPGAs", *Lecture Notes in Computer Science*, Vol. 2438, pp. 340-349, 2002. Springer-Verlag, Berlin Heidelberg.
- [16] F. Cardells, J. Valls, "Area-Optimized Implementation of Quadrature Direct Digital Frequency Synthesizers on LUT-based FPGAs", *IEEE Trans. on Circuits and Systems II*, vol. 50, no. 3, march 2003.