Seeking Optimal Granularity: Are We There Yet?

Koji Suginuma
Eizo Shimbun, Tokyo, Japan / Keio University, Tokyo, Japan

Abstract—**For media processing, many kinds of computer architecture have been proposed. Many of them focused on the granularity of calculation. The granularity spreads from sub-instruction level to inter processor level. Currently GPU is a commonly used architecture. This utilizes the idea of granularity very well. However, the efficiency of processing seems to be improvable. To reach higher performance, new shape of hardware should be investigated. Dynamic reconfiguration to adjust the granularity on the fly can be a good candidate. The optimal granularity for media processing must be considered for higher than HDTV resolution age.**

I. INTRODUCTION

Media processing has been considered to be fascinating topics for computer architecture. In early days one of the most prominent media processing was graphics calculation. One of pioneering work was the Geometry Processor by Clark [1]. Since then graphics processors utilize many kinds of parallelisms to gain performance.

Video processing was pioneered by RCA as DVI (Digital Video Interactive). Because this method was asymmetry, encoding was not realtime. A realtime decoder was Intel's i750 [2]; however, encoder was never disclosed. In 1993 MPEG-1 was released as a symmetric video coding scheme. MPEG-2 (1995) and MPEG-4 Part 10 (2003) are widely used for television braodcast. MPEG-1, MPEG-2 and MPEG-4 are capable to use inter-frame prediction for higher efficiency.

On the other hand, digital cinema community chose an intra-frame coding scheme, JPEG2000. Although digital cinema supports up to 4096 by 2160 pixels, MPEGs in 2004 could handle only up to 1920 by 1080 resolution. This might be one of reasons for JPEG2000 to be chosen.

Now, we are heading to Ultra High Definition or High Frame Rate. The Ultra High Definition Television, UHDTV, claims maximum 7680 x 4320 at 60P [3] temporal and spatial resolution. The High Frame Rate system, HFR, targets to 240 frames for 1920 x 1080 resolution [4].

For professional acquisition, image compression is not a mandatory; however, it is necessary to deliver the contents to end users. Because transmission methods are under pressure to save bandwidth, high performance CODEC scheme is called for future broadcasting. One thing we need to care is that the number of pixels are up to 16 times at UHDTV. The latency must be reduced to quarter for HFR. It is unrealistic to comply with these demand by raising clock frequency 4 to 16 times. These demands call for new solution other than simple clock upscaling.

II. GRANULARITY

To gain the performance, introducing parallel processing is a well known method. There are many levels of parallelism. One aspect to express the degree of parallelism is the granularity. Sub-instruction level (SIMD), instruction level, thread level, and process level are most used 'units' to express parallelism.

Levels shown above are used for software point of view. These levels are tightly coupled with hardware but not fixed to map onto specific architecture.

III. PARALLEL ENTITY

Granularity is defined above from software point of view. In parallel, there must be hardware to accommodate these grains. This section looks about hardware side.

A. SIMD

Since MMX [5], SIMD technology has been widely applied to media processing. Although SIND was born long before MMX, its usage was limited in special purpose processors, e.g. PIXAR Image Computer [6].

SIMD works best for vector calculation. This means that the scheme cannot work for coarser grain parallel calculation. PIXAR Image Computer was heavily used for image processing because the calculation relies on vector processing.

B. Core

Recently, a new criterion, core, is introduced. Multi core architecture is different from traditional shared memory multi processor architecture because it shares main memory instead of small size memory that is used for inter-processor communication.

Multi-core or many-core architecture is common now. Even for a lap top computer, the mainstream machines are with multi-core processors. For high performance machines, four to six cores processors are used. Intel defines that more than ten cores should be called as many-core. It will not take long time to reach CPUs to be multi-core.

In 1999 GPU was introduced by NVIDIA; however, the industry needs to wait for Cg programming language for GPU operation. Cuda and OpenCL followed Cg to fill the gap in application. These programming languages opened door for general purpose processing on GPU. There are many GPU based solutions available for media processing.

GPU is a kind of multi-core; however, the number of core is much more than CPU. Recent GPUs equip with 300 to 400 cores. Although, the performance of GPU core is much less than CPU core, the superiority of core count works well on some applications.

C. Processor

Tightly and loosely coupled multi processors have long history in computation. For media processing, tightly coupled processors have been used where frames have connection each other. Loosely coupled processors are commonly used where frames are independent each other.

