Chapter 1

Introduction

1.1 Background

In the field of electronics and computer engineering, hardware and software are two common approaches for implementing functionality. Hardware approach viz. Application Specific Integrated Circuit (ASIC), is an outcome of huge design and fabrication efforts as well as heavy Non-recurring Engineering (NRE) cost. It provides a solution with highly optimized resources for performing critical tasks. But it is permanently configured to only one application and can not be changed at later stage. A software approach around General Purpose Processor (GPP) involves writing program to get the desired functionality. This approach provides the flexibility to change applications and perform a huge number of different tasks into one single chip. When compared with ASIC, cost of GPP based solution is less but it lacks in terms of performance, area efficiency and power dissipation. [1]
Figure 1.1: Domain of Reconfigurable Devices

Configurable devices are revolutionary devices that blend benefits of ASIC and GPP. As shown in Fig. 1.1 (Source: [2]) these devices provide better performance than GPP and higher flexibility than ASIC. Also designing functionality on reconfigurable devices takes few days and less cost compared to ASIC design. The main advantage of reconfigurable devices is its flexibility. This means that these devices can be programmed and reprogrammed many times. Thus, reconfigurable devices have provided a trade off between ASIC and GPP as it tries to achieve balance among cost, power, flexibility and design efforts. [2–4].

Currently, Field Programmable Gate Array (FPGA) are the state of art reconfigurable devices, with the key property of being capable of reconfiguration for an infinite number of times. Reconfiguring an FPGA means changing its functionality to support a new application and it is equal to
have some new piece of hardware mapped on the FPGA chip to implement new functionality. In other words FPGAs make it possible to have custom designed high density hardware in an electronic circuit, with the added bonus of having the possibility of changing it.

Since their introduction in 1980, FPGAs are gaining importance both in commercial as well as in research settings. The usage in former times was focused in rapid-prototyping systems for integrating test systems. After the test-phase often an ASIC approach substituted these systems for mass-production. Today FPGAs are used for a wide sector of applications [5]. The increased popularity of FPGAs results from their significantly increased capabilities. Contemporary FPGAs contain thousands of look-up tables, flip-flops and a large variety of built-in digital components, e.g., processors, memories, multipliers, transceivers and many more for implementing complex digital logic. Decreasing prizes for FPGAs and increasing mask-costs for ASIC design, also lower power-consumption of novel reconfigurable hardware and high flexibility opened a market for industry and a wide research area for scientific work.

FPGAs are configured by loading the bitstream. Conventional FPGAs are poor at handling functional diversity. Handling diversity requires reloading the FPGA. It is a slow process. While reloading, FPGA goes unused [2]. The evolution of the reconfigurable computing can be explained with the help of Figure 1.2. The reconfiguration time was found as one of the problems among communication bottleneck and inflexible net connections of first generation reconfigurable devices. To reduce the reloading time various methods were proposed in literature like configuration caching, perfecting configuration relocation and defragmentation, utilizing multiple contexts and using partial reconfiguration. Few FPGA vendors have sup-
CHAPTER 1. INTRODUCTION

Dynamic Partial Reconfiguration (DPR) is an ability to reconfigure selected areas of FPGA anytime after its initial configuration. This can be done while the design is operational and the device is active. Using a DPR methodology FPGA allows design modules to be swapped on the fly. It can be an important component to any design or application allowing designers more capabilities and resources that meet the eye. This enables a larger percentage of the application to be accelerated in hardware, hence reducing overall application execution time. This capability allows multiple design modules to time-share resources on a single device, even while the base design operates uninterrupted. Using DPR one can gain ability to adapt hardware algorithms, share hardware between various applications, increase resource utilization, provide continuous hardware servicing, upgrade hardware remotely etc [6].

Dynamic partial reconfiguration, a feature available with current FPGAs has attracted the attention of researchers working in the area of re-

Figure 1.2: Evolution of Reconfigurable Computing

ported the feature of Dynamic Partial Reconfiguration (DPR) in new generation FPGAs.
CHAPTER 1. INTRODUCTION

configurable devices and reconfigurable computing. This feature has potential to provide benefits in resource utilization, reconfiguration time and power dissipation. This thesis is an account of research work carried out in the field of DPR of FPGA for real time applications.

During this thesis different applications have been developed using DPR. The work has resulted into observation that, even if the time required to perform partial reconfiguration is smaller than that of the complete reconfiguration, it is still high for real time applications having tight timing constraints. It has formed the need of reducing reconfiguration time, which finally resulted into proposing a new reconfiguration controller. The implementation results indicate that the proposed reconfiguration controller is better than the reconfiguration cores reported in the literature from both reconfiguration throughput and resource requirement point of view. The plugging of the proposed reconfiguration controller in the real time applications brings in improvement in reconfiguration time to meet tight performance constraints. This is one of the important outcomes of this work. Further proposed efficient dynamic relocation core to perform dynamic allocation of reconfigurable block is one more outcome of this work. The details of these outcomes and other experiments performed during this work are presented in this thesis.

