TD Modbus n°2

Un PC de supervision est connecté à un système de pesage via une passerelle Modbus/TCP

1. Quel est le principe de l'arbitrage maître/esclave utilisé par Modbus ?

La liaison entre la bascule et la passerelle Modbus/TCP est assurée par une RS422 utilisant un câble en paires torsadées.

1. Indiquer l’avantage apporté par ce type de câble.
2. Donner le rôle du CRC dans la trame Modbus RTU

Le PC « PSI de Supervision des ateliers » accède au résultat du pesage des tôles effectué par la balance avec la fonction 0x03 Modbus : Read Holding Register (Lecture de N mots de sortie consécutifs)

Trame de requête Modbus RTU permettant d’interroger la bascule :

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 04 | 03 | 00 | 02 | 00 | 01 | 25 | 9F |

Souligner en traits pointillés l’ADU Modbus.

Analyser la trame de requête ci-dessus en donnant le nom, la valeur et la signification de chacun des champs.

Documentation à utiliser :

- Format trame RTU : cf. Annexe

- Code fonction Modus : cf. Annexe

Le résultat du pesage est stocké dans un registre Modbus (2 octets) à l’adresse 0x02 en kg au format d’un entier 16 bits.

1. Donner la trame de réponse de la balance (sans calculer le CRC) pour une pesée de 150 kg. Indiquer la valeur des octets en hexadécimal. Commenter chacun des champs de la trame de réponse.
2. Calculer le temps de transmission de la trame de requête en considérant les paramètres suivants de la liaison série : Vitesse : 9600 bits/s, Longueur des données : 8 bits, Parité : Paire, Nombre de bits de stop : 1

***Annexe : Extrait documentation MODBUS***

Présentation

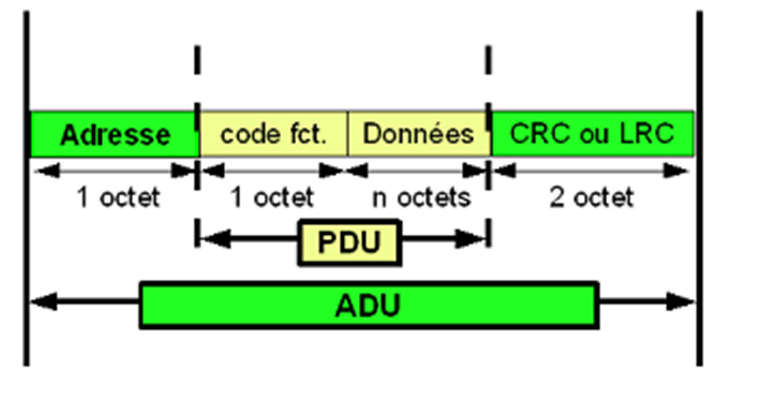
MODBUS est un protocole de communication couramment utilisé dans l'industrie pour faire communiquer des automates programmables.

Le protocole MODBUS est un protocole de dialogue basé sur une structure hiérarchisée entre un maître et plusieurs esclaves. Les esclaves possèdent une adresse comprise entre 1 et 64.

Échange Maître/ Esclave sur Modbus

Le maître interroge un esclave de numéro unique sur le réseau et attend de la part de cet esclave une réponse. Lorsque l'esclave envoie sa réponse, il place sa propre adresse dans le champ adresse afin que le maître puisse l’identifier.

Format d'une trame Modbus



Une trame Modbus est composée de deux parties :

* "ADU" Application Data Unit : Cette partie est fonction de la couche de communication sur laquelle s'appuie Modbus.

L'exemple ci-dessus correspond à un *ADU Modbus sur liaison série*.

* "PDU" Protocol Data Unit : Est composé des champs :
  + *"Code fonction"* : Ce champ est prédéfini par le protocole Modbus (voir paragraphe 2 de cette annexe). Ce code fonction définie l'action à exécuter sur l'esclave.

Par exemple le code fonction 0x0F (write Multiple Coils) permet de modifier les sorties numériques sur un esclave.

* + *"Données"* : Dans le champ "Données" sont présentes des informations relatives au code fonction (exemple : adresse de registre pour l'écriture sur les sorties) et des données à échanger entre le maître et l'esclave (exemple : les informations relatives aux sorties à mettre à l'état haut).

Codage des trames Modbus sur liaison série :

Deux types de codage peuvent être utilisés pour communiquer sur un réseau Modbus sur liaison série :

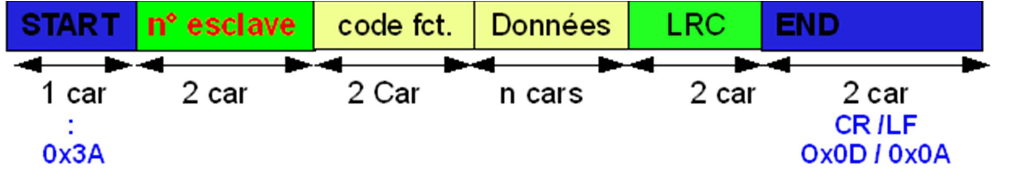
ASCII ou RTU

* Mode *ASCII* :

o Chaque octet composant une trame est codé avec 2 caractères ASCII (2 fois 8 bits).

