16. Purple Dragon book - Chapter 10.5 (Software Pipelining)

Team

- Pech David, Bc. xpechd00@stud.fit.vutbr.cz
- Ondrušek Libor, Bc. xondru12@stud.fit.vutbr.cz

Abstract

The chapter discusses method called “software pipelining” which in very basic terms suggests that the CPU can simulateously process several operations of different type. When the code is generated the target language instructions can be partly parallelised depending on their type and parameters.

In the first part the chapter is focused on the optimal decomposition of the input code to the target language instructions so that there is a minimal amount of nop instructions (the instructions signalising no operation, inserted to wait for a completion of the previous multi-cycle instruction).

The chapter goes on and shows more complicated patterns where a greater benefit from the software pipelining can be achieved – most notably loops. After unrolling a loop, a significant speed gain can be obtained if the pipelining technique is used.

There are of course certain limitations which the book covers in-depth. Order of the instructions (read vs. write) can be partly shifted to decrease the total cycles of the loop. The variables often depend on each other, even in the current iteration, result from the previous iteration is used and so on. The specific variable depency table has to be constructed to determine suboptimal resources utilisation.

We will focus on two main ideas of the chapter:

- High-level description of the pipelining with an example of the speed improvement if used properly
- Main rules which need to be kept in order to maximalise the potential of the optimization.

General Terms

Software pipelining

Keywords

Optimization, pipelining, performance, conditional statements

Bibliography

- **Aho, A.V., Lam, M.S., Sethi, R., Ullman, J.D.**: Compilers: Principles, Techniques, and Tools (2nd Edition)