Chapter 15

A Novel Puzzle Based Compaction (PBC) Strategy for Enhancing the Utilization of Reconfigurable Resources

Ahmed I. Saleh
Mansoura University, Egypt

ABSTRACT

Partially reconfigurable field programmable gate arrays (FPGAs) can accommodate several independent tasks simultaneously. FPGA, as all reconfigurable chips, relies on the “host-then-compact-when-needed” strategy. Accordingly, it should have the ability to both place incoming tasks at run time and compact the chip whenever needed. Compaction is a proposed solution to alleviate external fragmentations problem, trying to move running tasks closer to each other in order to free a sufficient area for new tasks. However, compaction conditions the suspension of the running tasks, which introduces a high penalty. In order to increase the chip area utilization as well as not affecting the response times of tasks, efficient compaction techniques become increasingly important. Unfortunately, traditional compaction techniques suffer from a variety of faults. This paper introduces a novel Puzzle Based Compaction (PBC) technique that is a shape aware technique, which takes the tasks shapes into consideration. In this regard, it succeeded not only to eliminate the internal fragmentations but also to minimize the external fragmentations. This paper develops a novel formula, which is the first not to estimate, but to exactly calculate the amount of external fragmentations generated by accommodating a set of tasks inside the reconfigurable chip.

INTRODUCTION

In spite of its flexible architecture and good internal arrangement, FPGA usually suffers from external fragmentations (Roberto, Francesco, Massimo, Marco, & Donatella, 2009). As several tasks enter and leave the chip, the chip area becomes fragmented into small islands, which are unable to host newly incoming tasks (Marconi, Lu, Bertels, & Gaydadjiev, 2008). This decreases the chip ability to host new tasks, which in turn harmly affects both the chip area utilization as well as the response times of running and waiting...
A Novel Puzzle Based Compaction (PBC) Strategy for Enhancing the Utilization of Reconfigurable Resources

Figure 1. How external fragmentations affect the chip area

As illustrated in Figure 1, the chip is too fragmented to host either task $T_x$ or $T_y$. Hence, they are waiting in the ready queue despite there being available sufficient noncontiguous cells to host them. Accordingly, response times for those waiting tasks become longer and chip utilization is lower than it could be.

Generally, task arrangement in reconfigurable computing may be done in either offline or online scenarios (Banerjee, Bozorgzadeh, & Dutt, 2005). The offline scenario is similar to the traditional packing problem (Fekete, Kohler, & Teich, 2001) (Fekete & Schepers, 2004) where both tasks’ size as well as its service time are known in advance. Accordingly, an optimal or near optimal task arrangements could be derived. However, such scenario becomes obsolete with the rapid development in the area of reconfigurable computing. New applications have been emerged in which each task has an arrival time and size that are unknown in advance. Hence, tasks of those applications need to be scheduled online, while the task placement is not the issue (Resano & Mozos, 2004). The challenge is then to find sufficient contiguous cells for those newly incoming online tasks.

Online task arrangement may be either static or dynamic (Banerjee, Bozorgzadeh, & Dutt, 2006). The static behavior conditions the no-preemption of running tasks, and so, it does not offer task rearrangement (Jean, Tomko, Yavagal, Shah, & Cook, 1999; Fekete, Kohler, & Teich, 2001; Bazargan, Kastner, & Sarrafzadeh, 2000; Tabero, Septien, Mesha, Mozos, & Roman, 2003). On the other hand, the dynamic behavior for online task arrangement allows the scheduler to preempt the running tasks. Hence, preempted tasks can be rearranged to create a sufficient space for the incoming tasks when those new tasks have to wait. The process of task rearrangement, which aims to minimize the amount of external fragmentations (by making tasks closer to each other) is called the chip “compaction” (El Farag, Hatem, & Sheheen, 2007; Thomas, Yi, Koen, & Georgi, 2010).

Whenever a set of online arriving tasks are to be hosted into a reconfigurable chip, the chip will certainly suffer from internal and/or external fragmentations. To the best of our knowledge, almost all introduced online task placement techniques suffer from internal fragmentations. The reason is quite simple; “as they assume rectangular tasks”. For non-rectangular tasks, the inclusive rectangle for the entire task is considered. Consequently, some area may get wasted, which is called internal fragmentations. In spite of its critical impact on the chip utilization, internal fragmentations problem has not been addressed yet. On the other hand, with the online task placement, as tasks are allocated and de-allocated over time, external fragmentations will unfortunately appear. External fragmentations may prevent a new task from being hosted into the chip although there exists a sufficient number of noncontiguous cells. Unlike internal fragmentations, external ones can be...