ELectronics EXpress

IEICE Electronics Express, Vol.9, No.11, 978–984

A cost conscious performance model for media processors

Yaohua Wang[a)], **Shuming Chen, Kai Zhang, Jianghua Wan, Hu Chen, Sheng Liu, and Xi Ning**

Computer School, National University of Defense Technology

#109, Deya Road, Changsha, 410073, China

a) *nudtyh@gmail.com*

Abstract: The combination of multi-core, SIMD and VLIW schemes is becoming prevailing in today's media processor architectures. To achieve a deep insight into this trend, we propose a power conscious performance model based on the rationale of Hill and Marty's model. Several representative media application kernels are evaluated on the proposed model. The evaluation result shows that: for none communication applications, a large number of small cores achieve optimal performance; for communication applications, architectures with reduced core count and increased core size is preferred. Meanwhile, by increasing the SIMD width, better power efficiency can be achieved for both types of applications at a small loss of performance.

Keywords: SIMD, VLIW, Multi-core

Classification: Integrated circuits

References

[1] M. Woh, S. Seo, S. Mahlke, et al., "AnySP: anytime anywhere anyway signal processing," *33rd Int. Symp. Computer Architecture*, pp. 128–139, 2009.

[2] M. D. Hill and M. R. Marty, "Amdahl's Law in the Multicore Era," *Computer*, vol. 41, no. 7, pp. 33–38, 2008.

[3] J. L. Hennessy and D. A. Patterson, *Computer Architecture: A Quantitative Approach*, Fourth Edition, Morgan Kaufmann, 2007.

[4] T. Y. Morad, U. C. Weiser, et al., "Performance, Power Efficiency and Scalability of Asymmetric Cluster Chip Multiprocessors," *Computer Architecture Letters*, vol. 5, 2006.

[5] S. Rixner, W. J. Dally, et al., "Register Organization for Media Processing," *Int. Symp. High Performance Computer Architecture*, 2000.

[6] E. Nilsson and J. Oberg, "Reducing peak power and latency in 2-D mesh NoCs using globally pseudochronous locally synchronous clocking," *CODES+ISSS*, pp. 176–181, 2004.

DOI: 10.1587/elex.9.978
Received April 01, 2012
Accepted May 23, 2012
Published June 12, 2012

1 Introduction

Multi-core architectures integrate multiple processing units into one chip to overcome the physical constraints of unicore architectures, and greatly increase the throughput and efficiency for media processors. Beside this multi-core scheme, the SIMD (Single Instruction Multiple Data) and VLIW (Very Long Instruction Word) schemes are also adopted inside each single core, due to the abundant LLP (loop level parallelism) existing in media applications. The combination of multi-core, VLIW and SIMD schemes is becoming a prevailing architecture of media processors. Examples includes the stream processors and GPUs. Processors in the academy area like AnySP [1] also exhibits this architecture.

Although widely used, given the lack of conventional wisdom concerning architectural parameters like SIMD width, VLIW length, and core count in media processors, it's not surprising that there are as many different designs as there are chips. Moreover, with the further development in both media applications and the technology trend, a larger amount of parallelism needs to be efficiently exploited by an increasing amount of hardware resources, leading to an escalating in the design space of media processors. Hence, an easy-to-understand model that offers valuable insight into primary architectural parameters, affecting the performance and power of media processors, would be especially valuable.

In this paper, we apply the rationale of Hill and Marty's performance model [2] to develop a power-conscious performance model for media processors, which combine the multi-core, VLIW and SIMD schemes. A correlative analysis of performance and power is performed on several representative media application kernels. The evaluation result shows that applications without communication overhead prefer a large number of cores with small SIMD width and VLIW length, while for applications with communication overhead, architectures with reduced number of cores and increased SIMD width and VLIW length in each single core can achieve optimal performance. Meanwhile, further increasing the SIMD width can achieve better power efficiency for both types of applications at a small loss of performance.

2 The performance model

The abstract architecture model, capturing the major features of current media processors, is shown in Fig. 1. In this architecture, hardware resources are divided into multi-cores (c). The communication among cores is done through the Router (R) of the NoC (Network on Chip) with a mesh topology. Each core has multiple SIMD lanes working under an unified instruction flow. The communications among different lanes are conducted through the DSH (data shuffle) unit [1]. A multi-banked memory (D-Mem) supplies data for SIMD lanes. Computation resources in each SIMD lane are organized in a VLIW manner.

