

## Software Engineering in der Praxis

### Aufgabenblatt 4: Zeitbehaftete Petri-Netze

#### Aufgabe 1: Herzschlag

Ein »cold-standby« Backupsystem besteht aus zwei Rechnern. Während der erste Rechner im normalen Betrieb ist, überwacht ein zweiter Rechner den ersten über sogenannte »heartbeats«, d. h. er sendet regelmäßig eine Anfrage an den ersten und falls innerhalb einer vorgegebenen Zeit keine Antwort eintrifft, erklärt der Backuprechner den Hauptrechner für tot und übernimmt dessen Aufgaben.

Der Hauptrechner kann drei Zustände einnehmen: *ohne Aufträge*, *arbeitend* oder *ausgefallen*, wobei er jederzeit ausfallen kann. Eine »heartbeat«-Anfrage kann er nur beantworten wenn er gerade nicht mit einem Auftrag beschäftigt ist jedoch muss eine Anfrage vor einem neuen Auftrag abgearbeitet werden. Jeder Arbeitsauftrag dauert mindestens 3 Zeiteinheiten (ZE).

Anforderungen:

1. Der Backuprechner darf die Arbeit nur übernehmen, wenn der andere Rechner nicht mehr im Betrieb ist.
2. Zwischen Ausfall des Hauptrechners und Übernahme durch das Backup-System dürfen nicht mehr als 8 ZE liegen.

Der Backuprechner sendet alle 2 ZE (nach Empfang der letzten heartbeat-Antwort) eine heartbeat-Anfrage.

#### Teilaufgaben

- a) Modellieren Sie das System mit dem TPN-Werkzeug *TINA/LAAS*!
- b) Wie müssen die übrigen Zeitintervalle (insb. die maximale Dauer eines Auftrages) gewählt werden, so dass alle Anforderungen erfüllt werden?
- c) Wie können Sie nachweisen, dass Ihre Umsetzung alle Anforderungen erfüllt?

## Aufgabe 2: Ampel

Der Verkehr an einer Kreuzung soll durch eine Ampel geregelt werden. Dabei sollen folgende Anforderungen erfüllt werden:

1. Die *rot/gelb*-Phase jeder Ampel dauert genau 1 Zeiteinheit (ZE), die *grün*-Phase 5 ZE und die *gelb*-Phase wieder 1 ZE.
2. Während eine Ampel einen Farbwechselzyklus durchläuft, muss die andere durchgehend *rot* zeigen.

Ihr Auftrag ist es, die Steuersoftware für die Ampelanlage zu entwerfen.

- a) Modellieren Sie die Zustände beider Ampeln (auf einem TINA-Arbeitsblatt) wobei sich diese lediglich über die »zeitliche Abfolge« synchronisieren!
- b) Welche »Schaltzeit« muss die *rot*-Phase haben, um alle Anforderungen zu erfüllen?
- c) Was geschieht, wenn die Schaltzeiten nicht exakt eingehalten werden können (modellieren Sie diesen Fall; z. B. *grün*: $[5,6]$ )?