

lea.schoenberger [☺] tu-dortmund.de
 nils.hoelscher [☺] tu-dortmund.de
 nick.pietrass [☺] tu-dortmund.de
 jan.pomplun [☺] tu-dortmund.de

Exercises for
 Embedded Systems
 Wintersemester 18/19

Exercise Sheet 2 (Theory)

(11 Points)

Please note: Solutions must be submitted (individually or in pairs) until 22.10.2018 at 10:00 AM (post box in OH16, ground floor, in front of room E16). Submitting solutions via mail is *not* possible. Discussion: 24-26.10.2018.

1 Specification and Modeling Languages (2 Points)

Name at least two requirements for specification and modeling languages for embedded systems.

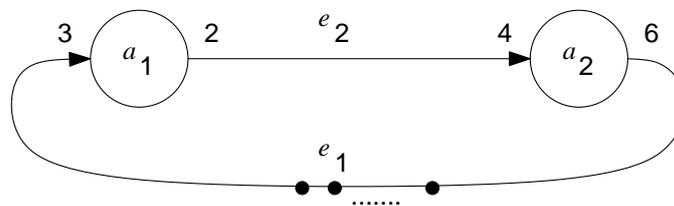
2 StateCharts (3 Points)

Mr. Smart developed a beer brewing machine, which offers multiple options for configuration. It can produce either beer (standard setting) or Radler. Beer can taste either hoppy (standard setting) or malty. The taste configuration can be changed by pressing the *taste button*. Moreover, three options regarding the alcohol content are available, namely, less than 4%, 4-5% (standard setting) or more than 5%, which can be selected by pressing the *alcohol button* (press once for more than 5%, twice for less than 4%, thrice for 4-5% etc.). If the machine is switched off and one again, the previous alcohol content setting is restored. For beer as well as for Radler, the bottle size must be chosen, i.e., big (standard setting) or small bottles, which can be switched by pressing the *bottle button*.

Depict the behavior of the beer brewing machine as a StateChart!

3 SDF (2 Points)

Suppose that the following SDF model is given:



Suppose that SDF actors execute in 1 unit of time. Generate a possible execution schedule, assuming a single execution unit (no parallel execution) and an initial set of 6 tokens for edge e_1 :

Time	Tokens on edges		Next actor execution
	e_1	e_2	
0	6	0	
1			
2			
3			
4			

Now, assume an initial set of 9 tokens for edge e_1 . Provide an example of parallel execution, i.e., an example for which both actors are active. Enter the names of the concurrently executing actors into the right column!

Time	Tokens on edges		Next actor execution(s)
	e_1	e_2	
0	9	0	a_1, a_2 or $(a_1$ and $a_2)$
1			
2			
3			
4			

4 KPN (4 Points)

Fibonacci numbers are defined by:
$$F(n) = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ F(n-2) + F(n-1) & \text{if } n > 1 \end{cases}$$

Create the following processes primitives:

- **Process Init1(input A, output B):** At the start, it sends just once the integer value “1” on its output channel. Afterwards, it executes in an infinite loop: Read one value from the input channel and put it on the output channel.
- **Process Init0(input A, output B):** At the start, it sends just once the integer value “0” on its output channel. Afterwards, it has the same behavior like process init1.
- **Process Dup(input A, output B, C):** It executes in an infinite loop: Read one value from the input channel and put the value on both output channels (remove “2x”).
- **Process Add(input A, B, output C):** It executes in an infinite loop: Read one value from each input channel. Add the two values. Put the result on its output channel.
- **Process Sink(input A):** In an infinite loop, this process reads one value from the input channel per cycle

Develop a process network with only these components to produce the sequence of Fibonacci numbers.