

Capitalisation technique CI

Gestion d'un variateur ATV312 par un automate PREMIUM (ou M340) via Modbus

Gamme / Produit / Référence : ATV312 – MODBUS – PREMIUM – M340 - UNITY

Introduction

Cette fiche détaille toutes les étapes pour mettre au point une liaison complète **Modbus** entre un automate **Premium** et un variateur **ATV312** grâce à l'utilisation de l'outil logiciel **UNITY**.

Cette fiche pourra aussi s'appliquer lors de la mise au point d'une liaison Modbus entre le variateur ATV312 et l'automate **M340** toujours à partir de Unity.

Présentation de Modbus

Le protocole de communication Modbus est un protocole de communication utilisé pour des réseaux d'automates.

Il fonctionne sur le mode **Maître / Esclave**. Il est constitué de trame contenant l'adresse de l'automate concerné, la fonction à traiter (écriture, lecture), la donnée et le code de vérification d'erreur appelé contrôle de redondance cyclique ou CRC.

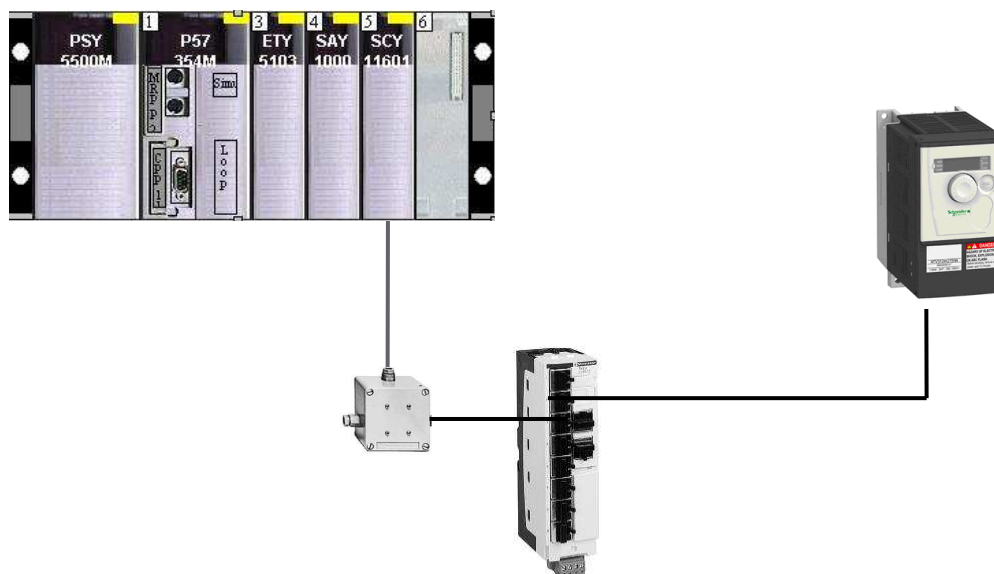
Une liaison multipoints **RS-485** relie maître et esclaves. Il s'agit d'une liaison série, de type asynchrone et différentielle qui permet un débit élevé (jusqu'à 10 Méga-bits par seconde) sur une distance importante (jusqu'à 1200m). Elle dispose de 2 bornes d'émission polarisées notées T(+), T(-) et de 2 bornes de réception polarisées notées R(+), R(-).

Deux mécanismes d'échange sont possibles :

- **Requête/réponse** : La requête du maître est adressée à un esclave donné.
La réponse est attendue en retour de la part de l'esclave interrogé.
- **Diffusion** : Le maître diffuse une requête à toutes les stations esclaves du bus.
Ces dernières exécutent l'ordre sans émettre de réponse.

Matériels nécessaires pour l'exemple

Ce document présente la communication Modbus entre un Premium et un variateur ATV312 via le répartiteur LU9GC3.



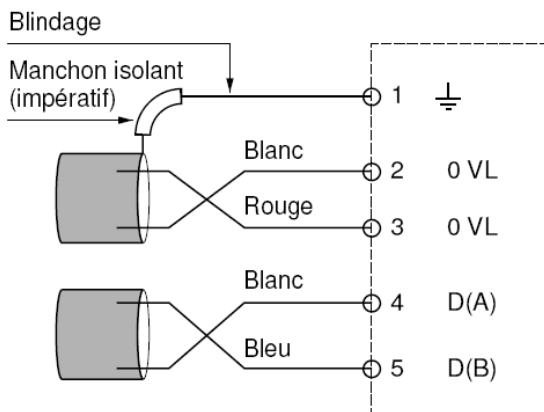
Voici la liste du matériel utilisé dans notre exemple :

- TSXP57354M + TSXSCY11601
- ATV312H075M2
- LU9GC3 et résistance de fin de ligne VW3A8306DRC
- VW3A8306R30
- TSXSCA50
- Logiciel Unity Pro

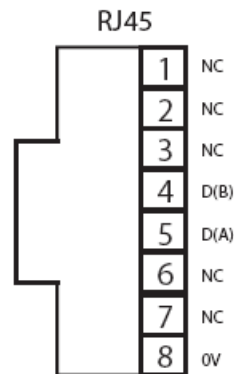
Raccordement du boîtier de dérivation TSXSCA50

Le câble référencé TSXCSA100/200/500 est utilisé pour le raccordement entre l'automate et le boîtier de dérivation. Il est également utilisé pour le raccordement entre les LU9GC3.

