

SAE Distribution d'eau potable



1) Objectif du Sae

L'objectif de ce SAE est de pouvoir communiquer avec un système de cuve à eau qui sert à la distribution et la régulation d'eau potable d'un village .Nous devons être capable de récupérer ces valeurs via une communication en ModBus et de faire un programme IHM pour pouvoir les visualiser sur un écran et pouvoir contrôler le débit des pompes.

2) Sommaire

- **1) Objectif du Sae**
- **2) Sommaire**
- **3) Montage de la platine**
- **4) Communication**
- **5) Programmation**
- **6) IHM / Ecostruxure Operator Terminal**
- **7) Autres problèmes rencontrés**
- **8) Conclusion**

3) Montage de la platine

Nous avons mis en place du montage sur une platine. Avec un rack de cartes automates Schneider comportant une CPU **P 34200**, une carte d'entrée logique **DDI 1602**, une carte de sortie logique **DRA 1605**, une carte d'entrées/sorties analogiques **AMM 0600** et une carte de communication Ethernet **NOC 0401**.

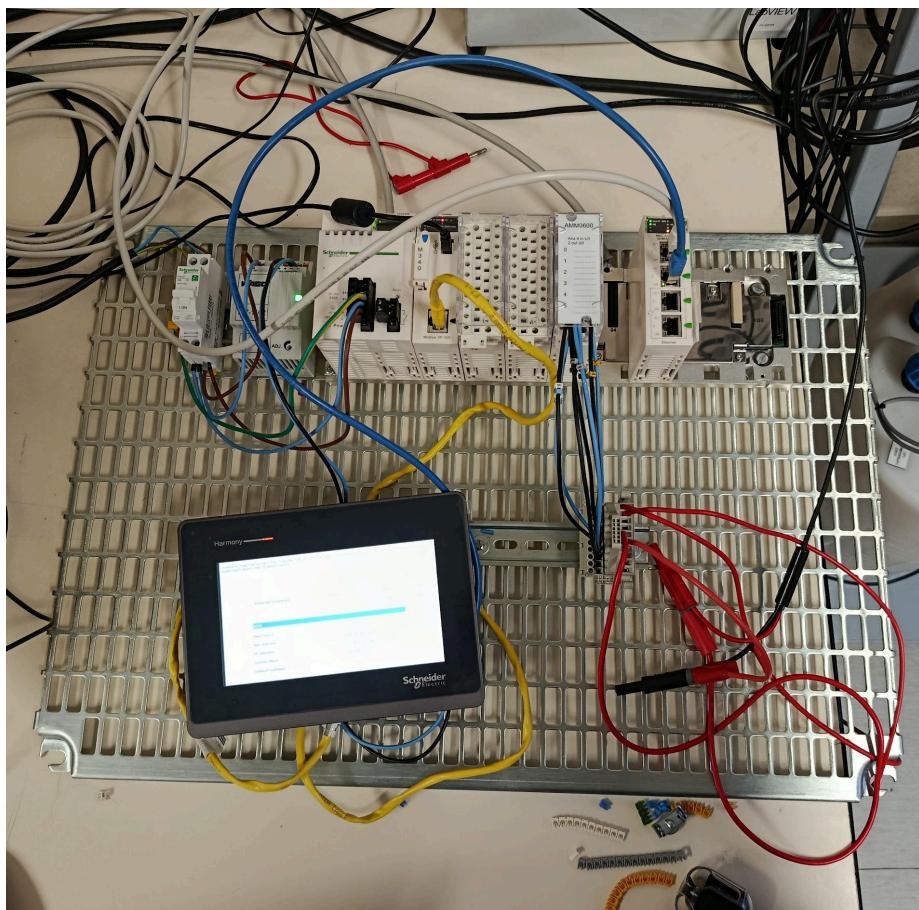
Nous avons aussi un IHM prénommée **HMIST6400** qui n'est pas accrochée sur la platine par manque d'accroche possible sur la platine.

Pour alimenter la platine, nous avons utilisé une prise dénudée que nous avons branchée en amont sur un disjoncteur de 2 ampères. Cet élément est important pour la sécurité de l'automate, cela peut éviter un potentiel casse.

Nous disposons notamment aussi d'un IHM relié sur une alimentation auxiliaire en 24V.

