

Rapport TP2 Téléinformatique

Paul BOULESTEIX 20198839

Laurent POLZIN 20345581

1-Format de trame et choix de conception :

Notre protocole adopte un format inspiré de HDLC. Chaque trame est construite selon :

1-

FLAG | Header | Données | CRC | FLAG

FLAG : 0x7E, soit le motif binaire 01111110, un de début, un de fin

Header : trois octets (commande,num_seq,lg_info):

- commande (1 octet) indique le type de trame (donnée, ACK, fermeture, etc.) ;
- num_seq (1 octet) contient le numéro de séquence de la trame ;
- lg_info (1 octet) donne la longueur des données utiles en octets.

Données : segment du fichier, max 100 octets.

CRC : CRC-16 CCITT (2 octets).

2-Mecanisme de contrôle : Go back N

Raisons du choix :

-Plus simple à étendre à un canal non fiable (pertes + erreurs).

-Plus facile à synchroniser avec les séquenceurs modulo N.

3- Choix de CRC

Nous avons choisi CRC-16 CCITT (polynôme 0x1021) car :

-il est standard dans les protocoles HDLC,

-il est plus léger que CRC-32 et suffisant pour détecter les erreurs simulées (bit-flip aléatoire),

-il s'intègre bien dans des trames petites

4- Bit stuffing :

L'algorithme est implémenté dans stuffing() et destuff() dans crc.py .

Pour le bit stuffing, on a suivi le « apres 5 « 1 » un « 0 » est inséré, le stuffing est effectué sur

Header + Données + CRC,

Démonstration (message out/destuff in):

```
message in : 01111101111011111011110  
message out: 0111110101111001111010111100  
destuff in: 011111010111100111110101111100  
destuff out message 011111011110111110111110
```

Résultat attendu après stuffing :

01111101011110011110101111100

Le flux retourné par destuff() est strictement identique au flux d entrée, ce qui valide l'algorithme.

2-Emulation du canal :

SE FIER AU TIMER EMETTEUR PAS RECEPTEUR (celui recepteur se lance quand je lance la commande pas quand l'envoi se lance ou quand il commence a recevoir, pas fiable ici)

protocole_py.py recv output_timeout.txt 0.05 0.10 200

python protocole_py.py send test.txt 0.05 0.10 200 0.1

```
[RECEPTEUR] Trame hors ordre (reçue seq=11, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=12, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=12, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=11, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=11, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=12, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=12, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=11, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=11, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=12, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=12, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=11, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=11, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=12, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=12, attendu=13), ignorée.  
[RECEPTEUR] Reçu CON_CLOSE seq=13, fermeture.  
  
[RECEPTEUR] Terminé.  
Durée totale : 18.630 s  
Trames DATA correctes: 13  
Trames corrompues : 10  
ACK envoyés : 156
```

```
[EMETTEUR] ACK reçu, last_good_seq=11
[EMETTEUR] ACK reçu, last_good_seq=12
[EMETTEUR] ENVOI CON_CLOSE seq=13

[EMETTEUR] Terminé.
Durée totale      : 14.769 s
Trames DATA envoyées: 173
Retransmissions   : 160
ACK reçus         : 59
```

3-Implémentaiton du protocole :

Forme demandée:

Transmission terminée.

Frames envoyées : XX

Frames retransmises : YY

ACK reçus : ZZ

Durée totale : TT s

Fait, voir screenshots

4-Tests experimentaux

Scénarios (pour chacun il faut une capturé d'écran selon le modèle du 3 avec Une brève interprétation (3–5 lignes) pour chaque résultat) :

SE FIER AU TIMER EMETTEUR PAS RECEPTEUR (celui recepteur se lance quand je lance la commande pas quand l'envoi se lance ou quand il commence a recevoir, pas fiable ici)

