UNIVERSITI MALAYSIA PERLIS

FPGA PROTOTYPING FLOW OPTIMIZATION BY USING FAILED PATH FIXES AND MANUAL CLOCK DISTRIBUTION TECHNIQUES

by

Salahuddin Bin Savugathali
(1730212442)

A report submitted in partial fulfillment of the requirements for the degree of Master of Science in Computer Engineering

School of Computer and Communication Engineering
UNIVERSITI MALAYSIA PERLIS

2019

ACKNOWLEDGMENT

Alhamdulillah, I would like to take this golden opportunity to thanks and deeply indebted to many individuals who contributed to this project toward the successful line. Special thanks to my research supervisor, Dr. Muslim bin Mustapa for motivation, support, giving the guidance and supervise throughout the whole project. His eagerness to listen to my problems, his educative comments and his advices for the timely are helpful for me to manage complete the project in time. Next, I would like to thank to my industry supervisor, Mr. Ooi Kok Peng and Mr. Gerrard Lee Kim Wah from the Atom CPU validation team in Intel Corporation in Penang, Malaysia as offers me an opportunity in working environment in the real industry area with the high end specification tools to complete my research. Through the process, I have learned many in terms of technical and management because of their useful tips and guidelines.

Next, I also thank for University Malaysia Perlis (UniMAP) especially School of Computer & Communication Engineering and include all lecturer for giving me the knowledge and help me in the past. Finally, I dedicate my appreciation to my parents and my friend for their kindness, moral support and encouragement during my study here. Special thanks also to those who had shared their knowledge and opinion with me

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ASIC	Application-specific Integrated Circuit
FPGA	Field-Programmable Gate Array
GCC	Gated Clock Conversion
SoC	System on Chip
STA	Static Timing Analysis
THS	Total Hold Slack
TNS	Total negative Slack
WHS	Worst Hold Slack
WNS	Worst Negative Slack

PENGOPTIMUMAN ALIRAN PROTOTAIP FPGA DENGAN MENGGUNAKAN TEKNIK PEMULIHAN LALUAN GAGAL DAN TEKNIK PEMBAHAGIAN JAM SECARA MANUAL

ABSTRAK

Tujuan kajian ini dijalankan adalah untuk kaji semula dan mencadangkan aliran baru dalam aliran prototaip FPGA dengan menggunakan alat Synopsys Protocompiler untuk menyelesaikan masalah yang dinyatakan.Ciri-ciri yang menarik dari reka bentuk berasaskan selak adalah kelewatan jalan kombinasi dapat dibiarkan lebih lama daripada kitaran jam kerana dapat meminjam masa dari jalan yang lebih pendek di logik berikutnya. Teknik Pinjaman Masa adalah pendekatan biasa yang digunakan untuk memenuhi pelanggaran masa dalam reka bentuk prototaip FPGA. Berdasarkan kajian terdahulu, lebih banyak usaha diperlukan untuk menghasilkan hasil yang berkesan untuk selesaikan pelanggaran masa, di mana membawa kita untuk mencadangkan Teknik Pemulihan Laluan Gagal. Teknik ini adalah untuk membetulkan laluan yang gagal dalam selak disebabkan oleh proses penukaran jam berpagar semasa peringkat sintesis yang boleh membawa kepada pelanggaran masa. Penyelesaian untuk prototaip litar ASIC / SoC dengan berjuta logik berpagar ke platform FPGA adalah dengan membahagikan reka bentuk ke dalam banyak FPGA. Penyelidikan ini akan memfokuskan kepada keperluan pembahagian yang diperlukan untuk prototaip litar SoC yang besar ke dalam banyak FPGA. Kehadiran cut clock di peringkat pembahagian akan mengakibatkan kegagalan dalam peringkat penghalaan kerana terlalu sesak. Oleh itu, satu pendekatan yang dicadangkan dalam kajian ini untuk menyelesaikan masalah ini ialah teknik Pembahagian Laluan Jam secara Manual,supaya kehendak pembahagian ke dalam banyak FPGA dapat dikecapi. Perbandingan masa negatif pada reka bentuk SoC sebelum dan selepas menggunakan gabungan teknik Pemulihan Laluan Gagal dan Pinjaman Masa dapat dikurangkan sebanyak 90%.Teknik Pembahagian Laluan Jam secara Manual pula dapat menyelesaikan masalah cut clock yang membawa kepada masalah kesesakan penhalaan semasa pembahagian satu litar ke dalam dua chip FPGA. Dengan teknik yang dicadangkan, semua keperluan pembahagian dipenuhi dan penghapusan cut clock 100% dicapai.

