Embedded Systems 2010/2011 – Assignment Sheet 7

Due: Tuesday, 14th December 2010, before the lecture (i.e., 10:10)

Please indicate your name, matr. number, email address, and which tutorial you are planning to attend on your submission. We encourage you to collaborate in groups of up to three students. Only one submission per group is necessary. However, in the tutorials every group member must be capable to present each solution.

---

Exercise 1: Schedulability Test (20 pts.)

Given the following periodic, synchronous task-set

<table>
<thead>
<tr>
<th>Task</th>
<th>C_i</th>
<th>D_i</th>
<th>T_i</th>
</tr>
</thead>
<tbody>
<tr>
<td>τ_1</td>
<td>2</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>τ_2</td>
<td>2</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>τ_3</td>
<td>7</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>τ_4</td>
<td>15</td>
<td>121</td>
<td>121</td>
</tr>
</tbody>
</table>

a) Give a tight upper bound on the number of time units, for which we have to simulate in order to proof if the task-set is schedulable or not.

b) Is this task-set schedulable with EDF? Justify your answer.

c) Use the response-time analysis as presented in lecture to determine whether or not the task-set is schedulable with RM scheduling.

Exercise 2: VHDL (30 pts.)

Write a behavioral and a structural architecture implementing a 4-bits adder, as well a behavioral and a structural architecture implementing a saturating 4-bits adder. Start with the following declaration.

```
entity adder is
    port(a3, a2, a1, a0, b3, b2, b1, b0: in bit; c3, c2, c1, c0, d: out);
end adder;
```

Let

\[ a = 8a_3 + 4a_2 + 2a_1 + a_0, \]
\[ b = 8b_3 + 4b_2 + 2b_1 + b_0, \]
\[ c = 8c_3 + 4c_2 + 2c_1 + c_0. \]
Denote with $+_{16}$ addition modulo 16. Denote with $+_{16}^{s}$ saturating addition modulo 16, that is:

$$a +_{16}^{s} b = \begin{cases} 
    a + b, & \text{if } a + b \leq 15 \\
    15, & \text{otherwise}.
\end{cases}$$

You must ensure that:

- $c = a +_{16} b$, for the 4-bits adder;
- $c = a +_{16}^{s} b$, for the saturating 4-bits adder;
- in all cases: $d = 1$ iff there was an overflow, i.e., $a + b > 15$.

When writing the structural architectures, you can use full adders, half adders, and gates as your starting subentities.