1.2 Motivation

When the idea of "dynamic hardware" was first enunciated by Esrtin in late fifties and early sixties [7], devices with required flexibility were not available yet. It is with the introduction of reconfigurable devices that the first hardware platforms for reconfigurable computing were possible. The
first devices used for this purpose had the ability of holding different configuration images at the same time, and were able to switch between them in few clock cycles. This type of devices was labeled *multicontext* devices.

As devices grew denser, holding multiple configuration contexts became impractical. Later devices are *single-context*, meaning that only one configuration image is held at a point in time. Single context devices are limited by the time it takes to load a new configuration every time a change in configuration is required. This time is referred to in the literature as *reconfiguration time overhead*. Reconfigurable architectures used multiple devices in order to partially hide the reconfiguration time overhead by performing computations in one device while the others were being reconfigured as needed. In this scheme, reconfiguration is seldom used, being more of change of application rather than a change in the application. Some other techniques like configuration caching, perfecting configuration relocation and defragmentation were proposed in literature to reduce *reconfiguration time overhead*.

Modern devices introduced two new key features that have enabled a myriad of applications that can use reconfiguration more intensively. First is dynamic partial reconfiguration. Partial reconfiguration reduces the reconfiguration time overhead by allowing partial configurations–thus smaller bitstreams–to be uploaded. The second feature is access to the configuration memory from within the device’s fabric. This means that a device can reconfigure itself. These two features have moved the paradigm of reconfigurable computing towards self reconfigurable computing systems. High performance, flexibility and adaptivity of these devices raise the interest in academic research and also in industrial fields of application.
CHAPTER 1. INTRODUCTION

As mentioned in Section 1.1 advances in FPGA devices, tools and techniques have allowed new application domains to be suitable for reconfigurable architecture beyond the initial applications of rapid prototyping and circuit emulation. Also the novel features introduced with these devices like DPR and self reconfiguration formed the motivation to study this feature in detail.

The continuous development in this field, as has been reported in the literature, indicated that there is a lot of scope for researchers to study and suggest new application domains that can benefit from this feature. The reduced reconfiguration time for DPR has motivated to study how performance critical run time applications can benefit from this feature. The literature review and the idea that such technique would be implemented in future in real world applications motivated to explore this research area. Run time applications based on DPR can be implemented with small devices and in low cost for the society, formed the motivation to define research problem in the area of reconfigurable devices for run time processes.

1.3 Problem Definition

Based on the study of research papers published so far, this research work is being carried out with the following objectives;

1. To study a new feature available with recent FPGAs; dynamic partial reconfiguration. Further, study its implementation flow by implementing simple case studies using the tools provided by vendors.

2. Explore application areas and domains that can take advantage of potential benefits offered by this feature. Implement case studies in
such areas and compare the important design metrics like resource utilization, power dissipation and reconfiguration delay of the new feature style against that of the conventional (non-DPR) implementation styles.

3. To study the effect of reconfiguration delay in real time applications. Though many researchers have designed various reconfiguration cores to reduce the reconfiguration time, the maximum limit is not yet met and also processor was required to handle the reconfiguration. Hence, it was proposed to develop new controller able to handle reconfiguration effectively and without a need of the processor.

4. To test the performance of the proposed reconfiguration controller in synthetic bench mark. Further, use the controller in real time applications where the performance constraints are tight and test the benefits of the controller in such applications.

5. Finally, extend the work by developing relocation core to perform dynamic allocation of reconfigurable modules. Though many researchers have designed various relocation cores to perform the relocation, reported relocation throughputs were low and also the relocation was handled by internal processor. Hence, it was proposed to develop new core to perform relocation effectively and without a need of the processor.

Based on the above objectives of the research work the entire research problem was broadly defined as;

1. To use various types of DPR in different applications and to explore different possibilities of gaining benefits from this feature. Also to study effects of DPR in real time applications.
2. To propose and develop a reconfiguration controller that can lower the reconfiguration time and remove the need of processor to perform reconfiguration. Also experimentally prove that the proposed reconfiguration controller is superior to conventional ones and best suited for real time applications.

3. To propose and develop a dynamic relocation core that can perform dynamic allocation of RM in minimum time without the involvement of processor in carrying out relocation.

1.4 Research Methodology and Results

The efforts to achieve the research objectives were carried out in systematic experimentation phases. Initially, the feature of dynamic partial reconfiguration and its implementation strategy and flow was studied thoroughly. Simple case studies based on DPR were implemented on FPGA using the Early Access PlanAhead (EAPR) flow provided by Xilinx. This work served as a platform to design new applications using this feature.

Study of the DPR and its functioning revealed that, it has potential to provide savings in resources, power dissipation and reconfiguration time while compared with non-DPR implementations. Also, different application areas and domains can be benefited by effectively using this feature in its implementation. Exploring new case studies based on DPR was seen as potential research domain in the last decade. Many researchers have studied this feature and proposed its use in various case studies. During this thesis also, different case studies were developed to use different types of DPR. All the case studies designed during this thesis are novel with re-
CHAPTER 1. INTRODUCTION

spect to the one proposed by other researchers so far and hence it is one of the contributions of this research work.

