

1 Comparison of x86 Simulators

1.1 Classifications

There are many ways to divide simulators into various types. One way is by the level of accuracy. For example, Functional simulators and Timing simulators. The biggest difference between these two is whether they prefer to be timing accurate or functional accurate. For example, a timing simulator would give real time performance of the CPU, it would keep track of clock cycles, so if an operation that takes 5 cycles on an actual x86 processor would take 5 cycles in the simulator. These simulators are very resource intensive because they must be accurate to the times and needs to give kernel and other programs CPU time. A lot of these simulators give up some accuracy of the simulation in order to reach the timing of an actual on-hardware piece of silicon. The other type of simulator, functional simulators, tend to be less resource heavy but will be more accurate because they do not need to stay true to the timings of an on-hardware chip. Another way to categorize these simulators is by whether they simulator application level programs or do they also simulate the kernel. When only running application level programs, the simulator must pass kernel or other system service request down to the host kernel. This can be a problem when running some simulations, as they may require large computation in the kernel level and thus would not lead to an accurate result. One last way to classify simulators is if they run by a trace or in real-time. In a trace-based simulator, the simulator runs off of a file that contains the required data. This data was created prior to the simulator and contains the data it needs to accurately simulate the program, for example a trace-based cache simulator would need a data file of all the memory access that the program took. In a real-time simulator, the sim runs the application in real time.

2 x86 Simulator Pros and Cons

2.1 Gem5

The Gem5 simulator is a very extensive simulator capable of simulating many different ISAs and GPUs very similar to the Multi2Sim simulator. It can simulator different hardware threads, pipeline stages, and issue widths. A study was conducted and it was found that the Gem5 simulator had an error percentage between 1% to 20%. From all of the simulators this simulator was the most extensive with support for fast forwarding and cache warmups and more.

2.2 Sniper

Sniper is a simulator that exceeds in simulating large multicore designs. Sniper is based on Graphite, another parallel simulator. Similar to the Gem5 simulator, a study was conducted and it was found that it had close to a 25% accuracy compared to an actual processor.

2.3 Multi2Sim

Multi2Sim is a simulator that allows the user to simulate both the CPU and the GPU. Like the Gem5, the simulator can simulate multiple ISAs. This simulator can handle different simulation models including functional, detailed, and event-driven. It does not simulate an OS, but uses dynamic thread creation. The simulator was validated using a reference AMD Radeon 5870 GPU and had an error between 5% to 30%.

2.4 PTLsim

The PTLsim is a cycle-level full-system x86 simulator with a pipeline that borrows from many modern processors. The test had an error of around 5% compared to actual hardware using rsync compared to an AMD Athlon 64 processor.