To achieve a clear performance model, the computation resource is abstracted as ABCU (abstract basic computation unit), which is capable for

DOI: 10.1587/elex.9.978
Received April 01, 2012
Accepted May 23, 2012
Published June 12, 2012

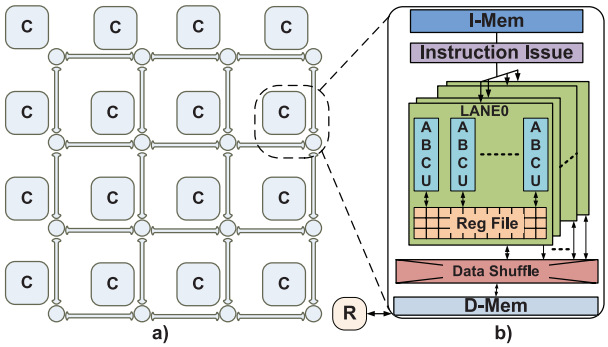

Fig. 1. The abstract model of typical media processors

conducting common computational operations, like ALU, Shift, MAC and Load/Store. Based on this abstraction, we model our target architecture with three parameters: the SIMD width (number of lanes) N_s, the VLIW length (number of ABCUs in a SIMD lane) N_v and the core count N_m.

We derive our performance model from Hill and Marty's, in which the speedup of a multi-core architecture depends on the parallel fraction of applications (p), single core performance, and the number of cores. For our target architecture, the single core shows different performance for scalar and parallel parts of applications: for the scalar part, little speedup can be achieved; while for the parallel part, both the VLIW and SIMD schemes can well exploit the abundant LLP in media application kernels, showing a large performance speedup noted as $Perf\,(N_v, N_s)$. Besides, the communication overhead among cores (C_{core}) is also considered. The overall performance model is shown in equation (1).

$$Overall = \frac{1}{(1-p) + \frac{p}{N_m \cdot Perf(N_v, N_s)} + C_{core}\,(N_m)} \tag{1}$$

Both the SIMD and VLIW schemes gain performance speedup mainly by well exploiting the LLP in media application kernels. The only difference is that the SIMD scheme can introduce additional communication overhead (C_{lane}) among SIMD lanes. Thus, $Perf\,(N_v, N_s)$ can be modeled in equation (2), in which α is the amount of LLP in media application kernels.

$$Perf\,(N_v, N_s) = \frac{1}{(1-\alpha) + \frac{\alpha}{N_v \cdot N_s} + C_{lane}\,(N_s)} \tag{2}$$

We model both C_{core} and C_{lane} with the scaling factor k multiplied with the product of communication amount and the cost of each communication (shown in equation (3) and (4)). The scaling factor k is used to normalize the communication overhead, which is set to be the inversion of the media application kernel's execution time. The communication amount depends on the computation pattern of media application kernels. Equation (5) gives the cost of each inter-core communication. NoC_{delay} is the Router delay of the NoC, and $Hop\,(N_m)$ represents the average hop counts. As indicated in [3], $Hop\,(N_m)$ can be modeled by equation (6). We choose the crossbar for the DSH unit due to its efficiency and popularity [1]. Generally, the crossbar can

DOI: 10.1587/elex.9.978
Received April 01, 2012
Accepted May 23, 2012
Published June 12, 2012

fulfill each inter-lane communication in one cycle.

$$C_{core}\left(N_m\right) = k \cdot Q_{core} \cdot T_{core} \tag{3}$$

$$C_{lane}\left(N_s\right) = k \cdot Q_{lane} \cdot T_{lane} \tag{4}$$

$$T_{core} = Hop\left(N_m\right) \cdot NoC_{Delay} \tag{5}$$

$$Hop\left(N_m\right) = 2 \cdot \left(\frac{N_m}{3} - \frac{1}{3 \cdot N_m}\right) \tag{6}$$

3 The power model

Estimating the power of the processor from a group of high-level parameters is nontrivial. The power can be various by employing different VLSI technology, optimization techniques and design methodology. We simplify this problem by assuming that total architecture power is proportional to the overall area (shown in equation (7)), as proposed in [4].

$$Power = \gamma \cdot Total_{area} \tag{7}$$

To achieve a reasonable area for different architecture configurations, the main components of the target architecture (shown in Fig. 1) are implemented in Verilog. Synopsys Design Compiler is used to synthesize these components in TSMC 65 nm technology at 700 MHz. The area result is shown in Table I. When we vary parameters N_s, N_v and N_m, these components show different scaling factors. Thus, the total area cost can be modeled as the summation of these main components' basic area cost multiplied by their corresponding scaling factors.

