INFORMATION METRICS FOR A PRIORI EVALUATION OF IC AREA ON VHDL DESCRIPTION

Marin Marinov

Fabless Ltd. , 7km Tzarigradsko chaussee, 1184 Sofia, Bulgaria

Tel: +359-2-974 3183 Fax.+359-2 –975 3762

E-mail: marin.marinov@fab-less.com

ABSTRACT

A programming hypothesis for digital circuit complexity evaluation, based on Halstead programming languages evaluation metrics, applied to VHDL descriptions of digital circuits, is proposed. The hypothesis provides relations between the integrated circuits gate count and the quantity of input/output operands, used in their VHDL description. Using the proposed approach an a priori evaluation of the circuit area can be done. An experimental proof of this relation on IP designs is presented. More specifically, experimental data for a number of Silway and Fabless companies IPs suggest that the accuracy deviations for the predicted IC area are within 10% of the actual synthesis figures.

Keywords

IC area, VHDL, Halstead metrics

1. INTRODUCTION

The need to satisfy the market with more and more complex digital integrated circuits, led to a dominating use of the design programming paradigm. With that respect, the use of the VHDL and Verilog hardware description languages becomes very popular, allowing reduction of the design cycle and at the same time increasing the integrated circuit (IC) complexity.

At the same time the lack of methods and metrics for a priori circuit complexity evaluation, and in turn the logic gates count of the circuit and its area, makes the process of evaluation of the integrated circuit project efficiency quite slow and difficult, while this is a crucial point for the project financing and decision making.

The presented study is dedicated to propose a method to avoid the "tautology", namely to provide the integrated circuit designer with metrics, which helps him to estimate (with reasonable or at least determined accuracy) a circuit area, by the help of circuit algorithm parameters only, which can be obtained on the stage of feasibility study of the circuit, i.e. a priori VHDL programming. Otherwise, when the VHDL model exists, the more exact and fast way of circuit area evaluation is to use the synthesizer results.

We experiment on number of IP cores from the digital VHDL libraries of Silway and Fabless companies, such as peripheral controllers, interface controllers and micro cores. Experimental data suggest accuracy of the predicted gate count within 10% with respect to the real count of the synthesized gates.

The remainder of this paper is organized as follows. In Section 2, we refer to some related work and propose our hypothesis on the relations between the VHDL model volume and the logic gates count. In Section 3, experimental data on several IP cores are presented to ground the proposed hypothesis. Finally, the discussion is concluded in Section 4.

2. RELATIONS BETWEEN THE VHDL MODEL VOLUME AND THE OF LOGIC GATES COUNT

In this section, we present some related work and propose our hypothesis on the relations between the VHDL model volume and the logic gates number.

Related work: A survey of existing metrics for circuit area estimation is given by Machado et al in [1] where a method for a posteriori evaluation of the area is proposed.

Xu and Kurdah [2] proposed a method for area evaluation of FPGA. Gerlach and Rosentiel [3] describe a methodology for circuit area study on high abstraction level, but they do not propose an algorithm for its calculation. Srinivasan et al [4] propose a method for area evaluation, based on statistical models, but again calculation algorithms are not presented.

Some bibliography about indirect methods for area estimation is found. Mostly these methods are dedicated to explore other circuit characteristics. Among them are Ferrandi et al [5], who propose a methodology for power estimation, Knieser and Papachristou [6], proposing system level C and VHDL hardware/software co-design methodology, based on data flow graphs. Sharma and Jain [7] propose a method to estimate the lowest bound for number of functional units and delay. Gasteier et al [8] show a method to replace complex operations in VHDL with synthesizable code. An analyzer is implemented as well.

As it is seen, most of the evaluation procedures proposed above are dedicated to a certain design methodology and its application. The common idea of the proposed methods is that they are all based on analyzing tools, which extract from the models the quantitative parameters needed to calculate the circuit area and the number of logic gates. This in practice substitutes or doubles the function of the design environment VHDL synthesizer, which provides these parameters by default.

With the permanent increase of integrated circuits complexity, the modern circuits migrate to integration of system on a chip (SOC). In that sense, the earlier proposed in [8] relations between logic gates count and pin count of the circuit are not valid for them.

The modern integrated circuits are first of all systems and mostly programmable. The last feature means that the greater part of the circuit area is occupied by memory - automata or program.

