Contrail Processors for Converting High-Performance into Energy-Efficiency

Toshinori Sato[1,2] Itsujiro Arita[1]
[1] Department of Artificial Intelligence
[2] Center for Microelectronic Systems
Kyushu Institute of Technology
{tsato,arita}@ai.kyutech.ac.jp

1 Introduction

Current trend of increasing popularity of portable and mobile computer platforms such as notebook PCs and smart cell phones is a driving force to investigate high-performance and energy-efficient microprocessors. For example, Java-2 MicroEdition (J2ME) works on cell phones. We can download a game and play it on our cell phone. Travellers guide and flight ticket reservation are available. Furthermore, mobile banking and trading are also provided. As computing power of mobile device increases, its energy efficiency becomes a first class design constraint. It is important to note that energy consumption is more important for mobile devices than power consumption because it decides the lifetime of batteries.

The energy consumed in a microprocessor is the product of its active power and execution time. Thus, to reduce energy consumption, we should decrease both of or either of them. The active power P_{active} and gate delay t_{pd} of a CMOS circuit are given by

$$P_{active} = fC_{load}V_{dd}^2 \quad (1)$$

$$t_{pd} \propto \frac{V_{dd}}{(V_{dd} - V_{th})^{\alpha}} \quad (2)$$

where f is clock frequency, C_{load} is load capacitance, V_{dd} is supply voltage, and V_{th} is threshold voltage of the device. α is a factor depending upon the carrier velocity saturation and is about 1.3–1.5 in advanced MOSFETs [2]. Based on Eq.(1), it is easily found that power supply reduction is the most effective way to lower power consumption. However, Eq.(2) tells us that supply voltage reduction increases gate delay, resulting in slower clock frequency. And thus, computing performance of microprocessor is diminished.

In order to mitigate the performance loss, we can exploit parallelism [1]. Two identical circuits are used in order to make each unit to work at half the original frequency while the original throughput is maintained. Since the speed requirement for the circuit becomes half, the supply voltage can be decreased. In this case, the amount of parallelism can be increased to further reduce the total power consumption. In this paper, we propose to utilize another kind of parallelism, which is thread level parallelism, for energy reduction with maintaining processor performance.

2 Contrail Processor Architecture

To reduce the energy consumption, we divide an execution of an application into two streams [8]. One is called *speculation stream* and consists of the main part of the execution. However, it exploits trace level value prediction [4, 7], and thus several regions of the execution are skipped. In other words, the number of instructions in the speculation stream is smaller than that in the original execution, resulting in energy reduction. In contrast, the other stream is called *verification stream* and supports the speculation stream by verifying each data prediction. The key idea is that the verification stream can execute slowly if the data prediction accuracy is considerably high. We can reduce the clock frequency of the datapath for the verification stream. Furthermore, the supply voltage is also reduced. From these consideration, its energy consumption is significantly reduced.

Each stream executes as a thread on a simultaneous multi-threading processor [3, 10], whose execution core consists of dual speed pipelines. The speculation stream is dispatched into a high-speed pipeline (speculation pipeline) and the verification stream is dispatched into a low-speed and low-supply-voltage pipeline (verification pipeline). In the ideal case, that means there are no mispredictions, the speculation stream finishes silently and waits for the verification process. In the case where a misprediction occurs, the execution of the speculation stream is squashed at the point where the misprediction is detected and processor state is recovered by the verification stream.

We call this *Contrail Processor Architecture*. What is a contrail ? A contrail is the condensation trail that is left behind by a passing jet plane. The speculation stream runs ahead just like a jet plain, and the verification stream is left behind by the speculation stream

and fades away just like a contrail. One of the differences from the previously proposed pre-computing architectures [6, 9] is that the contrail processor architecture does not rely on redundant execution. In the ideal case, the number of instructions executed is unchanged. Another is that its target is improving energy efficiency instead of improving performance.

