A NEW FLEXIBLE PR DOMAIN MODEL TO REPLACE THE FIXED MULTI-PR REGION MODEL FOR DPR SYSTMS

Edward Chen[1]*, Dorian Sabaz*[2]*, William A. Gruver*[1]*, and Lesley Shannon*[1]

School of Engineering Science[1]
Simon Fraser University
Burnaby, BC, Canada
{ekchen, gruver, lshannon}@sfu.ca

Intelligent robotics Corporation[2]
North Vancouver, BC, Canada
dorian@iroboticscorp.com

ABSTRACT

Currently Xilinx's Dynamic Partial Reconfiguration (DPR) model requires the size and number of Partially Reconfigurable Regions (PR Regions) to be fixed during the design phase. Only one PR Module may be active per PR Region at any given time. This paper presents a new DPR model that replaces the current multi-PR Region model with a single *Partially Reconfigurable Domain* (PR Domain). Multiple PR Modules of different sizes are able to *concurrently* reside in the PR Domain. PR Modules may be loaded to any portion of the PR Domain given that sufficient resources exist. Future PR Modules can be designed with dimensions up to those of the PR Domain. A case study with multiple PR Modules of varied sizes is used to demonstrate the adaptability and flexibility of the PR Domain model.

1. INTRODUCTION

Xilinx's Field Programmable Gate Arrays (FPGAs) allow Dynamic Partial Reconfiguration (DPR) of the programmable fabric. DPR allows a selected area of the target device to be reprogrammed while the remainder of the fabric remains active. Multiple modules can time-share the same physical area on the target device, virtually increasing its physical resources. Figure 1 illustrates the key concepts for Xilinx's DPR model [8]. A *Partially Reconfigurable Region* (PR Region) is a physical area of the FPGA that can be dynamically reconfigured to implement different tasks. A *Partially Reconfigurable Module* (PR Module) is defined as one of the possible tasks that can be implemented in a PR Region. The Static Region contains design components that do not change at runtime.

Xilinx's DPR model requires the size and number of PR Regions to be fixed during the design phase and cannot be changed at runtime. Multiple PR Modules can time-share a PR Region, but only one PR Module may be active per PR Region at any given time. This mutually exclusive restriction of the PR Modules per PR Region requires the corresponding PR Region to be floorplanned to the size of its largest PR Module. Furthermore, the dimensions of future PR Modules are constrained to the dimensions of the largest PR Region. If designers create a new PR Module that is larger than the largest PR Region, the new PR Module must be partitioned to occupy multiple PR Regions. This constraint dramatically increases system complexity and degrades performance.

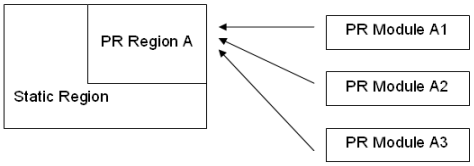

Fig 1. Relationship between a PR Region and its PR Modules

This paper presents a new DPR model that replaces the existing *multi-PR region* model with a s*ingle* flexible *Partially Reconfigurable Domain* (PR Domain). Multiple different-sized PR Modules are able to *concurrently* reside in the PR Domain. Additional PR Modules can be dynamically loaded without interfering with other PR Modules currently operating in the PR Domain. A PR Module is also not restricted to occupy the same area of the PR Domain; it may be loaded to whichever portion of the PR Domain that best optimizes system performance and maximizes resource utilization. Designers are able to implement new PR Modules with dimensions up to those of the PR Domain. The new model increases the design flexibility for PR Module placement at runtime and the design of new PR Modules.

2. RELATED WORK

Designers can implement a large design on a smaller FPGA by employing a DPR architecture. The two main approaches are 1-dimensional homogeneous [1] and 2-dimensional heterogeneous [7] partial reconfiguration frameworks.

The 1-dimensional homogenous approach utilizes a fixed, rectangular-shaped area on the target device subdivided

equally into multiple PR Regions. The size of each PR Region is set to the dimensions of the largest PR Module that will reside on the target device. The new PR Domain model extends this concept by combining all PR Regions into a single PR Domain to minimize resource overhead. Ullmann et al. [2] proposed a slot-based architecture where a fixed number of equally-sized slots are implemented as PR Regions. A common bus structure is used for inter-module communication. Rana et al. [4] extends this approach by connecting multiple partially reconfigurable FPGAs to a single embedded processor. The sizes of the PR Regions for all connected FPGAs are identical. The embedded processor determines the optimal location in which a task should be placed. Majer [5] proposed an Erlangen Slot Machine (ESM) based architecture that is able to combine the operations of multiple slots to seemingly create a larger PR Region. A complex crossbar structure is used for inter-slot communication. The new PR Domain model eliminates the need for a complex intra-module communication structure by allowing the entire PR Module to reside in a *single* PR Domain.

