An Efficient Methodology for Transaction-Level Design of Multi-core h.264 Video Decoder

Bingbing Xia, Fei Qiao, *Member, IEEE*, Huazhong Yang, *Senior Member, IEEE,* and Hui Wang
Tsinghua National Lab for Information and Technology, Tsinghua University, Beijing, China

***Abstract*—H.264 video decoder is a good choice for embedded instruments because of its higher compression ratio than MPEG2, as well as its higher requirements of run-time computational resource. Multi-core system is the future of the embedded processor design for its power efficiency and multi-thread parallelization, and can be used to fit well with the requirements for this decoder. To simulate and evaluate the performance of such application-specific multi-core systems effectively, a method based on the combination of TLM language (SystemC) and shared-memory parallel programming model (OpenMP) is given, and experiments show that it can effectively simulate the system in a short time and more importantly, it can be used to help analyze the efficiency of each task-parallelization strategy. After optimization, the speedup ratio for each slice decoding can get about 3.06 on average under 4-core multi-core systems.**

I. INTRODUCTION

Since the video processing algorithm is becoming more and more complex, the processing capability requirement is also on increase [1], it has reached at about 100 GOPS [2].For such processing capability requirement, embedded multi-core processor is one appropriate choice for its multi-thread parallelization and the flexibility, as well as the power efficiency, which matches well with the characteristics of video processing [3].

To design and simulate such multi-core video processing systems, the first thing to do is to model the total system as fast as possible to make the early choice for the task allocation onto each core. TLM (Transaction Level Modeling) is always used for this step [4], and is regarded as the golden tool for system-level SoC designs [5]. However, considering the complexity of the multi-core systems, the time cost of code-writing for the task-level parallelization is huge and the design will need very careful writing since one sentence is activated earlier or later will affect the tasks running on the other cores, so the debug work is trivial. Therefore, it will not fit well with the fast system-level modeling. From the software engineer's view, shared memory parallel programming model is used to model the software running on the multi-core systems and is proved to be successful [6]. However, such methods can't model the timing behavior of the system accurately. To cope with such problems, a new method based on the combination of these two methods is proposed to model the multi-core SoC for video processing.

The contributions of this paper are as follows:

1, A system-level design methodology based on the combination of SystemC and OpenMP is given for such multi-core video processing systems [10] [11].

2, A system-level design of h.264 video decoder is given with the proposed method.

II. PROPOSED DESIGN METHODOLOGY

The total system design flow is shown in Fig. 1. Taking h.264 video decoder as an example, the h.264 bitstream is put into the system, and simulated by the system model, through the basic task analysis and the iterative system evaluation for the time and power cost, the best task allocation strategy can be obtained at last.

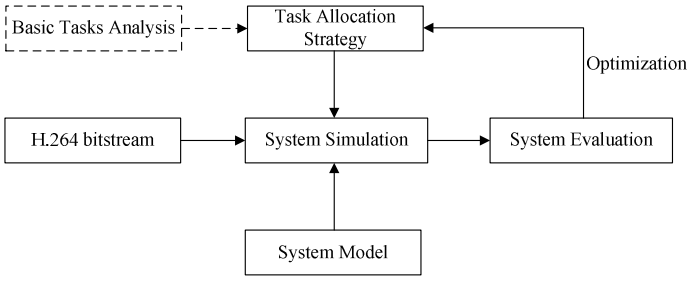

Fig. 1 Proposed Design Methodology

Particularly, the detailed system model is shown in Fig. 2. The main processor acts as the main controller, the shared memory is the common memory space used by these processors. For this multi-bus multi-core system, the number of the buses is the same as the number of the cores. To make this model more accurate, a cycle-accurate AHB bus model is referred to here(Based on the AMBA 2.0 Specification [12]), which is designed at the transactional level and used in my previous paper [13].For the modeling of these , both SystemC and OpenMP are used to realize the modeling of the parallelized tasks. All these together are integrated into the total system-modeling framework.

