

## 1 Protocole de transport simple : STOP-AND-WAIT

**Exercice 1** Soit un protocole de transport de données simple de type STOP-AND-WAIT qui est implanté selon l'architecture suivante (cf Figure 1) :

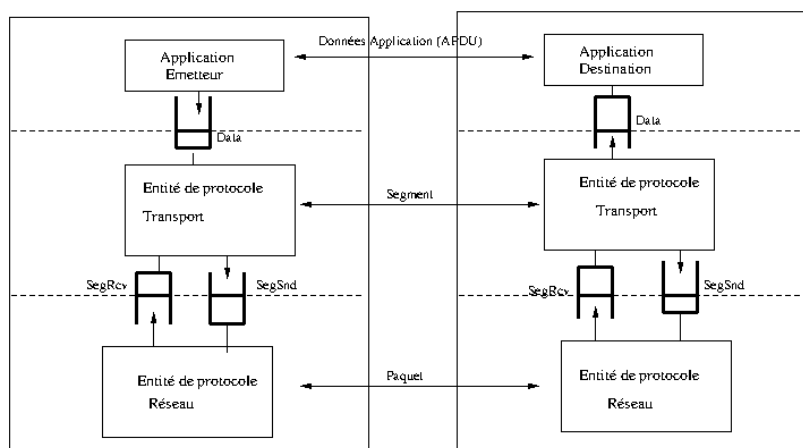


FIGURE 1 – Architecture

- Le protocole transport est considéré comme unidirectionnel du point de vue des données applicatives et le protocole réseau est de type *best effort*.
- Les différentes PDU sont les suivantes :

**Data** : donnée de l'application

**SegRcv** : segment reçu par la couche transport

**SegSnd** : segment émis par la couche transport

- Les événements reçus par la couche transport :

**transmit(Data)** : demande de transmission d'une donnée par l'application

**recv(SegRcv)** : réception d'un segment par la couche réseau

- Les actions réalisées par la couche transport :

**deliver(Data)** : remise de la donnée à l'application

**send(SegSnd)** : émission d'un segment vers la couche réseau

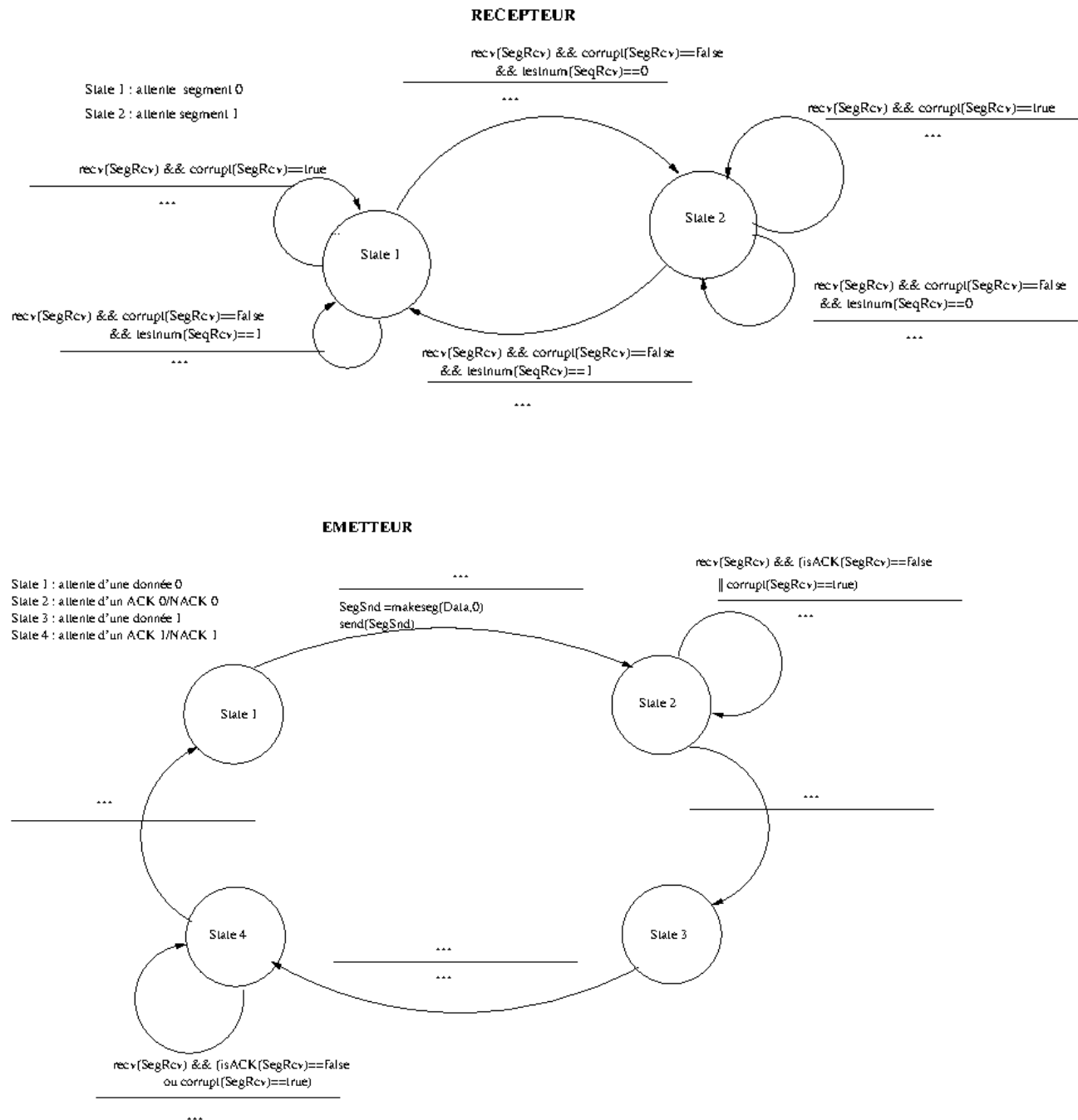
Ce protocole stop-and-wait utilise deux types d'acquittements :

