Saarland University
Department 6.2 – Computer Science
Prof. Dr. W. J. Paul
M. Sc. Petro Lutsyk

Computer Architecture – WS14/15

Exercise Sheet 7 (due: 16.12.14, 24 points)

Exercise 1: (open collector bus) (4 points)

Show how to use the *open collector bus* for computation of the following function:

$$\begin{aligned} or &: \mathbb{B}^n \to \mathbb{B} \\ or(a_{n-1}, \ldots, a_0) &= \bigvee_{i=0}^{n-1} a_i. \end{aligned}$$

Prove that your scheme works correctly.

Exercise 2: (bus contention) (6 points)

Recall in the lecture we saw example of self-destructing hardware, which though was provably correct within the digital model. Repeat the similar analysis for the bus from Figure 1. Is there a *bus contention* in this case? In case there is, what is the average time of a *short circuit*?

Exercise 3: (proper bus control) (2 points)

Recall the scheme of proper control of the *tristate bus* from the lecture. Explain (in words) why we need a gap between intervals T_i and T_{i+1} in case if

$$send(i) \neq send(i+1).$$

Exercise 4: (automata construction) (12 points)

In the lecture we defined a *finite state transducer* (also called a *control automaton*) as a 6-tuple of the form

$$M = (Z, z_0, I, O, \delta, \eta).$$

Let

$$\mathbb{Z} = [0:2], \quad z_0 = 0, \quad I = \mathbb{B}^2, \quad O = \mathbb{B}^2$$

and

$$\delta(y,x) = \begin{cases} 0 & y = 2 \vee y = 0 \wedge x = 11 \\ 2 & y = 0 \wedge x \in \{01, 10\} \vee y = 1 \wedge x = 01 \\ 1 & \text{otherwise} \end{cases}$$

and

$$\eta_1(y,x) = \begin{cases} 00 & y = 0 \\ 01 & y = 1 \\ 10 & y = 2 \end{cases}$$

$$\eta_2(y,x) = \begin{cases} 01 & y = 0 \wedge x \in \{00, 11\} \vee y = 1 \wedge x \in \{00, 01\} \\ 10 & y = 2 \\ 00 & \text{otherwise.} \end{cases}$$

Further, let

$$\begin{aligned} M_1 &= (Z, z_0, I, O, \delta, \eta_1) \\ M_2 &= (Z, z_0, I, O, \delta, \eta_2). \end{aligned}$$

We consider two implementations of a Moore automaton, where the output function does not depend on the input, and two implementations of a Mealy automaton, where the output function does depend on the input. In this exercise you need to provide the following gate level constructions:

(a) Moore(1) (i.e., a Moore automaton without pre-computation of the outputs) implementation of automaton M_1,

(b) Moore(2) (i.e., a Moore automaton with pre-computation of the outputs) implementation of automaton M_1,

(c) Mealy(1) (i.e., a Mealy automaton without pre-computation of the outputs) implementation of automaton M_2,

(d) Mealy(2) (i.e., a Mealy automaton with pre-computation of the outputs) implementation of automaton M_2.

Note, that you can provide the construction of circuits such as *nexts* and *out* as separate units and then use them as building blocks in the construction of the entire automata.

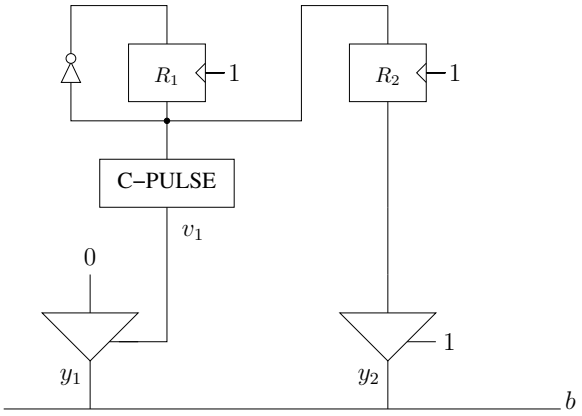

Figure 1: Example of the tristate bus control