Design Automations of Efficient Complexity Estimation for Digital Integrated Circuit Design

Feng Feng
Undergraduate student in Pillar of Engineering Product Development (EPD),
Singapore University of Technology and Design (SUTD)
Singapore
fengfeng.nj@gmail.com

Jiajia Chen
Pillar of Engineering Product Development (EPD)
Singapore University of Technology and Design (SUTD)
Singapore
jiajia_chen@sutd.edu.sg

***Abstract*—Computer Aided Design (CAD) tools are commonly adopted in the design, synthesis and simulation of digital circuits. Switch-level simulation, logic synthesis, timing analysis and architecture optimizations are essential CAD tools during digital IC design flow. However, the HDL coding, computation complexity and simulation time overhead makes the design turnaround time much longer. This drawback becomes critical in digital circuit high-level synthesis and optimization algorithms when design complexities, such as area cost and delay, need to be efficiently and accurately evaluated within the algorithmic iterations. To address this problem, this paper presents some design automation techniques used during our digital IC design algorithm development, which are helpful to reduce the design and verification turnaround time. One design example using our complexity evaluation model shows that the total algorithm computation time cost has been significantly reduced by 73.1% compared with using the conventional HDL development and simulation cycle.**

Keywords-Design Automation; Digital IC Design; Circuit Simulation

I. INTRODUCTION

Digital integrated circuits (IC) have been widely applied due to their fast speed, small sizes and high power efficiencies. As the scope of application continuously extends as well as the update speed unceasingly increase, demands for specification-assured digital IC designs have largely expanded, which requires shorter design turnaround time with higher accuracy and reliability. Computer aided design (CAD) tools in the applications of circuit design, synthesis and simulation has been widely adopted due to its amazingly high efficiency and reliability compared to traditional design methods. Further improvement in the integration and level of automation is one of the key factors of exploiting the advantage of CAD tools. This paper will present two methods to increase the efficiency in the CAD design flow. Section II discusses the automation of Hardware Description Language (HDL) coding process for circuit synthesis and simulation. Section III presents the acceleration of the iteration process between circuit design process and circuit performance evaluation. For better illustration of the methods mentioned in these two sections, we use design algorithm for multiple constant multiplication (MCM) block as an example. Section IV summarizes the proposed methods.

II. AUTOMATION OF HARDWARE DESCRIPTION LANGUAGE CODING

A. Traditional Digital IC Design Flow using CAD

Traditional computer aided digital IC design process mainly automate two processes, which are circuit design process and design simulation. Circuit design processes are realized by computer programs that are dedicated to designing specific circuit components. These programs take design requirements as inputs and produce the circuit diagram following specified algorithm or design methodologies. For example, the program for designing multiple constant multiplication (MCM) blocks takes quantized coefficients as inputs and generates the adder tree structure consists of adders, subtractors and shifters. The design simulation refers to the process of logic synthesis and simulation using tools such as Design Compiler and LeonardoSpectrum instead of fabricating a real chip for performance evaluation. Fig. 1 is a flow chart that summarizes the flow of a typical CAD design process.

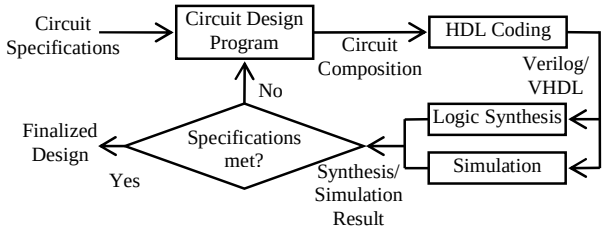

Fig. 1 Design flow of a typical CAD process for digital IC design

B. Automated Hardware Description Language Coding

Traditional computer aided digital IC design process enables quick evaluation of circuit performance without fabricating the circuit for every candidate design, which significantly reduces time and monetary cost. However, the hardware description language (HDL) coding process inevitably occupies long durations of time, especially for complicated circuits. Also, the coding efficiency greatly depends on the fluency of the HDL programmer. Therefore, we

This work and the material reported in this document are supported by the SUTD-MIT International Design Centre (IDC) at Singapore University of Technology and Design (SUTD).

can consider automating the coding process by introducing automatic HDL code generators.

For example, sysFIR [1] is a tool which takes the designed filter structure as input and generates the corresponding VHDL code automatically. Designers need to bind their design outputs from algorithmic design program to the input of the HDL generators. Fig. 2 is a flow chart that summarizes the design flow with an automated HDL coding process.

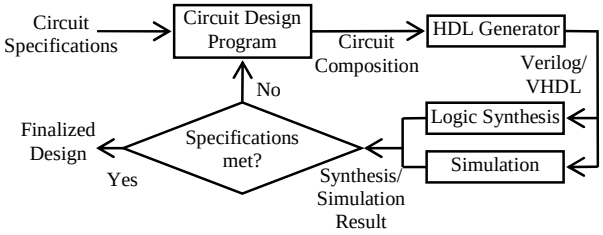

Fig. 2 Design flow of a CAD design with an automated HDL coding process

III. Acceleration of Circuit Design and Evaluation Iteration Process

During the design process, designers may need to alternate between designing, simulation, evaluation and design modification. The utilization of commercial synthesis and simulation tools are largely constrained by the existence of cell libraries, operations in command prompt and computation time. If we are able to integrate the circuit evaluation into the design process and reducing the times of synthesis and simulation, the overall design turnaround time can be further shortened.