Canal parfait (probErreur=0, probPerte=0) :

```

PS C:\Users\kopel\Desktop\TP2 TELEINFO\Devoir2_Teleinformatique\testnew> python protocole_py.py recv output_parfait.txt 0 0 0
[RECEPTEUR] Écriture dans 'output_parfait.txt'
[RECEPTEUR] Paramètres canal: err=0.0, perte=0.0, delaiMax=0 ms

[RECEPTEUR] Reçu DATA seq=0, attendu=0
[RECEPTEUR] Reçu DATA seq=1, attendu=1
[RECEPTEUR] Reçu DATA seq=2, attendu=2
[RECEPTEUR] Reçu DATA seq=3, attendu=3
[RECEPTEUR] Reçu DATA seq=4, attendu=4
[RECEPTEUR] Reçu DATA seq=5, attendu=5
[RECEPTEUR] Reçu DATA seq=6, attendu=6
[RECEPTEUR] Reçu DATA seq=7, attendu=7
[RECEPTEUR] Reçu DATA seq=8, attendu=8
[RECEPTEUR] Reçu DATA seq=9, attendu=9
[RECEPTEUR] Reçu DATA seq=10, attendu=10
[RECEPTEUR] Reçu DATA seq=11, attendu=11
[RECEPTEUR] Reçu DATA seq=12, attendu=12
[RECEPTEUR] Reçu CON_CLOSE seq=13, fermeture.

[RECEPTEUR] Terminé.
Durée totale      : 8.542 s
Trames DATA correctes: 13
Trames corrompues   : 0
ACK envoyés       : 14

```

```

PS C:\Users\kopel\Desktop\TP2 TELEINFO\Devoir2_Teleinformatique\testnew> python protocole_py.py send test.txt 0 0 0 0.3
[EMETTEUR] Fichier 'test.txt' (13 trames).
[EMETTEUR] Paramètres canal: err=0.0, perte=0.0, delaiMax=0 ms

[EMETTEUR] ENVOI DATA seq=0 (index=0)
[EMETTEUR] ENVOI DATA seq=1 (index=1)
[EMETTEUR] ENVOI DATA seq=2 (index=2)
[EMETTEUR] ENVOI DATA seq=3 (index=3)
[EMETTEUR] ENVOI DATA seq=4 (index=4)
[EMETTEUR] ENVOI DATA seq=5 (index=5)
[EMETTEUR] ENVOI DATA seq=6 (index=6)
[EMETTEUR] ENVOI DATA seq=7 (index=7)
[EMETTEUR] ENVOI DATA seq=8 (index=8)
[EMETTEUR] ENVOI DATA seq=9 (index=9)
[EMETTEUR] ACK reçu, last_good_seq=0
[EMETTEUR] ENVOI DATA seq=10 (index=10)
[EMETTEUR] ACK reçu, last_good_seq=1
[EMETTEUR] ENVOI DATA seq=11 (index=11)
[EMETTEUR] ACK reçu, last_good_seq=2
[EMETTEUR] ENVOI DATA seq=12 (index=12)
[EMETTEUR] ACK reçu, last_good_seq=3
[EMETTEUR] ACK reçu, last_good_seq=4
[EMETTEUR] ACK reçu, last_good_seq=5
[EMETTEUR] ACK reçu, last_good_seq=6
[EMETTEUR] ACK reçu, last_good_seq=7
[EMETTEUR] ACK reçu, last_good_seq=8
[EMETTEUR] ACK reçu, last_good_seq=9
[EMETTEUR] ACK reçu, last_good_seq=10
[EMETTEUR] ACK reçu, last_good_seq=11
[EMETTEUR] ACK reçu, last_good_seq=12
[EMETTEUR] ENVOI CON_CLOSE seq=13

[EMETTEUR] Terminé.
Durée totale      : 0.007 s
Trames DATA envoyées: 13
Retransmissions    : 0
ACK reçus         : 13

```

Rien à dire, fonctionne parfaitement !

