

Pipelined Streaming Computation of Histogram in FPGA OpenCL

Mohammad HOSSEINABADY¹, Jose Luis NUNEZ-YANEZ

Electrical and Electronic Engineering, Bristol University, UK

Abstract. The emergence of High-Level Synthesis (HLS) techniques and tools, along with new features in high-end FPGAs such as multi-port memory interfaces, has enabled designers to utilize FPGAs not only for compute-bound but also for memory-bound tasks. This paper explains how to efficiently parallelise histogram, as a memory-bound task, utilizing the OpenCL framework running on FPGA. We have run our implementation on three high-end FPGAs including Alpha Data 7v3, Alpha Data ADM-PCIE-KU3 and Xilinx KU115. The 256 fixed-width bins histogram running on 7v3, KU3 and KU115 platforms shows 8.38, 15.29 and 38.57 Giga bin Update Per Second (GUPS), respectively. The best result, i.e., 38.57 GUPS on KU115 platform defeats the Nvidia GeForce 1060 GPU with 31.36 GUPS. In addition, it shows better performance than the one obtained in the dual socket 8-core Intel Xeon E5-2690 with 13 GUPS and 60-core Intel Xeon Phi 5110P coprocessor with 18 GUPS. The proposed implementation is not sensitive to locally invariant (LI) data sets, while the performance of GPU and CPU implementations drops with LI data. Processing locally invariant data sets shows that our FPGA implementation can be up to 91.4% and 44.9% faster than that of the GeForce 1060 and 1080 GPUs, respectively. The source codes of the designs are available at https://github.com/Hosseinabady/histogram_sdaccel.

Keywords. FPGA, High-Level Synthesis, Stream Computing, Histogram

1. Introduction

FPGAs have been used as accelerators for compute-bound applications in different fields, including image and video processing platforms, scientific applications and embedded systems. The time consuming and tedious design process based on a Hardware Description Language (HDL), was one of the main hurdles to use FPGAs in mainstream platforms. To alleviate this issue, researchers and industry have proposed high-level synthesis (HLS) techniques [1] that receive an algorithm in a high-level language such as C/C++/SystemC/OpenCL [2,3] and then transform it into a Register Transfer Level (RTL) description which can be synthesised by logic synthesis tools into FPGA configuration bitstreams.

In this paper, we explain an efficient implementation of histogram, as memory-bound task, using HLS tools. Histogram is a fundamental statistical tool in the algorithms of various fields including image processing, scientific computing, data-base analysis,

¹Mohammad Hosseinabady, Electrical and Electronic Engineering, Bristol University, UK, BS8 1UB; E-mail: m.hosseinabady@bristol.ac.uk