

# Examen temps réel IN2

## Partie langage pour le temps réel

6 juin 2014

Tous les documents sont autorisés.

**Exercice 1 (10 points)** *L'objectif de cet exercice est d'écrire le protocole du bit alterné vu en cours dans le langage LUSTRE puis avec des automates temporisés. Programme LUSTRE de l'émetteur :*

```
node emetteur (const n: int; ack, b_ack : bool)
returns (m, b_m: bool)
var cpt, inter: int;
let
  cpt = 0 -> if (ack and b_ack=pre b_m) then 0 else pre cpt +1;
  inter = 1 -> ((cpt +1) mod n);
  m = ((cpt=0) or (inter=0));
  b_m = true -> if (cpt=0) then not pre b_m else pre b_m;
tel
```

*L'émetteur envoie le message  $m$  (lorsque  $m = true$ ) contenant le booléen  $b_m$ . Il reçoit le message  $ack$  (lorsque  $ack = true$ ) contenant le booléen  $b_{ack}$ . Si le booléen d'acquiescement est celui attendu, il émet le message suivant en changeant la valeur du booléen  $b_m$ . Si non, il ignore le message d'acquiescement. Si au bout de  $n$  unités de temps, il n'a pas reçu l'acquiescement correct, il réémet le message  $(m, b_m)$ .*

1. Donner le comportement du nœud pour la séquence d'entrée  $(3, f, f)(3, f, f)(3, f, f)(3, f, f)(3, t, t)(3, f, f)(3, f, f)(3, t, f)$ .
2. Ecrire le code  $C$  généré.
3. Ecrire le nœud LUSTRE pour le récepteur. Ce dernier renvoie immédiatement un  $ack$  à réception d'un message.  
`node recepteur (m, b_m: bool) returns (ack, b_ack : bool)`
4. Ecrire le système complet connectant l'émetteur et le récepteur via un médium de communication qui met 2 unités de temps pour transmettre chaque message.
5. Ecrire 3 automates temporisés : un pour l'émetteur, un pour le médium et un pour le récepteur. Détailler les synchronisations s'il y en a ainsi que les variables globales.