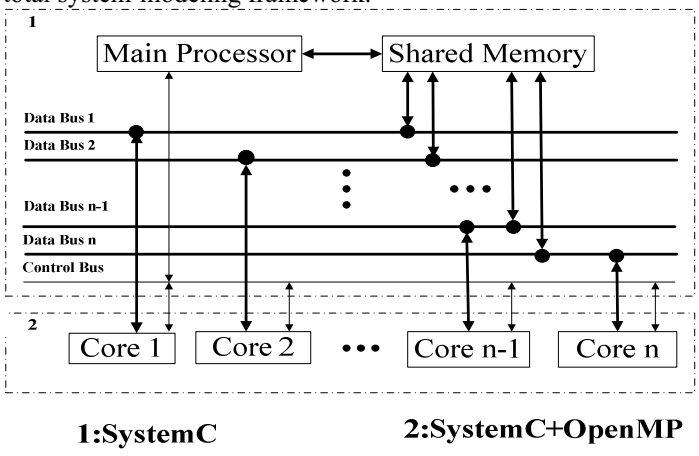

Fig. 2 System Architecture Based on the Proposed Methodology

Bingbing Xia is a PhD student in the NICS library, at the Electronic Engineering Department of Tsinghua University, Beijing, China. Email: xbb07@mails.tsinghua.edu.cn.

III. Simulation Results

Based on this proposed architecture, the total multi-core h.264 video decoder is simulated. Through the macro-block level parallelization, the best speedup ratio and power efficiency can be obtained by the optimization of various task-allocation strategies. The time is calculated by the counting of the running cycles and the power is calculated by the annotation of each function using the Wattch toolset [14]. And finally the simulation results for different video test sequences are shown in Fig. 3 and Fig. 4. Although the simulation results for different test sequences are not the same, the same conclusion can be drawn that 4 core will be the best choice.

