

Entwurf und Implementierung paralleler Programme

Aufgabe 1

- (a) Gegeben sei ein Puffer mit beschränkter Kapazität N . Falls der Puffer nicht voll ist, kann mit *put* ein Element in den Puffer gelegt werden. Falls der Puffer nicht leer ist, kann mit *get* ein Element aus dem Puffer geholt werden. Zu Beginn ist der Puffer leer.
- Modellieren Sie den Puffer durch einen parametrisierten FSP-Prozess und geben Sie das zugehörige LTS für den Fall $N = 3$ an. (Bei der Modellierung soll die Art der Pufferelemente nicht berücksichtigt werden.)
- (b) Der Puffer soll von einem Erzeuger (Producer) und einem Verbraucher (Consumer) als Zwischenlager bei der Übergabe von Daten verwendet werden. Modellieren Sie das Erzeuger-Verbraucher System in FSP. Geben Sie ein Struktur-Diagramm dafür an.

Aufgabe 2

Das Erzeuger-Verbraucher System von Aufgabe 1 soll in Java unter Verwendung der Schnittstelle *Runnable* implementiert werden. Es ist die systematische Vorgehensweise zur Implementierung von Synchronisationsbedingungen mit Monitoren aus der Vorlesung anzuwenden. Erstellen Sie zunächst ein Klassendiagramm für das System. In der Implementierung soll der Erzeuger die Kleinbuchstaben des deutschen Alphabets wiederholt produzieren. Der Puffer soll nach dem First-In-First-Out Prinzip arbeiten.