Nous avons rencontré des problèmes matériels avant tout, nous n'avions juste pas le bon rack pour pouvoir mettre toutes les cartes d'automates. Il nous manquait aussi les adaptateurs nous permettant la communication entre le système et l'automate. Pour palier a ce problemes nous avons "fabriquées" nos propres adaptateurs avec des fils à embouts.

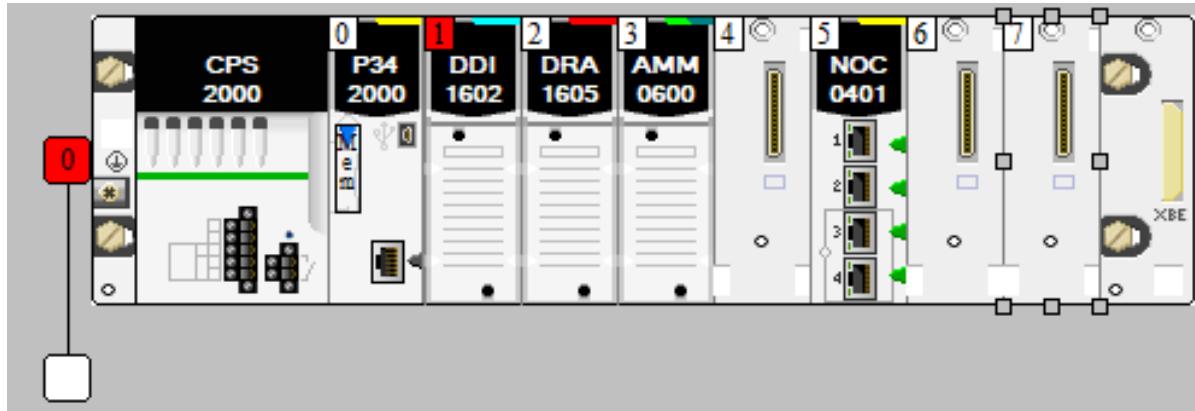
Voici une photo de la platine branché avec le système :



4) Communication

On a utilisé le logiciel **Control Expert Classic** pour programmer notre automate. On a tout d'abord reproduit le rack afin de pouvoir communiquer, nous avons ensuite formé un sous-réseau pour que notre automate, notre IHM et notre ordinateur puissent communiquer.

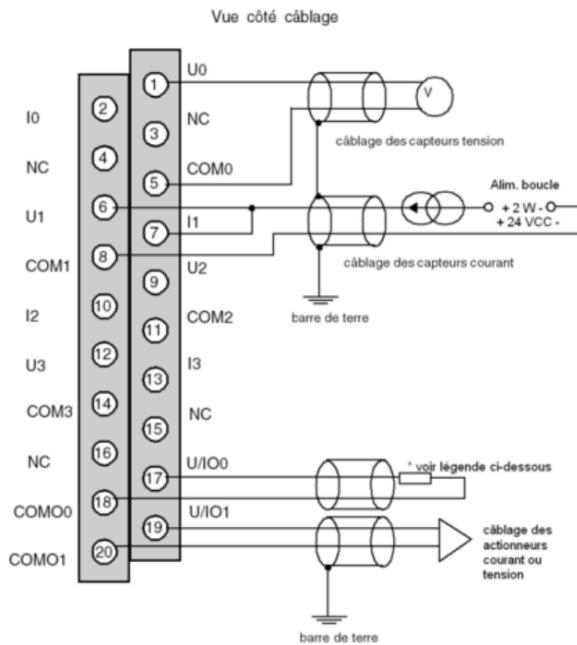




Le **CPS2000** alimente le rack, la **P342000** reçoit le programme, la **DDI1602** est une carte d'entrées logique, la **DRA1605** est une carte pour les sorties logique, la **AMM0600** est une carte d'entrées/sorties analogique et la **NOC0401** est une carte pour communiquer avec des éléments externe en Ethernet.

Lors de la réalisation de notre projet, nous nous sommes essentiellement servis de la P34, de l'AMM et de la NOC.

Nous n'avions jamais utilisé une carte analogique, du coup nous avons fait des recherches sur Internet pour savoir comment la brancher.



Ux Entrée pôle + de la voie x
 COMx Entrée pôle - de la voie x
 UI/Ox sortie pôle + de la voie x
 COMOx sortie pôle - de la voie x
 * La boucle de courant est auto-alimentée par la sortie et ne nécessite aucune alimentation externe.