The hypothesis: The information contents of the digital circuit can be identified with the entropy of the logic gates states. On the other hand, these logic gates are described in the VHDL program model, whose information contents, according to proposed in [9] Halstead metrics, is determined by the model volume – V, as follows:

$$V = N \log n \qquad (1)$$

where: **N** is the length of the VHDL model program implementation

and **n** is the vocabulary of operators – $\mathbf{n_1}$ and operands – $\mathbf{n_2}$, used in this particular model implementation.

The drawback of using the program model volume – V is that, as it is seen from the formula above, the volume is a function of the operands but also of VHDL operators, for whose determination a circuit VHDL description is needed. This description, however, is not available at the stage of project feasibility study. Thus, usage of metrics – potential volume V* is more advisable:

$$V^* = (2+n^*_2) \log.(2+n^*_2) \qquad (2)$$

where: n^*_2 – the minimum number of operands, used in the program model, associated with the number of independent and unique input / output signals and

2 – is the minimum number of operators, namely for the function and for the assignment.

As it could be seen the potential volume of the algorithm depends only on the minimum number of unique operands, which can be associated with the input / output signals of the circuit.

As on the stage of the project feasibility study the only description of the target circuit is its specification, from which the operands n^*_2 could be derived, the metrics - potential volume V^* turns to be a good choice for an estimating characteristics.

The potential volume V^*, however, is a characteristics of the language-independent implementation of the algorithm. Therefore, in order to bring in the language-dependable correction of the volume we should introduce as well the metrics– language level λ, as a measure for the language capability to contain the information about the algorithm in the potential volume of its description.

$$\lambda = L.V^* \qquad (3)$$

where $L = V^*/V$ – level of the program model.

After introducing the basic notions of the Halstead metrics, on the basis of which we'll build the preliminary estimation method for logic gates count constituting the circuit, we can go back to the initial thesis for the analogy between the volume of the VHDL program model, describing the circuit, and the logic gates count - G, realizing it.

We'll base the hypothesis for this analogy on the dependency between the logic gates count and the volume of the VHDL program implementation, corrected with the parameter λ, characterizing the language level as follows:

$$G^* = \lambda \, . \, V \qquad (4)$$

or considering the formula (3) we get:

$$G^* = V^{*2} \qquad (5)$$

where G^* is the calculated value of the logic gates number in the circuit.

In [10] the validity of Halstead metrics for VHDL language has been proven. For that purpose a "Halstead Metrics Tool" has been developed. The tool contains a syntactic analyzer with an algorithm to count the VHDL operands and operators, as basic parameters for metrics calculation.

At the root of this analyzer there are procedures for extracting the following syntactic constructions:

Operands, including all the language syntaxes (functions and procedures), manipulating the data. They include the functional operators and the VHDL key words. In VHDL the procedures can be defined through outer packages, and for that reason these procedures are considered as an expansion of the language operators set.

Operands, including all signals, variables and constants, , functionally transformed during program execution. Conceptually unique and independent signals are those, stated in the "port" part of the VHDL "entity" declaration. The internal signals and variables in the VHDL models are considered derivatives, i.e. they are formed after influence of operators on the input signals and consequently they are not taken into consideration when counting the number of operands.

Structural constructions, including neutral syntactic constructions, defining the infrastructure of the VHDL project, despite of the model dimension (the program), the abstraction or the complexity level. Most of them ca be found in the model only once (usually as markers for the syntactic analysis. That's why these constructions are ignored when analyzing the data.

With the help of this program the basic parameters of the VHDL programs are being extracted and calculated, namely $\mathbf{n_1}$- operators' vocabulary and $\mathbf{n_2}$ – operands' vocabulary, appearing in a given VHDL program. On the basis of these parameters, extracted from numbers of synthesizable VHDL core descriptions, the program length and volume have been calculated and compared to their real values. The coverage of calculated and real values of Halstead's program characteristics have been shown.

3. GROUNDING THE PROGRAM HYPOTHESIS

In order to ground the proposed dependency, VHDL descriptions of synthesizable cores for number of controllers designed in Silway and Fabless companies have been analyzed. As a tool for extracting the syntactic characteristics of the VHDL descriptions the analyzer described in Section 2 has been used.

With the help of the analyzer the main characteristics of the VHDL program models have been obtained – the operator and operand vocabularies, and in accordance with (1), (3) and (4) the volume of the program, the VHDL language level and the logic gates count in the circuit have been calculated.

In Table 1, the results obtained from the analysis of the VHDL models of several synthesizable cores are presented, more specifically: PWM (a Pulse-Width-Modulation controller IP), RTC (a Real-Time Clock IP), WD (a Watch-Dog controller IP), and UART (an Universal .Asynchronous Receiver Transmitter IP).

