An investigation of Mobile Device Supporting Online Self-dynamic Partial Reconfiguration on a Testbed System

Yuma Chihara and Akira Yamawaki

Kyushu Institute of Technology, Kitakyushu, Japan

*Corresponding Author: yama@ecs.kyutech.jp

Abstract

A mobile device has become widely used not only for telephone and email, but also for entertainment.

Some entertainments such as 4K require heavy process for video and high-quality games. Therefore, a mobile device needs higher performance and lower power consumption. To achieve them, hardware implementation instead of processing heavy software is effective. However, it is impossible to prepare a wide variety of hardware on a mobile device. Therefore, we considered installing a reconfigurable device on a mobile device to support the Dynamic Partial Reconfiguration (DPR).

The mobile device we have proposed is equipped with a reconfigurable device such as an FPGA inside the System on Chip. The website like Apple Store distributes the DPR application. The DPR application consists of software and hardware data. When executing the application, the software runs first. Then, the mobile device uses the hardware data corresponding to the software and configures reconfigurable hardware on FPGA. Since hardware can perform parallel processing, it is expected to achieve higher performance and lower power consumption than software.

In this paper, we aimed to investigate the feasibility of the mobile device supporting DPR. So, we create this mobile device with the FPGA and support online the DPR by using Lightweight IP (LwIP). LwIP is lightweight TCP / IP stack for embedded devices. Then, the correct operation of the created the mobile device was confirmed. Furthermore, the time required for the DPR was sufficiently short, confirming the feasibility of a mobile device with the FPGA.

Keywords: mobile device, FPGA, partial reconfiguration.

1. Introduction

A mobile device has widely used not only for telephone and email, but also for entertainment. In Japan, the time of using a mobile device for entertainment in 2018 is longer than that in 2012[1]. Some entertainments such as 4K video and high-quality games are heavy to process. Therefore, we think a mobile device must achieve higher performance. Since a mobile device runs by batteries, higher power consumption leads to a decrease in time operating device. Therefore, power saving must also be considered. To achieve them, it is effective to implement hardware instead of processing heavy software. However, we cannot prepare a wide variety of hardware on a mobile device. Therefore, we considered installing a reconfigurable device on a mobile device to support Dynamic Partial Reconfiguration (DPR).

The mobile device we have proposed is equipped with a reconfigurable device such as an FPGA inside the System on Chip. The website like Apple Store distributes the DPR application. The DPR application consists of software and hardware data. When executing the application, the software runs first. Then, the mobile device uses the hardware data corresponding to the software and configures reconfigurable hardware on the FPGA. Since hardware can perform parallel processing, it is expected to achieve higher performance and lower power consumption than software.

In this paper, we aimed to investigate the feasibility of the mobile device I proposed. So, we create this mobile device with the FPGA and support the online DPR by using Lightweight IP (LwIP). LwIP is lightweight TCP / IP stack for embedded devices.

Fig 1. The DPR application

2. DPR mobile device

2.1 System of application delivering

The website like Apple Store distributes the DPR application. The DPR application consists of software executable file and data of hardware as shown in Fig. 1. The CPU in the mobile device executes software executable file that is light process. And, FPGA execute heavy process.

2.2 System of the mobile device I propose

The mobile device we have proposed equips with a reconfigurable device such as an FPGA inside the System on Chip. So, our mobile device can execute the DPR. The DPR is a function that can reconfigure a specific circuit (Reconfiguration Module, RM) without stopping the entire circuit [2-3]. Figure 2 shows how to use the mobile device we have proposed. When executing the application, the software runs first. Then, the mobile device uses the hardware data corresponding to the application and configures reconfigurable hardware on the FPGA. When executing another application, the mobile device similarly configures reconfigurable hardware on the FPGA

Fig 2. How to use mobile device proposed

3. Online self-dynamic reconfiguration

3.1 Hardware platform

Figure 3 shows the block diagram of the hardware platform of the DPR mobile device created in this study.

