Saarland University
Department 6.2 - Computer Science
Prof. Dr. W. J. Paul
M. Sc. Christian Müller

Computer Architecture I - WS 07/08

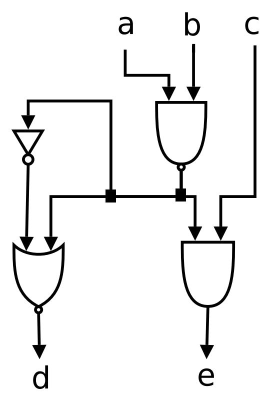
Exercise Sheet 2

Organizational stuff

Important! To receive credit points for this course, students of CS and CuK must register for it in the HISPOS[1] system of the examination office by December 1, 2007.

Excercise 1: (circuit definition, graph theory)

1. Get familiar with the formal definition of a *graph*.[2]
2. Consider the following circuit S 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. Describe S formally as a graph, i.e. show the set of its edges, nodes, etc.

3. Give the in/out-degree numbers of each node.

Excercise 2: (circuit definition)
The depth $de(g)$ of a gate g in a circuit S is defined as follows:

$$de(g) = \begin{cases} 0 & : \quad g \text{ is an input of } S \\ max\{l(p) \mid p \text{ is a path into } g \text{ in } S\} & : \quad \text{otherwise} \end{cases}$$

Where $l(p)$ is the length of a path p defined over the number of gates in p. Prove that into a gate of depth u there are at most 3^u paths.

Excercise 3: (decomposition lemma)
Let $a \in \mathbb{B}^n$ and $\langle a \rangle$ its binary representation. For $k \in \{0, \ldots, n-1\}$ prove:

$$\langle a[n-1:0] \rangle = \langle a[n-1:k] \rangle \cdot 2^k + \langle a[k-1:0] \rangle$$

---

[1] https://www.lsf.uni-saarland.de
[2] You can use, e.g., Wikipedia or a lot of computer science books.

Saarland University

Department 6.2 - Computer Science

Prof. Dr. W. J. Paul

M. Sc. Christian Müller

Computer Architecture I - WS 07/08

Exercise Sheet 2

Excercise 4: (sign extension)

Let $a \in \mathbb{B}^n$ and $\langle a \rangle$ its binary representation. Prove that $\langle a \rangle = \langle 0^n a \rangle$ for $n \in \mathbb{N}$.

Excercise 5: (full adder)

Full adder is a circuit computing the sum of three input bits:

$$fa : \mathbb{B}^3 \to \mathbb{B}^2, fa(a, b, c) = (s_1, s_0)$$

with $\langle a \rangle + \langle b \rangle + \langle c \rangle = \langle s_1 s_0 \rangle$. E.g., $f(0,0,0) = (0,0), f(0,1,0) = f(1,0,0) = (0,1), f(1,0,1) = (1,0)$, etc. Implement this circuit using logical gates.

Excercise 6: (carry chain adder)

Consider the implementation of a *carry chain adder*.[3] Prove that the carry chain adder computes the following function cca:

$$cca : \mathbb{B}^n \times \mathbb{B}^n \times \mathbb{B} \to \mathbb{B}^{n+1}$$
$$cca(a[n-1:0], b[n-1:0], c_{in}) = c_{out}s[n-1:0]$$

with $\langle a[n-1:0] \rangle + \langle b[n-1:0] \rangle + \langle c_{in} \rangle = \langle c_{out}s[n-1:0] \rangle$ where $c_{out}s[n-1:0]$ is just a concatenation of bit c_{out} and bitvector $s[n-1:0]$.

Excercise 7: (carry save (or 3/2) adder)

Carry Save Adder[4] (or also called "3/2 adder") is a circuit computing the following function:

$$csa : \mathbb{B}^n \times \mathbb{B}^n \times \mathbb{B}^n \to \mathbb{B}^{n+1} \times \mathbb{B}^{n+1}$$
$$csa(a[n-1:0], b[n-1:0], c[n-1:0]) = (s[n:0], t[n:0])$$

with $\langle a[n-1:0] \rangle + \langle b[n-1:0] \rangle + \langle c[n-1:0] \rangle = \langle s[n:0] \rangle + \langle t[n:0] \rangle$. Prove this.
Hint: use the decomposition lemma (exercise 3) in order to transform the sum of three bitvectors into a sum of two bitvectors. Argue that the carry save adder realizes the same transformation.

Excercise 8: (O-notation)

1. Get familiar with the O-notation.
2. Give the definitions of sets $\mathcal{O}(f), o(f), \Omega(f), \omega(f), \Theta(f)$ for some function f.
3. Which of the following statements are true and which are not. Explain!
   (a) $n^2 \cdot log(n) \in o(n \cdot log(n)^3)$
   (b) $\frac{n^2-n}{4} \in \mathcal{O}(n^2)$
   (c) $n^{10!} \in \omega(2^n)$
   (d) $\forall f : \mathcal{O}(f) \cap \Omega(f) = \Theta(f)$

---

[3] Take a look into your lecture notes or into [MP00] (Figure 2.11).
[4] Figure 2.26 in [MP00] or your lecture notes.