The 2-dimensional heterogeneous approach defines multiple PR Regions of different dimensions during the design phase. Runtime schedulers determine the appropriate placements of PR Modules based on their dimensions, system objectives, and runtime conditions. Hubner et al. [6] proposed a new architecture that allowed the 2D placement of interconnected, different sized PR Modules into different PR Regions on the target device. The modules use point-to-point connections that support adaptive communication protocols over communication networks based on currently instantiated modules. A further step is to implement an integrated runtime routing algorithm for individual connections of the PR Modules. In Marconi et al. [7], a novel heterogeneous 2D Block-Partitioned Model was proposed. Multiple PR Regions of different sizes are implemented on the target device, and an innovative online placement algorithm is developed to map the tasks into the appropriate locations.

The 2-dimensional approach increases system complexity by fragmenting the partially reconfigurable fabric. To implement modules larger than the largest PR region, multiple PR Regions must be connected together using additional communication infrastructure than inherently required by the PR Module. Conversely, if a PR Module is significantly smaller than the smallest PR Region, these resources are wasted as no other PR Module can share these resources. The new PR Domain model allows multiple PR Modules to concurrently operate within a contiguous partially reconfigurable fabric to minimize fragmentation and system complexity.

3. THE PROPOSED PARTIAL RECONFIGURABLE DOMAIN

The limitations of Xilinx's current DPR model are imposed by the Computer-Aided-Design (CAD) flow, not the programmable fabric itself. As such, by abstracting certain information from the tools, the new PR Domain model can be achieved.

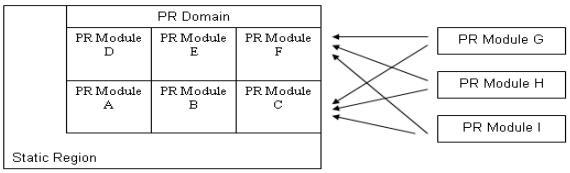

Fig 2. Relationship between a PR Domain and its PR Modules

The PR Domain is conceptually similar to a single large PR Region, yet multiple PR Modules can *concurrently* operate within it. Shown in Figure 2, the number of PR Modules that can concurrently reside is limited only by the resources available in the PR Domain. A PR Module is not restricted to always occupy the identical physical area of the PR Domain, but any area that meets the resource and I/O requirements. Either *PR Mod G, H,* or *I* can replace *PR Mod F* or *C* at runtime without interrupting the other PR Modules or the Static Region. Designers can set the dimensions of the PR Domain to the entire programmable fabric, excluding the Static Region, to achieve maximum resource flexibility. Any new PR Modules can be designed with dimensions up to those of the PR Domain. Currently, the PR Modules that are being used only include LUTs and FFs. However, the PR Domain can be specified to include embedded hard-cores (e.g. BRAMs, multipliers, and DSP blocks) so future work on schedulers will include selecting PR Module placements that properly leverage heterogeneous architectures.

The PR Domain does, however, share some of the design restrictions imposed on an individual PR Region for Bus Macros (BMs). Although the exact placements of BMs can be design-dependent, their placements in the PR Domain are fixed during the design phase and cannot be changed at runtime. At runtime, designers can selectively use the BMs that best optimize PR Module performance and leave the remainder unused.

4. PARTIAL RECONFIGURABLE DOMAIN IMPLEMENTATION

In Xilinx's DPR CAD flow, a *single DPR project* is developed with multiple PR Regions that are floorplanned during the design phase. Multiple PR Modules can be designed for each PR Region. In the newly proposed model, the Static Region and the PR Domain are created together in a DPR project, while multiple *independent*

DPR projects are used for the development of PR Modules. The complete DPR system consists of the DPR project containing the Static Region and the PR Domain, and the multiple PR Modules developed in independent DPR projects. The remainder of this section first describes the high-level implementation procedures, followed by the constraints of the new PR Domain model.

4.1. Implementation Overview

A multi-layered approach is used to implement the new PR Domain model. Each layer provides a portion of the functionality of the overall DPR system. This approach is illustrated in Figure 3.

