Draft

ClockHands: Rename-free Instruction Set Architecture for Out-of-order Processors

Luis D. Pena
EECE.5821

Summary

What is the problem the paper tries to solve?

The paper addresses the problem of speeding up irregular programs while improving power efficiency. Irregular programs are key in the execution, optimization, and compilation of interpreted and high-level programming languages, gaming, and social network analysis, to name a few applications. According to the paper, the only solution that the industry offers to speed up these processes is to integrate more massive out of order cores, which improve processor's single thread performance, which is key to reduce latency. However, although this unique solution seems to solve the problem, it does so with poor power efficiency.

What are the main ideas or insights of the paper?

The paper tries to solve the problematic with a different approach, abandoning the idea of increasing out of order cores and focusing his attention on the ISA, and how existing instruction set architectures (ISA) manage register operands and how this affect power efficiency. Specifically, the paper reviews two existing ISAs, Conventional RISC-V, and STRAIGHT. The paper points out that RISC-V specify register operands by register name. Due to register renaming poor power efficiency emerges, preventing an increase in the front-end width. On the other hand, STRAIGHT manage register operands by inter-instruction distance, which eliminates register renaming, but due to its strong constraint in instruction placement, a large increase in number of instructions is generated, which do not help with power efficiency either, since more instructions implies consuming more power.

The paper then propose a novel solution called ClockHands and compare its power efficiency with RISCV and STRAIGHT. ClockHands try to take the good of both worlds, since as opposed to RISCV it does not require register renaming, and in contrast to STRAIGHT it has looser constraints on instruction placement which reduce number of instructions generated at levels like RISCV.

Strengths

Does the paper solve the suggested problem well?

Methodology wise, the paper did a good job measuring each ISA against a common ruler or benchmarks including SPEC CPU 2006/2017. for this purpose, implementing cycle-accurate simulator, FPGA implementation, and first-step compiler for ClockHands was necessary. This effort also allowed instruction count comparison of STRAIGHT and ClockHands which provided an empiric evidence of the cause of increased instruction count in STRAIGHT, and a method of verification that this issue does not

occur in ClockHands, but in the contrary that is like RISCV in terms of instruction counts, which is good since fewer instructions count improve power consumption.

The solution presented ClockHands, simulation was fundamental to estimate power consumption. Overall, ClockHands did what it promised, reduce power consumption. Results presented show the following remarks:

1. On a machine with an eight-fetch width:
   a) results showed that Clock-hands consume 7.4% less energy than RISC.
   b) And its performance was comparable to RISC.
2. simulating a futuristic up-scale processor with a 16-fetch width.
   a) results showed that Clock-hands energy reduction increases up to 24.4%.
   b) which shows that Clock-hands enables a more power efficient and wider front-end?

Weakness

Although they offer cycle-accurate simulation I think would've been beneficial to include different simulations. Simulation will always be that simulation, and although it can predict the trends it can never mimic reality, that is why having multiple methods to validate a proposed architecture is always welcome and simply good practice.

Also, taking in consideration that each ISA is unique and perform differently it would've been good to compare ClockHands with others ISA besides RISCV and STRAIGHT, since ClockHands took their strength and build upon them implementing a solution which didn't includes their weaknesses. I think there is room for improvement there.

What have you learned and what were your likes and dislikes in the paper?

Although we covered the topic of how the different variables affect performance, CPI, ISA, etc. in class, it was refreshing to see in practice how the ISA can affect performance, specifically power consumption. Also, I liked also that I could relate with the terminology, and that the terms and concepts that we covered in class were relevant to understand the paper. In addition, I'm taking an advance digital design class in which we are introduce FPGA design, which I'm very excited about, and it was good to see that this paper used FPGA in the problematic of solving Computer architecture problematic, which is a plus.

Overall, I didn't have major dislikes, the paper was well written, however it was a little be confusing at first, since in the abstract section the paper first states the following "a recently proposed architecture called STRAIGHT specifies operands by inter-instruction distance, thereby eliminating register renaming". Then a paragraph after they state "ClockHands does not require register renaming as in STRAIGHT" which is a contradiction, or maybe a bad choice of words, I think they meant to say ClockHands does not require register renaming like RISCV does, this idea was later clarified as we read the paper.