



HAL
open science

A Neural Model for RT-Level Power Estimation on FPGAs

Yehya Nasser, Jean-Christophe Prevotet, Maryline H elard

► **To cite this version:**

Yehya Nasser, Jean-Christophe Prevotet, Maryline H elard. A Neural Model for RT-Level Power Estimation on FPGAs. 13 eme Colloque du GDR SoC/SiP 2018, Jun 2018, Paris, France. hal-02021099

HAL Id: hal-02021099

<https://hal.science/hal-02021099v1>

Submitted on 15 Feb 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destin ee au d ep ot et  a la diffusion de documents scientifiques de niveau recherche, publi es ou non,  emanant des  tablissements d'enseignement et de recherche fran ais ou  trangers, des laboratoires publics ou priv es.

A Neural Model for RT-Level Power Estimation on FPGAs

Yehya Nasser, Jean-Christophe Prévotet and Maryline Hélard
Univ Rennes, INSA Rennes, IETR, CNRS, UMR 6164, F-35000 Rennes
Rennes, France
yehya.nasser@insa-rennes.fr

Abstract

Power optimization is required all along the design flow but particularly in the first steps where it has the strongest impact. In this work, we propose new power models based on neural networks that predict the power consumed by digital operators implemented on Field Programmable Gate Arrays (FPGAs). These operators are interconnected and the statistical information of data patterns are propagated among them. The obtained results make an overall power estimation of a specific design possible. A comparison is performed to evaluate the accuracy of our power models against the estimations provided by the Xilinx Power Analyzer (XPA) tool. Our approach is verified at system-level where different processing systems are implemented. A mean absolute percentage error which is less than 8% is shown versus the Xilinx classic flow dedicated to power estimation.

1. Introduction

Power consumption has become a critical performance metric in a wide range of embedded systems, specially in reconfigurable devices like FPGAs. Moreover, a huge amount of these devices will clearly have an impact on the environment and on the worldwide energy consumption in particular. One attempt to reduce the consumed power of such devices consists in exploring various hardware architectures, very soon in the design process, in order to reach an optimal architecture in terms of power consumption. Today, this exploration is not sufficiently implemented in most of the design tools.

For example, design flows for circuits such as FPGAs do not natively take into consideration power issues at system level. The exploration flow generally consists in implementing several versions of an architecture in the FPGA and get power estimation results after the timing simulation step. Power estimation results are obtained by presenting stimuli data to the circuit and by evaluating the activity of internal signals. Although very accurate in practice, these results are obtained very late in the design process and require a lot of previous implementation steps, which can be very time consuming. It is also

prohibitive when one wants to rapidly test various architectures according to given parameters.

Avoiding hardware implementation for the purpose of estimating the power at early design phases is the best switchable solution for researchers and engineers demanding a faster power estimation at high-level of abstraction. Unfortunately, high-level abstract models are generally not accurate enough and prevent the evaluation and comparison of different hardware solutions.

In our work, we aim to propose a new hardware exploration methodology that tries to combine both accuracy and fast power estimation. This is achieved by estimating the power of specific operators after the hardware implementation on a real platform and by exploiting this information in high-level models. These pre-characterized models can be easily integrated and simulated to estimate the power consumption of an overall design.

2 Related Works

Power estimation techniques can be classified into two classes, high-level and low-level approaches. High-level power estimations are based on power models which are used to evaluate power consumption without requiring any implementation and physical details. High-level input parameters are sufficient to obtain a good estimate of the power consumption. The other class deals with low-level power estimation approaches that take into consideration all physical details. These are generally far more accurate as described in [5].

In this paper, we focus on high-level power estimation. In [9], a dynamic power estimation model for FPGA is proposed. It is based on the analytical computation of the switching activity generated inside the components and takes into consideration the correlation between the inputs. A mean relative error of 9.32% for adders and 5.67% for multipliers are obtained compare to measures on a real platform.

A hybrid power modeling approach that accurately and quickly assesses gate-level power consumption is presented in [8]. It consists in integrating valuable and accurate physical information with a LUT-based power model to ensure that the correct optimization techniques are implemented. The main disadvantages of the LUT-based

macro-modeling are the use of a huge amount of memory to store the input-output statistics and the corresponding simulated power values. Concerning the analytical power models, the main drawback is the simulation time, thereby the need of the computation of the outputs statistics after a time consuming simulation.

