Microprocessors and Microsystems 35 (2011) 217–229

Contents lists available at ScienceDirect

Microprocessors and Microsystems

journal homepage: www.elsevier.com/locate/micpro

System design of full HD MVC decoding on mesh-based multicore NoCs

Ning Ma *, Zhonghai Lu, Lirong Zheng

iPack Vinn Excellence Center, School of Information and Communication Technology, Royal Institute of Technology (KTH) Forum 120, 164 40 Stockholm-Kista, Sweden

ARTICLE INFO

Article history:
Available online 2 November 2010

Keywords:
Application-specific
Homogeneous NoC
Exploration framework
Full HD MVC decoding
Multicore architecture
Communication and computation

ABSTRACT

Future multimedia applications such as full HD (1920 × 1080) multiview video coding (MVC) present great challenges on computing architectures. Even if with the state-of-the-art ASIC technology which can process single view HD decoding, dealing with multiple views would require times of computation capacity in proportion to the number of views, which is difficult to achieve. In this paper, we explore the system-level design space for full HD MVC applications mapped onto mesh-based multicore Network-on-Chip (NoC) architectures. To this end, we establish a simulation framework capable of simulating the combination of communication networks with computing cores. We investigate two task assignment schemes: picture-level assignment and view-level assignment. With an eight-view MVC decoding, we explore the design options with respect to network size, single-core performance and link bandwidth under both task assignment schemes. Our studies show that, to achieve a certain decoding performance, the computation capability and communication capacity should be balanced in the system. Also, to realize the eight-view HD decoding, the system only requires twice or less than twice of the single-core processing capacity required by single view decoding, thanks to the parallel computation and communication enabled by the multicore NoC architectures. Our results exhibit feasibility and potential of efficiently implementing the full HD MVC decoding on multicore NoC architectures.

1. Introduction

In the past five decades, the scale and performance of integrated circuits have constantly increased following Moore's Law. However, the performance demand grows more rapidly since new performance-eager applications keep booming up. Video processing is one of those applications. Huge data volume is one of the most significant characteristics of video streams, especially for high resolution (HD) videos. To reduce the amount of data, various data compression techniques are developed to decrease the redundancy so as to lower the data volume for transmission and storage. As the video coding standard develops from MPEG2 to H.264, the compression ratio has been greatly improved with more complex algorithms. On the other hand, as a result of pursuing for vivid user experience such as stereo video coding, free-view video coding, 3-D video coding etc., data volume further increases. *Multi View Coding (MVC)* [1–4] is a key technology enabler for those video applications providing vivid user experience. The views from different angles are recorded by several cameras at the same time. Efficient yet complicated compression algorithms are used for the coding process in order to reduce the streaming throughput. Since the views are simultaneous from multiple cameras, both view redundancy and view dependency exist, complicating the view processing. Especially for full high-definition (FHD, 1920 × 1080) MVC processing, efficient solutions are critical.

Application-specific integrated circuits (ASICs) have been traditionally used for high performance video processing. However, lack of flexibility and poor scalability make it hard to adapt to the rapidly developed applications. As the scale of integrated circuits continuously grows, system-on-chip (SoC) design plays an important role on reducing the crucial time-to-market. Various intellectual property (IP) cores are integrated together on a single chip to enable rapid development. With the integration of more and more powerful IP cores, the communication between IP cores becomes the bottleneck in traditional bus-based interconnects. To address this problem, network-on-chip (NoC) approach emerges as a scalable and modular solution [5–8]. General network aspects, such as topology, routing algorithms, switching techniques, congestion control, reliability etc, have been extensively studied. Simulation frameworks are also developed as design space exploration tools [9,10]. However, current NoC research has mainly focused on communication [8]. Since a NoC in a wide sense is a SoC, communication and computation should be studied coherently [11]. The two subsystems should be balanced in order to achieve high performance of the whole system with efficiency. The state-of-the-art ASICs are capable of decoding FHD single view videos. However, if ASICs are used for FHD MVC decoding, multiple times of computation capacity is required. This will challenge the ASIC

* Corresponding author. Tel.: +46 735724280.
E-mail address: mning@kth.se (N. Ma).

0141-9331/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.micpro.2010.10.003

design. Under such circumstances, multicore NoC architectures provide an promising alternative.

In this paper, we employ mesh-based multicore NoC systems to implement the FHD MVC decoding application. The complex decoding algorithms are mapped to processing cores, and the huge data exchange is realized by the communication network. To explore the parallelism in MVC decoding for NoC implementation, we design and compare the *picture-level* and *view-level* task assignment schemes. The design space of the NoC systems is also explored with respect to the selection of network size (Γ), suitable link bandwidth (B_L), required performance of single cores (Φ_p) and the task assignment scheme (Λ). The problem can thus be described as *"Given a target application and its mapping to mesh-based multicore NoCs, find a suitable combination of* (Γ, B_L, Φ_p, Λ) *to satisfy performance requirements under design constraints."*. In order to explore the system design space, we also establish a simulation framework modeling the features of MVC decoding for computation and circuit-switched communication on 2D mesh homogeneous multicore NoCs. The relation between the system performance and (Γ, B_L, Φ_p, Λ) is studied based on the simulation framework, and finally design options are obtained.