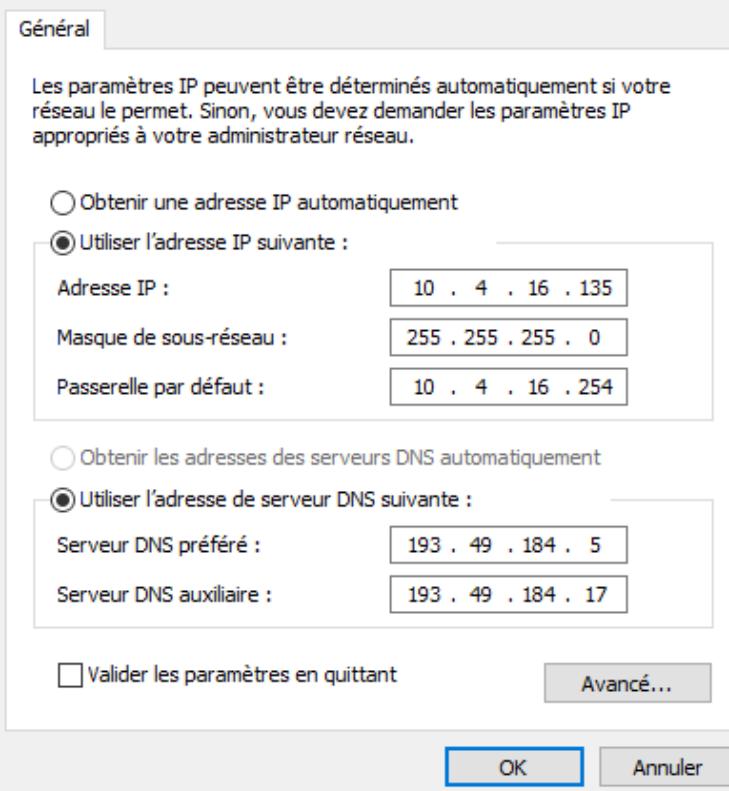
Nous avons aussi dû configurer le NOC pour pouvoir communiquer avec les autres équipements, nous avons donc créé un sous réseau 10.4.16.xxx

Le NOC :



L'ordinateur (pour transférer le programme sur l'IHM) :

Propriétés de : Protocole Internet version 4 (TCP/IPv4) X



L'adresse de l'IHM : 10.4.16.140

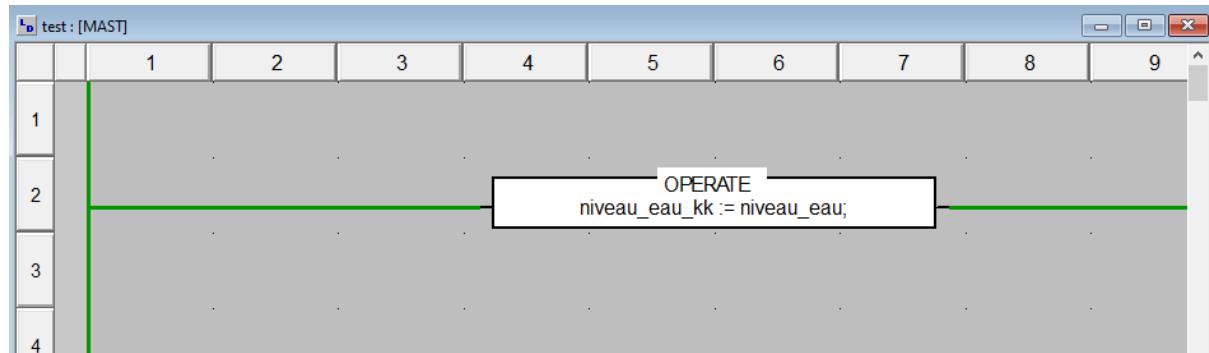
5) Programmation

En premier lieu nous avons créé diverses variables pour récupérer des informations de notre système, et d'autres pour tester notre communication avec notre IHM.

