

Applications de l'industrie connectée

Niveau 1

Paramétrer une Interface Homme-Machine (IHM)
et appréhender les bases de la communication

Tutoriel applicatif 2023

se.com/fr

Life Is On Schneider Electric



Sommaire

03 | Monter en compétence dans l'industrie

03 Notre vision de l'industrie du futur

03 Les métiers concernés

03 Faire monter en compétence les étudiants sur les technologies de l'industrie

04 Découvrir le nouveau parcours pédagogique Schneider Electric

04 | Familiarisation avec l'environnement industriel

05 | Tutoriel applicatif communication Modbus Série / Modbus TCP et IHM

05 Objectif : apprendre les bases de la communication Modbus Série et Modbus TCP entre un automate machine et une IHM

05 | Mise en application

05 Mise en contexte

06 Mise en œuvre de l'application

0- Consignes de sécurité

07 Analyse fonctionnelle

08 Schéma de principe

09 Schémas de câblage

10 Réalisation du câblage

11 | Écriture du programme

11 Les étapes clés

12 Descriptif du programme applicatif

43 Mise en service

44 | En synthèse

Où trouver les applications et les logiciels ?

Que dois-je retenir ?

Pour aller plus loin

Monter en compétence dans l'industrie

Notre vision de l'industrie du futur

- Digitale pour l'efficacité,
- Électrique pour décarboner,
- Humaine.

Schneider Electric répond aux enjeux d'une industrie 4.0 plus performante, plus durable et vous accompagne dans votre projet avec des solutions digitales de bout en bout couvrant quatre domaines de compétences :

- excellence opérationnelle,
- performance des équipements,
- cybersécurité industrielle,
- gestion de performance énergétique.

Les métiers de l'industrie évoluent et requièrent une montée en compétence pour que les élèves apprennent sur les technologies sur lesquelles ils travailleront demain.

Les métiers concernés

- **Conducteur de machine industrielle :** je dois m'assurer du bon fonctionnement et des réglages des machines ou poste(s) dont j'ai la charge.

- **Pilote de ligne de production industrielle :** je dirige les opérations des lignes de production et je mets en service les installations dont j'assure les flux de production et le fonctionnement.

- **Technicien de maintenance industrielle :** je m'assure que l'équipement de l'entreprise soit en permanence en état de marche.

- **Concepteur de machine industrielle :** je programme l'automatisme des machines qui composent les lignes de production.



Hugo Décrypte vous fait visiter l'Usine du Futur de Schneider Electric !

Schneider Electric France S'abonner

Faire monter en compétence les étudiants sur les technologies de l'industrie

Les évolutions technologiques rapides font évoluer les métiers dans l'industrie et requièrent une montée en compétence pour que les élèves se forment sur les technologies sur lesquelles ils travailleront demain.

C'est pourquoi nous avons créé un parcours de montée en compétence spécialement conçu pour les enseignants qui souhaitent acquérir une expérience d'apprentissage industrielle.

Pour exercer ces métiers, un élève doit être familier avec les technologies.

- L'automatisme
- La variation de vitesse et les moteurs
- La robotique/cobotique
- La détection
- La sécurité machine
- Les réseaux de communication et la cybersécurité
- Le traitement des données, la supervision et le contrôle commande

Découvrez le nouveau parcours pédagogique Schneider Electric

Schneider Electric a créé **un parcours de montée en compétence** spécialement conçu pour les enseignants qui souhaitent se développer et offrir une expérience d'apprentissage industrielle. Notre parcours de montée de compétence comprendra **une série de trois niveaux** vous permettant une prise en main progressive des technologies.

Niveau 1 — Niveau baccalauréat professionnel : **être familier avec l'environnement industriel et les technologies Schneider Electric.**

Niveau 2 — Niveau Brevet de technicien supérieur (BTS) / Bachelor universitaire de technologie (BUT) : **être autonome sur les technologies Schneider Electric.**

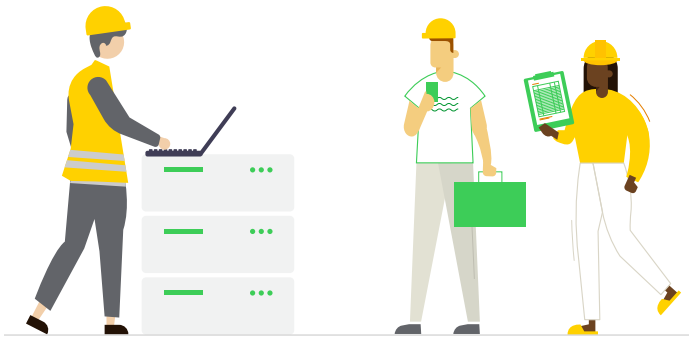
Niveau 3 — Niveau BUT, école d'ingénieurs : **être expert d'une technologie.** (disponible au 2e trimestre 2024).

Familiarisation avec l'environnement industriel

Je souhaite acquérir les compétences qui me permettent d'être familier avec l'environnement industriel et les technologies.

Équipez-vous de nos packs pour développer des applications et montez en compétence.