Schéma TSXSCA50 :



Brochage du câble RJ45 (VW3A8306):



Paramétrage de l'ATV312 en mode « non séparé »

Pour être commandé par une voie de communication, il est indispensable de renseigner les caractéristiques du bus de communication dans le menu **COM**.

Par la suite, il est nécessaire de configurer dans le menu **CTL**, les voies de commandes en modbus. Dès qu'une connexion est établie et qu'une requête est envoyée, l'ATV312 commute sa commande sur le bus actif.

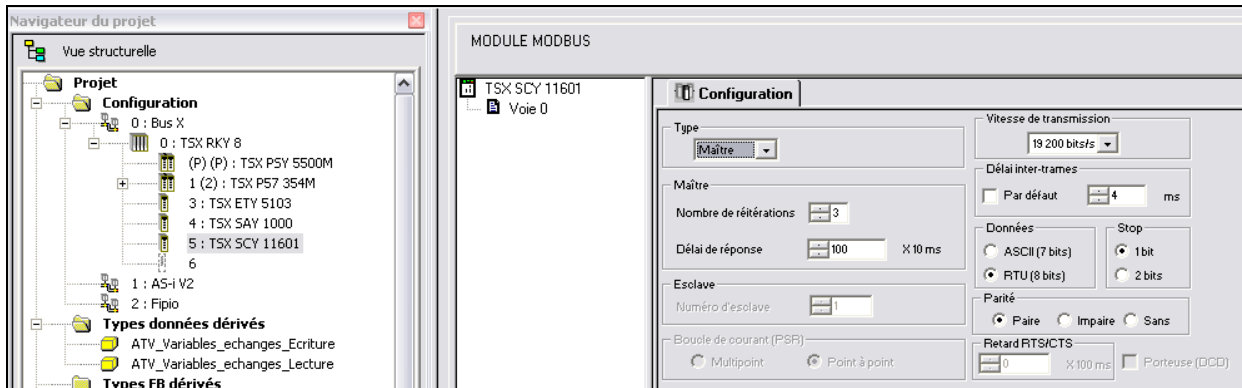
L'état du variateur passe alors de RDY à NST, indiquant que l'ATV a reconnu une connexion Modbus et qu'il est prêt à être commandé par ce canal.

- Dans le menu COM, ajuster successivement les paramètres :
 - Add : adresse esclave @ **1**
 - tbr : vitesse **19.2** (19200 bds)
 - tFO : format de communication Modbus **8E1** (8 bit de données - parité paire - 1 bit de Stop)
- Dans le menu CTL, ajuster successivement les paramètres :
 - LAC : niveaux d'accès aux fonctions avancées **L3**
 - Fr1 : configuration de la consigne **Mdb** (Modbus)

Configuration de l'automate en Modbus

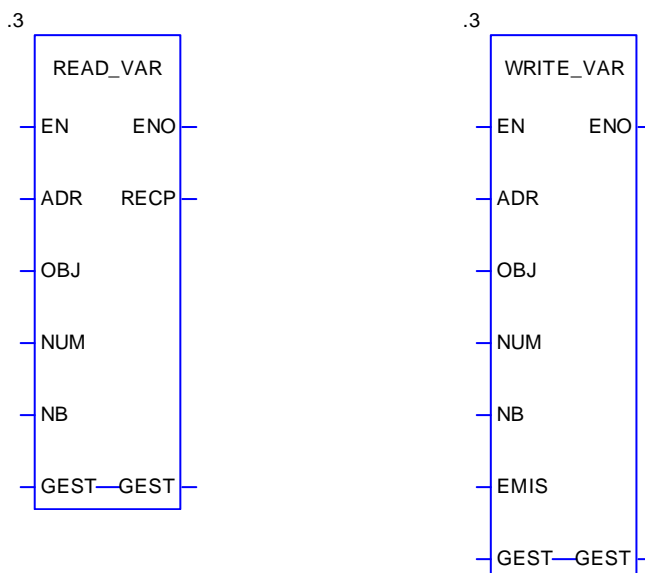
Sous Unity après avoir décrit le rack de l'automate, sélectionner la carte Modbus TSXSCY11601 pour renseigner les paramètres de communication :

- Vitesse de transmission : 19200 bits/s
- Données : RTU (8 bits)
- Stop : 1 bit
- Parité : Paire



Comment créer une requête WRITE_VAR ou READ_VAR ?