Table I. The hardware implementation result

Components	**Area** (mm^2)	Description
I-Mem (IM)	0.153264	16 KB instruction memory
Instruction Issue (IS)	0.033832	derived from 4 instruction slots
ABCU	0.471037	Supporting MAC, ALU, Load/Store, Logic
Register File (RF)	0.088872	16-Entry 32 bit wide
Data Shuffle (DSH)	0.022092	derived from 4 in 4 out crossbar
D-Mem (DM)	0.100573	16 KB data memory
Router (R)	0.038227	Router of Nostrum NoC [6]

We assume a 16 KB IM, which is enough for most of application kernels. The IS unit dispatches N_v instructions to N_v slots, so that the area cost increases in a square manner with N_v. As shown in Fig. 1, the total number of ABCUs are increased with the product of N_v and N_s. For RF, as indicated in research [5], the area is increased in a linear manner with register size (proportional to $N_v \cdot N_s$) and in a square manner with the number of ports (proportional to N_v). The area of the DSH unit is increased in a N_s^2 manner. DM shows the same scaling property as ABCUs, and R has a per-core characteristic. Based on the above components, the area of the entire multi-core architecture can be modeled by multiplying the summation of the

DOI: 10.1587/elex.9.978
Received April 01, 2012
Accepted May 23, 2012
Published June 12, 2012

scaling components with N_m (shown in equation (8)).

$$Total_{area} = (IM + IS \cdot N_v^2 + ABCU \cdot N_v \cdot N_s + RF \cdot N_v^3 \cdot N_s + DSH \cdot N_s^2 + DM \cdot N_v \cdot N_s + R) \cdot N_m \quad (8)$$

4 The power conscious performance evaluation

Several representative media application kernels are selected and evaluated on the proposed model. These kernels are compiled on a simple scalar compiler. α is obtained by calculating the proportionality of loop body's execution time. The corresponding values of α for application kernels including the 8*8 Discrete Cosine Transformation (DCT), Quantization (Quant), 4T4R MIMO Decode and Motion Estimation (ME) are 0.98, 0.96, 0.95 and 0.99 respectively. For ME, as communication is needed, k is also calculated, and the value is 8.54e-7. The Q_{core} (Q_{lane}) of ME exhibits a $log_2^{N_m}$ ($log_2^{N_s}$) characteristic, due to its "reduction to scalar" operations. To deeply reveal the influence of the communication overhead, we have also added two synthetic kernels, whose communication amount is 10x and 100x of ME.

For a better visibility, we reduce the architecture parameters to two by assuming a fixed number of ABCUs N. Thus, we only need to vary the N_s and N_v, with N_m equals to $N/(N_s \cdot N_v)$. The default parameters in our performance model are set as follows: $p = 0.99$, $N = 2048$, $NoC_{delay} = 2$, $T_{lane} = 1$, $\gamma = 1$.

4.1 Evaluation for *un_comm* kernels

We begin our analysis with *un_comm* application kernels (DCT, Quant, MIMO Dec), which have no communication overhead. The performance trend of different architectures for these kernels are the same. As shown in Fig. 2 (a), optimal performance can be achieved by architectures with a large number of small cores having small VLIW length and SIMD width. The

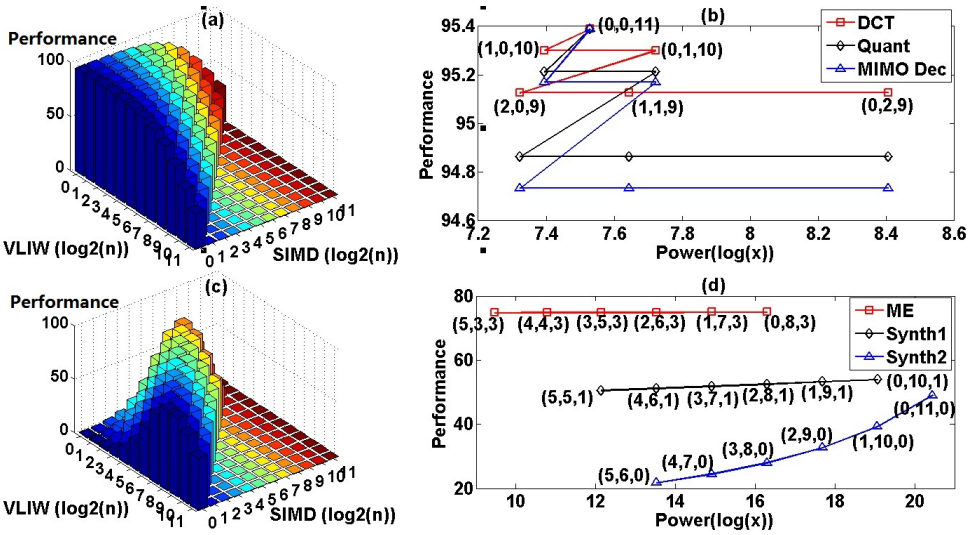

Fig. 2. The evaluation result for application kernels

DOI: 10.1587/elex.9.978
Received April 01, 2012
Accepted May 23, 2012
Published June 12, 2012

