別表６
（３）

Keio University

Adaptable Architectures for Acceleration of Protocol Processing using FPGAs

September 2013

Akagic Amila

別表６
（３）

Keio University

Thesis Abstract

No.

Registration Number	■ “KOU” □ “OTSU” No. *Office use only	Name	Akagic Amila

Thesis Title
Adaptable Architectures for Acceleration of Protocol Processing using FPGAs

Thesis Abstract

No.

The emergence of multi-Gigabit Ethernet and ever-increasing volume of network traffic on the Internet has begun outpacing server capacity to manage incoming data. In recent years, the network traffic exhibits constant increase, due to the confluence of many market trends. Today, data centers are considering employment of new technologies, such as 40- and 100-Gb Ethernet, however their adoption rate is still rather small. The major concern is that the potential for such high bandwidths would not be exploited, due to the communication overhead that consumes high levels of processor's processing power. One major source of processing overhead is the TCP/IP stack. This problem has been addressed in various methods. One method is to dedicate one or more cores for TCP/IP processing exclusively. However, with the new paradigm shift to multicore processors, it is hard to guarantee the high throughput for inherently sequential processes, such as cyclic redundancy checks. Other methods include protocol processing *offloading* onto a specialized hardware and using special large packets known as jumbo frames. This has been specially beneficial in storage applications that transfer large blocks of data.

The future networks also seem to take a new direction toward so called *programmable networks*, which will allow greater agility, programmability and flexibility. In this thesis, we take another step in this direction by utilizing programmable hardware to achieve the same goals. At first, we target one of the challenging aspects of iSCSI processing, which is processing of digests or Cyclic Redundancy Checks (CRC). CRCs are often characterized as computationally intensive, and thus often substituted with less efficient error detection schemes. We propose a *non-adaptable* and *fully-adaptable* CRC accelerators based on a table-based algorithm, which has been rarely used in hardware implementations. The *non-adaptable CRC accelerator* is suitable for acceleration of a specific application, and has no ability to adapt to a new standard or an application. The *fully-adaptable CRC accelerator* has ability to process arbitrary number of input data and generates CRC for any known CRC standard during run-time. We modify table generation algorithm in order to decrease its space complexity.

We also address the problem of efficiently implementing IP-based iSCSI Offload Engine which operates on the top of the TCP/IP protocol stack. Based on the analysis of iSCSI traffic, CPU utilization and throughput of software-based Open-iSCSI, we propose a new architecture which offloads data transfer and related non-data functions to an FPGA based adapter. The resulting architecture relieves the host CPU from computational burden imposed by software implementations. The iSCSI Offload Engine allows very low utilization on the host CPU of approximately 3%. Our work is a step toward the goal of using hardware accelerators to enable higher levels of agility, programmability and flexibility in future networks.