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

Computer Architecture I - SS09

Exercise Sheet 3 (due: 25.05.09)

Exercise 1: ((de)coding DLX instructions) (2 points)

The assembler instructions of the DLX processor are (see "Instruction Set Architecture (ISA)" in the section "Layouts" on the lecture's web page):

- r-type: Mnemonic / destination register / source register 1 / source register 2
  Example: add R1 R2 R3 (add the values of registers R2 and R3 and store the result in R1)
- i-type: Mnemonic / destination register / source register / 16-bit constant
  Example: add R1 R2 123
- j-type: Mnemonic / constant
  Example: j 500

1. What does the following instruction do (highest bit is on the left side)? Write it in DLX assembler and describe in words.

   01101100101001000000000001100100

2. Code the following two assembler instructions as bit strings:

   $add\ R1\ R2\ R3$
   $ori\ R25\ R12\ 3637$

Exercise 2: (n-decoder) (3 point)

In the lecture we constructed an n-decoder (Figure 1). The n-decoder is a circuit with inputs $x[n-1:0]$ and outputs $Y[2^n-1:0]$ such that for all i

$$Y_i = 1 \leftrightarrow \langle x \rangle = i$$

Prove the correctness of the n-decoder.

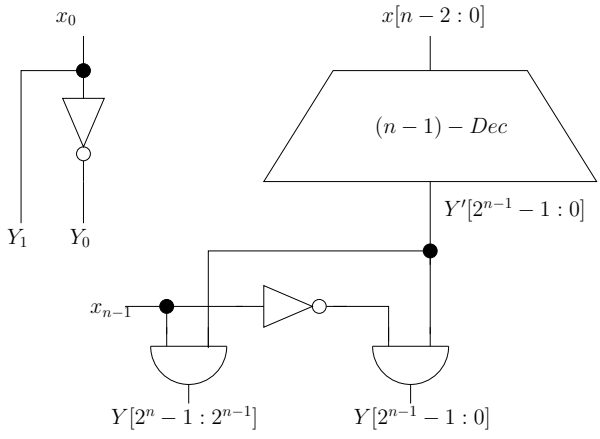

Figure 1: Circuit n-decoder.

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

Computer Architecture I - SS09

Exercise Sheet 3 (due: 25.05.09)

Exercise 3: (3-port RAM) (4 points)

In the lecture we designed a 1-port a x 1 - RAM. Construct a 3-port RAM (Figure 2) specified as follows:

$$\begin{aligned} DoutA(S) &= h.S(AdA(h)) \\ DoutB(S) &= h.S(AdB(h)) \\ h'.S(x) &= \begin{cases} Din(h) & : w(h) \wedge x = AdC(h) \\ h.S(x) & : otherwise, \end{cases} \end{aligned}$$

where $h.S : \{0,1\}^a \to \{0,1\}^d$.

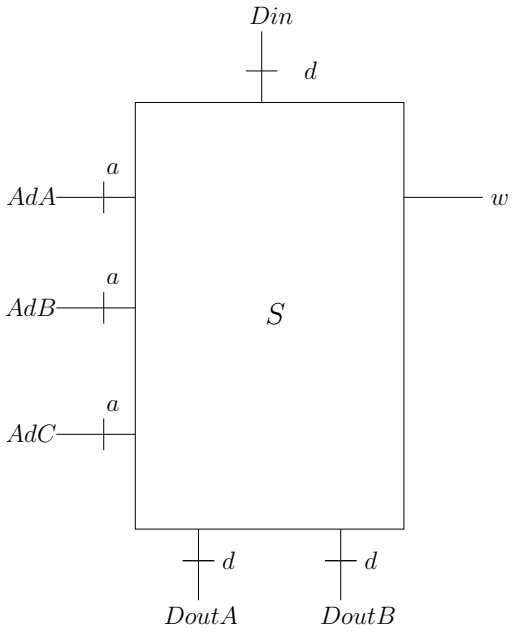

Figure 2: 3-port RAM.