
TD8 Automates

Exercice 1 : Automates à deux états

On s'intéresse aux protocoles de liaison implémentés lors du TD4, plus précisément à la version 2.

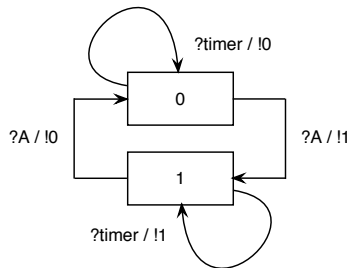


FIG. 1 – Comportement de l'émetteur

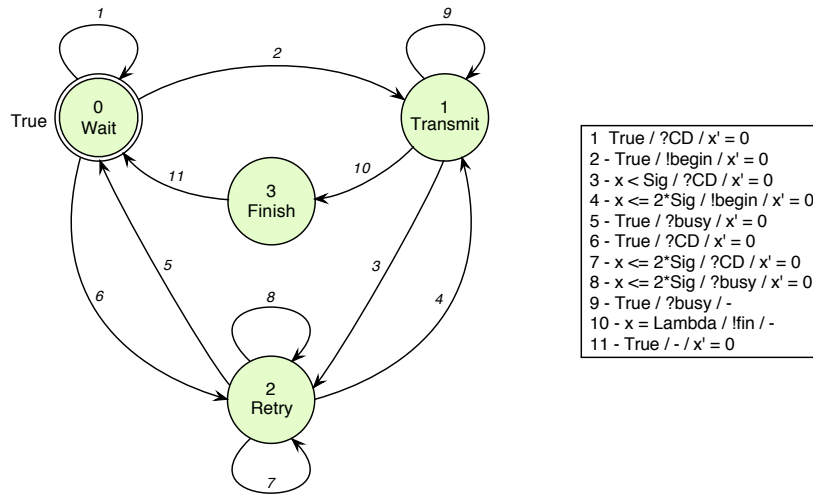
1. Quels sont les types de messages qui transitent sur le canal (bidirectionnel) ?
2. Une simplification consiste à considérer que les messages forment un ensemble fini. Que peut-on considérer comme ensemble fini de messages ? Appelons M cet ensemble.
3. Peut-on représenter le comportement du canal par un automate à états finis sur l'alphabet M ?
4. On suppose que le canal est borné (il contient au plus n messages). Pour $n = 1$, donner une modélisation (sous forme d'automate à états finis) d'un canal fiable (en considérant les messages de M).
5. Le canal est maintenant bruité (non fiable). On considère uniquement les pertes : pour $n = 1$, donner une modélisation (sous forme d'automate à états finis) d'un canal avec pertes (en considérant les messages de M).
6. L'automate de la figure 1 représente le comportement de l'émetteur :
 - $?m$ (resp. $!m$) désigne la réception (resp. émission) du message ou événement m .
 - $?m!n$ signifie « après réception du message/événement m , envoyer n ».Quelles sont les simplifications faites dans cette modélisation ?
7. Donner une représentation du récepteur sous forme d'automate.
8. Déduire des questions précédentes une représentation du système global. A-t-on des situations de blocage ? A-t-on des doublons ? Cela ne fait pas apparaître les situations d'erreur rencontrées lors des expérimentations faites avec ce protocole (TD4, version v2). Expliquer. Dans cette modélisation, quelle simplification masque ces situations d'erreur ?

Exercice 2 : CSMA/CD avec des automates temporisés

La modélisation suivante est basée sur le modèle des automates temporisés. Le système est composé de deux *Sender* (2(a)) et un *Bus* (2(b)). λ et Sig représentent respectivement la durée d'émission d'une trame et le temps maximum de propagation entre deux stations du réseau. x et y sont des horloges. L'état 0 est initial et les transitions sont de la forme : garde/événement/action. Par exemple, la transition 3 du *Sender* :

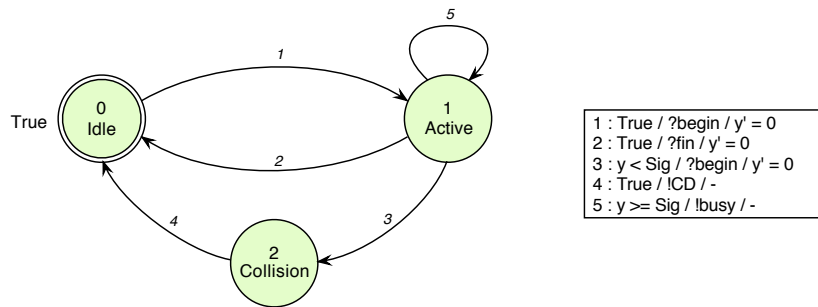
- est gardée par l'expression booléenne $x < Sig$
- correspond à l'événement réception de CD
- effectue l'action $x' = 0$ (la nouvelle valeur de l'horloge x est 0).

La synchronisation des événements est de type *Rendez-vous* : en même temps ont lieu les événements $!begin$ et $?begin$.



- 1 - True / ?CD / x' = 0
- 2 - True / !begin / x' = 0
- 3 - x < Sig / ?CD / x' = 0
- 4 - x <= 2*Sig / !begin / x' = 0
- 5 - True / ?busy / x' = 0
- 6 - True / ?CD / x' = 0
- 7 - x <= 2*Sig / ?CD / x' = 0
- 8 - x <= 2*Sig / ?busy / x' = 0
- 9 - True / ?busy / -
- 10 - x = Lambda / !fin / -
- 11 - True / - / x' = 0

(a) Sender



- 1 : True / ?begin / y' = 0
- 2 : True / ?fin / y' = 0
- 3 : y < Sig / ?begin / y' = 0
- 4 : True / !CD / -
- 5 : y >= Sig / !busy / -

(b) Bus

1. Donner les simplifications effectuées sur la modélisation proposée.
2. Donner un scénario avec 2 *Sender* qui émettent chacun un message sans collision.
3. Donner un scénario avec 2 *Sender* qui émettent chacun leur message après avoir provoqué une collision.