FPGA PROTOTYPING FLOW OPTIMIZATION BY USING FAILED PATH FIXES AND MANUAL CLOCK DISTRIBUTION TECHNIQUES

ABSTRACT

The purpose of this research is to review and propose a new flow in FPGA Prototyping Flow using a Synopsys Protocompiler tool to solve the stated problem. A fascinating property of a latch-based design is that the combinational path delay is allowed to be longer than the clock cycle as it can borrow time from the shorter paths in the subsequent logic states. Time Borrowing technique is a common approach used to satisfy timing violation in an FPGA prototyped design. However, based on previous studies, more efforts are required to produce an efficient result in closing the timing violation, where lead us to propose a Failed Path Fixes technique. This approach is meant to fix the failed path in a latch due to the gated clock conversion (GCC) process during the synthesis stage which could lead to the timing violation. A solution for prototyping a multi-million logic gates of ASIC/SoC circuit into the FPGA platform for verification purpose is by partition the design into multi-FPGA. This research is focusing on the required partition requirement to successfully prototype the large SoC circuit into the multi-FPGA. The presence of cut clocks in a circuit after partition stage will result in the failure in routing stage due to the congestion error. Therefore, an approach proposed in this research to resolve this challenge is Manual Clock Distribution technique, so that design is able to meet the partition requirement to complete the prototyping process into multi-FPGA. The combination of our Failed Path Fixes and time borrowing technique are able to solve the timing violation problem by eliminating the unnecessary path created by protocompiler tool. Comparison of numbers of negative slack before and after our proposed technique is applied resulting 90% improvement. The manual clock distribution technique proposed has been able to solve the cut clock issue that leads to routing congestion problem when partitioning a circuit into two FPGA chips. With our proposed technique, all partition requirements are met and 100% cut clock elimination is achieved.

CHAPTER 1 : INTRODUCTION

1.1 Research Background

Nowadays, the Application-Specific Integrated Circuit (ASIC) and System on Chip (SoC) design have become one of the mainstream in the electronic design among the semiconductor industry (X. Li, Hou, Geng, Wang, & Zhang, 2012). Due to the continuous improvement of integrated design technologies which leads to developing more complex and high-performance design, more resolution is required for the verification.

Design verification is an end-stage process of ensuring that everything on the integrated design works as planned (Aboagye, Patel, & Vig, 2012) and meets the customer requirement. According to International Business Strategic (IBS) (Tang et al., 2016), cost of developing a full design showing that verification stage is expanding at an aggressive rate and be the highest rate among the overall design cost.

As a result of tremendous increases in the size, complexity and consumed the cost of ASIC/SoC in verification stage, the software developer can no longer wait for the chip to be fabricated for the integration of the hardware/software phase in order to meet the ever-shrinking time-to-market window. Therefore, various methods are introduced for the ASIC/SoC verification process.

