Travis Ritter and Jonathan Shaefer

CMPE220, 01

**Literature Review #2**

The article we decided to do a review on is titled “Design and implementation of an elastic processor with hyperthreading technology and virtualization for elastic server models.” This article’s purpose is to present design methodology and implementation of an elastic-natured 32-bit RISC-pipelined processor inspired by MIPS. Elastic meaning it scales required computation power based on certain conditions. The article introduces the material by stating that despite the fact that cloud computing has made computer computations much more efficient, the demand for computation power is ever rising. They then dive more into the meaning of elasticity, with respect to this paper, saying that it refers to the ability to scale hardware resources using concepts of virtual memory and hyperthreading. These concepts are implemented at a hardware level, which they say provides smoother execution on their processor. Not only is it smoother, but it provides salient features of parallel execution and a shorter instruction set. The article then dives into how microprocessors are generally made. There are two major units the control unit, and the data path unit. They work in tandem to pass data along the processor, so it can be used for its specific purpose. What makes this processor unique is the use of hyperthreading technology (HTT), which forms multiple threads within the core unit and uses the operating system (OS) as an interface. These threads allow for simultaneous execution that lowers and sometimes eliminates latency. This hyperthreading tends to improve performance, they say, by providing as many independent functions in the pipeline as possible. Also, this processor uses virtual memory in the pipeline. This hybrid-natured pipelining with virtual memory allows for independent instruction manipulation, and simultaneous thread execution. Their processor is faster, because instead of using software based implementations of hyperthreading, they use hardware based implementations that are considerably faster.

The paper then moves on to talk about their instruction set architecture (ISA), saying that it is a hybrid ISA, and uses control models to standardize the process. This hybrid ISA was designed with respect to the hyperthreading discussed previously, and it allows for the processor to keep the maximum number of independent instructions in the pipeline. To be able to do this the ISA includes aspects from RISC, CISC, and VLIW which are all popular computer architectures. When it comes to their instructions, each one is made up of multiple smaller instructions. So, either the instruction can be processed all at once, or it can be broken into its independent parts. These parts are then executed at different stages in the pipeline. This method of instruction execution is referred to as ‘symmetrical’ and it allows for the server architecture to be dynamically configured based on the user’s needs or preferences. The actual instruction set is very similar to MIPS, there is add, addi, lw, sw, beq, bne, etc. There are some differences, since this processor has a Load Byte (lb) and Store Byte (sb) instruction, which allow for more specific memory allocation. They then talk about the finite state machine or FSM, which is a uniform state machine model that can be monitored, and more importantly, customized. The data path they use for this architecture is a massive parallel threaded multi-core platform, in other words it allows for many things to be sent through the data path simultaneously. A functional aspect of the data path, in the control unit, while technically this processor does not have a control unit, it does have a unit that generates control signals based on the current state of the FSM. This is termed as the control unit. The control unit, as it is called, automatically updates on its own, which is referred to as automation. This automated process is said to be more precise and efficient as the data path is utilized more.

The next thing that is mentioned is major units of the processor that is being designed. Major units meaning the consequential aspects of the processor, that do the most work. These are the arithmetic logic unit (ALU), data memory, register files, instruction memory, and the program counter (PC). The ALU is used for general calculations and instruction manipulation. The ALU in this processor is unique because, under normal conditions the original ALU is the only operational one, but when under high loads, a second, virtual, ALU is utilized and acts as the master ALU. The next unit is data memory, which is a unit that is used primarily for resource management to support the hardware. They then talk about register files, and how each register has more than one access point, to allow for simultaneous threads to access the enclosed data at the same time. Each register file is divided in two the top half is used by the core (hardware) only, while the second half holds external information used by the virtual processor. This is presumably faster than a traditional register file with only one access point. The next unit is the program counter (PC), whose sole purpose is to point to the next instruction. Unlike traditional machines, this processor has a core and a virtual PC, that are used together to help achieve better memory and resource management.

The paper then goes on to describe the test and benchmarks done on this processor, and the results of said tests. The first thing they tested was the clock speed of the processor, clock speed just being the rate at which the processor cycles through pipeline stages. According the authors, this processor passed all these benchmarks and tests with sufficiency. Also, they also state that the cost of creation for these processors is lower than others, because it uses fewer physical cores, by offsetting work onto virtual cores. There are then multiple tables and figures detailing the performance of this chip versus a standard MIPS processor core. The results show that this processor, uses less power, and completes more instructions per cycle. However, this processor is much more complex than a regular MIPS based processor, which is not necessarily an advantage.

In conclusion, this new processor not only meets benchmark requirements, but is also an improvement over them. This is all done without the cost of manufacturing rising significantly. Also, this processor is more suited for modern server models that use machine learning to improve capability. The authors then go on to state future goals, mainly more thorough power and physical design analysis before it would be ready for widespread usage.

**Works Cited**

Bir, Parth, et al. “Design and Implementation of an Elastic Processor with Hyperthreading Technology and Virutalization for Elastic Server Models.” *The Journal of Supercomputing: An International Journal of High-Performance Computer Design, Analysis and Use*, vol. 76, no.9, 2020, p.7394. EBSCOhost, doi:10.1007/s11227-020-03174-5.