- acquittement positif qui permet d'acquitter un segment non corrompu ;
- acquittement négatif qui permet de signaler que le segment reçu est corrompu et donc de demander sa retransmission.

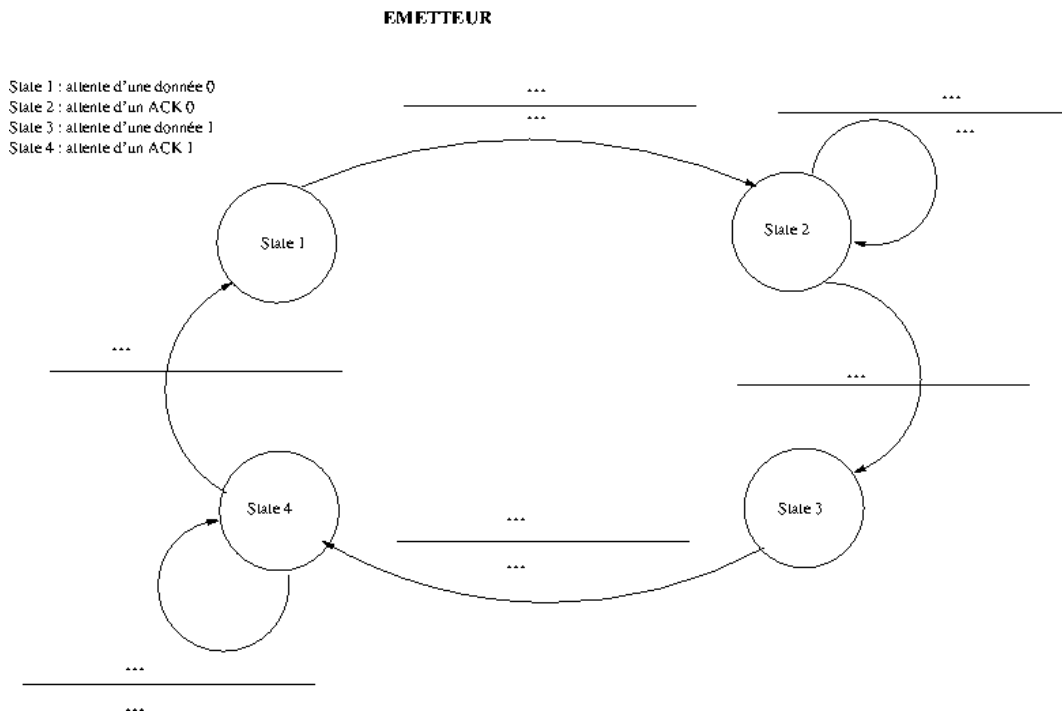
Vous disposez des fonctions suivantes :

- **makeseg(Buffer, [NumSeg])** : retourne le segment **SegSnd** à transmettre. Buffer peut être de type NACK, ACK ou Data
- **isACK(SegRcv)** : retourne un booléen et détermine si le segment reçu est un ACK
- **corrupt(SegRcv)** : retourne un booléen et détermine si le segment reçu est corrompu
- **extract(SegRcv)** : retourne la donnée applicative du segment
- **testnum(SegRcv)** : retourne le numéro du segment reçu