Nom	Adresse	Type	Description
Pompe_remplissage	%I0.3.0	INT	La vitesse de remplissage de la pompe
Niveau_eau	%Q0.3.0	INT	La valeur actuelle du niveau d'eau

Pompe_remplissage_IHM	%MW2	INT	La vitesse de remplissage mais convertie en variable mémoire pour pouvoir la communiquer à l'IHM
Niveau_eau_IHM	%MW1	INT	Le niveau d'eau actuel mais convertie en variable mémoire pour pouvoir la communiquer à l'IHM
Truc_IHM	%M1	EBOOL	Variable de test pour vérifier que l'IHM communique bien avec l'automate
Muche_IHM	%M2	EBOOL	Variable pour tester différentes option dans le logiciel pour programmer notre IHM

On a donc créé une un programme en LADDER pour “load” les valeurs de nos variables réels dans nos variables mémoires :

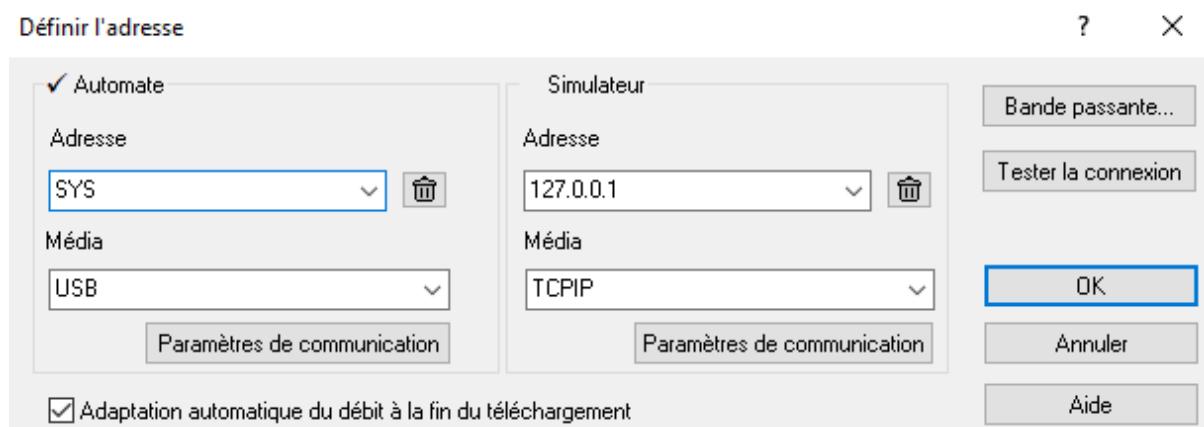


On a aussi créé une table d'animation pour visualiser ces valeurs :

