Formal Aspects of VLSI Research Group
University of Utah, Department of Computer Science

Dynamic Reordering of High Latency Transactions in Time-Warp Simulation Using a Modified Micropipeline*

ARMIN LIEBCHEN
GANESH GOPALAKRISHNAN
University of Utah
Dept. of Computer Science
Salt Lake City, Utah 84112

Keywords: Asynchronous Design, Micropipelines, Dynamic Instruction Reordering, Time Warp Simulation

Abstract. *Time warp* based simulation of discrete-event systems is an efficient way to overcome the synchronization overhead during distributed simulation. As computations may proceed beyond synchronization barriers in *time warp*, multiple checkpoints of state need to be maintained to be able to rollback invalidated branches of the lookahead execution. An efficient mechanism to implement state rollback has been proposed in [1]. In this environment, a dedicated Roll-back Chip (RBC) maintains multiple versions of state by responding to a set of control instructions interspersed with the regular stream of data-access instructions. As these control instructions have latencies that are orders of magnitude more than the latencies of data-access instructions, a strict ordering of the instructions may lead to large inefficiencies.

This paper describes a dynamic instruction reordering scheme that optimizes multiple pending instructions to achieve higher throughput. A modified asynchronous *micropipeline*, called the *Asynchronous Reorder Pipeline* (ARP) has been chosen to implement this scheme. ARP can be easily adapted for supporting dynamic instruction reordering in other situations also. After outlining the design of the ARP, we present its high level protocol, and a correctness argument. We then present two new primitive asynchronous components that are used in the ARP: a *lockable C-element* `LockC`, and an *exchange pipeline stage* `ExLatch`. Circuit level simulation results are presented to justify that `LockC` – a critical component of our design – functions correctly. The newly proposed primitives, as well as the ARP itself, are useful in other contexts as well.