

Objectifs pédagogiques :

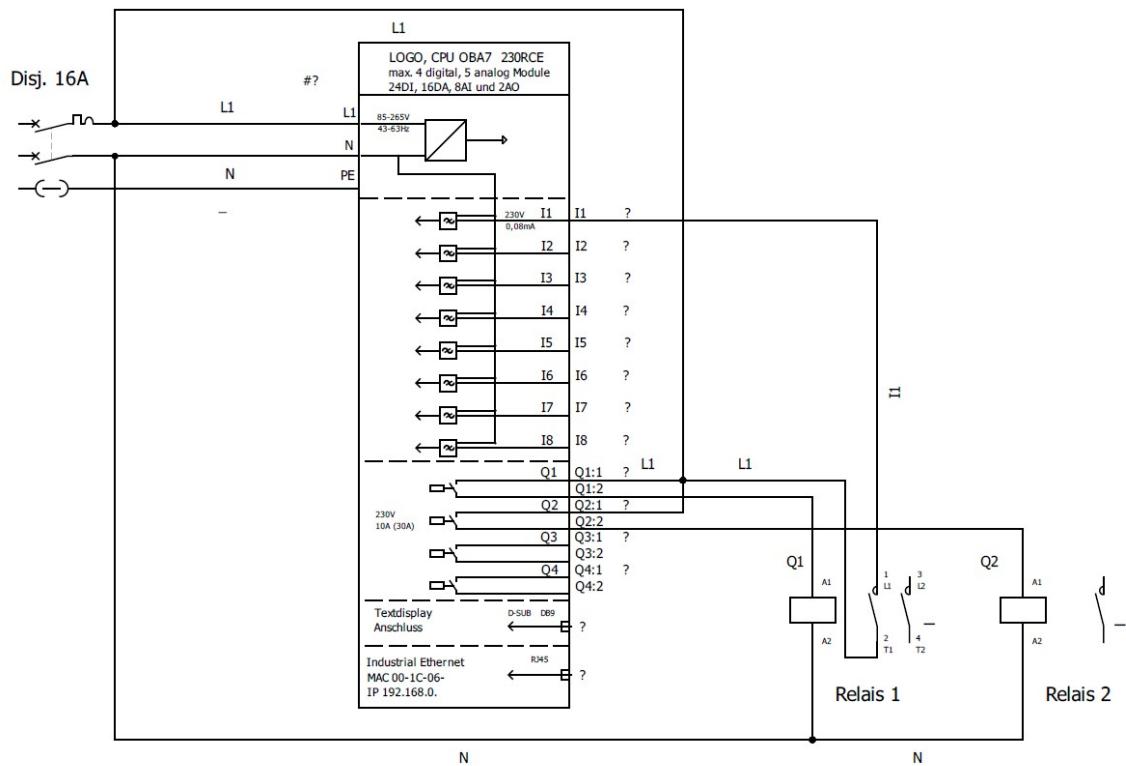
L'objectif de ce TP est de découvrir le fonctionnement d'un applicatif simple utilisé dans l'industrie : le modbus.

Safety First !

Attention, la maquette est câblée en 220V : Ne pas toucher les connections, fils et borniers !

Partie 1 : Prise en main de la maquette

Le schéma électrique de l'automate est le suivant :



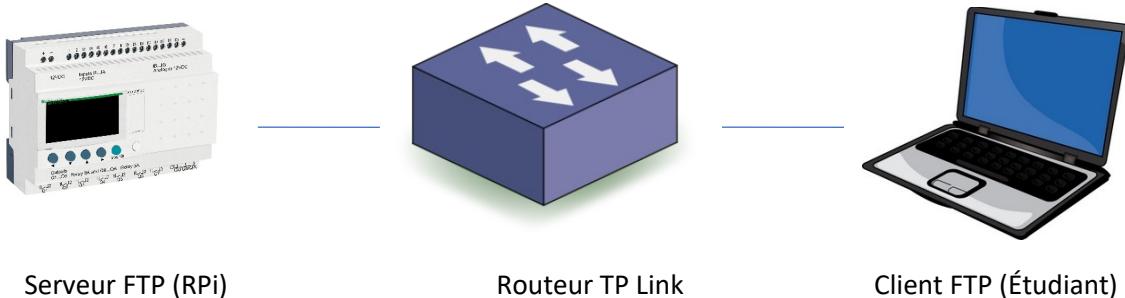
- Quelle tension est utilisée pour alimenter la maquette ?
 - Comment couper l'alimentation électrique ?
 - Combien de sortie possède l'automate ? Combien d'entrée ?
 - Deux relais sont câblés. Sur quelles sorties sont-ils câblés ?
 - Que ce passe-t-il lorsque la bobine du relais 1 est alimentée ?