Because inter-process communication takes longer time, light-processes, threads, are favorable for shorter latency. Although software may hide the difference on inter-processors communication scheme, a programmer must understand its characteristics. This makes it difficult to describe portable programs between machines.

IV. Efficiency

Because a media-processing algorithm demands many degree of parallelism, single architecture cannot accommodate all procedures in the algorithm. For example, DCT works well on SIMD; however, DCT is not good on loosely coupled multi processors. Inter frame comparison suits for a shared memory multi core system, but it may not work well with other architecture because of communication limitation. This kind of granularity mismatch hinders efficient execution.

Here is one example. A transcode software can achieve 40 fps on a 12-core machine. The software achieves 60 fps with GPU. The 12-core CPU runs at 3 GHz and the GPU runs at 600 MHz. The GPU contains 192 cores.

If simple scaling can be applied, the GPU should reach more than 100 fps; however, the reality does not allow the GPU to reach 3x speed. Many granularity mismatches exist when the algorithm is mapped to GPU architecture. The same thing can be said to multi-core architecture.

V. Resource allocation

It is hard to fit a media-processing algorithm into single architecture. If the algorithm is divided into many pieces and each piece is accommodated into the best suitable parallel scheme, the efficiency will be improved. This ideal implementation cannot be done on a CPU because of its fixed structure. New hardware should be developed.

In the real world, memory traffic is one of the hardest obstacles to overcome. The memory bus will be a bottleneck for fluent data transition. However, there is a hope. The embedded DRAM becomes less difficult than a decade ago. Distributed RAMs in a processor is reasonable solution. Also, careful cache construction can reduce memory traffic dramatically. Combining with these techniques, we may accommodate multiple degree of parallelism in one processor.

VI. Reconfigurability

To change the degree of parallelism for each stage of calculation, a reconfigurable architecture becomes a candidate. Typical traditional reconfigurable architecture uses many computing elements and just change the connection between them. Sea of ALU was one of the methods in earlier reconfigurable processor. This method is good for simple calculations but does not work for complex processing.

For future media processor, more architectural freedom must be incorporated. Not only lower level elements, but also higher level elements and data paths must be configured easily. Keeping the best shape for an operation reduces mismatch between algorithm and architecture. Dynamic reconfigurable method is a good candidate.

Dynamic reconfigurable technologies are still under development. No commercial success has been reported. Even though no commercial success, this technology is one of most promising way to overcome hefty cost of LSI development. If we have a generic dynamic reconfigurable processor, we can off load hardware cost. Only we need to do is to concentrate on software development that consists of HDL and program.

VII. Conclusion

The advent of new video standard demands CODEC hardware to be higher performance. Conventional architecture may not be able to accommodate new CODEC methods in efficient way. The 8k x 4k format supplies 16 times more pixels than 2k x 1k HDTV. The 240 fps format demands CODEC to reduce latency to one fourth. For mass production, it is unrealistic to raise clock frequency 4x or 16x. Increase parallelism must be a good solution for new formats.

To fit into conventional architecture, the granularity must be in fixed level for entire process. To achieve higher performance, hardware should match to the natural granularity of a computation. To cope with the dynamic change of granularity requirement, hardware needs to metamorphose during computation. Dynamic reconfigurable technology will bring the solution.

Acknowledgment

The author would like to thank to my ex-colleagues with Sony Corporation.

Reference

[1] J. Clark, "A VLSI Geometry Processor for Graphics," *Computer.*, vol. 13, no.7, pp. 59-68, Jul. 1980.

[2] S. Vinekar, "DVI TM TECHNOLOGY & i750 VIDEO PROCESSORS," *Hot Chips2,* 1990

[3] SMPTE, *SMPTE ST 2036-1:2009 Ultra High Definition Television — Image Parameter Values for Program Production*, New York, 2009

Y. Kuroki, "Development of the High Frame Rate 3D System," *1st Brazil-Japan Symposium on Digital Television Advances*, submitted for publication.

[4] M. Kagan, "P55C Micro-Architecture – The First Implementation of the MMXTM Technology," *Hot Chips 8*, 1996.

[5] A. Levinthal and T. Potter, "Chap – A SIMD Graphics Processor," *SIGGRAPH Computer Graphics,* vol. 18, issue 3, pp. 77-82, July 1984