Table 1. VHDL descriptions analysis

IP Name	G	V	G/V
PWM	5336	4055	1.32
RTC	13902	11102	1.25
Watchdog	5876	4539	1.29
UART	25980	21302	1.22
Average			≈1.27

The correlation between the real logic gates count G on the scheme and the volume of the VHDL model, which according to the programming hypothesis should correspond to the VHDL language level - λ, is on the average around 1.27. In [10] it has been proven that the VHDL language level λ is 1.23, which compared to the resulting λ value from table 1 gives a deviation of about 3%. The calculated values of the logic gates count of the mentioned controllers - G^*compared to their real values – G, obtained after the synthesis of the VHDL description of the same controllers, are shown in Table 2.

Table 2. Calculated vs. real logic gates counts

IP Name	G	G^*	G*/G
PWM	5336	5679	106%
RTC	13902	13995	101%
Watchdog	5876	6012	102%
UART	25980	23832	92%

The results show satisfactory coincidence (within the limits of ± 10%) between the calculated and real values of the logic gates counts in the analyzed controller cores.

The hypothesis, thus proven, for the connection between the VHDL description parameters, and still earlier, between the digital circuit algorithm parameters and the quantity of constituting logic gates, gives a possibility for a relatively reliable (within the limits of the experimentally proven accuracy of ± 10%) a priori evaluation of the integrated circuit area depending on the corresponding technology capabilities (integration density). This means that even before the VHDL description of the circuit appears, on the basis of the extracted from its specification vocabulary of the minimum number of operands - $\mathbf{n_2^*}$**,** associated to the quantity of the circuit input/output signals, it is possible to estimate the logic gates count of the circuit, and from there – its area.

4. CONCLUSIONS

In the presented paper, we introduced and experimentally proved the viability of a programming hypothesis on estimating the digital circuit complexity, using the Halstead metrics. This metrics, typically used for programming languages evaluations, has been adapted and applied on digital circuits

VHDL descriptions. Its unique dependencies between the circuit complexity and the circuit algorithm characteristics allow estimating the area of the considered IC at very early design stages. The digital circuit complexity has been expressed by the number of constituting logic gates, while the circuit algorithm has been characterized by the quantity of input / output operands. Experimental data suggest that the accuracy of the predicted IC complexity (in terms of gates count) is within 10%. The area estimations directly affect the projected cost of the IC considered, which is of primary importance for taking an economically advisable and effective decision for the project financing.

5. ACKNOWLEDGMENTS

Our thanks to my colleague MSc C. Bojkov from Fabless Ltd, Dr. A. Popov from TU Sofia and Dr. G. Kouzmanov from TU Delft for their valuable assistance. Thanks to my wife for her support.

6. REFERENCES

[1] Mochado, F., Torroja,Y., Casado, F., Riesgo, T., Torre, E.de la, Useda, J. *A simple method to estimate the area of VHDL RTL description.* E.T.S.I. Industriales – Universidad Politecnica de Madrid – Division de Ingeneria Electronica C/Jose Gutierrez Abascal 2, 28006 Madrid http://www.upmdie.upm.es

[2] M.Xu, F.Kurdah. Area and Timing Estimation for Lookup Table Based FPGA. *European Design and Test Conference*, 1996.

[3] J.Gerlach, W.Rosentiel. A Scalable Methodology for Cost Estimation in a Transformational High-Level Design Space Exploration Environment. Design, Automation and Test in Europe, Paris, February 1998.

[4] A. Shinivasan, G.D. Huber and D.P. La Potin, Accurate Area and Delay Estimation from RTL Descriptions, IEEE Transactions on VLSI systems, vol.6 no.1,pp.168-172, March 1998.

[5] F.Ferrandi, F.Fummi, E.Macii, M.Poncino, D.Sciuto, Power Estimation of Behavioural Descriptions. Design, Automation and Test in Europe, Paris, February 1998.

[6] M.J.Knieser,C.A.Papachristou, COMET: A Hardware-Software Codesign Methodology, EURODAC'96

[7] A.Sharma, R.Jain, Estimating Architectural Resources and Performance for High-Level Synthesis Applications, IEEE Transactions on VLSI systems,vol.1, no2, June 1993.

[8] M.Gasteier, N.When, M.Glesner, Synthesis of Complex VHDL operators. EuroDAC'93

[9] M.Halstead. Elements of Software Science. Elsevier North Holland,1977.

[10] M.Marinov, C.Bojkov, VHDL models complexity evaluation by using Halstead metrics. Fabless Ltd. (in printing)