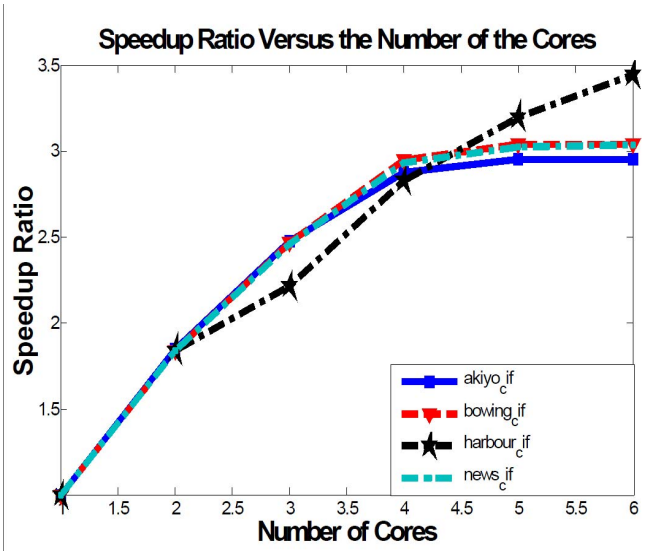

Fig. 3 Speedup Ratio Versus the Number of the Cores

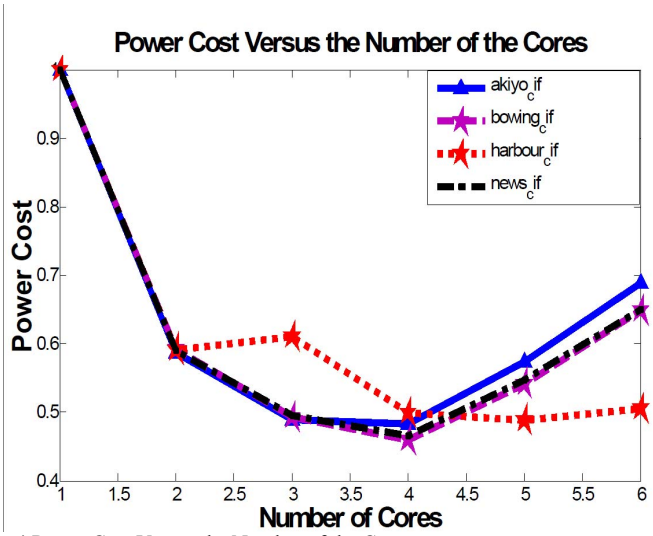

Fig. 4 Power Cost Versus the Number of the Cores

Simulation time cost of this methodology in comparison with other methods for the same h.264 video decoder is shown in table I. It shows that the simulation time for the RTL code using Modelsim is about 773s per frame, and with the Simplescalar toolset this cost is about 84s per frame [15], the proposed methodology has the least simulation time cost which means that it will be the best choice for the initial system-level simulation. Meanwhile, the code size is also the least, which can be used for fast prototyping of such complex systems.

TABLE I
Comparisons for Simulation Time Cost and Code Size

	Modelsim	Simplescalar	SystemC+Open MP (Proposed)
TimeCost	773s/frame	84 s/frame	30.26 s/frame
CodeSize	58838 Lines	52521 Lines	26496 Lines

IV. Conclusion

An effective transaction-level design methodology based on the combination of SystemC and OpenMP is given in this paper, and a TLM model of multi-core h.264 video decoder is implemented under this framework as a case study to show its efficiency. Since this will be much faster than other relative methods, it can be used for the system-level fast prototyping of such multi-core systems, especially for industrial product prototyping.

Acknowledgment

This work was supported in part by National Natural Science Foundation of China under Grant #60871005; in part by Ph.D. Programs Foundation (Young Scholar) of Ministry of Education of China under Grant #200800031073; in part by Research Platform of Graduate Course in Tsinghua University, “Structural Integrated Circuit Design (70230183)”.

References

[1]Gwo Giun Lee, Yen-Kuang Chen , Mattavelli, M. , Jang, E.S. "Algorithm/Archietcture Co-Exploration of Visual Computing on Emergent Platforms: Overview and Future Prospects", IEEE Transactions on Circuits And Systems For Video Technology Vol 19, No.11, 2009,pp: 1576 - 1587

[2]Kunio Uchiyama, "Power-Efficient Heterogeneous Parallelism forDigital Convergence", *IEEE Symposium on VLSI Circuits*,Honolulu,USA,18-20 June 2008

[3]Ling Guan, Sun-Yuan Kung, Jan Larsen. "Multimedia Image And Video Processing", CRC Press, USA, 2001

[4]Bart Vanthournout, TLM Working Group Chairman, ”An Insider’s View on the Making of the New TLM-2.0 Standard”

[5]Thorsten Grotker,etc.”System Design with SystemC”, Hingham, MA, USA: Kluwer Academic Publishers, 2002

[6] Bruce P.Lester, ”The Art of Parallel Programming”, published by: Pearson Education, Inc

[7] ITU-T Rec. H.264 / ISO/IEC 11496-10, “Advanced Video Coding”, Final Committee Draft, Document JVTF100, December 2002

[8]Jike Chong, Nadathur Satish, etc.” Efficient Parallelization of H.264 Decoding with Macro Block Level Scheduling”, ICME 2007, pp:1874-1877

[9] Nishihara,K.;Hatabu,A.; ”Parallelization of H.264 video decoder for embedded multicore processor”, ICME 2008, pp: 329-332

[10] http://www.systemc.org

[11]Mitsuhisa Sato,”OpenMP: Parallel programming API for shared memory multiprocessors and on-chip multiprocessors”, ISSS 2002

[12] ”AMBA™ Specification (Rev 2.0)”, By ARM Ltd

[13]Bingbing Xia, Fei Qiao,etc.”A fault-tolerant structure for reliable multi-core systems based on hardware-software co-design”, ISQED 2010, pp: 191-197

[14]David Brools, Vivek Tiwari, Margaret Martonosi. ”Wattch: A Framework for Architectural-Level Power Analysis and Optimization”.*Proceedings of the 27th International Symposium on Computer Architecture*, June 2000 , Vancouver, BC, pp:83-94

[15] www.simplescalar.com