Réseaux et protocoles
Maitrise d'informatique ©

REPR
Tp 4
99/00

Synchronisation

.

But

Nous vous proposons de réaliser une application qui nous permette de prendre en compte l'ensemble des problèmes soulevés lors des trois TP précédents.

1 Présentation

On désire programmer, deux fonctions `emi_flot_connect` et `rec_flot_connect` à ajouter fichier *fonct.c* (accessible dans le répertoire habituel des TP REPR) afin de mettre en œuvre une transmission de caractères avec contrôle de flux et synchronisation au démarrage (gestion de connexion).

Rôle joué par les signaux `DTR` et `RTS`

- `DTR` est utilisé pour la synchronisation de début et pour la fermeture de connexion.
- `RTS` sert à acquitter le caractère précédent. Il n'a pas pour rôle de synchroniser l'émetteur sur le récepteur au démarrage comme vu lors du dernier TP.

2 Automate

Question 1 *A-t-on besoin d'un test sur* `DE` *dans* `emi_flot_connect` *? Discutez de sa nécessité.*

Question 2 *Décrivez l'automate associé aux procédures* `emi_flot_connect` *et* `rec_flot_connect`*. Il doit modéliser l'ouverture de connexion, la fermeture de connexion ainsi que le contrôle de flux. Lors du démarrage, on devrait pouvoir lancer les deux procédures constituant l'application dans n'importe quel ordre.*

3 Réalisation

Question 3 *Programmez l'automate du protocole. On fera une programmation par attente active en se basant explicitement sur l'automate obtenu à la question précédente.*

Question 4 *Quelles modifications faut-il apporter à l'automate et au code pour prendre en compte les erreurs de transmission (parité, congestion et format) ?*