top-six high performance architecture configurations (noted as a tuple ($log_2^{N_s}$, $log_2^{N_v}$, $log_2^{N_m}$)) and their power costs are illustrated in Fig. 2 (b). Applications with larger LLP achieve an higher overall performance for each configuration. Better power efficiency can be achieved by increasing the SIMD width while reducing the VLIW length or core count.

4.2 Evaluation for *comm* kernels

Fig. 2 (c) shows the performance trend of the *comm* kernel ME. Compared with *un_comm* kernels, the communication overhead shifts the high performance configurations from a large amount of small cores to a moderate number of middle sized cores. To further reveal the effect of the communication overhead, Fig. 2 (d) gives the high performance configurations (noted as a tuple ($log_2^{N_s}$, $log_2^{N_v}$, $log_2^{N_m}$)) for ME and the two synthetic kernels. As we can see, kernels with a larger amount of communication prefer a smaller N_m. Besides, increasing N_s can achieve better power efficiency at a small loss of performance.

4.3 Putting it all together

To thoroughly consider both the *comm* and *un_comm* applications, we rebuild the overall performance model with equation (9), in which δ stands for the proportion of *un_comm* application kernels. We choose DCT for the *un_comm* part, and ME for the *comm* part. Fig. 3 (a) shows the performance trend when δ is 0.5. As we can see, the optimal configurations are those with moderate number of middle sized cores. To reveal the effect of the proportion of both application types, we vary δ (delta) from 0.3 to 0.9. the top-six high performance configurations are listed in Fig. 3 (b). As it shown that with the decrease of δ, N_m is becoming smaller. It can be concluded that a smaller number of larger sized cores is suitable for a higher proportion of *comm* application kernels. Moreover, the power efficiency of larger N_s is still maintained.

$$Overall^* = \delta \cdot un\_comm + (1 - \delta) \cdot comm \tag{9}$$

To eliminate the side effect of the default values used for parameters. We have also varied the value of p, N, NoC_{delay}, T_{lane} and γ. The variation of

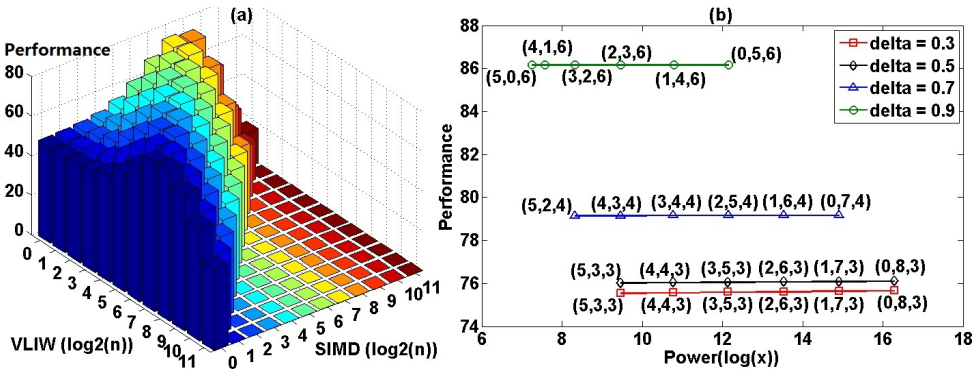

Fig. 3. The overall evaluation

DOI: 10.1587/elex.9.978
Received April 01, 2012
Accepted May 23, 2012
Published June 12, 2012

these parameters does affect the absolute performance of different architecture configurations, but it does not change the conclusions of this paper.

5 Conclusion

This paper proposes a power conscious performance model for modern media processors. It reveals that none communication applications prefers a large number of small cores, while the communication overhead in applications can reduce the optimal number of cores to a moderate extent. When cooperatively considering both types of applications, a moderate number of middle sized cores, with wider SIMD inside, can achieve better performance and power efficiency.

Acknowledgments

We thank Mark Hill of the University of Wisconsin for his feedback and encouragement on an early version of this article. This work is sponsored by NSF of China (61070036, 61330007 and 60906014).

DOI: 10.1587/elex.9.978
Received April 01, 2012
Accepted May 23, 2012
Published June 12, 2012