Prof. Dr. Christian Steger   Sebastian Altmeyer, M.Sc.
Prof. Dr. Reinhard Wilhelm   Michael Gerke, M.Sc.
                                                          Dipl.-Inf. Hans-Jörg Peter

Embedded Systems 2010/2011 – Assignment Sheet 11

Due: Tuesday, 8th February 2011, *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: Design Space Exploration   (40+10 pts.)

A manufacturer is planning to develop a new monitoring device that should be capable of storing data over time. Naturally, the production costs should be low while the memory capacity should be high at the same time. The following memory chips are available on the market:

Chip i	Type	Energy consumption e_i [mW]	Price p_i [USD]	Memory m_i [MB]
1	A	70	2	16
2	A	40	4	16
3	A	80	6	32
4	B	150	20	64
5	B	100	24	64
6	B	260	40	128

The chosen system architecture implies the following constraints on the choice of the chips:

- There are exactly two type A sockets and two type B sockets available. Sockets can also be unpopulated.

- The total energy consumption should not be higher than 320 mW.

- At least a total capacity of 140 MB is required.

The marketing department established a selling price of 2 USD per 1 MB. In this context, perform the following tasks:

(a) Encode the given optimization problem as an integer linear programming problem. (20 pts.)

(b) Find the optimal solution.   (20 pts.)

   You can obtain 10 bonus points when you use the open source tool `lp_solve`. In this case, also provide a printout of your input file in the *lp-format*.

Exercise 2: Modeling with Timed Automata (20 pts.)

For each of the following scenarios, provide a small timed automaton.

(a) A certain operation brings a machine from a state A into another state B. Nondeterministically, it may take 10 to 20 time units. (5 pts.)

(b) A register circuit tries to sample a value from the bus on the event *tick*. A change of the value on the bus is indicated by the event *change*. Let t be the (unknown) point in time at which the event *tick* occurs. The behavior of the register is as follows:

   - Iff there is **no** change of the value on the bus in the time interval $[t - setup, t + hold]$, the value is successfully sampled (and the timed automaton is in the final state *success*).

   - Iff there is a change of the value on the bus in the time interval $[t - setup, t + hold]$, the automaton does nondeterministically end up either in the final state *error* (the wrong value was sampled) or in the final state *success* (the right value was sampled).

   In the scenario, there is only one *tick* event, but there can be arbitrarily many *change* events. (5 pts.)

(c) A train runs over a track with constant speed $v \in \mathbb{Q}$ (unit: meters per second). After $b \in \mathbb{Q}$ meters, a braking maneuver should be initiated. (5 pts.)

(d) The timing of two asynchronous ECUs can drift up to 0.15% per clock pulse. (5 pts.)

Exercise 3: Timed Automata Analysis (20 pts.)

Consider the network of timed automata in Figure 1.

(a) Construct the product timed automaton of the network. (10 pts.)

(b) Draw the region graph of the product timed automaton. (10 pts.)

Exercise 4: Timed Automata Simplification (20 pts.)

Consider the timed automaton in Figure 2 with the clocks x and y, and the events a, b, c, d, e, f, g, h, i, j, and k.
Provide a simplified timed automaton by identifying and removing redundant clock constraints and clock resets. After your simplifications, the timed behavior (the events with their time of occurrence) should remain exactly the same.
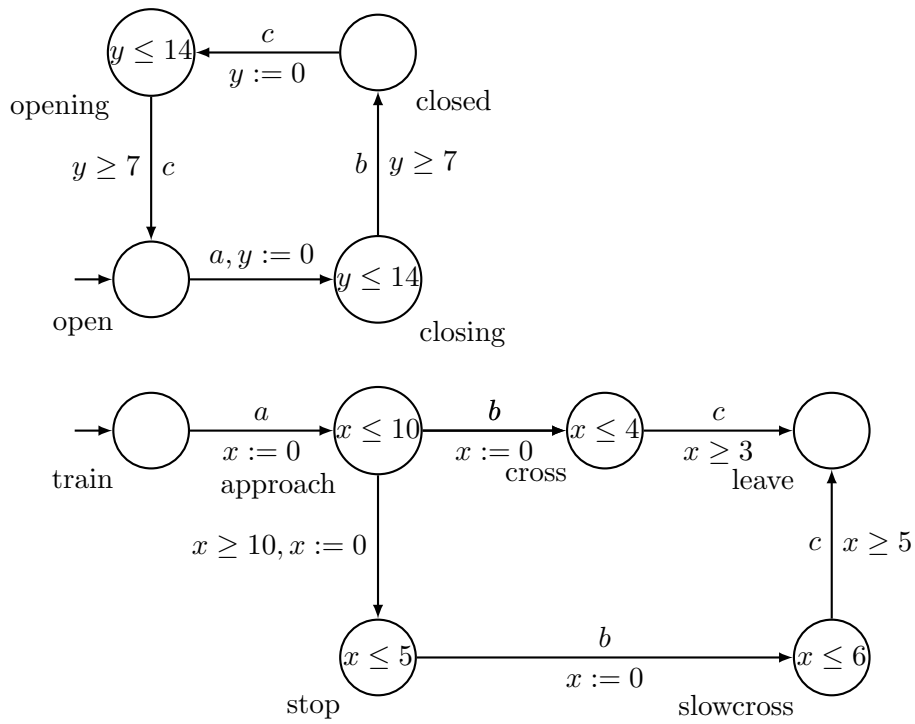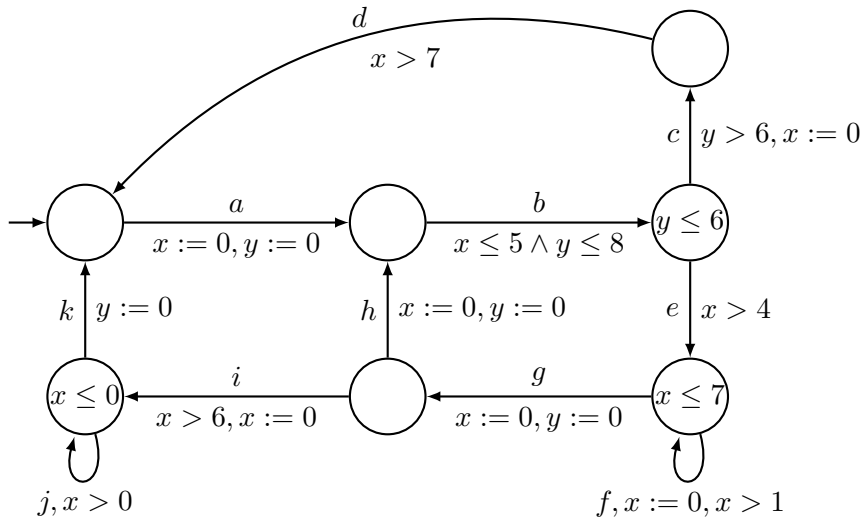
Figure 1: Network of timed automata for Exercise 3.

Figure 2: Timed automaton for Exercise 4.