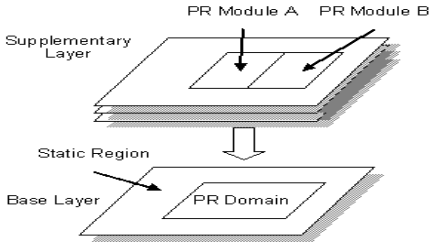

Fig 3. Multi-layered design of PR Domain

The *base layer* includes the Static Region and the PR Domain. The Static Region contains static components that remain fixed for the duration of the DPR system. The PR Domain accommodates PR Modules that are dynamically loaded at runtime. The *supplementary layers* includes multiple PR Regions used for the development of PR Modules. Both the base layer and the supplementary layers are designed using PlanAhead™, Xilinx's hierarchical floorplanning CAD tool that can also be used to develop DPR systems. The size and number of PR Regions in a supplementary layer is flexible but they must encompass the entire area of the PR Domain of the base layer. This ensures a one-to-one mapping of the partially reconfigurable fabric where the supplementary layer is superficially *imposed* on the base layer. Also, the placement of the BM(s) in the supplementary layers must map to the location of the BM(s) of the base layer.
Multiple supplementary layers can be implemented to generate different configuration of PR Regions. Different floorplanning combinations of PR Regions can be used to generate a PR Module that can be placed at different locations in the PR Domain. This concept is illustrated in Figure 4.

With the first supplementary layer, PR Mod A can only reside in the left half of the PR Domain. However, together with the second supplementary layer, the system has the flexibility of placing PR Mod A in either left or right half of the PR Domain. Similarly, PR Mod B can also be placed at either location. This concept can be extended to include complex placements strategies of multiple PR Modules to ensure efficient resource utilization of the PR Domain.

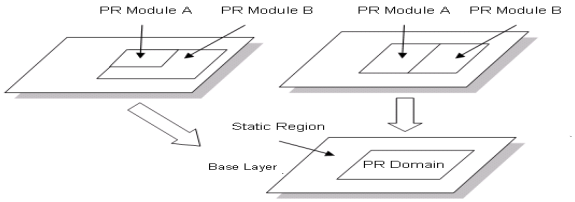

Fig 4. Different placement strategies for PR Modules

4.2. PR Domain Design Constraints

In the existing DPR model, the Static Region is first placed and routed. The Static Region may use routing resource inside the PR Regions as required to achieve better system routability. No contention of routing resources will occur between the Static Region and the PR Modules since the PR Modules have knowledge of the routing resources still available. In the newly proposed model, PR Modules are designed in multiple independent projects and may potentially cause contention between the PR Module of the supplementary layer and the PR Domain of the base layer.
To address this issue, the *routing private* attribute for the PR Domain is enabled in PlanAhead. Nets from the Static region are then prohibited from using the local routing wires of the PR Domain. Only routing wires within the PR Domain that cross into the Static Region can be used. Future work includes allowing the Static Region to freely use local routing wires and providing this information to the supplementary layers.

5. APPLICATION OF PR DOMAIN MODEL

To demonstrate the flexibility and effectiveness of the PR Domain, a case study of the newly proposed DPR Model is implemented. The DPR system is implemented on the Xilinx's ML505 development board with the Virtex-5 xc5vlx50t FPGA running at 100MHz. ISE9.1, EDK9.1 and PlanAhead 9.2.7 are used to implement the DPR system. The host machine is a Windows PC running XP-SP2 with 4GB of RAM. The high-level block diagram is illustrated in Figure 5.

The PR Domain has the physical dimensions of 84 slices on the target device. The PR Domain is designed to communicate with a maximum of two PR Modules through its four BMs. Xilinx's soft processor, the MicroBlaze, continually polls the identity of the two PR Modules residing in the PR Domain through the BMs. This information is displayed serially through UART. The PR Domain is partially reconfigured at runtime with different placements of various PR Modules.

A total of five PR Modules created in multiple supplementary layers are created for this case study. Each

PR Module can occupy different portions of the PR Domain. Their specifications are outlined in Table 1.

