Simulating the 62xx Pipeline for Teaching Computer Architecture

Oscar Yáñez-Suárez

Department of Electrical Engineering
Universidad Autónoma Metropolitana – Iztapalapa, México

e-mail: yaso@xanum.uam.mx

Abstract

The advances in digital technology have rendered obsolete the contents, scope and reach of many traditional undergraduate curricula on the subject. A broad range of topics, from programmable logic and logic synthesis to pipelining, superscalar execution or parallel and distributed systems, for example, must be included in the course contents in order to keep them in pace with the new demands of the field. As adding extra quarter/semester courses to the digital electronics branch of the curriculum is usually not an option, effective teaching methods and strategies are required to adequately expose the students to both the fundamentals and current frontiers of the digital world.

Under this perspective, the use of the 62xx processor architecture as a reference in digital systems teaching is a reasonable choice, as it combines basic concepts of computer architecture into a powerful, yet simple to understand state-of-the-art processor. Load/Store architectures, sufficiency of indirect addressing, or instruction set orthogonality are some fundamentals that can be naturally observed and exemplified, together with hardware/software pipelining, VLIW or fast multiplies. Moreover, even when obviously the purpose of the architecture is supporting high performance DSP, the signal processing aspects of it can be safely ignored (or transferred to a course in discrete linear systems and applications) by realizing that the 62xx is basically a general purpose RISC machine.

For the students to experiment and clearly understand all the mechanisms involved in the operation of this family of processors, a visual simulator of the 62xx pipeline has been developed. 62XXVIEW is a Win32 application that loads and disassembles 62xx programs stored in standard hex format, and fully simulates its execution while graphically showing the progression of the fetch and execute packets through the system pipeline. The visual feedback of the program has been proven to be a major aid in the understanding of the processor main design principles.

The program is being used in an entry-level, quarter-length Computer Architecture course at our university, with satisfactory results.