I. INTRODUCTION

Network communication is one of the most important terms for recent computing environment. The bandwidth of end-to-end communication grows from 100 Mbps to 1 Gbps or more in the current and near future. Due to the increasing requirements for establishing faster communications, the designers have to consider the performance of packet processing. TCP/IP is one of the major protocols for network data transactions. In order to make good use of high-speed network communications, TCP/IP packet processing within the termination nodes becomes a major bottleneck in delivering high-speed network computing [1]. An approach to solve this problem is to implement the TCP/IP packet processing by custom hardware [2], we call it TCP/IP Core. In pervasive computing environment, in addition, improving energy efficiency is an inevitable constraint for system designers. Therefore, low-power packet processing will strongly be required for mobile network-connected computing devices. Motivated by this, our work presented in this paper focuses on the power consumption of a TCP/IP Core. Our purpose is to design a TCP/IP Core and analyze its power consumption for power-aware TCP/IP hardware designs. In the following sections, we describe a brief summary of our design and power analysis results.

II. TCP/IP HARDWARE IMPLEMENTATION

Figure 1 shows the TCP/IP protocol stack that consists of independent several layers. The TCP belongs to the transport layer as well as UDP, and works for ensuring connection reliability. On the other hand, the IP belongs to the network layer and its main purpose is to handle addresses information for sending or receiving data correctly. In order to implement the TCP/IP operation in hardware, we partition the TCP and IP functions into several modules, respectively. This module partitioning has been done based on the packet header structure. In each field of the TCP and IP header, the information which notice how the transmitted or received data should be treated is included. Based on the packet-header information, we partitioned the TCP core into four modules as follows:

- **port_ctr**: this module works for data input, or output, form, or to, the Application layer.
- **data_ctr**: this module divides and reconstructs the data to be conducted based on sequence numbers obtained from the data header. Moreover, a timer is included in this module to control data streaming.
- **window_ctr**: this module measures the amount of current load of the TCP processing unit. This can be done by monitoring the number of empty entries of the data buffer.
- **checksum**: this module checks errors in the data. If one or more error(s) is detected, it sends back retransmission requests.

Actually, the TCP core includes two types of units: one for data transmission and one for receive. On the other hand, the IP is organized by the following four modules.

- **service_info**: this module deals with IP packet header information, and recognizes the type of current service.
- **frag_ctr**: this module checks the occurrences of fragmentations. If yes, then must be identified each data.
- **checksum**: In this module, the work is to check header, and to compute time to live for packet.
- **add_ctr**: In this module, the work is to decide source and determination address.

Designing a TCP/IP Core for Power Consumption Analysis

Kenichi Tanamachi, Koji Inoue, and Vasily G. Moshnyaga

Department of Electronics Engineering and Computer Science, Fukuoka University
8-19-1 Nanakuma, Jonan-ku, Fukuoka 814-0180, JAPAN
In order to analyze the power consumption of a TCP/IP Core, we have done an RTL design by using Verilog-HDL. The HDL source files are translated into standard-cell levels by the synopsys Design Compiler with a 0.35 µm CMOS library. Furthermore, we have extracted wire capacitances based on layout design data generated by an auto P&R tool. Table 1 and Table 2 report the design results for TCP and IP core, respectively. The area and power are the values reported from the synthesis tool. This means that the area does not include space for wires. Moreover, the power consumption presented in the tables does not take account of the wire capacitances and switching activity. To improve the accuracy of power consumption, we have extracted the total wire capacitance for each module because the power consumption of CMOS LSI is in promotional to switched load capacitances. Note that we have not finished the layout of IP core design and it is our on going work.

First, we discuss the power consumption of the TCP core. As can be seen from Table 1, the checksum component has a significant impact on the total power consumption, which account for 60-65 % of total. This component includes checksum circuit in order to ensure trust data communications. This part requires a large number of transistors. The second power hungry module is the data_ctr that consumes about 22-30% of total power. This is because the data_ctr has complex operations, such as a management of divided input data based on sequence number, acknowledge checking for stable communications, etc. Next, we look at the design results for the IP core. From Table 2, it is observed that the checksum unit is the most power consuming function as same as the TCP design, almost half of total power is consumed. The frag_ctr unit also relatively consumes large percentage of power. By the two modules, the frag_ctr and the checksum, 77% of the total IP power is dissipated. Finally, we compare the power consumption of TCP core and that of IP core. When we focus only on the power reported by the logic-synthesis tool, the power consumed by the TCP is almost double of that of the IP core. This is because the TCP unit need to employ the independent circuits for both data transitions and receives. Consequently, we conclude that designers should make the best effort to lower the power consumption of TCP’s checksum and data management functions. Furthermore, for IP design, checksum and frag-control function should be focused on.

III. DESIGN RESULTS

In this paper, we have introduced our preliminary design results. The goal of our research is to develop fast, low-power TCP/IP hardware for future pervasive computing. In this evaluation, we have not taken account of switching activity for each node that is one of the most important factors for power consumption. Our on going work is to capture input stream data to real TCP/IP layers in order to improve the accuracy of our measurements.

Acknowledgment

This research was supported in part by the Grant-in-Aid for Creative Basic Research (14GS0218) and the Encouragement of Young Scientists (A)(14702064). The VLSI chip design in this study has been done in the chip fabrication program of VLSI Design and Education Center (VDEC), the University of Tokyo in collaboration with Rohm Corporation and Toppan Printing Corporation.

REFERENCES