Par exemple, la valeur 01 est codée avec les codes ASCII des symboles ‘0’ et ‘1’ (soit 0x30 et 0x31).

o Format de l’ADU pour la trame Modbus ASCII

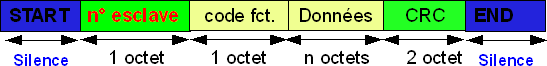


LRC : C’est la somme en hexadécimal modulo 256 du contenu de la trame hors délimiteurs (START et END), complémentée à 2. Le LRC est transmise en ASCII.

* Mode *RTU* :
  + Les octets composants la trame ne sont pas codés.

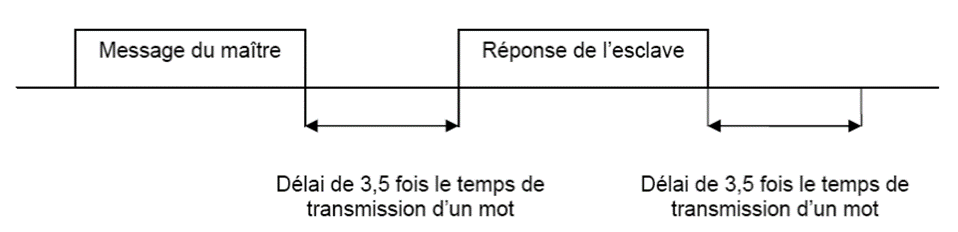
Par exemple, la valeur (15)10 aura pour octet 0x0F ou (00001111)2.

* + Format de l’ADU pour la trame Modbus RTU

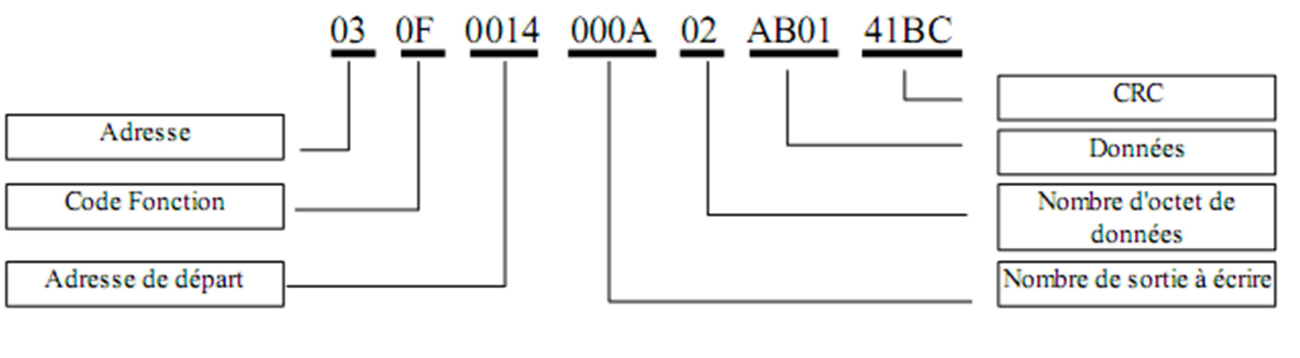


Silence : Les trames RTU ne comportent ni entête, ni délimiteur de fin.

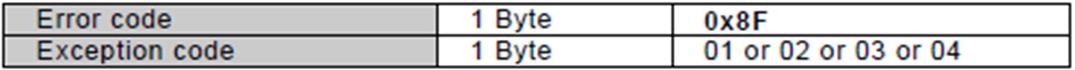
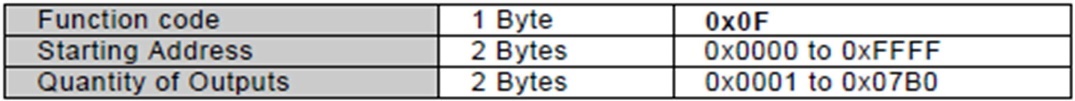
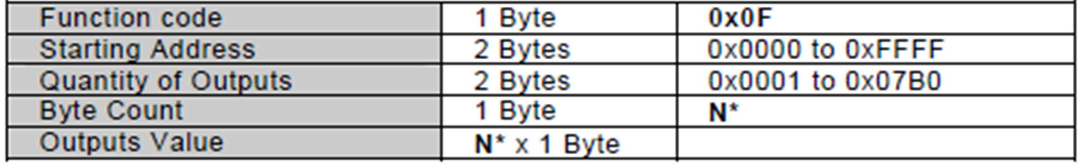
La station réceptrice contrôle le temps séparant deux caractères consécutifs, s'il est supérieur à 3 caractères et demi, elle considère que le prochain caractère est un début de trame.



Exemple d'une requête modbus RTU pour écrire une série de 10 sorties commençant à l'adresse 20 sur l'esclave à l'adresse 3:



Fonctions Modbus



**Error**

\***N** = Quantity of Outputs / 8, if the remainder is different of 0 ⇒ N = N+1

**Response PDU**

This function code is used to force each coil in a sequence of coils to either ON or OFF in a remote device. The Request PDU specifies the coil references to be forced. Coils are addressed starting at zero. Therefore coil numbered 1 is addressed as 0.

The requested ON/OFF states are specified by contents of the request data field. A logical '1' in a bit position of the field requests the corresponding output to be ON. A logical '0' requests it to be OFF.

The normal response returns the function code, starting address, and quantity of coils forced.

**Request PDU**

**15 (0x0F) Write Multiple Coils**

