Run-Time FPGA Partial Reconfiguration for Image Processing Applications

Shaon Yousuf and Ann Gordon-Ross
NSF Center for High-Performance Reconfigurable Computing (CHREC) [1]
ECE Department, University of Florida
{yousuf, ann}@chrec.org

The dynamic reconfigurability of SRAM-based FPGAs is advantageous to space-based systems and applications by providing the flexibility to dynamically load and unload system functionality on demand, thus providing benefits such as reduced payload and energy consumption. The two main types of dynamic reconfiguration are full reconfiguration and partial reconfiguration (PR). Full reconfiguration reconfigures the entire FPGA device and PR reconfigures only a portion of the FPGA device (a PR region or PRR). PR is advantageous over full reconfiguration because, unlike full reconfiguration, which halts entire system execution, PR only halts the reconfigured PRR while the remainder of the device continues executing. PR reduces overall memory requirements (PR bitstream sizes are smaller than full bitstreams) and can increase performance (only the reconfigured PRR is halted) as compared to full reconfiguration for applications that do not require all functionalities at the same time (i.e. software defined radios).

PR system design is challenging, requiring specialized design flows and extensive designer expertise in order to fully realize PR benefits. To ease some design challenges, previous work proposes two PR design flow methodologies: a special-purpose and a multi-purpose design flow. The special-purpose design flow creates a PR design specifically tailored for a target application, which potentially maximizes PR benefits, but requires a priori knowledge of application dependent specifications. Alternatively, the multi-purpose design flow creates a generic PR design flexible enough for a multitude of applications, but due to application unknowns, can limit PR benefits.

However, even with these design flows, PR system design becomes increasingly difficult as application size and functionality increases. One such difficulty involves application partitioning (determining the specific functionality mapped to each PRR) under constrained system performance. Whereas PR has the potential for increasing system performance, poor application partitioning can decrease system performance. Alternatively, optimal application partitioning involves significant formulation and planning, requiring an application designer to have in-depth knowledge of the application's inner workings as well as considerable target FPGA device knowledge.

Since optimal application partitioning requires exploration of an exponential design space, analyzing potential PR design benefits for different application types can provide valuable insights and ease PR design for similar typed applications. A key enabling technology for low-power and high-performance image transmission in on-line satellite communications are image processing applications. JPEG compression is the international standard for still color image compression and provides a basis for various motion-capture compression techniques. Due to JPEGs high-impact, an in-depth study of PR system design and PR benefits influences a myriad of system domains.

In this paper, we explore a special-purpose PR design flow for the JPEG compression algorithm. First, we provide an overview of a hardware implementation of the JPEG sequential discrete cosine transform (DCT)-based encoding process – the JPEG hardware compressor. Next, we create a PR design of this compressor and discuss the issues and challenges faced during the PR design process. Finally, we highlight PR benefits for the PR JPEG hardware compressor as compared to a non-PR implementation using numerical analysis of device slice requirements on a Virtex-4 device.

[1] This work was supported in part by the I/UCRC Program of the National Science Foundation under Grant No. EEC-0642422. We also gratefully acknowledge tools provided by Xilinx.