The potential of the contrail processor architecture on energy-efficiency is estimated as follows. We decide that clock frequency and supply voltage for the verification pipeline are half of those for the speculation pipeline. We assume that half of the original execution of an application is ideally predicted and is distributed uniformly as explained in Figure 1(a). This is a reasonable assumption, since it is reported that 59% of dynamic traces can be reused with the help of the value prediction [4]. Under the assumption, the execution is divided into the speculation and verification streams on a contrail processor with three contexts (1 and 2 for the speculation and verification streams respectively) as depicted in Figure 1(b). The predicted regions are skipped in the speculation stream and execute in the verification streams with enlarging their execution time. Energy consumption is calculated as follows. For the speculation stream, it becomes half of the original execution since the number of instructions reduced by half. In contrast, for the verification streams, the sum of every execution time remains unchanged since the execution time of each instruction increases by double while the total number of instructions reduced by half. Its energy consumption is decreased by the reduction of the clock frequency and the supply voltage. Based on Eq.(1), it is reduced to $\frac{1}{8}$. Thus, total energy savings is 37.5%. In addition, the application gains higher performance. On the other hand, when the contrail processor has only two contexts, the execution time becomes slightly longer as explained in Figure 1(c). In this model, every thread constructing the verification stream is kept in a FIFO queue when it can not obtain its dedicated context. It is true that the effectiveness of the contrail processors (energy savings) depends on the value prediction accuracy and the size of each predicted region. However, we have confirmed that the potential of the contrail processors on energy saving is substantial.

3 Summary

Currently, it is expected that multi-threading and dual-power functional units are key techniques for energy reduction [5]. In this paper, we proposed such an energy-efficient processor architecture based on value prediction and multi-threading techniques. These techniques exploit thread level parallelism, resulting in mitigating performance loss caused by the supply voltage reduction. From preliminary estimation, the proposed architecture has a potential of approximately 40% energy savings with maintaining processor performance.

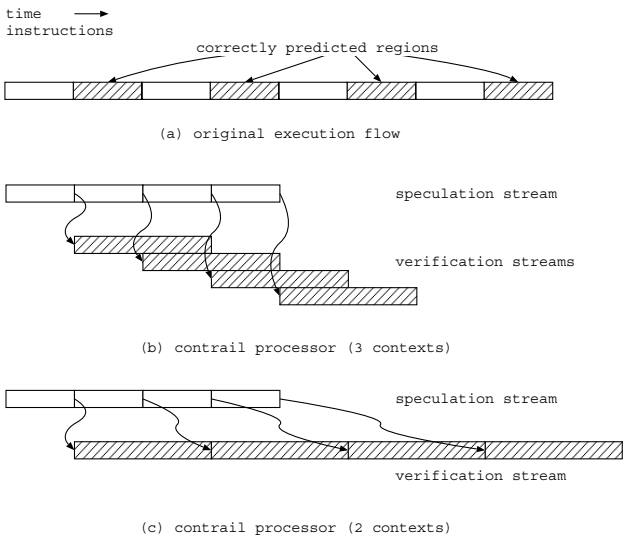

Figure 1: Execution on a contrail processor

Acknowledgments

This work is a joint study with Semiconductor Technology Academic Research Center (STARC).

References

[1] A.P.Chandrakasan and R.W.Brodersen, "Minimizing power consumption in digital CMOS circuits," Proc. of IEEE, vol.83, no.4, April 1995.

[2] T.Hiramoto and M.Takamiya, "Low power and low voltage MOSFETs with variable threshold voltage controlled by back-bias", IEICE Trans. on Electronics, vol.E83-C, no.2, February 2000.

[3] Intel Corporation, "Introduction to Hyper-Threading technology", White paper, September 2001.

[4] M.L.Pilla, A.T.da Costa, F.M.G.Franca, P.O.A.Navaux, "Predicting trace inputs with dynamic trace memoization: determining speedup upper bounds", 10th Int. Conf. on Parallel Architectures and Compilation Techniques, Work in Progress session, September 2001.

[5] J.Rattner, "Electronics in the Internet age", 10th Int. Conf. on Parallel Architectures and Compilation Techniques, Keynote Address, September 2001.

[6] A.Roth and G.Sohi, "Speculative data-driven multithreading", 7th Int. Symp. on High-Performance Computer Architecture, Junuary 2001.

[7] R.Sathe, K.Wang, and M.Franklin, "Techniques for performing highly accurate data value prediction", Microprocessors and Microsystems, vol.22, no.6, November 1998.

[8] T.Sato, "Study on application of high-performance processor architectures for energy-efficiency", Proposal for STARC joint research program, September 2000 (in Japanese).

[9] K.Sundaramoorthy, Z.Purser, and E.Rotenberg, "Slipstream processors: improving both performance and fault tolerance" 9th Int. Conf. on Architectural Support for Programming Languages and Operating Systems, November 2000.

[10] D.M.Tullsen, S.J.Eggers, and H.M.Levy, "Simultaneous multithreading: maximizing on-chip parallelism", 22nd Int. Symp. on Computer Architecture, June 1995.