TD4 FDS – Master 2 ISC, 2006/07

Exo 1

Le problème classique des philosophes affamés décrit n philosophes qui alternent entre l'état "refléchir" et l'état "manger". Leur repas a lieu autour d'une table sur laquelle se trouvent n assiettes, avec une fourchette entre chaque deux assiettes voisines.

Pour pouvoir manger, un philosophe doit demander sa fourchette gauche ainsi que sa fourchette droite. Après les avoir obtenues, il peut manger et ensuite retourner à l'état "refléchir".

1. Décrivez des modules RML `Philo` et `Fork` pour décrire le comportement de chaque philosophe et de chaque fourchette.
2. Est-ce que votre solution peut générer des blocages ?

Exo 2

On considère le protocole du bit alterné, qui gère la communication entre 2 agents `Sender` et `Receiver`, qui s'échangent des messages en utilisant des canaux avec pertes. On suppose qu'il y a un canal de transmission `Trans` de `Sender` vers `Receiver`, ainsi qu'un canal d'acquittements `Ack` de `Receiver` vers `Sender`.

`Sender` se comporte de la manière suivante :

a) Il accepte un nouveau message m et l'envoie sur `Trans`, taggé par le bit $b \in \{0, 1\}$. En même temps il démarre une horloge `Timer`.

b) – S'il reçoit un *timeout* de `Timer`, alors il renvoie le même message taggé par b sur `Trans`.
   – S'il reçoit l'acquittement b sur `Ack`, alors il est de nouveau prêt à recevoir un message (qu'il va envoyer sur `Trans` taggé par $1 - b$).
   – S'il reçoit l'acquittement $1 - b$ sur `Ack`, il fait rien.

Le comportement de `Receiver` est dual :

a) Lorsqu'il reçoit un message m sur `Trans`, taggé par b, il répond en envoyant b sur `Ack`, et démarre `Timer`.

b) – S'il reçoit un *timeout* de `Timer`, alors il renvoie le même acquittement b sur `Ack`.
   – S'il reçoit un (nouveau) message taggé par $1 - b$ sur `Trans`, alors il délivre le message m (et confirme le nouveau message en envoyant $1 - b$ sur `Ack`).
   – S'il reçoit un message taggé par b sur `Trans`, il fait rien.

1. Définisez les modules RML `Sender`, `Receiver` et `Timer`.
2. Décrivez une modélisation du protocole par un système de transitions fini.