Saarland University

Department 6.2 - Computer Science

Prof. Dr. W. J. Paul

M. Sc. Artem Alekhin

Computer Architecture I - SS09

Exercise Sheet 4 (due: 8.06.09)

Exercise 1: (DLX constant) (3 points)

In the lecture we considered Instruction Set Architecture of the sequential DLX processor. The computation of the constant in the arithmetic-logical or shift operations can be defined as:

$$C_o(c)[31:0] = \begin{cases} sxtimm(c) & : j-type(c) \vee i-type(c) \\ *^{27}SA(c) & : r-type(c) \end{cases}$$

where c is a current DLX configuration, $sxtimm(c)$ returns the sign extended immediate constant, $SA(c)$ is a shift amount, and * means "no matter" value.

In this exercise you have to design a circuit that computes this constant.

Exercise 2: (32-bit constants) (3 points)

We need more than one instruction to load a 32-bit constant into a register of the DLX processor. Since we know the value of the constant during writing a programm, we can build 32-bit constants with two instructions. Give a rule(s) how to construct these instructions. *In these rules you have to consider more than one case!* (as before use the instruction set given on the lecture's web page).

Exercise 3: (DLX assembler programming) (7 points)

Let n, i, $a_i \in \mathbb{N}$. Write a DLX assembler programm computing the maximum of the numbers $a_1, ..., a_n$. At the start of your program n will be stored in register $GPR[1]$ and a_i in a memory cell $M(i-1)$ for $i \in \{1, ..., n\}$. Your program should store its result in the memory cell $M(n)$. You should not take care about delay slots in your programm. Comment your program in a way everyone can understand what it does.

Exercise 4: (next PC computation) (7 points)

In the lecture we considered the next PC computation circuit and the simulation relation for the sequential DLX processor. In this exercise you have to prove the following statement:

$$sim(c^i, h^i) \longrightarrow c^{i+1}.pc = h^{i+1}.pc$$