Canal bruité (probErreur≈0.05, probPerte≈0.1) :

```
[RECEPTEUR] Trame hors ordre (reçue seq=3, attendu=10), ignorée.  
[RECEPTEUR] Reçu DATA seq=4, attendu=10  
[RECEPTEUR] Trame hors ordre (reçue seq=4, attendu=10), ignorée.  
[RECEPTEUR] Reçu DATA seq=5, attendu=10  
[RECEPTEUR] Trame hors ordre (reçue seq=5, attendu=10), ignorée.  
[RECEPTEUR] Reçu DATA seq=6, attendu=10  
[RECEPTEUR] Trame hors ordre (reçue seq=6, attendu=10), ignorée.  
[RECEPTEUR] Reçu DATA seq=7, attendu=10  
[RECEPTEUR] Trame hors ordre (reçue seq=7, attendu=10), ignorée.  
[RECEPTEUR] Reçu DATA seq=8, attendu=10  
[RECEPTEUR] Trame hors ordre (reçue seq=8, attendu=10), ignorée.  
[RECEPTEUR] Reçu DATA seq=9, attendu=10  
[RECEPTEUR] Trame hors ordre (reçue seq=9, attendu=10), ignorée.  
[RECEPTEUR] Reçu DATA seq=10, attendu=10  
[RECEPTEUR] Reçu DATA seq=2, attendu=11  
[RECEPTEUR] Trame hors ordre (reçue seq=2, attendu=11), ignorée.  
[RECEPTEUR] Reçu DATA seq=3, attendu=11  
[RECEPTEUR] Trame hors ordre (reçue seq=3, attendu=11), ignorée.  
[RECEPTEUR] Reçu DATA seq=4, attendu=11  
[RECEPTEUR] Trame hors ordre (reçue seq=4, attendu=11), ignorée.  
[RECEPTEUR] Trame corrompue/destuff/CRC invalide: CRC invalide : reçu 0x283C, calculé 0x533B  
[RECEPTEUR] Reçu DATA seq=8, attendu=11  
[RECEPTEUR] Trame hors ordre (reçue seq=8, attendu=11), ignorée.  
[RECEPTEUR] Reçu DATA seq=9, attendu=11  
[RECEPTEUR] Trame hors ordre (reçue seq=9, attendu=11), ignorée.  
[RECEPTEUR] Reçu DATA seq=10, attendu=11  
[RECEPTEUR] Trame hors ordre (reçue seq=10, attendu=11), ignorée.  
[RECEPTEUR] Reçu DATA seq=11, attendu=11  
[RECEPTEUR] Reçu DATA seq=12, attendu=12  
[RECEPTEUR] Reçu CON_CLOSE seq=13, fermeture.  
  
[RECEPTEUR] Terminé.  
Durée totale      : 13.931 s  
Trames DATA correctes: 13  
Trames corrompues   : 3  
ACK envoyés        : 81
```

```

[EMETTEUR] ACK reçu, last_good_seq=9
[EMETTEUR] ACK corrompu ou invalide : Flags not found in frame
[EMETTEUR] ACK reçu, last_good_seq=9
[EMETTEUR] ACK reçu, last_good_seq=9
[EMETTEUR] ACK corrompu ou invalide : CRC invalide : reçu 0x4134, calculé 0x7604
[EMETTEUR] ACK reçu, last_good_seq=9
[EMETTEUR] ACK reçu, last_good_seq=9
[EMETTEUR] ACK reçu, last_good_seq=9
[EMETTEUR] ACK reçu, last_good_seq=9
[EMETTEUR] ACK reçu, last_good_seq=10
[EMETTEUR] ACK reçu, last_good_seq=11
[EMETTEUR] ACK reçu, last_good_seq=12
[EMETTEUR] ENVOI CON_CLOSE seq=13

[EMETTEUR] Terminé.
Durée totale      : 8.809 s
Trames DATA envoyées: 91
Retransmissions    : 78
ACK reçus         : 67

```

Beaucoup moins bon comme attendu, le délai reste raisonnable bien que beaucoup plus grand, on remarque ici le defaut du gobackn de tout réenvoyer. De plus, comme le délai est quand même bon, on prend QUAND MEME moins de temps que si l'on avait un délai inapproprié

Canal instable (probErreur≈0.10, probPerte≈0.15, delaiMax=300 ms)

```
[RECEPTEUR] Trame hors ordre (reçue seq=12, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=8, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=8, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=10, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=10, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=11, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=11, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=12, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=12, attendu=13), ignorée.  
[RECEPTEUR] Trame corrompue/destuff/CRC invalide: CRC invalide : reçu 0xFB37, calculé 0x8587  
[RECEPTEUR] Reçu DATA seq=9, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=9, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=12, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=12, attendu=13), ignorée.  
[RECEPTEUR] Err: 'Connexion réinitialisée par l'hôte distant',graceful ending  
  
[RECEPTEUR] Terminé.  
Durée totale : 47.273 s  
Trames DATA correctes: 13  
Trames corrompues : 21  
ACK envoyés : 261
```

```
[EMETTEUR] ACK reçu, last_good_seq=10  
[EMETTEUR] ACK reçu, last_good_seq=11  
[EMETTEUR] ACK reçu, last_good_seq=12  
[EMETTEUR] ENVOI CON_CLOSE seq=13  
  
[EMETTEUR] Terminé.  
Durée totale : 36.725 s  
Trames DATA envoyées: 302  
Retransmissions : 289  
ACK reçus : 63
```

