A 26mW 6.4GFLOPS Multi-Core Stream Processor for Mobile Multimedia Applications

You-Ming Tsao[1], Chih-Hao Sun[1], Yu-Cheng Lin[1], Ka-Hang Lok[1], Chia-Jung Hsu[2], Shao-Yi Chien[1], and Liang-Gee Chen[1]

[1]Graduate Institute of Electronics Engineering and Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan; [2]UMC, Hsinchu, Taiwan

Abstract

A 26mW 6.4GFLOPS multi-core stream processor for mobile applications is implemented in 90nm CMOS technology. A unified stream processing architecture with power-aware frequency scaling and adaptive task scheduling techniques are proposed to reduce the power consumption and increase the performance to achieve the performance of 200Mvertices/s and 400Mpixels/s in 3D graphic applications.

Introduction

Due to the rapid growth of mobile phones and improvement of display and semiconductor technology, more and more multimedia features are embedded in mobile devices, such as image/video processing and 3D graphics. The stream processing architecture, where the data accessing operations are separated from the kernel function execution, was reported in [1] for both image and video signal processing. The 3D graphics applications, where mesh-based 3D models are transformed and scanned to convert into pixels, can also be treated as one kind of modified stream processing. However, the high performance oriented architecture of [1] is not fully suitable for the power constrained platforms like mobile phones.

In this work, different from the other prior arts [2][3] focusing only on graphics applications, we develop a multi-core stream processor with two unified stream processing kernel cores and one general purpose RISC core for both graphics and video processing applications. Three key techniques, unified stream kernel (USK), adaptive task scheduling (ATS) and five-clock-domain power-aware frequency scaling (PAFS) are proposed for this chip to achieve high performance, high hardware efficiency and low power consumption in the system level. Down to each of the stream kernel, the techniques proposed in [4], such as configurable memory array (CMA), adaptive multi-thread (AMT), fixed-point/floating-point reconfigurable processing elements and video accelerating instruction set are also adopted in the kernel micro-architecture.

Processor Architecture

Fig. 1 shows the system block diagram of the proposed multi-core stream processor. The system and application tasks are executed on the five-pipeline-stage 32-bit RISC CPU. The stream fetching unit (SFU) receives the stream load/store commands and the accessing pattern configurations from the application task to fetch the stream data, such as vertices/pixels for graphics or macroblocks (MB) for video via the memory controller (MC) from the external memory. A modified configurable memory array (CMA), which can be configured as cache memory or buffer, is served as a low-power and high-bandwidth inter-switch. The stream processing unit (SPU)

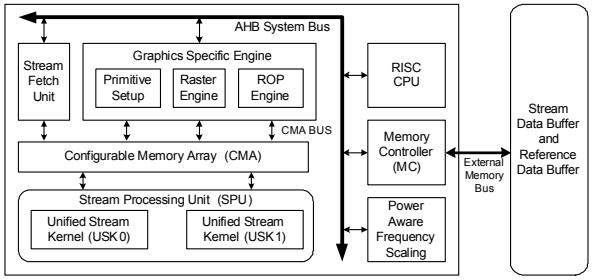

Fig. 1. Architecture of the multi-core stream processor.

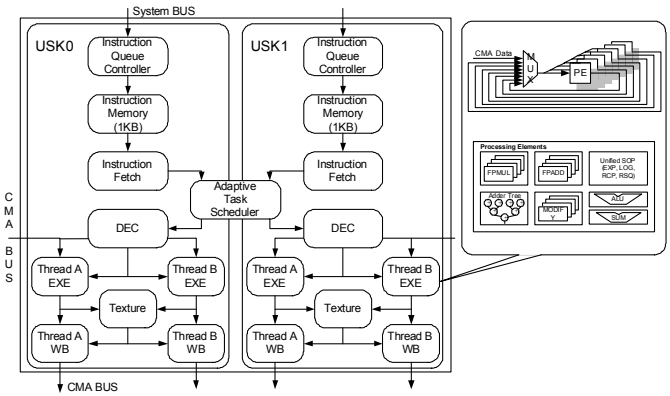

Fig. 2. Architecture of the dual unified stream processing kernels.

contains dual unified stream kernels (USK) to provide the maximum parallel processing capability in instruction level and thread level. For 3D graphic applications, the dedicated graphic tasks, such as primitive-setup, rasterizaton and final color/depth operations, are off-loaded to the graphic specific engines (GSE). There are five clock domains in the system and each clock domain is controlled by the power-aware frequency scaling unit (PAFS) to find the balance point of power and performance by adaptively scaling the five clock signals.

Unified Stream Kernel

The unified stream kernel (USK) architecture inherits the micro-architecture from [4]. When working at 200MHz, with VLIW instruction-level parallelism (ILP), it achieves 16GOPS with 8-bit addition and 32-bit multiplication and 6.4GFLOPS in single-precision. Fig. 2 shows the USK architecture. The dual-core unified stream architecture adopts an adaptive task scheduling (ATS) scheme to adaptively adjust the processing resource to increase the utilization and performance. The ATS receives two instruction streams and dispatches to the two decoding (DEC) units by inspecting the USK and CMA status

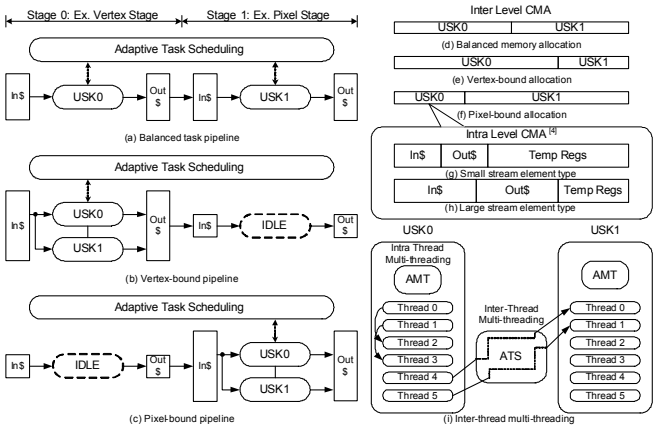

Fig. 3. Illustration of adaptive task scheduling techniques.