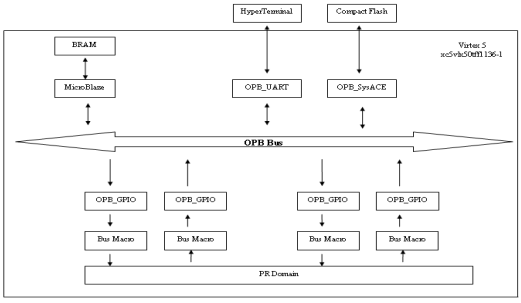

Fig 5. Block diagram of sample system

Table 1. PR Module sizes and locations

PR Module	Percentage of PR Domain	Placement on PR Domain
PR_0	100	Entire
PR_1	50	Left half
PR_1	50	Right half
PR_2	50	Left half
PR_2	50	Right half
PR_3	33	Left 1/3
PR_3	33	Middle 1/3
PR_3	33	Right 1/3
PR_4	66	Left 2/3
PR_4	66	Right 2/3

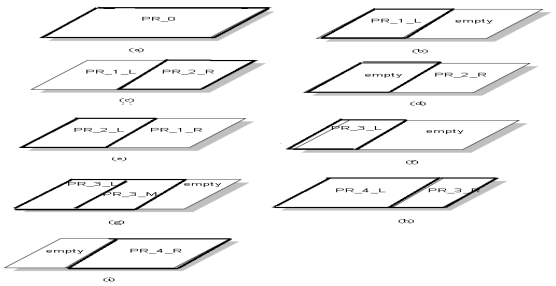

Fig 6. PR Module placements in the PR Domain

To illustrate the flexibility of the PR Domain, the sequence of placement configurations is shown in Figure 6. The operation of the Static Regions remains uninterrupted throughout the configurations. The identity of the PR Modules currently residing in the PR Domain is continuously displayed serially through UART. Newly placed PR Modules are highlighted in bold in Figure 6.

6. CONCLUSION

Xilinx's current DPR model requires the size and number of PR Regions to be fixed during the design phase and cannot be varied at runtime. Multiple PR Modules of different dimensions time-share a PR Region on the target device. This mutually exclusive characteristic of the PR Modules in a PR Region results in inefficient use of available resources, particularly if the largest PR Module is rarely in use.

This paper presents a new DPR model that replaces the multi-PR Region model with a single PR Domain. Multiple PR Modules of different sizes are able to concurrently reside in the PR Domain. A PR Module can be dynamically loaded without interfering with the runtime operation of other PR Modules currently in the PR Domain. PR Modules are not restricted to occupy the same physical resources within the PR Domain each time they are loaded. The designs of new PR Modules are only constrained by the size of the overall PR Domain. Different placement strategies of the PR Modules can be realized through the use of multiple supplementary layers to ensure that the PR Modules are optimally placed within the PR Domain. A case study with multiple PR Modules of varied sizes and placements demonstrated the functionality and flexibility of the PR Domain. Future work includes the design of schedulers for the optimal placement of the PR Modules, investigation into the efficient use and placements of PR Modules requiring embedded hard-cores, and automating the newly proposed model to facilitate the development of an improved DPR system.

7. REFERENCES

[1] M. Hübner and J. Becker. Tutorial on Macro Design for Dynamic and Partially Reconfigurable Systems. Presented at *Architecture of Computing Systems (ARCS)*, Zurich, Switzerland, March, 2007.

[2] J. Becker, D. Adam, M, Hübner. Network tool support and architectures in adaptive reconfigurable computing, *International Conference on Very Large Scale Integration (VLSI),* pages 134-139, Atlanta, GA, Oct, 2007

[3] M. Ullmann, M. Hübner, B. Grimm, J. Becker. An FPGA Run-Time System for Dynamical On-Demand Reconfiguration, Presented at *Internaional Parallel and Distributed Processing Symposium (IPDPS) Workshop*, Santa Fe, New Mexico, April, 2004

[4] V. Rana, M. Santambrogio, D. Sciuto,B. Kettelhoit. Dynamic Reconfigurability in Embedded System Design. *International Symposium on Circuits and Systems*, Pages 2734-2737, New Orleans, Louisiana , May 2007

[5] M. Majer, An FPGA-Based Dynamically Reconfigurable Platform: From Concept to Realization, *International Conference on Field-Programmable Logics and Applications*, pages 1-2, Madrid, Spain, Aug 2006

[6] M. Hübner, C. Schuck, M. Kiihnle, and J. Becker. New 2-Dimensional Partial Dynamic Reconfiguration Techniques for Real-time Adaptive Microelectronic Circuits. *Proceedings of the IEEE Computer Society Annual Symposium on Emerging VLSI Technologies and Architectures,* page 6, Karlsruhe, Germany, March, 2006

[7] T. Marconi, Y. Lu, K.L.M. Bertels, G. N. Gaydadjiev, Online Hardware Task Scheduling and Placement Algorithm on Partially Reconfigurable Devices, *Proceedings of International Workshop on Applied Reconfigurable Computing (ARC),* pp. 306-311, London, UK, March 2008

[8] Xilinx Early Access Partial Reconfiguration with PlanAhead 9.2 Users Guide