Saarland University
Department 6.2 - Computer Science
Prof. Dr. W. J. Paul
M. Sc. Artem Alekhin

Computer Architecture I - SS09

Exercise Sheet 6 (due: 22.06.09)

Exercise 1: (delay slot) (5 points)

Assume that we have a DLX machine with delayed branch and you as *compiler* for such a machine. In programmes for such a machine the delay slots should be filled either with NOP instructions (means: "no operation") or with some useful operations.

Take the program you wrote in the previous Exercise Sheet 4 (Exercise 3) (the correct version of the programm will be shown in the tutorial this week) and fill out the minimum of delay slots by NOP-operations. The smallest number of NOP-operations bring you the maximum amount of points.

Exercise 2: (fast forwarding circuit) (7 + 8 points)

In the lecture we built a forwarding circuit capable of result forwarding from 3 stages. The construction is obviously generalizes an s-stage forwarding with $s > 3$. The actual data selection is then performed by s cascaded multiplexors. Thus, the delay of this implementation of an s-stage forwarding engine is $O(s)$.

However, these s multiplexors can also be arranged as a balanced binary tree of depth $\lceil \log(s) \rceil$. The signal $top.j$ indicates that a circuit stage j provides the current data of the requested operand (please, use the last naming convention for circuit and register stages introduced in the lecture). These signals $top.j$ can be used in order to control the multiplexor tree.

1. Construct a circuit TOP, which generates the signals $top.j$ using a parallel prefix circuit.

2. Construct an s-stage forwarding engine based on the multiplexor tree and the circuit TOP. Show that the implementation has delay of $O(\log(s))$.