Some previous works have introduced artificial intelligence based on neural networks for power estimation. These are described in [3]. In [3] the neural network is used to estimate the output statistics and the power consumption of complementary metal-oxide-semiconductor (CMOS) circuits. The average absolute relative error of the proposed method is about 5.0%, at circuit level. Using this method, a lot of circuits have to be simulated to cover most applications. Therefore it lacks of generality.

This work is inspired from [1], and in the continuity of [6] and [7]. In [1] a power macro-model for (Intellectual Property) IP power estimation is proposed that is based on lookup table (LUT) power models. The IP power model used to estimate the power consumption of an IP-based digital system which is based on signal statistics propagation. In [6], IP power models based on neural networks have been proposed. Compare to this previous study, this work aims to consider only operators characterization instead. The idea is to provide a more generic and flexible approach to estimate power consumption by allowing the design of a complete system from basic operators.

3 Power Estimation Methodology

3.1 Proposed approach

The proposed methodology is based on the assumption that any hardware system can be represented by a group of hardware components that are dedicated to a specific function. The main idea consists in estimating the consumption of a global digital system, based on an accurate power estimation of its sub-components. Each component has been fully characterized and is available in a dedicated library. Once the components have been fully characterized and that accurate models are built, designers have the possibility to construct their architecture by connecting components in a design entry tool that is similar to schematics or with a hardware description language. The full design may then be simulated at high-level to evaluate the performances and obtain an accurate evaluation of the design power profile. Note that, our methodology makes it possible to test various combinations of components and then of architectures very easily, by simply modifying the components in the high-level design entry tool.

3.2 Model Definition

The purpose of the characterization phase is to develop a power model of a component by extracting the relevant information that has a direct impact on power.

In order to evaluate the power consumption of a given circuit, we propose to provide each operator with a power model that depends on the activity rate of its inputs. Each

operator model consists actually of two sub-models $M1$ and $M2$ that are described in Fig. 1. $M1$ constitutes a first model that predict the power consumption for given (signal rates) SR or (α) and (percentage logic high) PLH or (p) of all inputs. The signal activity of the inputs is expressed in terms of millions of transitions per second (Mtr/sec). For every input, PLH refers to the percentage of time during which the signal is at HIGH level, on a period of 1 s. The global model provides then an average of the energy consumed during this period.

The second sub-model ($M2$) makes it possible to estimate the signal activity of the outputs as well as the percentage logic high according to the operator's inputs. This second model is particularly useful when designers want to propagate the signals' activity as well as the PLH among all connected operators in their design.

In order to obtain a global power estimation for the entire design, it is then possible to sum-up the power contribution of each operator, at high level. A global power value can be computed according to eq. 2.

$$P_{Global} = M_{1,1}(\alpha_1, p_1) + \sum_{i=2}^{N-1} M_{1,i}(\beta_{i-1}, h_{i-1}). \quad (2)$$

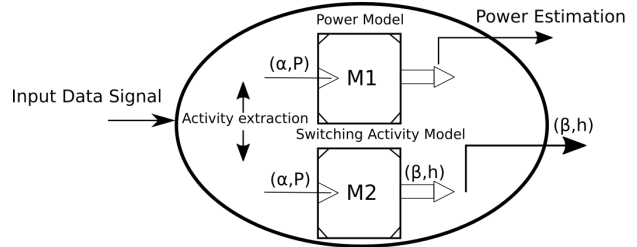


Figure 1. Operator Model

Both $M1$ and $M2$ sub-models have been implemented using neural networks that are classical tools that have shown their efficiency in lots of domains. They especially perform very well in classification and also in non-linear regression problems.

4 Experiments and results

Two types of power estimation have been performed. The first corresponds to the classic power estimation that is achieved in the last steps of an FPGA design flow (after place and route). Regarding Xilinx FPGAs, this step is implemented using XPA. The second power estimation has been performed using our neural networks models. In this case, simulation is implemented in Matlab and takes place at high-level.

4.1 Per-operator Verification

At operator-level, a verification phase is performed to calculate the accuracy of the proposed models. As shown in Fig. 2, XPA has been used to estimate the power consumption of a 16 bit adder and a 16 bit multiplier. For the

same operators, the MATLAB tool has then been used to estimate the dynamic power consumption using the neural approach.

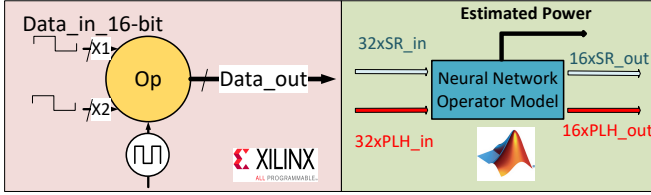


Figure 2. Neural network accuracy measurements

The relative error has been calculated according to 10000 data samples. The results shown in this table indicate a relative error that is very small (around 0.01%). This shows that neural networks have learned the behavior of XPA and are able to model it very accurately.