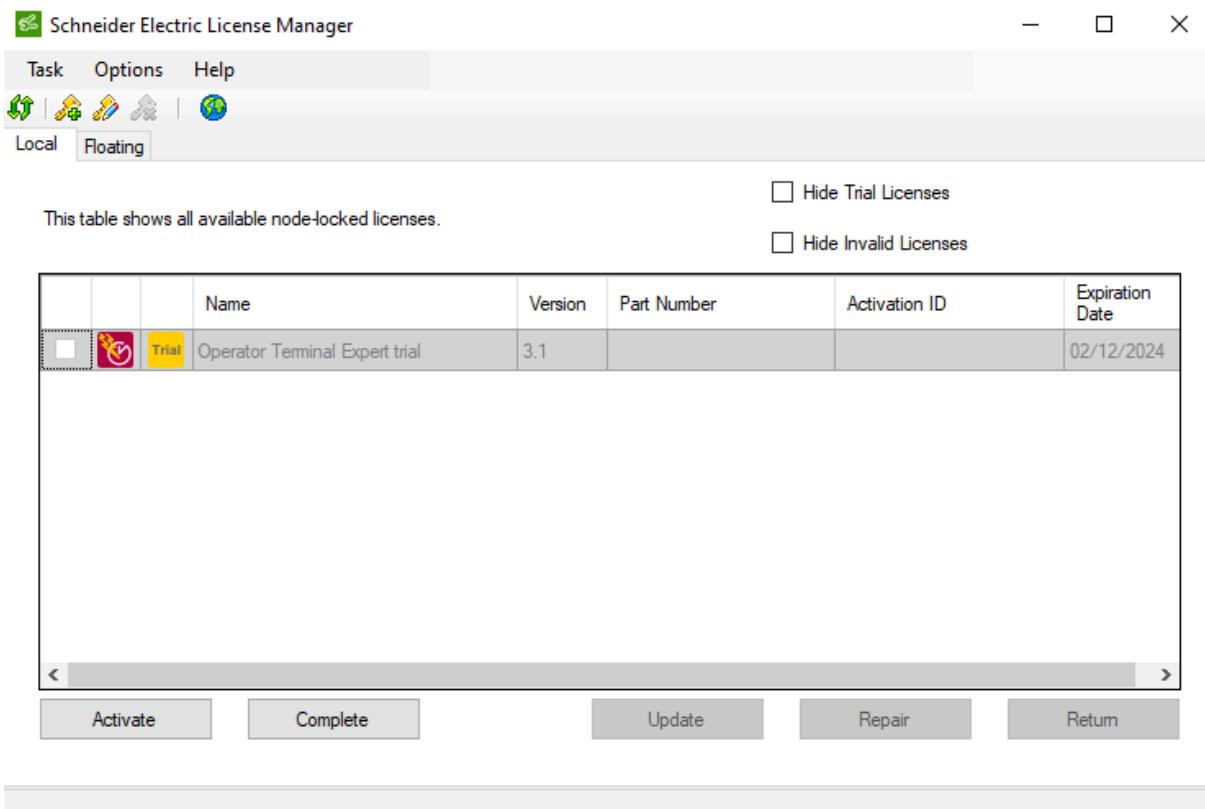
Test_Out

Modification	Forcer					
Nom	Valeur	Type				
niveau_eau	5526	INT				
Pompe_remp脂issement	0	INT				
truc	0	EBOOL				
bidule	0	EBOOL				
niveau_eau_kk	5526	INT				
Consigne_pp	3000	INT				

On a eu des problèmes de connexion, on a voulu transférer le projet par Ethernet, mais la connexion ne voulait pas s'effectuer pour une raison qu'on a pas cherché à régler car nous avons directement opté pour une communication par USB car nous avions le câble adéquat.



Après avoir changé la façon de communiquer, il n'y avait plus de problème avec le programme.



n°2

Nom	Valeur	Type	Commentaire
niveau_eau	516	INT	

Niveau d'eau : 50 cm = 500 int

6) IHM / Eco Structure Operator Terminal

Après avoir connecté votre PC à internet, et réussi à lancer control expert. Nous avons pu commencer à créer et configurer notre écran IHM, pour commencer nous avons repris la table de mnémoniques que nous avons fait sur Control expert et nous avons repris les mêmes noms et adresses et les avons intégrées au logiciel.

Les voici :

Niveau_eau	INT	GenericModbusSerial... %MW1	DefaultScanRate (10... 0)
Pompe_de_remplissage	INT	GenericModbusSerial... %MW2	DefaultScanRate (10... 0)
truc	BOOL	GenericModbusSerial... %M0	DefaultScanRate (10... 0)
muche	BOOL	GenericModbusSerial... %M1	DefaultScanRate (10... 0)

Truc et Muche sont deux variables en booléen qui nous ont servi de test pour la communication entre l'automate et l'ihm. Nous reviendrons dessus plus tard.

Une fois les variables créée nous avons pu commencer l'écran que voici :



Nous avons ajouté 2 affichages numériques.

Le premier affichage numérique est là pour nous montrer la valeur du niveau de l'eau nous ne rentrons pas de valeur dedans il sert juste à montrer la mesure du niveau de la cuve.

Voici la configuration de la saisie de valeur:

Fonction	Forme	Favori
Basique	Détail	
Valeur actuelle	Niveau_eau.Value	<input type="checkbox"/>
Format	Dec	<input type="checkbox"/>
Integer	5	<input type="checkbox"/>
Nombre de décimales	0	<input type="checkbox"/>
Activer le mode de saisie		<input checked="" type="checkbox"/>

Nous avons mis dans la rubrique valeur actuelle notre variable niveau eau. Cela permet de récupérer la valeur actuelle dans l'automate et cette valeur est donc renvoyée sur l'ihm et dans l'affichage numérique. Nous pouvons lire directement la valeur sur l'IHM.

Pour le second afficheur numérique, elle sert à appliquer une consigne à notre pompe. Notre souci principal étant ce bouton car nous avons besoin de saisir une valeur et que cette valeur s'applique dans l'automate pour que celui-ci active la pompe à la puissance demandée.

Basique	Détail
Valeur actuelle	Pompe_de_remplissage.Value
Format	Dec
Integer	5
Nombre de décimales	0
Activer le mode de saisie	truc.Value

Tout comme pour le niveau d'eau nous avons mis la variable associée à l'action de l'affichage dans la rubrique valeur actuelle.