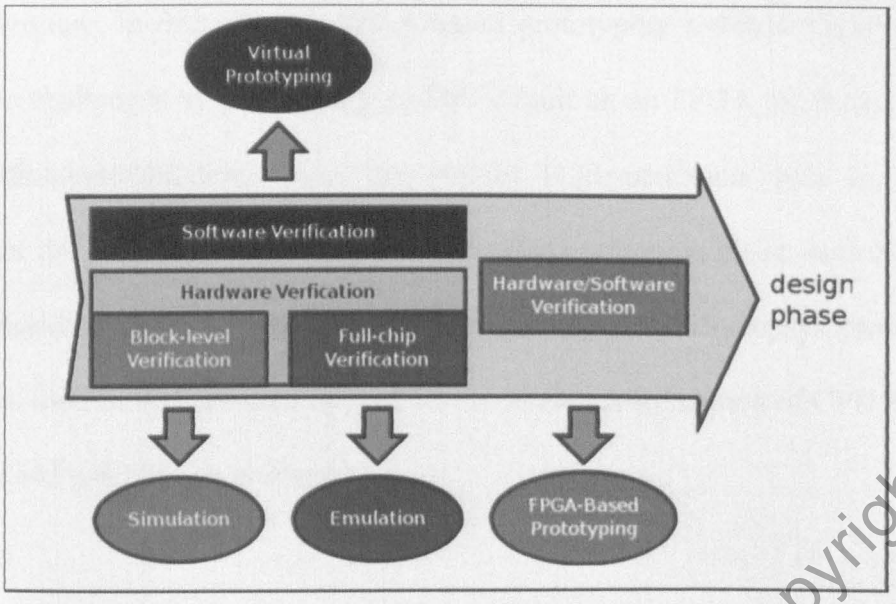

Figure 1.1: Verification technology
Source: (Tang et al. 2016)

Figure 1.1 represents various methods available to verify the ASIC/SoC circuits where each method plays its own role. For simulation, the SoC/ASIC design functionalities are verified through the simulator at the block level. However, the drawback of using simulation technique is the execution speed is limited to 1KHz. While for emulation, an entire SoC/ASIC simulated to address the limitation of the simulators, but due to the unstable RTL it requires an additional detailed debug analysis. This technique has much faster execution speed than the simulation which is 1MHz. The virtual prototype represents a fully functional software model of SoC/ASIC which is an earliest available software verification tool. However, these approaches are not cycle-accurate and not able to ensure the functionalities of the software in the hardware. FPGA-based prototyping provides a real-time speed in a cycle-accurate and bit accurate model of the SoC/ASIC verification. In mean time, the RTL also stable and offers the best execution speed for 10MHz.

Therefore, in this research FPGA-based prototyping technique will be used to address the challenges by prototyping an SoC circuit on an FPGA platform pre-silicon stage verification/validation. There are various implementation tools and platform available in the market to automate an FPGA-based prototype phase such as Cadence Protium Rapid Prototyping Platform, Synopsys and S2C. A Synopsys protocompiler tool will be used in this research to perform the prototyping process of CPU based SoC circuit on HAPS-80 FPGA platform.

