Joint Uplink and Downlink Performance Profiling of LTE Protocol Processing on a Mobile Platform

David Szczesny, Ruhr-Universität Bochum, Germany

Sebastian Hessel, Ruhr-Universität Bochum, Germany

Anas Showk, Ruhr-Universität Bochum, Germany

Attila Bilgic, Ruhr-Universität Bochum, Germany

Uwe Hildebrand, Comneon GmbH, Germany

Valerio Frascolla, Comneon GmbH, Germany

ABSTRACT

This article provides a detailed profiling of the layer 2 (L2) protocol processing for 3G successor Long Term Evolution (LTE). For this purpose, the most processing intensive part of the LTE L2 data plane is executed on top of a virtual ARM based mobile phone platform. The authors measure the execution times as well as the maximum data rates at different system setups. The profiling is done for uplink (UL) and downlink (DL) directions separately as well as in a joint UL and DL scenario. As a result, the authors identify time critical algorithms in the protocol stack and check to what extent state-of-the-art hardware platforms with a single-core processor and traditional hardware acceleration concepts are still applicable for protocol processing in LTE and beyond LTE mobile devices.

Keywords: *ARM Processor, Hardware/Software Co-Design, LTE, Protocol Processing, Software Evaluation, Virtual System Prototyping, Wireless Technologies*

INTRODUCTION

Next generation mobile communication systems like 3GPP's (3rd Generation Partnership Project) Long Term Evolution (LTE) will provide higher data rates at simultaneously reduced latency to account for new features and services like video streaming or online gaming over wireless links (Dahlman et al., 2008). Consequently, the radio architecture will experience another increase of its processing complexity (Berkmann et al., 2008) that already

DOI: 10.4018/jertcs.2010100102

today amounts to several giga operations per second in latest feature-rich mobile handsets (Berkel, 2009).

The workload and thus the power consumption are dominated by the RF front-end and the digital baseband processing (Hausner & Drewes, 2008). For this reason many researchers are focusing on circuits and algorithms that are located in the LTE physical layer (Manolakis et al., 2008) where higher layer information is sometimes adopted for link level performance simulations (Martín-Sacristán et al., 2009). Nevertheless, more attention should be drawn to the data plane of higher protocol layers, since its processing demand will also scale with the increased data rates in LTE and even more in LTE-Advanced.

Traditionally, the protocol processing is realized in software and executed on embedded processors like the ARM family (Lueftner et al., 2007). Only cryptographic algorithms, which are characterized by a high computational effort, are offloaded to hardware in mobile platforms (Gehrmann & Stahl, 2006). It is expected that even more protocol stack functionality will be supported by hardware in future in order to identify suitable system architectures that offer the required performance while keeping the processor clock frequency and thus the energy budget at a reasonable level. This is of particular importance in mobile devices with a limited battery lifetime. Therefore, a thorough investigation of the processing demand is mandatory for an effective hardware/software partitioning of the LTE protocol stack.

Another related aspect in the design of upcoming mobile phone platforms deals with the question how such a hardware/software co-design can be done efficiently. Especially for a performance analysis on system level several commercial and academic design tools exist (Gajski et al., 2009). One approach comprises the use of a virtual system prototype (Cockx, 2000; Eckart & Schnieringer, 2006; Brandenburg et al., 2007). This is a software image of the hardware platform that allows for an accurate performance analysis of the software stack in an early design phase before the final silicon is available.

In this article we analyze the performance of LTE protocol processing on an ARM based mobile phone platform by applying a virtual system prototype (VSP) whose structure is shown in Figure 1. The protocol stack model in the software stack contains the most complex part of the LTE layer 2 (L2) data plane in uplink (UL) and downlink (DL) direction. Video applications are used to generate the data load in both directions during simulations, while the Dhrystone application runs in parallel to the video processing in the protocol stack for a more realistic scenario. Both applications run on top of a real-time operating system (RTOS). The virtual hardware platform, however, consists of the processor architecture and a base station/physical layer peripheral (eNodeB/L1). The latter emulates a communication between a base station and the analyzed mobile device. It therefore enables appropriate simulations in a fully controlled and closed environment where the protocol stack can be analyzed on transport block level at different transmission conditions associated with different computational demands on the terminal side.

The article is organized as follows: We first describe the virtual hardware platform and give details on the processor architecture and the eNodeB/L1 peripheral. Afterwards, an insight to the LTE L2 UL/DL protocol stack model and an introduction to the freeRTOS™ real-time operating system are provided. Then we briefly explain the profiling workflow, followed by the presentation of the simulation results. Finally, we give a conclusion in the last section.

VIRTUAL HARDWARE PLATFORM

A virtual hardware prototype is used in this work for agile investigations and execution of an LTE protocol stack model. Implementation, simulation and analysis of such a software model of a system on chip (SoC) are carried out using