Abstract

In recent years, the market for special-purpose devices designed for advanced applications has grown at a tremendous rate. As a result, the demand for embedded microprocessors, a necessary component of these devices, is stronger than ever. The nature of devices such as Personal Digital Assistants (PDAs), mobile phones, printers, and networking equipment requires that these embedded processors meet high performance levels while simultaneously satisfying strong constraints on power consumption and cost.

Instruction-Level Parallelism (ILP) is one of the major forces increasing the performance of high-end workstation processors. Such ILP architectures are highly complex and exhibit a large amount of power dissipation. However, parallelism is also a well-known power-saving technique that can be used to improve the energy efficiency of a system. ILP can thus be a very attractive technique for embedded processors that require increased performance at a low energy consumption.

This work focuses on the design of synergistic hardware-compiler ILP architectures, such as EPIC or VLIW machines, for low-power embedded processors. Such synergism minimizes the hardware overhead of multiple-issue pipelines, while maintaining the performance benefits of ILP. Introducing parallelism into a processor drastically alters its architecture. To understand and quantify how such modifications can reduce or nullify the expected benefits, and also to assess where the tradeoffs should be made, a new EPIC-like low-power processor, *DEVIL*, is proposed. Its implementation is the subject of a detailed experimental evaluation.

DEVIL includes a fetch mechanism that supports variable instruction lengths and allows the compiler to explicitly encode parallelism within an instruction bundle. It will be shown that this mechanism allows savings of 50% on average in the code size with respect to a standard VLIW fetch mechanism while keeping performance unchanged.

DEVIL, with its 2-issue pipeline, achieves a speed-up of 1.5 on average compared to a 1-issue processor. This performance enhancement allows DEVIL to work at a lower voltage and a lower clock frequency while keeping the same level of performance of a scalar processor. It will be demonstrated that DEVIL can execute a task a the same speed than a scalar processor while requiring an energy consumption approximatively 38% smaller.

ILP architectures generally suffer from a large amount of code expansion. This negative effect is reduced thanks to *DEVIL*'s instruction fetch mechanism. However, *DEVIL* still suffers from a code size penalty, compared to a scalar processor. To counter this unfortunate fact, a step is made towards predication techniques. It will be shown that a full-predication support with an adequate instruction fetch mechanism allows to generate parallel code that is 12% faster and 25% smaller.