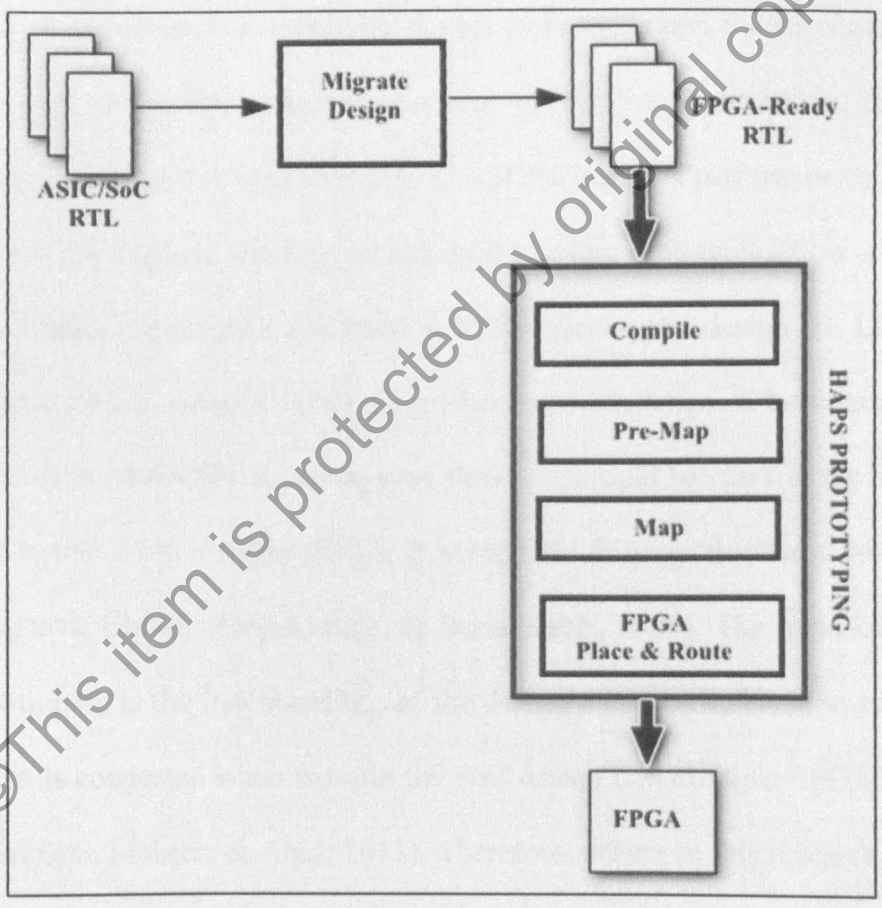

Figure 1.2: Prototyping flow
Source: *HAPS ProtoCompiler User Guide* (2016)

Figure 1.2 shows a prototyping flow using the Synopsys Protocompiler tool. ASIC/SoC design's RTL must be reworked to meet the FPGA based requirements. The requirements are top-level pads required to be adapted for the FPGA tool flow, gated-

clock and complex generated clocks in SoC/ASIC must be transformed in FPGAs and memories required to be handled with FPGA memory resources. To obtain a prototyped design ready, few stages are visited on the prototyping flow which begins with a compilation of the RTL and subsequently through the pre-map, map and finally Place and Route stage. The generated bit configuration file at the end of prototyping flow will be configured into the hardware before executing the verification test.

As the FPGA prototyping technique able to close the challenges for the verification of more complex ASIC/SoC design on limited time, it does come at a price in terms of area, performance and power consumption (Teng & Anderson, 2013) in the FPGA prototyped circuit. In this research, one of the design's performances challenges that will focus the negative slack generated during design prototyping flow which could lead to the timing requirement violation in FPGA prototyped design (B. Li, Chen, & Schlichtmann, 2012). Another design's performance challenge to be concentrated in this research is in Multi-FPGA prototyping flow. As a rapid increase in the SoC circuit size which cannot fit in a single FPGA, it is required to be partitioned across multiple FPGA (Taghavi, Ghiasi, Ranjan, Raje, & Sarrafzadeh, 2004). The main challenge in FPGA partitioning is the functionalities of the design might be defected as routing on a single FPGA is congested when we split the SoC design into multiple FPGA's (Mariem Turki, Marrakchi, Mehrez, & Abid, 2013). Therefore, efforts in this research have been devoted to addressing the timing violation in both single and multiple FPGA based prototyping and routing congestion problem during partitioning the circuits into multiple FPGA.

1.2 Research Question

There are several questions are arisen which is led to the problem statement for this research such as:

- What are the challenges occurred during the verification stage of an ASIC/SoC design using an FPGA prototyping technique?
- What are the effects for the design if not satisfy the design requirement?
- What is the root cause of the problem in a verification stage?
- What are the progress met by the existing techniques?

1.3 Problem Statement

FPGA prototyping is a technique proposed as a solution for validating the SoC design in a pre-silicon stage. FPGAs prevent high non-recurring engineering costs by allowing incremental design debugging (Teng & Anderson, 2013). Therefore, an SoC design can be validated in a virtual environment with sophisticated simulation, emulation, and formal verification tools which is FPGA based prototyping.

Implementing a design into an FPGA-based platform is a challenging process in order to guarantee satisfaction in the design requirement since the original ASIC/SoC

design had to be converted into an FPGA based design before the design can be validated.

As mentioned earlier in the Research Background section, timing violation is a major consideration in digital design. Timing check is divided into two; setup and hold timing. The existence of negative slacks which is less than an ideal value of 0ns could violate the functionalities of the FPGA prototyped design. It is necessary to verify that FPGA prototype design has met the timing requirement and able to work with the intended clock speed. Therefore, this research will focus on reducing the timing violation as modern SoC/ASIC timing closure critically depends upon the effectiveness of the timing fixes and its implementation (Bhargava, Kapoor, Iqbal, & June, 2014). An ASIC/SoC design timing closure is the process where the design needs to be altered to meets its timing requirement.

