Preface

Rapid advances in VLSI technology have enabled fabrication of billions of transistors on a single chip. Technology scaling has allowed more than one processor core to be integrated in a single chip. The current computing systems, including desktops, laptops, and mobile phones have many-core processor chips. Software applications are parallelized to execute on multiple processing units/cores concurrently to reduce the overall execution time. Although sophisticated compute systems have been developed for fast execution, software applications belonging to certain domains take a very long time (days to months) for producing results.

Bioinformatics is one such domain in which applications take a long time to execute. It is also a multidisciplinary research field consisting of computer science, mathematics, and statistics, which deals with storing, analyzing, and interpreting large biological data. The recent advancements in biological research has resulted in generation of large amounts of digital data. Most of the bioinformatics algorithms are both data intensive and compute intensive. Speedups can be obtained using specialized hardware along with the existing processor architectures.

General-purpose processors are often augmented with hardware accelerators for compute intensive application to reduce the computation time. Typical hardware accelerators consist of a large number of small execution units which facilitate parallel execution. Very often they represent transformation of loops in a sequential code (temporal iteration) to spatial unrolling of the loop (spatial iteration) to reduce the computation time through concurrent execution. Some of the popular hardware accelerators are GPUs, FPGAs, and CELL processors. The accelerators are programmable and hence can be used for a variety of related applications. FPGA-based accelerators are known to be effective in speeding up certain kinds of applications when compared to other accelerators. Since FPGAs are configurable, they can be customized to implement a variety of processing elements as accelerators. But speedup may not always be possible as FPGAs run at slower clock frequency vis-a-vis processors and the resources available in the FPGA might not be sufficient for the implementation of a significant number of copies of the processing elements. FPGAs with heterogeneous mix of coarse grained hard blocks along with

programmable soft logic, can facilitate implementation of a much larger number of processing elements and thus achieve higher speedups.

FPGAs have evolved and the hardware blocks useful for many applications are being implemented within the FPGA fabric as Hard Embedded Blocks (HEBs). Introduction of HEBs in FPGA can significantly increase the performance of FPGA-based accelerators. Modern FPGAs contain specialized embedded units like memory units, array multipliers, DSP computation units, etc. In fact many FPGAs have embedded processor cores. Based on the application a matching FPGA with the right HEBs is chosen. For example in Xilinx Virtex-4 FPGAs, the SX-series has only LUTs, in the LX-series there are DSP units as HEBs and the FX-series have Power-PC embedded in them. The user can choose the best suited FPGA architecture according to his/her needs. It is easy to predict that many more such hard blocks will be embedded into future FPGAs.

The evaluation approach to identify and incorporate HEBs is complex as there are many parameters and constraints such as area, granularity routing resources, etc., which need to be considered in an integrated manner to get efficient implementation. Incorporating HEBs in FPGA involves a clear tradeoff as this may occupy a significant area and hence may reduce the configurable logic. Further, they may not be usable for many applications. On the other hand, they may give very significant speedups for certain applications. Clearly, the challenge is to identify kernels that are useful for a class of applications and justify designing customized HEBs that are effective in significantly speeding up an important class of applications. There is a need to develop a methodology to explore the FPGA fabric design space to evaluate the nature and number of HEBs based on other constraints.

This book presents an evaluation methodology to design future FPGA fabrics incorporating hard embedded blocks to accelerate applications. This methodology is useful for selection of blocks to be embedded into the fabric and for evaluating the performance gain that can be achieved by such an embedding. The use of the methodology is illustrated by designing FPGA-based acceleration of two important bioinformatics applications: Protein docking and Genome assembly. This book explains how the respective HEBs are designed and how hardware implementation of the application is done using these HEBs. The impact of use of HEBs on accelerating these two applications is shown. The methodology presented in this book may also be used for designing HEBs for accelerating software implementations in other domains as well.

We thank Prof. Dominique Lavenier for letting us collaborate with the SYMBIOSE lab, IRISA, Rennes, France. We are also grateful to him for providing us guidance on this project. We thank Pierre Peterlongo for his valuable inputs on Mapsembler.

Hong Kong — B. Sharat Chandra Varma
New Delhi, India — Kolin Paul
New Delhi, India — M. Balakrishnan

Springer

http://www.springer.com/978-981-10-0589-3

Architecture Exploration of FPGA Based Accelerators for BioInformatics Applications

Varma, B.S.C.; Paul, K.; Balakrishnan, M.

2016, XV, 122 p. 54 illus., 40 illus. in color., Hardcover
ISBN: 978-981-10-0589-3