➔ Appeler le professeur

Partie 2 : Essais Modbus

Les entrées/sorties de l'automate peuvent être lues/pilotées en Modbus. Dans cette partie, nous allons construire un client Modbus en python pour piloter l'automate et pour lire les entrées.

Schéma du réseau :



Connectez vous au réseau de l'automate. Vérifiez que vous pouvez réaliser une commande ping sur l'adresse de l'automate (Il vous faudra déterminer l'adresse de l'automate à l'aide de son IHM).

Ci-dessous, vous trouverez la table d'adressage Modbus de l'automate :

2.8.1.20 File -> Properties - Modbus address space

In this tab, you can view the mapping relationship between ModBus address and LOGO! resource.

Type Name	LOGO! Resource	Modbus Address	Direction	Unit
I	1 - 24	Discrete input 1 - 24	R	bit
Q	1 - 20	Coil 8193 - 8212	R/W	bit
M	1 - 64	Coil 8257 - 8320	R/W	bit
V	0.0 - 850.7	Coil 1 - 6800	R/W	bit
AI	1 - 8	Input Register 1 - 8	R	Word
VW	0 - 850	Holding Register 1 - 425	R/W	Word
AQ	1 - 8	Holding Register 513 - 520	R/W	Word
AM	1 - 64	Holding Register 529 - 592	R/W	Word

Créer un client Modbus (en python) qui doit être capable de modifier l'état de la sortie 1. (La librairie [ModbusTcpClient](#) (nommée pyModbusTCP dans pip est pratique)

Modifier votre client Modbus pour :

1. Éteindre le relais 1
2. Lire la valeur de l'entrée 1
3. Mettre en route le relais 1
4. Lire la valeur de l'entrée 2

Modifier votre script python pour :

- Qu'il accepte deux paramètres : état du relais 1 , état du relais 2
- Que la valeur de sortie du script soit la valeur de l'entrée 1
- ➔ Appeler le professeur

Partie 3 : Analyse des trames

Installer Wireshark sur votre ordinateur.

Lancer l'écoute des paquets qui sont envoyés à l'automate.

Exécuter votre script.

Analyser une trame MODBUS :

- Déterminer les adresses MAC des deux équipements
- Le port source et le port destination
- Le code utilisé dans la couche IP pour indiquer que la communication sera du TCP
- Trouver la charge utile de la trame
- Calculer l'efficacité de la trame : nombre d'octet de charge utile par rapport au nombre d'octet total de la trame

Sécurité (Appeler l'enseignant pour avoir une discussion sur ce point après avoir répondu aux questions suivantes) :

- Est-ce qu'un identifiant / mot de passe est utilisé pour se connecter ?
 - Quels sont les risques pour le système ?
 - Quels sont les solutions qui peuvent être mises en place pour sécuriser le réseau ?
- ➔ Appeler le professeur

Partie 4 : Création du système SCADA

Maintenant que vous êtes capables de piloter l'automate et de lire les données, nous allons construire une système de contrôle et d'acquisition des données simpliste (SCADA).

Connecter le raspberry pi au réseau que vous avez réalisé.

Utiliser un client SCP pour vous connecter sur le Rapsberry Pi. Identifiant : pi , mot de passe : raspberry

Déployer votre script python sur le Raspberry Pi dans le dossier /home/pi

Utiliser un client SSH pour vous connecter au Raspberry Pi . Modifier le crontab (crontab -e) pour :

- Groupe 1 : Mettre en route le relais 1 aux minutes 0, 10, 20, 30, 40, 50 de chaque heure.
Éteindre le relais 1 aux minutes 5, 15, 25, 35, 45, 55 de chaque heure.
 - Groupe 2 : Mettre en route le relais 1 aux minutes 2, 12, 22, 32, 42, 52 de chaque heure.
Éteindre le relais 1 aux minutes 7, 17, 27, 37, 47, 57 de chaque heure.
 - Groupe 3 : Enregistrer l'état de l'entrée 1 dans un fichier plat dans /home/pi/status_1.txt
- ➔ Appeler le professeur

Attention : avant d'éteindre le Rpi, se connecter en SSH et lancer la commande

`sudo shutdown -h now`