In this section, therefore, we propose an approximated circuit evaluation method which is integrated into the algorithmic circuit design programs as an independent module. It is an approximated evaluation of design performance and serves as the feedback to modify the design. For example, dynamic power consumption is correlated with complexity of the circuit, as well as critical path delay because longer delay will cause more switching activities. For design of power-efficient MCM blocks, the evaluation criterion can be estimated as the

α×(# of logic gate in the design)+β×(# of logic gate along the critical path)

The parameters α and β are the significance weightages of contribution to dynamic power consumption, by total gate cost and critical path lengths respectively. The values of α and β can be collected by running power simulation on many similar benchmark designs. Apparently, the more prudent the evaluation parameters are, the more accurate the evaluation result would be. However, the drawback is the longer time used to develop the evaluation module. Therefore, there should be a tradeoff between the quality of the evaluation method and the speed of the development process. There can be multiple trading-off strategies based on different design contexts and requirements. Here we propose a general evaluation model design strategy. If a design largely fulfills the design requirements, it is not necessary for the evaluation model to be highly accurate. However, for a design that marginally fulfills the requirements, the accuracy of the evaluation model needs to increased. To formulate the relationship, for a circuit with an estimated deviation ε_{req} in percentage from the design requirement, the evaluation model should be accurate enough such that

$$|\varepsilon_{eval}| < |\varepsilon_{req}| \quad (1)$$

where ε_{eval} is the deviation of the evaluation model from the real simulation in percentage.

We take designing an MCM block based on a novel design algorithm as an example. The overall circuit optimization objective is to design an MCM block with smaller total logic gate count than that of the most famous algorithms in MCM block designing. After a quick round of investigation, we found that the deviation of the logic gate count of our method from the most competitive algorithm is 3.8% ($|\varepsilon_{req}| = 3.8\%$), which means that our evaluation model should not be over 3.8% deviated from the real simulation result. By the cost of full/half adders/subtractors with their complexity weightages in 0.18μm CMOS libraries introduced in [2] , we successfully controlled the average error of evaluation model under 2.7% ($|\varepsilon_{eval}| < 2.7\%$), which is considered acceptable. With the help of the pre-evaluation model, the majority of design evaluation work is done within the Matlab program as part of the algorithm and hence largely reduced the design and simulation turnaround time. Fig. 3 is a flow chart that summarizes the design flow with accelerated circuit design and evaluation iteration processes.

We used the proposed automation techniques in one of our works which is to design low-complexity digital FIR filter. A total of 20 benchmark filter samples need to be evaluated during the algorithm iteration. Each benchmark always needs to evaluated and modified more than 3 times to finalize the solution. Instead of performing the conventional HDL development and simulation for more than 20×3=60 times, we used the evaluated complexity by our proposed estimation model. Only after the designs are finalized, we develop the HDL and perform the logic synthesis. With the same equipment and software setup, the total development and computation time using our approach has been significantly reduced by 73.1% over the conventional exact simulations. The quality of the solutions did not drop much as the simulation results trend of our proposed algorithm still shows noteworthy improvements over other competing algorithms.

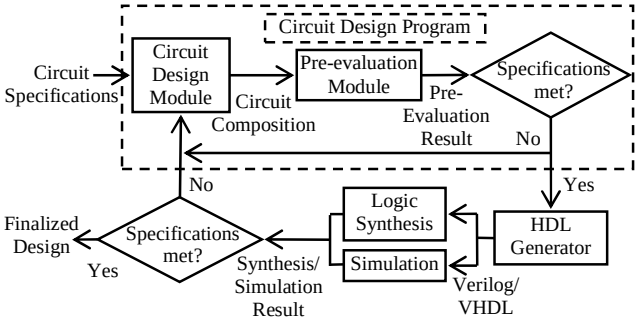

Fig. 3 Design flow of a CAD design with an accelerated circuit design and evaluation iteration process

IV. Conclusion

It is of great significance to shorten the design turn-around time of digital IC design. With increased design efficiency, designers are able to focus less on repetitive labor work and more on core designing parts, while the finalized design could be released more quickly. This paper has investigated two typical places where design process can be automated or accelerated, and there is much larger space for further design process optimization.

References

[1] M. Faust, (2013), *Design Methodologies for Complexity Reduction of FIR Filters*, Ph.D. Thesis, Nanyang Technological University, Singapore.

[2] P. Tummeltshammer, J. C. Hoe, and M. Püschel, "Time-multiplexed multiple-constant multiplication," *IEEE Trans. on Compute Aided Designs*, vol. 26, no. 9, pp. 1551-1563, Sep. 2007.