Rohan Jhaveri

Reading Assignment 2

Topic: An efficient algorithm for exploring multiple arithmetic units.

R. Tomasulo

This paper addresses the need to optimize the actual performance of arithmetic operations, especially floating point. It describes hardware algorithm for dynamic scheduling of instructions that allows out-of-order execution, designed to efficiently utilize multiple execution units. It addresses the concept of register renaming in hardware, reservation stations for all execution units, and a common data bus (CDB) on which computed values broadcast to all reservation stations that increases Instruction Level Parallelism(ILP).

The CDB connects reservation stations directly to functional units maintain the precedence and encouraging concurrency. Functional units can access the result of any operation without involving a floating-point-register, allowing multiple units waiting on a result to proceed without waiting to resolve contention for access to register file read ports by using reservation tables. The reservation stations control when an instruction can execute, rather than a single dedicated hazard unit. Instructions are issued sequentially so that the effects of a sequence of instructions, such as exceptions raised by these instructions, occur in the same order as they would on an in-order processor, but they are being executed out-of-order. Tomasulo's Algorithm uses register renaming to correctly perform out-of-order execution.

The issues with this algorithm is that each result commits to its reservation station register, and gets written back to its architectural register also. The write to the reservation station register gets broadcast, so that instructions waiting for the result know to grab it, but otherwise, the result lands in an architectural register also. This means that results appear in the architectural register file out of order, and an exception would expose the fact that the machine ran out-of-order. Also, the basic algorithm does not provide for speculation beyond branches. Therefore, we must add some branch prediction scheme together with hardware to deal with speculative execution of instructions.