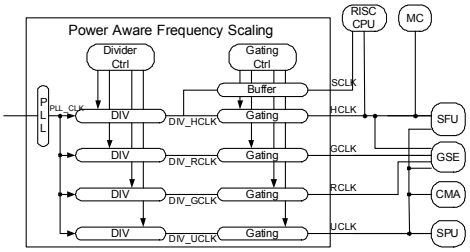

Fig. 4. Architecture of power-aware frequency scaling.

and achieves the peak processing speed of 200Mvertices/s and 400Mpixels with only 26mW in power consumption.

Adaptive Task Scheduling

An application usually can be partitioned into a series of cascaded sub-tasks. From a multi-thread processing point of view, concurrently executing these kinds of heterogeneous sub-tasks is called as inter-thread parallelism. The ATS improves the hardware utilization and performance with the inter-thread multi-threading. If we take the heterogeneous sub-tasks of vertex processing and pixel processing in a graphics application as an example, Fig. 3 shows the block diagram of the three ATS conditions. For a balanced task pipeline in Fig. 3(a), the workloads in both of the USKs are balanced. For a non-balanced task pipeline like vertex-bound/pixel-bound conditions in Fig. 3(b) and 3(c), ATS relocates the USK to the bottleneck task. Furthermore, the in/out cache memories can be reconfigured into different partitions to resolve the unbalanced pipeline latency using CMA techniques, as shown in Fig. 3(d)—(f).

Power-Aware Frequency Scaling

In order to further reduce the power consumption, power-aware frequency scaling (PAFS) with finer clock domain control is designed as shown in Fig. 4. The clock dividers (DIV) provide the frequency scalability for each clock domain. The four clock gating cells gate the unused clocks when the relative modules are idled. The only clock without gating cell, SCLK, is passed to the RISC CPU wakeup logic and is also the clock source of PAFS. The PAFS monitors the system power budget and all the function units to provide two-level power saving. The first level is the frequency scaling level. When the

Process Technology		UMC 90nm CMOS 1P9M LowK
Supply Voltage		1.0V core, 2.5V I/O
Clock Frequency		50-200MHz, 5 CLK Domains
Power Consumption		26mW (Stream Processing Unit 16 mW)
SRAM	RISC CPU	I$: 8KB D$: 8KB
	SPU	IM: 2KB
	CMA	10KB
Performance	Arithmetic	16 GOPS 6.4 GFLOPS
	Graphics Throughput	200M vertics/s, 400M pixels/s

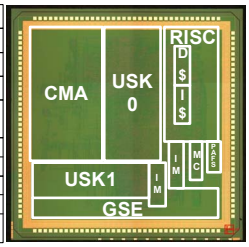

Fig. 5. Chip specification and micrograph.

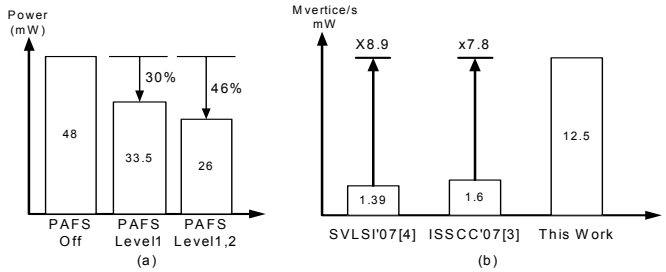

Fig. 6. Power consumption comparison.

PAFS finds the system is in low power state, the frequency of the low-priority unit can be scaled down. The second level is dynamic gating level. When the idled function unit signals the PAFS as idle state, the relative clock tree will be gated.

Implementation

This work is implemented using UMC 90nm 1P9M process. The measured chip specifications and micrograph are shown in Fig. 5. Fig. 6(a) shows the results of power reduction, where 46% of power consumption can be saved with both levels of PAFS. Performance index of Mvertices/s per mW is used to evaluation the power efficiency, and it shows that 7.8 times improvement can be achieved when compared with the state-of-the-art vertex processor [3] in Fig. 6(b).

Acknowledgments

The authors thank UMC University Program team for process supporting, Chip Implementation Center (CIC) for EDA tool supporting, and the valuable discussions with members of DSP/IC Design Lab and Media IC and System Lab are also appreciated.

References

[1] B. Khailany, et al., "A Programmable 512 GOPS Stream Processor for Signal, Image, and Video Processing," ISSCC Dig. Tech. Papers, pp.272-273, Feb. 2007.

[2] J.-H Woo, et al., "A 152mW/195mW Multimedia Processor with MPEG/H.264/JPEG and Fully Programmable 3D Graphics for Mobile Applications," Symposium on VLSI Circuits Dig. Tech. Papers, pp.220-221, Feb. 2007.

[3] B.-G Nam, et al., "A 52.4mW 3D Graphics Processor with 141Mvertices/s Vertex Shader and 3 Power Domains of Dynamic Voltage and Frequency Scaling," ISSCC Dig. Tech. Papers, pp.278-279, Feb. 2007.

[4] Y.-M Tsao, et al., "Low power programmable shader with efficient graphics and video acceleration capabilities for mobile multimedia applications," Symposium on VLSI Circuits Dig. Tech. Papers, pp. 61-72, Jan. 2007.