Malheureusement cela n'a pas suffit, comme nous étions bloqués, Nous avons dû faire beaucoup d'essais avant de comprendre comment faire pour pouvoir saisir une valeur dans l'afficheur numérique.
Nous avons donc essayé beaucoup de paramètres différents nous avons une case à cocher pour activer le mode de saisie mais malheureusement celle ci ne suffit pas. un autre paramètres est à activer et il se situe sur la ligne valeur actuelle du moins plus précisément dans le choix de la valeur actuelle.

Sélecteur de variable

- □ ×

Propriété source

Pompe_de_remplissage.Value

Rechercher

Dossier	Nom	Nom du type
None	Niveau_eau	INT
▶ None	Pompe_de_remplissa...	INT
None	truc	BOOL
None	muche	BOOL
None	Auto	BOOL

➤ Ajouter un variable

Sens Bidirectionnel

Quand on choisit la variable que nous voulons associées à notre objet, il est impératif de choisir le sens qui est donc bidirectionnel, pourquoi cela car via l'ihm nous envoyons une valeur directement à l'automate pour la consigne de la pompe et à l'inverse l'automate envoie constamment la valeur de la variable *ici consigne pompe* constamment à l'ihm.

Une fois ce paramètre trouvée (ce qui a pris beaucoup de temps)
Nous avons pu passer à la suite.

Ensuite nous avons ajouté une lampe pour voir si la communication fonctionnait bien, malheureusement nous avons pris peu de temps pour essayer de la faire fonctionner du coup la lampe ne sert à rien sur l'écran de l'IHM.

Pour pouvoir être sûr que notre communication fonctionne bien, Nous sommes passées par un bouton ON/OFF qui passe de l'état 0 à 1 quand celui-ci est appuyé. De ce fait, en lui mettant la bonne variable dans le paramètres fait pour nous, nous avons pu tester la communication entre l'IHM et l'automate.

Nous avons associées la variable de test au bouton. Cela se fait de la même manière que les autres boutons ou affichage.

Maintenant nous allons vous expliquer comment nous avons paramétré le type de communication.

Pour la communication nous avons été obligé de passer par un modbusSIO (Serial Input/Output) qui nous permet de pouvoir faire une communication en modbus mais avec des câbles ethernet.

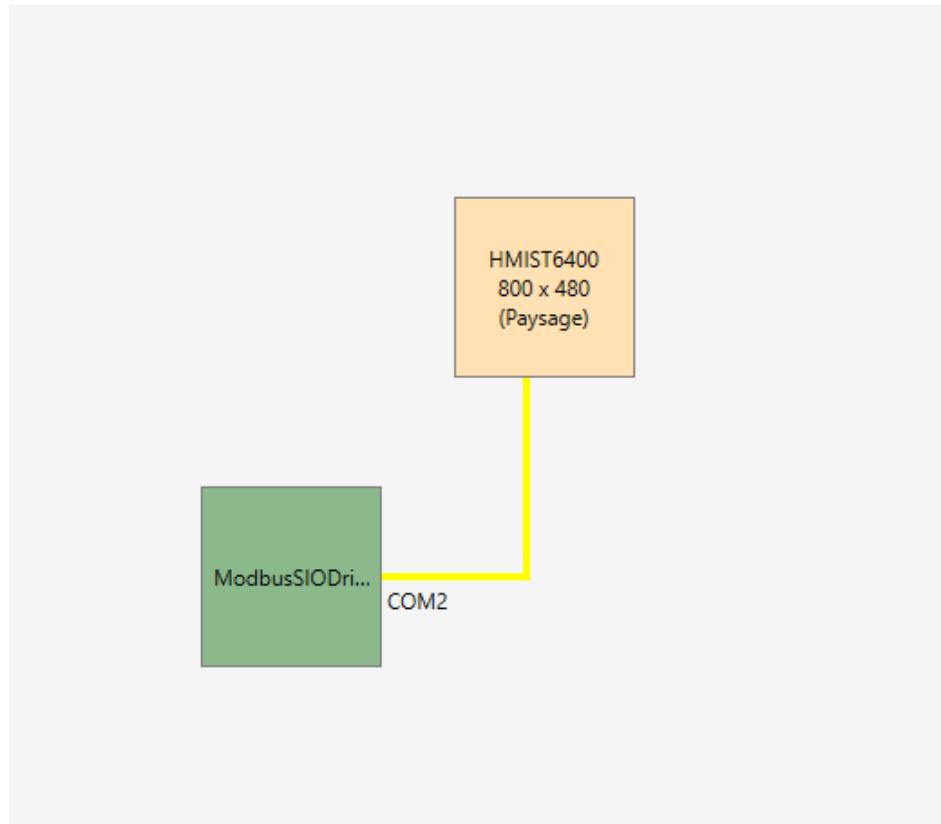
Voici la configuration de celui ci:

