Techniques for Accurate Performance Evaluation in Architecture Exploration

George Hadjiyiannis, Srinivas Devadas

Abstract — We present a system that automatically generates a cycle-accurate and bit-true Instruction Level Simulator (ILS) and a hardware implementation model given a description of a target processor. An ILS can be used to obtain a cycle count for a given program running on the target architecture, while the cycle length, die size, and power consumption can be obtained from the hardware implementation model. These figures allow us to accurately and rapidly evaluate target architectures within an architecture exploration methodology for system-level synthesis.

In an architecture exploration scheme, both the ILS and the hardware model must be generated automatically, else a substantial programming and hardware design effort has to be expended in each design iteration. Our system uses the LSDL, machine description language to support the automatic generation of the ILS and the hardware synthesis model, as well as other related tools.

Keywords — architecture exploration, instruction set, cycle-accurate simulation, hardware synthesis.

I. INTRODUCTION

Embedded systems typically require low cost and low power consumption. To reduce manufacturing cost and power consumption, it is important to match the architecture of the processing engine to the application at hand. A simple way of designing such a processing engine is architecture exploration by iterative improvement (see Figure 1). In this approach, the application code is analyzed, and an initial target architecture is generated and described in a machine description language. The application code is then compiled for this target architecture and executed on an Instruction Level Simulator (ILS) where performance measurements and utilization statistics are gathered. A hardware model of the target architecture is used to derive the length of the cycle and the physical costs (such as die size and power consumption). These measurements allow one to evaluate the architecture and make improvements. A new architecture is generated based on these improvements and the process repeated until no further improvements can be made.

Such a synthesis scheme can only be effective if the design evaluation tools (compiler, ILS, hardware model, assembler and disassembler) can be automatically generated from the machine description. Automatic generation of the design evaluation tools allows rapid evaluation of candidate architectures, increasing the coverage of the design space while shortening the design time and, thus, the time-to-market. The machine description language forms the most important part of the system. Ideally, it should support the automatic generation of all the design tools rather than be-