Fig 3. Hardware platform

This platform was created on vivado2016.4 using HDL description. The CPU in Fig. 3 is an FPGA embedded processor. The ICAP uses the reconfigurable hardware data obtained from the lwIP and execute the DPR according to the hardware data. The reconfiguration area generates LEDs blinking patterns according to the hardware rewritten by the DPR.

3.2 lwIP (light-weight Internet Protocol)

The lwIP (light-weight Internet Protocol) is a widely used open source TCP/IP stack designed for embedded systems. In addition, it is designed to work with or without an OS and with or without thread support. lwIP can also use UDP.

In this study, we created a mobile terminal side program on an FPGA with an embedded processor using UDP-based lwIP. We also created a hardware platform on the FPGA for dynamic reconfiguration using lwIP.

3.3 The GUI interface

Fig 4. GUI interface

The GUI interface consists of check boxes for selecting hardware (circuit data), as shown in Figure 4, and buttons for selecting whether to download hardware or perform dynamic partial reconstruction, as shown in Figure 4. I set up four pieces of hardware to see how it works.

3.4 Whole system

Fig 5 shows whole system I created. The GUI interface is running on PC. If buttons are pushed on the GUI, UDP

Fig 5. Whole system

Fig 6. How to use this device

based program written in C language send a command to the mobile phone. The command is to identify whether it is the download or the DPR. After the mobile device receive the command by lwIP, the mobile device execute download by lwIP or execute the DPR (Figure 6). After the DPR, the LEDs blinking according to the hardware rewritten by the DPR. Pattern of LEDs blinking tells us whether this system is working correctly.

4. Experiment

4.1 Experiment environment

We tested the mobile device with FPGA we created to see if it worked correctly and investigate whether the mobile device with FPGA is effective. The platform developed by Vivado 2016.4 is built on the Xilinx ZYNQ FPGA which is equipped on Digilent ZYBO. An embedded processor in ZYNQ is used for download hardware data by using lwIP and for the DPR. On PC, The GUI interface is running.

4.2 Experimental result

To verify the operation, we downloaded three circuits with different LED blinking patterns and an empty circuit, and had each circuit perform the DPR sequentially. Operational verification confirmed that the DPR actually works correctly.

In addition, online circuit data download and dynamic reconfiguration times were measured (Figure.7).

As a discussion of the measured data, we first look at the download times in Fig. 7. This is thought to be due to the fact that wireless communication (Wi-Fi) is used for

Fig 7.Measure for each circuit

downloading, which is affected by network congestion. Looking at the DPR times in Fig. 7, the DPR times showed almost no variation in value. This is because the DPR is done in the FPGA and is not affected by the network. For all circuits, the download time and the DPR time are almost the same. The reason for this is that the size of the circuit is determined in advance when designing the reconfiguration area in vivado2016.4. The size of the circuit used in this study was all 527KB. If the size of the reconstructed data is the same, the download time and the DPR time will not change, which means that the system works correctly in terms of time and the measurement results can be obtained

correctly.

5. Conclusions

Mobile device has been required higher performance and lower power consumption. To achieve them, it is effective to implement hardware instead of processing heavy software. In order to realize a high-performance, high-performance, and power-saving mobile device, we investigated the implementation of the online DPR using lwIP. As a result, it was found that the desired system can be realized.

References

- (1) Ministry of Internal Affairs and Communications, "WHITE PAPER Information and Communications in Japan(2019)", pp 10
- (2) L. Pezzarossa, M. Schoeberl, and J. Sparso, "A Controller for Dynamic Partial Reconfiguration in FPGA-Based Real-Time Systems", 2017 IEEE 20th International Symposium on Real-time Distributed Computing (ISORC), pp. 92-100, 2017.
- (3) A. Hassan, H. Mostafa, Hossam A. H. Fahmy, and Y. Ismail, "Exploiting the Dynamic Partial Reconfiguration on NoC-Based FPGA", 2017 New Generation of CAS (NGCAS), pp. 277-280, 2017.