4.2 Per-system Verification

At system-level, verification of several scenarios have been performed. Figure 3 describes a function composed of single operators. As in the per-operator approach, XPA has been used to evaluate the power of the global system as well as the SR and PLH parameters. In parallel, another global model has been built under Matlab and consists in interconnecting both $M1$ and $M2$ models for each operator and perform an high-Level simulation. This simulation aims to propagate the parameters throughout all operators and provides an overall power estimation.

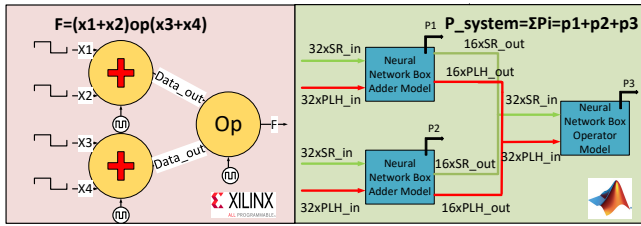


Figure 3. System level power estimation XPA vs NN

Note that two functions have been implemented : $F_i = (a \text{ op } b) \text{ op } (c \text{ op } d)$, where op can either be an adder or a multiplier. As in eq. 4, a mean absolute percentage error is calculated over 5000 data samples at system-level to evaluate the accuracy of the approach. P_{XPA} is the power consumption calculated using the XPA tool and P_{NN} is the power consumption estimated by the power model.

$$\%MAPE_{System-level} = \frac{100}{n} \sum_{i=1}^n \frac{|P_{XPA} - P_{NN}|}{P_{XPA}} \quad (4)$$

According to the results given in Table 1, it can be seen that our method provides a good accuracy at system-level with an error that is less than 8%.

Arithmetic Function	Neural models (MAPE %)
F1(op1=+, op2=+, op3=+)	7.3699
F2(op1=*, op2=*, op3=*)	3.5158

Table 1. Models Accuracy for Different Functions

5 Conclusion

In this work, we have presented a new methodology, in which neural networks are the key models used for High-Level estimation of the consumed power. Two types of neural networks have been used to estimate power and signal activity (signal rate and PLH). Both types show a very good accuracy when considering simple operators. When using these models to evaluate more complex functions, the obtained results are less accurate on the power estimation due to the fact of the modeling errors and the interconnections power.

References

- [1] D. Y. Arafat and R. Teresa. Power estimation for intellectual property-based digital systems at the architectural level. *J. King Saud Univ. Comput. Inf. Sci.*, 26(3):287–295, Sept. 2014.
- [2] I. M. Author. Some related article I wrote. *Some Fine Journal*, 99(7):1–100, January 1999.
- [3] W. Q. Y. Cao, Y. Yan, and X. Gao. Neural network macro-model for high-level power estimation of cmos circuits. In *2005 International Conference on Neural Networks and Brain*, volume 2, pages 1009–1014, Oct 2005.
- [4] A. N. Expert. *A Book He Wrote*. His Publisher, Erewhon, NC, 1999.
- [5] L. Fei, C. Deming, H. Lei, and C. Jason. Architecture evaluation for power-efficient fpgas. In *Proceedings of the 2003 ACM/SIGDA Eleventh International Symposium on Field Programmable Gate Arrays, FPGA '03*, pages 175–184, New York, NY, USA, 2003. ACM.
- [6] J. Lorandel, J. C. Prévotet, and M. Hélar. Efficient modelling of fpga-based ip blocks using neural networks. In *2016 International Symposium on Wireless Communication Systems (ISWCS)*, pages 571–575, Sept 2016.
- [7] Y. Nasser, J. C. Prévotet, M. Hélar, and J. Lorandel. Dynamic power estimation based on switching activity propagation. In *2017 27th International Conference on Field Programmable Logic and Applications (FPL)*, pages 1–2, Sept 2017.
- [8] A. Nocua, A. Virazel, A. Bosio, P. Girard, and C. Chevalier. A hybrid power estimation technique to improve ip power models quality. In *2016 IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SoC)*, pages 1–6, Sept 2016.
- [9] J. Ruzica and C. Carlos. A complete dynamic power estimation model for data-paths in fpga dsp designs. *Integr. VLSI J.*, 45(2):172–185, Mar. 2012.