C'est le « pire », tout va mal en gros, on a tout les types d'erreur qui s'enchainent, trame pas en ordre, erreur crc, etc

Influence du délai sur les temporisations :

Fixez timeout = 200 ms et variez delaiMax :

- 50 ms < timeout → aucune retransmission,

```

[RECEPTEUR] Reçu DATA seq=12, attendu=13
[RECEPTEUR] Trame hors ordre (reçue seq=12, attendu=13), ignorée.
[RECEPTEUR] Reçu DATA seq=3, attendu=13
[RECEPTEUR] Trame hors ordre (reçue seq=3, attendu=13), ignorée.
[RECEPTEUR] Reçu DATA seq=4, attendu=13
[RECEPTEUR] Trame hors ordre (reçue seq=4, attendu=13), ignorée.
[RECEPTEUR] Reçu DATA seq=6, attendu=13
[RECEPTEUR] Trame hors ordre (reçue seq=6, attendu=13), ignorée.
[RECEPTEUR] Trame corrompue/destuff/CRC invalide: CRC invalide : reçu 0x283C, calculé 0x5F76
[RECEPTEUR] Reçu DATA seq=9, attendu=13
[RECEPTEUR] Trame hors ordre (reçue seq=9, attendu=13), ignorée.
[RECEPTEUR] Reçu DATA seq=10, attendu=13
[RECEPTEUR] Trame hors ordre (reçue seq=10, attendu=13), ignorée.
[RECEPTEUR] Reçu DATA seq=11, attendu=13
[RECEPTEUR] Trame hors ordre (reçue seq=11, attendu=13), ignorée.
[RECEPTEUR] Reçu CON_CLOSE seq=13, fermeture.

[RECEPTEUR] Terminé.
Durée totale      : 5.709 s
Trames DATA correctes: 13
Trames corrompues   : 4
ACK envoyés        : 91

```

```

[EMETTEUR] ACK reçu, last_good_seq=10
[EMETTEUR] ENVOI CON_CLOSE seq=13

```

```

[EMETTEUR] Terminé.
Durée totale      : 2.560 s
Trames DATA envoyées: 102
Retransmissions    : 89
ACK reçus         : 60

```

retransmissions liées surtout à :

- trames corrompues (CRC invalide),
- trames perdues (prob_loss);

Mais le bas délai compense beaucoup

- 180 ms ≈ timeout → quelques faux timeouts,

```
[RECEPTEUR] Reçu DATA seq=10, attendu=13
[RECEPTEUR] Trame hors ordre (reçue seq=10, attendu=13), ignorée.
[RECEPTEUR] Reçu DATA seq=11, attendu=13
[RECEPTEUR] Trame hors ordre (reçue seq=11, attendu=13), ignorée.
[RECEPTEUR] Trame corrompue/destuff/CRC invalide: CRC invalide : reçu 0x4454, calculé 0x17AA
[RECEPTEUR] Reçu DATA seq=10, attendu=13
[RECEPTEUR] Trame hors ordre (reçue seq=10, attendu=13), ignorée.
[RECEPTEUR] Reçu DATA seq=11, attendu=13
[RECEPTEUR] Trame hors ordre (reçue seq=11, attendu=13), ignorée.
[RECEPTEUR] Reçu DATA seq=12, attendu=13
[RECEPTEUR] Trame hors ordre (reçue seq=12, attendu=13), ignorée.
[RECEPTEUR] Reçu DATA seq=10, attendu=13
[RECEPTEUR] Trame hors ordre (reçue seq=10, attendu=13), ignorée.
[RECEPTEUR] Reçu DATA seq=12, attendu=13
[RECEPTEUR] Trame hors ordre (reçue seq=12, attendu=13), ignorée.
[RECEPTEUR] Reçu CON_CLOSE seq=13, fermeture.

[RECEPTEUR] Terminé.
Durée totale      : 13.619 s
Trames DATA correctes: 13
Trames corrompues   : 12
ACK envoyés        : 135
```

```
[EMETTEUR] ACK reçu, last_good_seq=11
[EMETTEUR] ENVOI CON_CLOSE seq=13
```

```
[EMETTEUR] Terminé.
Durée totale      : 12.618 s
Trames DATA envoyées: 144
Retransmissions    : 131
ACK reçus          : 82
```