The second problem we addressed in this research is multiple FPGA prototyping, where the higher routing congestion is more than the maximum congestion level (congestion level 4). Whenever the congestion level exceeds level 4, it will fail the FPGA partitioned design (*HAPS ProtoCompiler User Guide*, 2016). A larger SoC/ASIC design must be partitioned into the multiple FPGA before the routing stage is visited (Kevin Morris, 2009). there are few partition requirements must be achieved during the FPGA partitioning stage, which is; zero unrouted nets, zero cut clocks, low number of feedthrough, low number of multi-hop nets, the minimum number of FPGA interconnect nets, and each of the FPGA utilization must be less than 65% (*HAPS ProtoCompiler User Guide*, 2016). In summary, there are two problems to be focussed in this research:

- Timing violation in an FPGA prototyping design with a negative slack less than an ideal value of 0ns.

- Routing congestion level more than a maximum level set to be at level 4 and violation of the partition requirement.

1.4 Objectives

The objectives are set to solve the problems mentioned. There are three objectives set in this research which are:

- Analyze the SoC/ASIC design to discover the root cause for the requirement violation in an FPGA prototyping design.

- Design a technique that can be used to optimize the timing by reducing the negative slack to be close to the ideal value of 0ns.

- Design a technique to reduce the routing congestion in a FPGA partitioned design to be less than maximum congestion level 4 and satisfy the partition requirements.

1.5 Research Scope

This research has been conducted with the aim to reduce the timing violation and optimize the clock distribution techniques for single and multiple FPGA partition prototyping. This research is set to solve the real problem exist in the semiconductor industries where latch-based design is commonly used compared to the flip-flop-based design. A level-sensitive latch-based design which is consisting of more than 20 million gates is used to prototyped into a HAPS-80 series FPGA platform which is using the latest Xilinx Virtex® UltraScale FPGAs. The Prototyping process is executed by using Synopsys Protocompiler tool.

1.6 Thesis Organization

This research is organized into five chapters. Chapter 1 presents the introduction.the research background, research question, problem statement, objective, of this thesis are listed in chapter 1. Chapter 2 present the literature review including some so the theory related to FPGA Prototyping Flow and summarizes some of the research work in the area of timing and partitioning stage in FPGA Prototyping Flow. Furthermore identified research gap is discussed in this chapter. Chapter 3 presents the methodology.the overall research work-flow and flow charts of algorithm are shown in this chapter. The simulation tool for FPGA prototyping are listed. The proposed methods of Failed Path Fixes and Manual Clock Distribution techniques are discussed in this chapter.Chapter 4 presents the results and discussion of the proposed techniques. Percentage of improvement using the Failed Path Fixes and Manual Clock Distribution

techniques is also presented in this chapter. Finally, chapter 5 concludes this research works and direction for future works.

CHAPTER 2 : LITERATURE REVIEW

The challenges most designer care to confront is when mapping the functionalities of ASIC/SoC design onto FPGA by prototyping it. There are plenty of challenges had been highlighted before and, in this research, reducing the timing violation will be focussed essentially on a single FPGA partitioning. Apart from that, the next hurdle will be discussed in this section is the partitioning strategy for a multiple FPGA used by the previous researcher to achieve the workable design on FPGA. Moreover, in this section, a basic fundamental of timing violation, root cause for the failure of the timing to meet its specification for FPGA prototyping and also the techniques available and implied by previous researchers will be discussed further. All of the previous research will be summarized in the table before we proposed our work.

2.1 Timing Violation

Timing is the main concern of the performance verification process. For an FPGA prototyping, the concept of timing is relating to availabilities of data before the clock event which is called as setup and will remain stable for a specific time after the clock event known as a hold (VivadoUG1, 2013). Timing slack is the difference between the required time and the arrival time of a signal to a specific point in a design. Path with negative slack is generally known as a violated path. Two types of timing check are required to identify the timing violation, which is named as setup timing check and hold timing check which will be discussed later in this chapter.