Prof. Bernd Finkbeiner
Dr. Calogero Zarba
Moritz Hahn

Midterm

Embedded Systems

Problem 1 (Embedded Systems) 5 points

(a) Give a concise definition of the term “embedded system.”

(b) List at least 10 products that contain embedded systems.

(c) List at least 3 characteristic requirements of embedded systems.

Problem 2 (Statecharts) 15 points

Draw a statechart modelling a chess clock. Your statechart should include the following events:

- `tick`: an internal event generated once per second.
- `white_moves`: an external event indicating that player White’s button of the clock is pressed.
- `black_moves`: an external event indicating that player Black’s button of the clock is pressed.
- `white_wins`: an internal event generated when White wins (because Black runs out of time).
- `black_wins`: an internal event generated when Black wins (because White runs out of time).

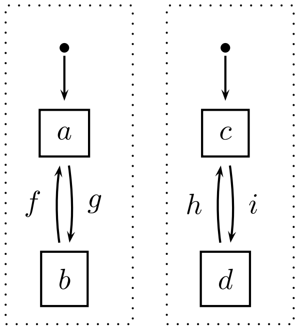
At each instant of time, exactly one of the two players is *active*. Initially, player White is active. For each player, the clock maintains two time variables: *grace*, which is initially set to 30 seconds, and *pool*, which is initially set to 5 minutes. As long as there is time left in the active player’s *grace*, one second is deducted from *grace* for every `tick`; once the grace time has expired, one second is deducted from *pool* for every `tick`. If the pool time expires, the active player loses the game. When the active player’s button is pressed, the *grace* is reset to 30 seconds, the *pool* is incremented by 12 seconds, and the other player becomes active.

Problem 3 (Statecharts) 5 points

Draw a statechart that exhibits a potential infinite superstep when using Harel’s classic STATE-MATE semantics and the asynchronous time model.

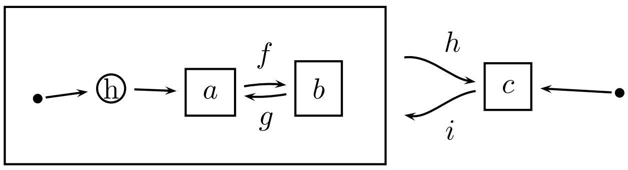
Problem 4 (Statecharts) **10 points**

Draw a statechart equivalent to the one shown below, but that uses no concurrency. For simplicity, assume that at most one event can occur at each unit of time.



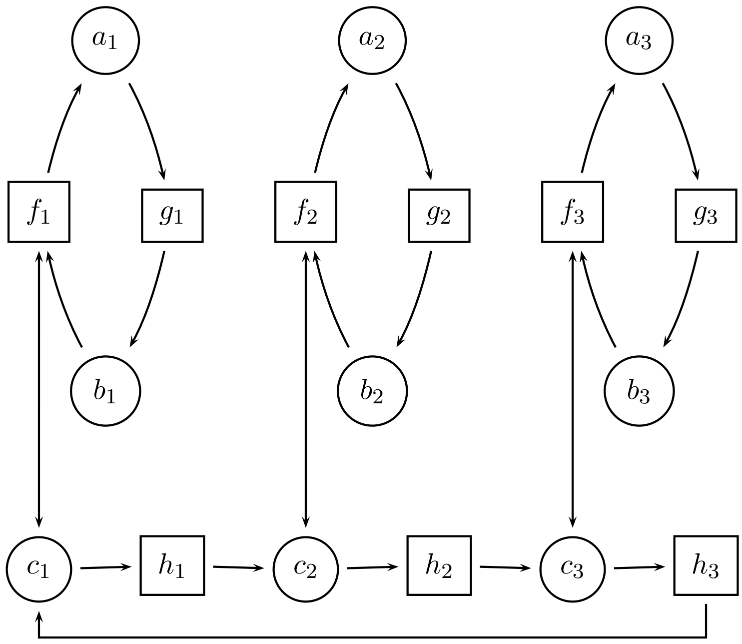
Problem 5 (Statecharts) **10 points**

Draw a statechart equivalent to the one shown below, but that uses no hierarchy and history mechanism. For simplicity, assume that at most one event can occur at each unit of time.



Problem 6 (Petri nets) 10 points

The Petri net shown below models a simultaneous chess exhibition where a Grandmaster plays with 3 amateurs.



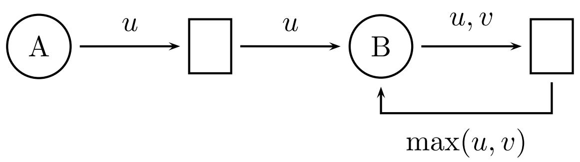
The intuitive meaning of the places and transitions of the net is as follows:

- a_i: Amateur i has the right to move at table i;
- b_i: Grandmaster has the right to move at table i;
- c_i: Grandmaster stands in front of table i;
- f_i: Grandmaster moves at table i;
- g_i: Amateur i moves at table i;
- h_i: Grandmaster shifts from table i to the next.

State at least four place invariants of the net. For each invariant, briefly explain its intuitive meaning.

Problem 7 (Petri nets) 20 points

Let $S = \langle x_1, \ldots, x_n \rangle$ be a multiset of natural numbers. In the colored Petri net shown below, place A is initially marked with n tokens labeled by the natural numbers in S. After a finite number of transitions, the net enters enters a deadlock.



(a) When the deadlock occurs, place B contains exactly one token. What number is that token labeled with?

(b) Modify the net so that the token is labeled with $\min(S)$.

(c) Modify the net so that the token is labeled with $\text{sum}(S) = x_1 + \ldots + x_n$.

(d) Modify the net so that the token is labeled with $\text{avg}(S) = \text{sum}(S)/n$. For this modification, you are allowed to assign priorities to the transitions of the net. When more than one transition is activated, assume that only the transition with the highest priority can fire.

Problem 8 (Kahn process networks) 15 points

Specify a Kahn process network that generates the following sequence of natural numbers:

$$
\begin{aligned}
f(0) &= 1\,, \\
f(1) &= 1\,, \\
f(n) &= 2f(n-1) + nf(n-2)\,, && \text{for } n \geq 2\,.
\end{aligned}
$$

Draw the process network and give C-like code for each process. Use atomic processes that can add two numbers, multiply numbers, increment a number, or duplicate a number. You may also use processes that initially generate a constant, and simply forward their input afterwards. Finally, the sequence f should be sent to a dedicated process that acts as a sink.

Problem 9 (SDF networks) **5 points**

Determine whether the SDF network below is schedulable. If yes, compute a feasible schedule using Lee/Messerschmitt scheduling algorithm.