Nom	ModbusSIODriver1
Type	Driver
Basique Favori	
SIO Type	RS-422/485(2wire)
Speed	19200
Data Length	8 bits
Parity Bit	Pair
Stop Bits	1 bit
Flow Control	NONE
Timeout(sec)	3
Retry	2
Use default wait to send time	<input checked="" type="checkbox"/>
Wait To Send(ms)	3
RI/VCC	NONE

Il n'y a rien a souligner de spécial aucun paramètres a était changée.

Ce protocole de communication nous permet de passer toute nos informations en filaire et sans utiliser d'adresse IP sur l'ihm ce qui nous simplifie grandement la tâche.

Voici l'architecture du système que nous pouvons voir sur le logiciel Eco Structure Operator Terminal :



L'architecture comprend l'IHM, le mode de communication et le port de l'IHM utilisé pour la communication.

C'est un schéma très simple qui ne détaille pas vraiment l'architecture de communication.

Une fois le logiciel maîtrisé et tout les soucis réglé, nous avons pu faire l'écran final. Le voici :



Cet écran comprend un mode auto/manu et une activation de la saisie sur la consigne pompe.

Pour ce qui concerne le mode auto manu le mode manu est quand le bouton est pas coché alors on peut rentrer les valeurs que l'on souhaite. Le mode auto lui fait en sorte que *tant que le niveau d'eau est pas => 6000 alors la consigne pour = 5000 dès que le niveau d'eau = 6000 alors la consigne = 0*

Ce système est très simple, par manque de temps nous n'avons pas pu faire quelque chose de plus sophistiqué.

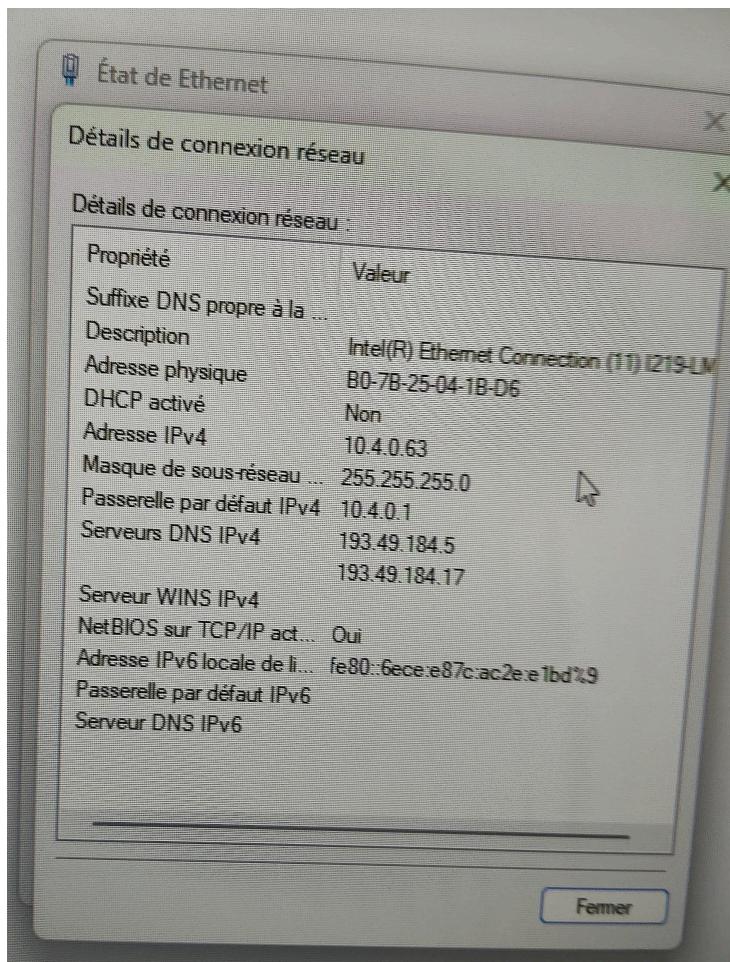
l'activation de la consigne via un switch était un test d'un paramètre de l'afficheur numérique qui nous permet de dire que si une valeur est pas à 1 alors le mode de saisie est pas actif. De ce fait nous avons laissé ce bouton qui permet de saisir ou non la consigne du moteur.