Correspond aux assumptions

- 300 ms > timeout → nombreuses retransmissions.

```
[RECEPTEUR] Trame hors ordre (reçue seq=12, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=10, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=10, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=11, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=11, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=12, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=12, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=10, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=10, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=11, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=11, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=12, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=12, attendu=13), ignorée.  
[RECEPTEUR] Reçu DATA seq=10, attendu=13  
[RECEPTEUR] Trame hors ordre (reçue seq=10, attendu=13), ignorée.  
[RECEPTEUR] Reçu CON_CLOSE seq=13, fermeture.
```

```
[RECEPTEUR] Terminé.
```

```
Durée totale : 13.998 s  
Trames DATA correctes: 13  
Trames corrompues : 2  
ACK envoyés : 82
```

```
[EMETTEUR] ACK reçu, last_good_seq=10  
[EMETTEUR] ACK reçu, last_good_seq=11  
[EMETTEUR] ACK reçu, last_good_seq=12  
[EMETTEUR] ENVOI CON_CLOSE seq=13
```

```
[EMETTEUR] Terminé.
```

```
Durée totale : 13.353 s  
Trames DATA envoyées: 88  
Retransmissions : 75  
ACK reçus : 42
```

Pareil que le 2eme, pas bon et lééégerement pire mais pas tant que ça

Je note de mon coté que ce délai est à varier selon la qualité du canal, dans un canal parfait, l'on peut baisser tout cela car on veut alors privilégier la rapidité, bien qu'on aura jamais de canal parfait dans la vraie vie, cela montre que les ajustements des 2 paramètres en question ont un impact grand (en faisant tourner le canal parfait à timeout 200ms et delaimax à 180ms il prend à peu près 8 sec au lieu du presque instantané si paramétré avec presque aucun délai !

5 et 6-Bit Stuffing :

```
Flux original : 011111101111101111110111110
Après stuffing : 011111010111100111110101111100
Après destuff : 011111101111101111110111110
CRC(original) : 9302
CRC(destuffed) : 9302
Trame corrompue : 011111101111100111110101111100
Corruption détectée par destuff : Trame corrompue : bit stuffed différent de '0' (reçu '1' à '6')
```

Tests concluants montrés dans la partie 1

7- Discussion et analyse :

TODO :

1.Expliquez comment votre implémentation réagit réellement quand un ACK est perdu

Cela cree un timeout, ce qui entraîne un renvoi, le tout généralement hors ordre

```
[EMETTEUR] TIMEOUT sur index=2, retransmission depuis index=2
[EMETTEUR] RE-ENVOI DATA seq=2 (index=2)
[EMETTEUR] RE-ENVOI DATA seq=3 (index=3)
```

```
[RECEPTEUR] Reçu DATA seq=2, attendu=12
[RECEPTEUR] Trame hors ordre (reçue seq=2, attendu=12), ignorée
[RECEPTEUR] Reçu DATA seq=3, attendu=12
[RECEPTEUR] Trame hors ordre (reçue seq=3, attendu=12), ignorée
[RECEPTEUR] Reçu DATA seq=4, attendu=12
[RECEPTEUR] Trame hors ordre (reçue seq=4, attendu=12), ignorée
```

2. Quelle différence observez-vous entre votre code et le comportement théorique

attendu? Comme les ordinateurs sont très rapides et il n'y a pas réellement de transmission, beaucoup de nos paramètres seraient impossibles en vrai et certains qui ont moins d'impact ici ex les délais sont cruciaux quand on regarde la chose de manière théorique en vue de l'appliquer « en vrai »

3. Si vous deviez optimiser la fiabilité, que changeriez-vous ?

L'on pourrait ajouter du code de correction d'erreur au lieu de retransmission mais cela aurait un grand impact sur la vitesse de transmission