Abstract

The effective performance of computer systems is to a large measure determined by the synergy between the processor architecture, the instruction set and the compiler. In the past, the sequencing of information within processor architectures has normally been synchronous: controlled centrally by a clock. However, this global signal could possibly limit the future gains in performance that can potentially be achieved through improvements in implementation technology.

This thesis investigates the effects of relaxing this strict synchrony by distributing control within processor architectures through the use of a novel asynchronous design model known as a *micronet*. The impact of asynchronous control on the performance of a RISC-style processor is explored at different levels. Firstly, improvements in the performance of individual instructions by exploiting actual run-time behaviours are demonstrated. Secondly, it is shown that micronets are able to exploit further (both spatial and temporal) instruction-level parallelism (ILP) efficiently through the distribution of control to datapath resources. Finally, exposing fine-grain concurrency within a datapath can only be of benefit to a computer system if it can easily be exploited by the compiler. Although compilers for micronet-based asynchronous processors may be considered to be more complex than their synchronous counterparts, it is shown that the variable execution time of an instruction does not adversely affect the compiler's ability to schedule code efficiently. In conclusion, the modelling of a processor's datapath as a micronet permits the exploitation of both fine-grain ILP and actual run-time delays, thus leading to the efficient utilisation of functional units and in turn resulting in an improvement in overall system performance.