In the pursuit of using DPR in real time applications, the case study of real time video signal processing was implemented based on external DPR. After observing that the reconfiguration time is high, internal DPR was carried out using conventional reconfiguration cores. Processor based system was explicitly developed to carry out internal DPR. Though the reconfiguration time was reduced than the external DPR, it still was high to meet tight performance constraints of video signal processing. This has formed motivation to develop a reconfiguration controller that can improve the reconfiguration time and can perform reconfiguration without processor. A reconfiguration controller is developed and extensively tested for benchmark circuit to observe performance and other important parameters. The experimental results indicate that it provides higher performance than reported in literature. It should be noted that, so far no work has reported reconfiguration core for DPR without processor’s need.

Further, to observe effect of proposed reconfiguration controller in real time applications, the controller is plugged into two such applications. The one is real time audio-video signal processing and the other is an architecture for evolvable hardware. Experimental results for real time audio-video signal processing prove the usability of the proposed controller in real-time constrained applications, providing lower reconfiguration time for video and audio streaming. This experiment also proved that the reconfiguration process is carried out without using a processor, thus pure FPGA-based HW architectures do not need a processor to take advantage of dynamic partial reconfiguration. Further experimentation describes how a computationally intensive Evolvable Hardware sys-
tem based on dynamic reconfiguration has been improved introducing the proposed controller. Experimental results have shown that the novel controller brings benefits in the individual reconfiguration process as well as in the overall running time to complete the evolutionary process. The development of reconfiguration controller is one of the major contributions of this research work, made in the field of reconfigurable devices supporting DPR for real time applications.

It was realized that dynamic relocation of reconfigurable module is required in the Evolvable Hardware system used in previous experiments. After studying the relocation cores suggested by other researchers, it is being concluded that use of these cores will degrade the overall performance of the Evolvable Hardware system. Moreover, to be able to do complete intrinsic evolution, relocation in the Evolvable Hardware system should be carried out without a processor. This has formed motivation to develop a relocation core that can improve the relocation time and can perform relocation without processor. A relocation core is developed and tested to observe performance and other important parameters. The experimental results indicate that it provides higher performance than reported in literature. The development of relocation core is one of the major contributions of this research work, made in the field of reconfigurable devices supporting DPR for real time applications.

1.5 Organization of Thesis

The chapter scheme for the thesis follows the logical flow of research work. The thesis is organized in seven chapters as follows;

Chapter 2 - FPGA Technology and Dynamic Reconfiguration
CHAPTER 1. INTRODUCTION

This chapter starts with exploring architecture details of FPGAs. The chapter illustrates DPR; a feature that is available with recent Xilinx FPGA devices. Further, taxonomy of dynamic reconfiguration is given. Finally, low level configuration details viz. configuration ports and configuration bitstream format, of Xilinx Virtex-4 FPGA are presented.

Chapter 3 – Literature Survey This chapter begins with presenting possible merits of employing DPR in real world applications. After presenting different application scenarios that justify use of DPR, it presents literature survey of the work done by researchers in various application domains. Further, this chapter presents work done by researchers in order to enhance reconfiguration throughput. Finally, work done by researchers with respect to the dynamic allocation of Reconfigurable Module (RM) is presented.

This chapter highlights previous research work done in the area of dynamic partial reconfiguration and reconfiguration management. The chapter summarizes previous work and finally indicates how research objectives are formed. The chapter clearly indicates a logical base for the choice of the research work.

Chapter 4 – Applications Developed using Dynamic Partial Reconfiguration Applications developed using DPR during this thesis are presented in this chapter. The detail explanation on system architecture, experimental set-up for applications is given along with the implementation results in tabular form followed by comments.

Chapter 5 – Proposed Reconfiguration Controller Major contribution made in the field of dynamic partial reconfiguration, development of novel reconfiguration controller is presented in this chapter. Detailed design steps, development phases and implementation results of the reconfiguration
controller are presented. The details of the experiments performed to check the behavior and performance of the reconfiguration controller, the implementation results in both tabular and graphical form followed by comments are presented in this chapter. This chapter reports the comparison between the proposed controller and reported by researchers.

Further, this chapter presents the performance benefits achieved by using proposed reconfiguration controller in real time applications like audio-video signal processing and evolvable hardware. The detail explanation on system architecture, experimental set-up for applications is given along with the implementation results in tabular form followed by comments.

**Chapter 6 – Dynamic Relocation** This chapter presents one of the contributions of this research work in the form of development of dynamic relocation core. After presenting the design and development of this new relocation core, implementation results confirming its high throughput are presented. Further, this chapter reports comparison between the proposed relocation core and reported by other researchers.

**Chapter 7 – Conclusions and Future Scopes** The contributions as a result of research work carried out are highlighted in this chapter. The major contributions of this research work are;

1. Exploring various types of DPR by developing different applications using the feature

2. Development of novel reconfiguration controller to enhance reconfiguration throughput

3. Use of the novel reconfiguration controller in real time applications
4. Development of relocation core for dynamic allocation of RM

This work opens up various research directions that are presented at the end of this chapter.
When I Asked God for Brain and Brown
He Gave Me Puzzles in Life to Solve