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

# Computer Architecture – WS14/15

**Exercise Sheet 2 (due: 11.11.14, 26 points)**

## Exercise 1: (computational limits) (4 points)

Estimate the physical limit for the computational frequency of a processor made out of one spoon of water. Hint: use the uncertainty principle and the mass-energy equivalence.

## Exercise 2: (smart decoder) (6 points)

Give a construction of a "smart" decoder, i.e. a circuit which fulfills the following specification:

- input $x \in \mathbb{B}^n$
- output $y \in \mathbb{B}^{2^n}$, s.t.

$$y[i] \leftrightarrow \langle x \rangle = i$$

Prove that your construction

(a) mets the specification.

(b) has a delay logarithmic in $n$.

## Exercise 3: (fast OR-trees) (4 points)

Let $n = 2^k$ for $k > 0$. Construct a circuit which implements the following function:

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

Optimize the delay of your construction using gates from

(a) the upper half of table 1.

(b) the entire table 1.

| gate | symbol | cost | delay |
|---|---|---|---|
| INV | $\neg$ | 1 | 1 |
| AND | $\wedge$ | 2 | 2 |
| OR | $\vee$ | 2 | 2 |
| NAND | $\overline{\wedge}$ | 2 | 1 |
| NOR | $\overline{\vee}$ | 2 | 1 |
| XOR | $\oplus$ | 4 | 2 |

Table 1: Cost and delay of gates

## Exercise 4: (circuit definition) (4 points)

Consider the circuit in figure 1, consisting of an inverter, of an AND, NAND and of a NOR gate. Let $a$, $b$, $c$ be inputs and $d$, $e$ outputs of $S$.

(a) Describe $S$ formally as a graph, i.e. show the set of its edges, nodes, etc.

(b) Give the in/out-degree numbers of each node.

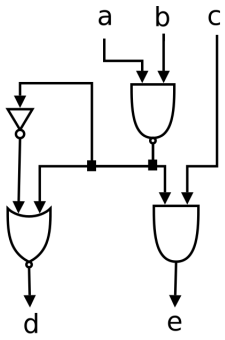


Figure 1: Circuit $S$

**Exercise 5: (SRAM construction) (8 points)**
Recall the principles of static RAM construction from the lecture.

(a) Specify and construct an $(n, a)$-RAM with two ports $A$ and $B$ which can both be used to read and write. In case of simultaneous writes to the same address the write specified by port $A$ should win.

(b) Provide an extra control input *inv*. Activation of *inv* sets all register contents to $0^n$ and overrides possible concurrent ordinary writes[1].

[1]Such RAMs are used for instance to store the status of cache lines. Invalid data is signaled by value $0^n$. After power up caches contain no meaningful data, and this is signaled in their status if one activates *inv* together with *reset*.