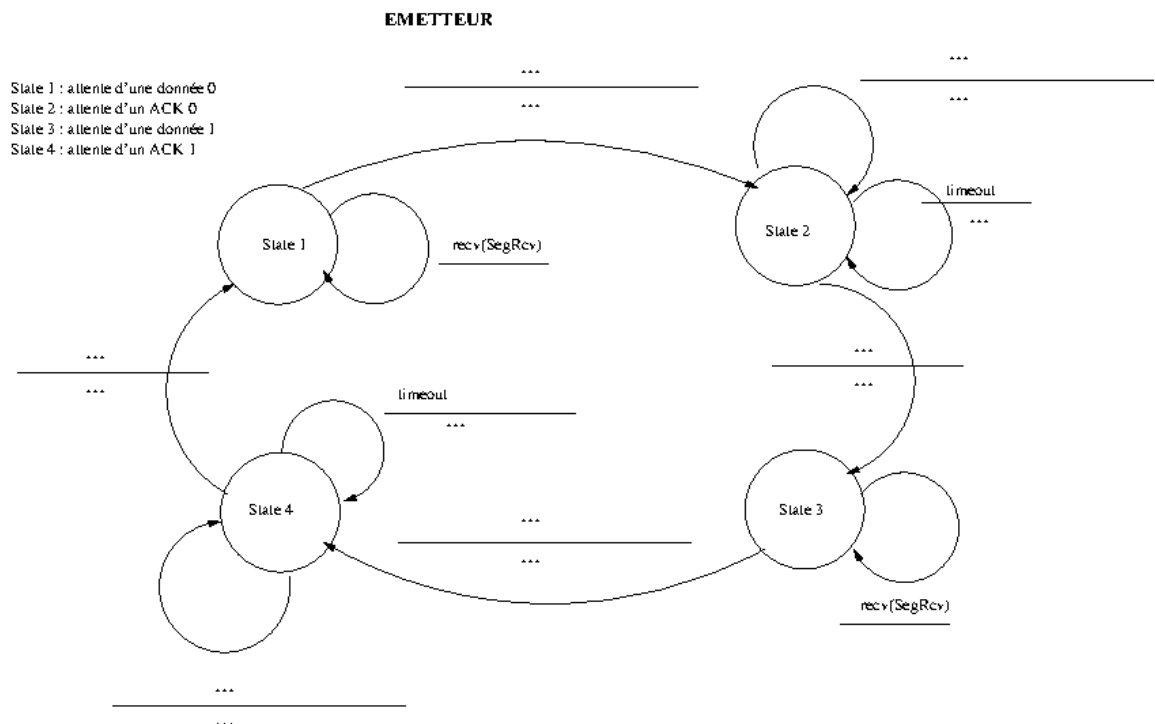
1. On considère dans un premier temps que le canal de transmission peut corrompre les données transmises et les acquittements, mais qu'il ne peut pas perdre ou détruire des données.



2. Modifier l'automate précédent de l'émetteur en supprimant le NACK. Comment dans ce cas, l'émetteur peut savoir que la donnée est corrompue ?



3. Modifier l'automate précédent de l'émetteur en prenant en compte le fait que les données et les acquittements peuvent se perdre.



**Exercice 2** Dans le cas du protocole STOP-AND-WAIT, déterminer le taux d'utilisation de la liaison. On définit ce taux comme étant le ratio du temps de transmission sur le temps total pour émettre la donnée et obtenir un acquittement.

1. Donner une formule générique du taux d'utilisation  $U$ , en fonction de  $a$ ;  $a = T_{prop}/T_{trans}$ . Soit,  $T_{trans}$ , le temps de transmission de la donnée et  $T_{prop}$ , le temps de propagation du signal. On négligera le temps d'attente dans les buffers et les files d'attente des routeurs. La taille de l'acquittement est supposée faible.
2. Soit une liaison satellite géostationnaire, une taille de données de 4000 bits et un débit de 1 Mbit/s. On considère une vitesse de propagation des signaux d'environ 300 000 km/s. L'émetteur et le récepteur sont terrestres et situés dans la zone de couverture du satellite. Rappeler l'altitude d'un satellite géostationnaire, puis déterminer le taux d'utilisation de la liaison.
3. Soit une liaison câble coaxial de 10km et un taux de transmission de 1Mbit/s. On considère que la vitesse de propagation des signaux est d'environ 200 000 km/s dans les câbles. La taille de la donnée est toujours de 4000 bits. Déterminer le taux d'utilisation de la liaison.
4. Qu'en déduisez-vous ?

## 2 Protocoles UDP/TCP

**Exercice 3** Quelles caractéristiques sont vraies pour UDP ?

- a) Contrôle de flux
- b) Sans connexion
- c) Démultiplexage
- d) Ordre des paquets

**Exercice 4** Parmi les éléments suivants, quels sont ceux qui sont inclus dans un en-tête TCP mais pas dans un en-tête UDP ?

- a) port d'origine
- b) numéro de séquence
- c) taille des fenêtres
- d) port de destination
- e) numéro d'accusé de réception

**Exercice 5** Quelle est la spécificité des numéros de séquence en TCP ?