7) Autres problèmes rencontrés

Le premier souci que nous avons pu rencontrer était que les logiciels n'étaient pas installés sur le pc que nous avons utilisés pour faire tout le projet. Un gros souci pour l'avancement du projet du coup. Nous avons installé les logiciels qui était control expert et Eco Structure Operator Terminal.

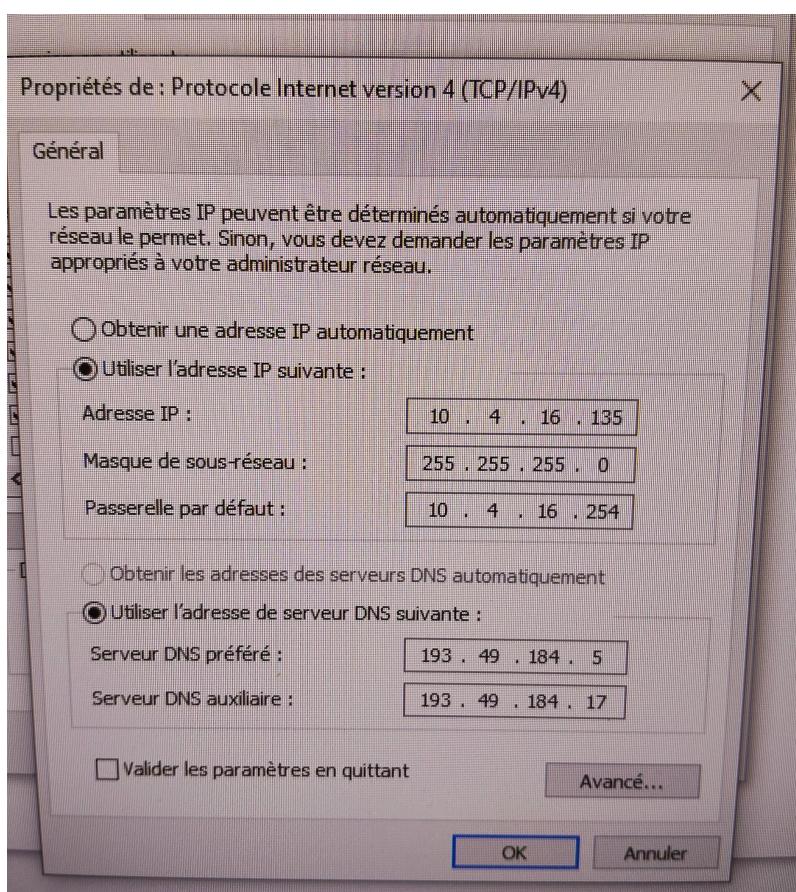
Le second souci rencontré était lors de la mise en place de la machine, pour pouvoir nous connecter au réseau de la salle nous devions changer l'adresse IPv4 de la carte réseau pour pouvoir lancer Ecostruxure et pour faire des recherches sur Internet.

Adresse utilisée dans la salle : 10.4.1.63



Cette adresse nous servait à nous connecter à internet, nous étions obligé de nous connecter à internet pour pouvoir lancer le logiciel control expert.

Une fois le logiciel lancé nous pouvons repasser sur une autre adresse ip qui était celle ci.



Elle nous permet de faire la communication entre le pc et le NOC donc l'automate.

Les autres soucis que nous avons pu rencontrer ont pu être expliquées précédemment.

8) Conclusion

Même si le travail à effectuer pouvait paraître simple à mettre en place, nous avons eu beaucoup de soucis matériel et logiciel. Nous avons dû apprendre un nouveau logiciel presque de A à Z. Nous avons pu passer de la partie pratique à faire une platine entière avec une alimentation à trouver une solution pour récupérer les données. C'était un projet très intéressant menant à une partie logicielle comme manuel. Nous avons pu voir aussi les spécificités du réseau à devoir changées sans cesse d'adresse IP

Ce projet nous a beaucoup enrichi car nous avons dû apprendre un logiciel seul, du coup les recherches étaient de mise. Nous avons pu mettre directement en pratique.