

## Notes sur le protocole Modbus des blocs Slin et Slout du Millenium M3

Les blocs FBD Slin et SLout permettent de lire et d'écrire des données 16-bits dans un programme via une application externe reliée au M3 via le câble USB Crouzet.

Le protocole entre cette application externe et le M3 est dérivé du protocole industriel ModBus.

Cela permet de surveiller certaines valeurs du programme ou de modifier des paramètres avec une application distante.

24 valeurs 16-bits peuvent être lues (via les blocs SLout) et 24 valeurs peuvent être écrites (via les blocs Slin). Les valeurs Slin et SLout ce ne sont pas les mêmes.

A noter que l'icone du bloc Slin semble indiquer que l'on peut lire et écrire ces valeurs. Je n'ai pas essayé de relire des valeurs écrites.

La documentation Crouzet (l'aide en ligne du logiciel Millenium M3) est très brève sur le protocole utilisé. Les exemples fournis ne sont pas suffisants et certains sont même faux.

Les points listés ci-dessous ne sont pas mentionnés dans la documentation Crouzet et sont pourtant indispensables pour écrire une application communiquant avec les blocs Slin et SLout.

Leur connaissance doit permettre d'écrire une application communiquant avec le M3 plus rapidement et surtout d'éviter les tâtonnements initiaux imposés par les erreurs de l'Aide Crouzet.

En clair, au début on ne comprend pourquoi notre programme n'arrive pas à communiquer.

1) Si le M3 trouve des éléments incorrects dans la trame reçue (par exemple, un *checksum* erroné), il ignore simplement la trame.

2) Quand il reçoit une requête, le M3 vérifie que l'adresse des données est dans l'intervalle valide (sinon la requête est ignorée), mais il n'est pas nécessaire que le bloc Slin ou SLout connecté à cette adresse existe dans le programme tournant dans le M3.

Par exemple, on peut donc lire des adresses SLout même si aucun bloc SLout n'est présent dans le programme.

3) Les trames du protocole sont encodées en « hexadécimal ASCII ».

Les lettres hexadécimales (de A à F) doivent être en majuscule, sinon le M3 ignore le message.

Par exemple, pour échanger la valeur décimale sur 8 bits 195, il faut déjà la convertir en hexadécimal : C3 (C = 12,  $12 \times 16 + 3 = 195$ ), puis on envoie chaque chiffre sous forme de caractère : « C » puis « 3 ». Mais envoyer « c » puis « 3 » ne sera pas compris par le M3.

4) Dans les réponses du M3, certaines caractères (mais pas tous) ont leur bit le plus significatif (bit 7) à 1. C'est un comportement incohérent car :

- Cela les rend impossible à afficher dans les outils de trace (alors que c'est l'objectif du mode ASCII de Modbus)
- Ces bits à 1 ne sont pas pris en compte dans le calcul du *checksum* de la trame. Il faut donc les remettre tous à 0 avant tout traitement y compris le calcul du *checksum* de vérification.

5) Les exemples de l'Aide Crouzet

Les exemples de Crouzet pourraient être utilisés pour tester l'interface, mais malheureusement certains sont faux et ne peuvent pas fonctionner.

SLin :

- La trame de requête donnée en exemple est invalide : elle utilise l'adresse 17 alors que l'adresse utilisable la plus petite est 25. Si cette trame est envoyée au M3, il l'ignore.
- La trame de réponse donnée ne montre pas que certains caractères ont leurs bits forts à 1

SLout :

La trame de requête est correcte, mais certains caractères de la réponse du Millenium auront leur bit fort à 1, contrairement à ce qui est donné en exemple.

6) L'Aide Crouzet indique qu'en cas de trames erronées, le M3 doit être redémarré. En fait, le M3 est plutôt robuste face aux erreurs dans les trames. Il les ignore simplement, ce qui semble la meilleure chose à faire pour un automate industriel.

7) Pour SLin, l'Aide Crouzet mentionne un débit et une parité. Cela n'a pas de sens puisque le M3 utilise un câble USB. Les détails de la liaison série sont gérés par le circuit de conversion situé dans le câble.

Quelques recommandations :

Vitesse : Le traitement des requêtes SLin et SLout par le M3 est assez lent. Il est logique que le M3 les traite de manière moins prioritaire que le processus qu'on lui a confié. Il faut donc mettre des temporisations adaptées dans l'application (quelques secondes) pour lui laisser le temps de répondre et d'envoyer sa trame.

Pour la même raison, il est inutile d'envoyer des requêtes à fréquence élevée : certaines seront probablement ignorées.

Checksum : Bien que mes programmes vérifient toujours le *checksum* des trames reçues du Millenium, je n'ai jamais détecté d'erreur.