- **ADR** : définit l'adresse de l'esclave. Elle est composée successivement du rack, de l'emplacement, la voie et l'adresse et se note **ADDR('r.e.v.@')**.
Dans notre exemple, le rack est à l'adresse 0, l'emplacement du module TSXSCY11601 est 5 et on passe par la voie 0 pour établir la communication Modbus : **ADDR('0.5.0.@')**
- **Attention** : dans le cas de l'automate M340, **ADDR** est remplacé par **ADDM**
- **OBJ** : définit le type d'objet à écrire. Les variables sont en '%MW'.
- **NUM** : définit l'adresse de la variable. Pour obtenir les différentes adresses, se reporter au guide des variables de communication pour chaque équipement.
- **NB** : définit le nombre de variables successives à lire ou à écrire.
- **RECP** ou **EMIS** : définit le mot (ou tableau de mots) de réception ou d'émission.
- **GEST** : définit le tableau de mot de gestion.



N.B : pour plus d'information se rapporter au guide de mise en œuvre de communication Premium.

Configuration de l'ATV312

ADR : adresse esclave @ **1**

OBJ : **'%MW'**

NUM : il faut inscrire l'adresse de la 1^{ère} variable uniquement.

NB : **2**

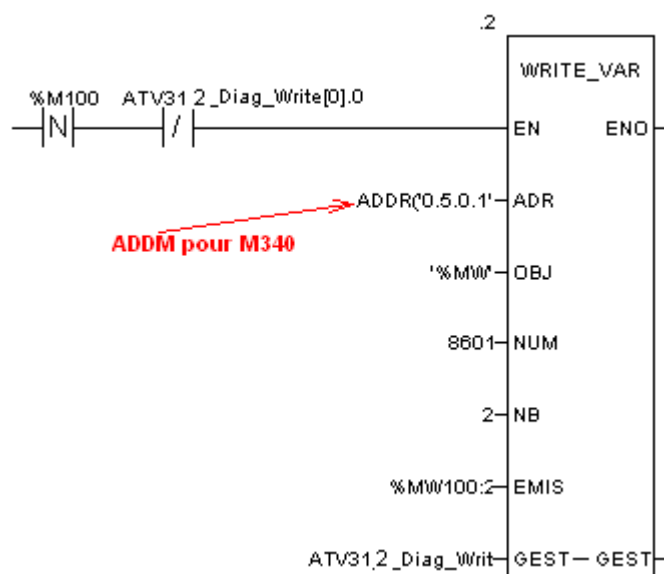
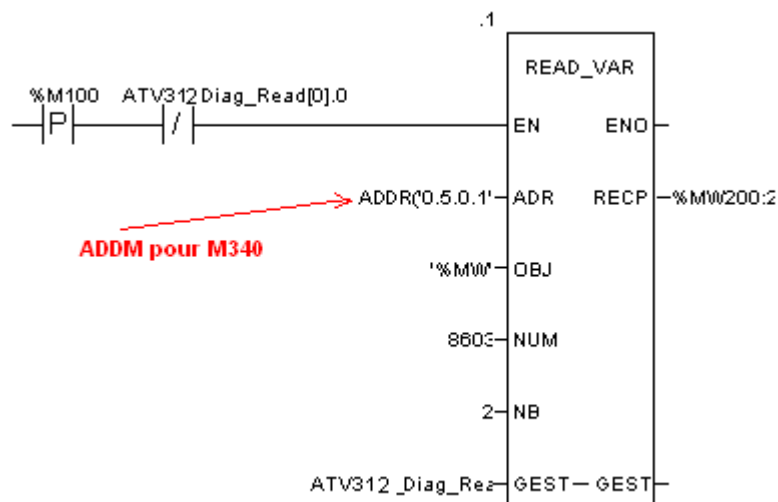
- Lecture : mot d'état ETAD **8603** et la consigne de vitesse RFRD **8604**.
- Ecriture : mot de commande CMDDD **8601** et la consigne de vitesse LFRD **8602**.

RECP : tableau de 2 mots **%MW200:2** pour la requête READ_VAR.

EMIS : tableau de 2 mots **%MW100:2** pour la requête WRITE_VAR.

GEST : tableau de 4 mots :

- **%MW500:4** pour la requête READ_VAR
- **%MW504:4** pour la requête WRITE_VAR.



Exploitation de la communication de l'ATV312

Une fois la fonction I/O scanning configurée, il suffit de réaliser un programme qui permet de lancer la commande du Driver COM de l'ATV312

Après avoir transférer le programme et vérifier la connexion des esclaves. Il suffit d'éditer une table d'animation des variables I/O scanning et piloter par la communication les équipements.

- **DRIVECOM ATV312**

