

Modellierung und Analyse eingebetteter und verteilter Systeme

Übungsblatt 2

Ausgabe: 13. Oktober, Abgabe: 20. Oktober

Hinweis: Die Besprechung findet am 24.10.2014 statt.

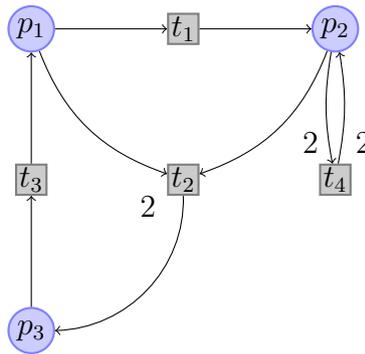
Aufgabe 2.1 (6 Punkte) Erweiterter Mealy-Automat

- Zeichnen Sie den durch die Tabelle gegebenen Mealy-Automaten.
- Stellen Sie auch den zugehörigen erweiterten Mealy-Automaten graphisch dar.

Zustand	Eingabe	Ausgabe	Folgezustand
q_0	s_0	—	—
q_0	s_1	—	q_1
q_0	s_2	—	q_2
q_0	r_v	—	r_0
q_1	s_0	—	q_0
q_1	s_1	—	—
q_1	s_2	—	q_2
q_1	r_v	—	r_1
q_2	s_0	—	q_0
q_2	s_1	—	q_1
q_2	s_2	—	—
q_2	r_v	—	r_2
r_0	—	0	q_0
r_1	—	1	q_1
r_2	—	2	q_2

Aufgabe 2.2 (6 Punkte) Petri-Netze: Invarianten

Gegeben sei folgendes Petri-Netz:



- Stellen Sie die Inzidenzmatrix C des Petri-Netzes dar. Beachten Sie, dass t_2 zwei Tokens erzeugt und t_4 jeweils Tokens erzeugt und konsumiert.
- Zeigen Sie, dass die Markierung $m(p_1) = 1$, $m(p_2) = 1$ und $m(p_3) = 1$ eine Stelleninvariante des Petri-Netzes ist.
- Fünf Philosophen leben gemeinsam unter einem Dach und speisen gemeinsam an einem runden Tisch, an dem jeder seinen festen Platz mit eigenem Teller hat. Im Haushalt der Philosophen sind nur 5 Gabeln g_1, \dots, g_5 vorhanden. Zwischen zwei benachbarten Tellern liegt genau eine Gabel g_i . Das einzige Problem der Philosophen - außer dem der Philosophie - ist, dass ihnen ein kompliziertes Spaghetti-Gericht serviert wird. Dazu benötigt jeder Philosoph genau 2 Gabeln, so dass keiner der direkt benachbarten Philosophen gleichzeitig speisen kann.

Das Leben eines Philosophen kann als alternierende Folge von Aktionen beschrieben werden: *hungrig werden*, *essen*, *denken*. Wenn ein Philosoph *hungrig* wird, greift er zuerst nach seiner linken Gabel und dann nach seiner rechten Gabel. Sind beide Gabeln frei, fängt ein Philosoph mit der Verspeisung an. Nach einer endlichen Zeit wird ein Philosoph satt, dann legt er die beiden Gabeln zurück und fängt an zu *denken*. Modellieren Sie das *Five Dining Philosophers* Problem (s. [1]) mit Hilfe eines Petri-Netzes.

Literatur

- [1] Dijkstra, E.W., *Hierarchical ordering of sequential processes*, Acta Informatica, Vol.2, pp.115-138, 1971.