The rest of the paper is organized as follows. Section 2 discusses related work about the implementation of multimedia processing and the exploration of MVC decoding. Section 3 gives a brief introduction about the features of MVC. In Section 4, the NoC architecture and task assignment schemes for MVC decoding are proposed and detailed. The simulation framework is described in Section 5, including core modeling and router modeling. The system design options are explored in Section 6. We summarize the main findings and limitations in Section 7. Finally, Section 8 concludes the paper.

2. Related work

Video processing demands tremendous computational capability especially for high-definition (HD) video processing. In [12–15], high performance ASICs were implemented for H.264 HD single view video decoding. In order to obtain enough decoding speed for HD video application, compression algorithms of the video coding standard were studied in detail to explore the parallelism for hardware acceleration. The parallelism exploration was mainly focused on macroblock level and even lower level considering the huge volume of data and the complex data dependence. In [16], a system-on-chip was implemented by integrating dual cores and several high performance hardware accelerators to support 720p multi-standard video decoding.

Multicore architecture is a trend to gain higher processing performance and flexibility. In [17], a 167-processor computational platform was implemented for high performance applications. HD video encoding, JPEG encoding and other DSP operations could be performed in that platform. Reference [18] presented a NoC architecture for H.264 decoder. Different decoding processes were mapped to several NoC nodes. Implemented in FPGA, the platform was capable of decoding SDTV (standard-definition television) binary stream at a clock frequency of 66 MHz. In [19], the multimedia application (MPEG-2 encoder) was mapped and implemented on NoC architectures. However, most of the mapping schemes for multimedia applications are based on pipelining the processing procedures. The scalability may be constrained by the number of stages that the processing procedure can be partitioned into. Besides, the sophisticated task partitioning and mapping algorithms can be the obstacles for dynamically adjusting the tasks mapped to each node. In our work, we adopt the scheme of parallel processing, which means that each node can finish the whole processing procedure independently, and the dependent data exchange is mapped to the communication network. The advantage is the fine and easy scalability.

For MVC, a cache-based architecture was proposed and implemented for motion and disparity estimation (ME/DE) in [2], which is the computation intensive part of video coding. Based on the memory organization strategy, the processing capability supported the integer ME for HDTV 720p four-view MVC. Reference [3] gave a framework for heuristic scheduling of MVC on multicore architecture. Frame level and group of pictures (GOP) level parallelism were utilized for task mapping to processing cores. Based on the assumption that the cost of communication was negligible, the exploring algorithm was designed using directed acyclic graph. However, system architectures for FHD MVC applications have rarely been studied yet, where large computation capacity is required and communication networks are also crucial for the whole system. In [11], the authors reviewed the major design methodologies for NoC architecture design. The algorithms of task mapping, scheduling, routing, and etc. were summarized for various network modeling methods. Besides, the optimization techniques of power and energy were also discussed.

3. Multiview video coding

Multiview video coding (MVC) has been standardized by Joint Video Team (JVT) as Annex H of H.264 [20]. MVC inherits the sophisticated compression techniques of H.264 at the macroblock level, such as variable block-size motion compensation, quarter-sample-accurate motion compensation, various spatial predication modes for intra-coding, etc. Besides, hierarchical B-picture structure is adopted for temporal prediction to obtain higher coding performance [21]. View scalability and temporal scalability are embedded in the MVC standard to adapt user preference, decoder capability and network bandwidth. Significantly, the dependence among views is exploited to reduce the corresponding data redundancy. Thus, pictures from different views may be used as references for current processing picture. Thanks to these new features, users can select the views they are interested in to display, and the suitable volume of data can be transmitted according to the supplied network bandwidth.

The hierarchical structure of MVC stream is decomposed in Fig. 1, where reference relations are also labeled with arrows. The MVC stream consists of many independent groups of pictures (GOP). In each GOP, pictures from different views are interlaced. Both view dependency and temporal dependency exist. Basic coding units named as macroblocks (MB), normally blocks of 16 × 16 pixels, constitute each picture. The picture can be I frame, which only uses intra-frame coding, P frame, which uses forward pictures for inter-frame coding, or B frame, which uses both forward and backward pictures for inter-frame coding.

The typical MVC prediction structure is shown in Fig. 2, where each picture is represented by a square. Eight views and eight pictures in each view (8 × 8) constitute one group of pictures (GOP).

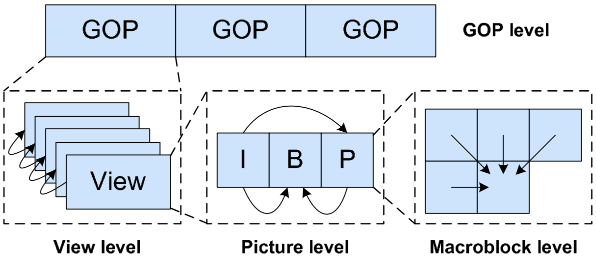

Fig. 1. The hierarchical structure of MVC.