- Programmer un contrôleur machine
- **Paramétrer une interface Homme-Machine (IHM) et appréhender les bases de la communication**
- Découvrir l'IoT



- Appréhender les bases de la régulation
- Appréhender la variation de vitesse
- Appréhender la motion
- Découvrir la sécurité machine

Découvrez nos tutoriels en vous inscrivant ici

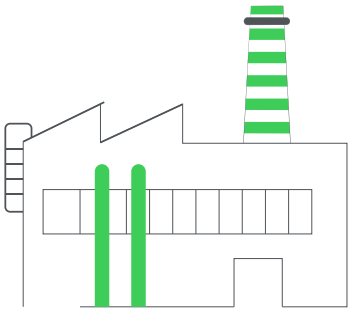
Liste des compétences susceptible d'évolution et d'enrichissement

Tutoriel applicatif communication Modbus Série / Modbus TCP et IHM

Objectif : apprendre les bases de la communication Modbus Série et Modbus TCP entre un automate machine et un terminal de dialogue (Interface Homme Machine)

Mise en application

Mise en contexte



La partie commande de gestion d'un feu de croisement (tutoriel automatisme séquentiel) doit être améliorée avec l'ajout d'une Interface Homme-Machine (IHM).

La communication entre l'IHM et l'automate peut être de type :

- Modbus série
- Modbus TCP

Cette IHM doit permettre de visualiser les états des deux feux de croisement (voie A et voie B) et déporter le bouton « appel piéton ».

Les durées restantes, pour l'allumage des différents feux sont calculées et affichées en fonction des l'allumage des feux. Les temporisations (**Vert**, **Orange** et **Rouge**) sont modifiables depuis un écran de paramétrage.
Développer les applications pour des terminaux graphiques de type HMISTU et HMIST6.

Mise en œuvre de l'application

Les produits nécessaires au développement de l'application comprennent :

Référence	Libellé	Quantité
MD2HMISTUX3	Lot de 3 IHM STU Education écran tactile 5.7pcs 320x240 Ethernet	1

OU

Référence	Libellé	Quantité
MD2HMIST6X3	Lot de 3 IHM ST6 Education écran tactile 7pcs 800x480 Ethernet	1

Matériel non fourni dans les packs ou optionnel :

Référence	Libellé	Quantité
NSYTBP292412	Thalassa - boîte industrielle - 291x241x128mm - PC	1
XB5PRJ45	Harmony XB5 - prise RJ45 type Ethernet - Ø22	2
ACTPC6UULS20WE	Cordon RJ45 long 2m	1
ACTPC6UULS05WE	Cordon RJ45 long 0,5m	2

Consignes de sécurité

Les consignes de sécurité sont dans les guides de chaque équipement listé ci-dessous :

- Modicon M221 - Logic Controller, Guide de Programmation - SoMachine Basic
- ABLM1A24025 - Documentation technique
- HMISTU855 - Notice de montage
- HMIST6400 - Notice de montage

Analyse fonctionnelle

Logigramme de l'analyse fonctionnelle :

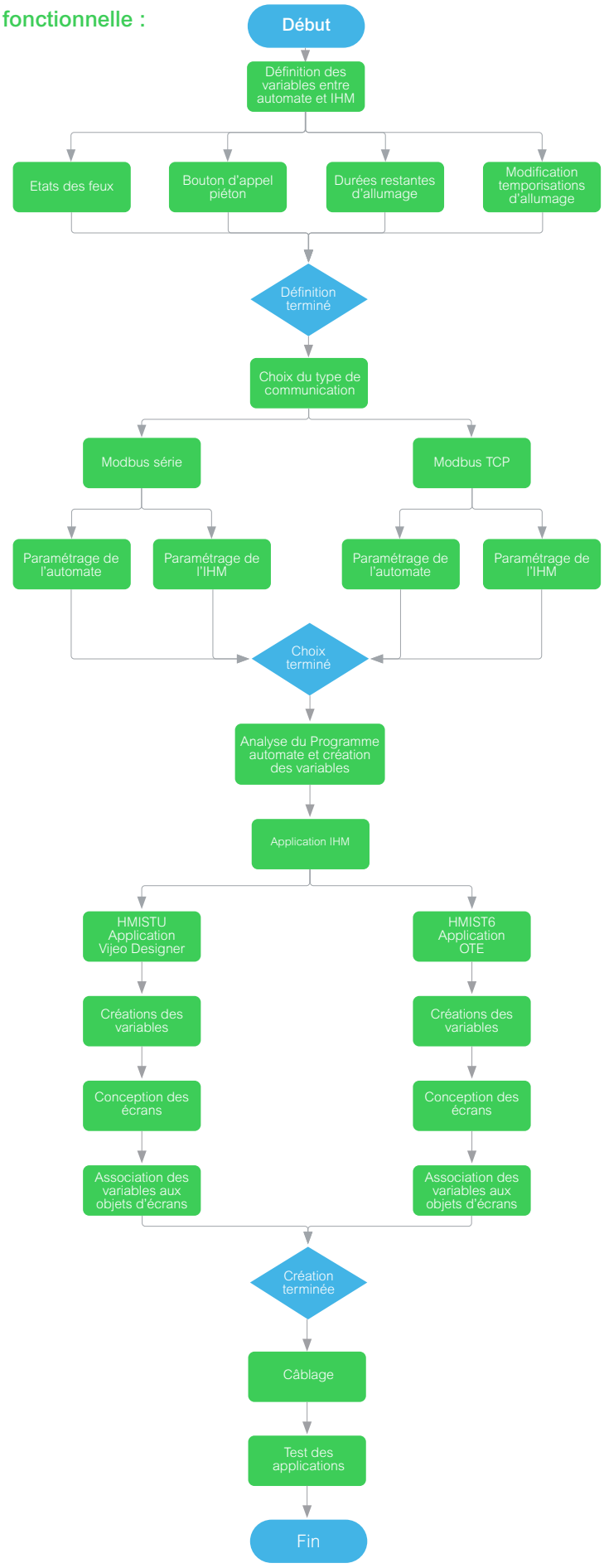


Schéma de principe

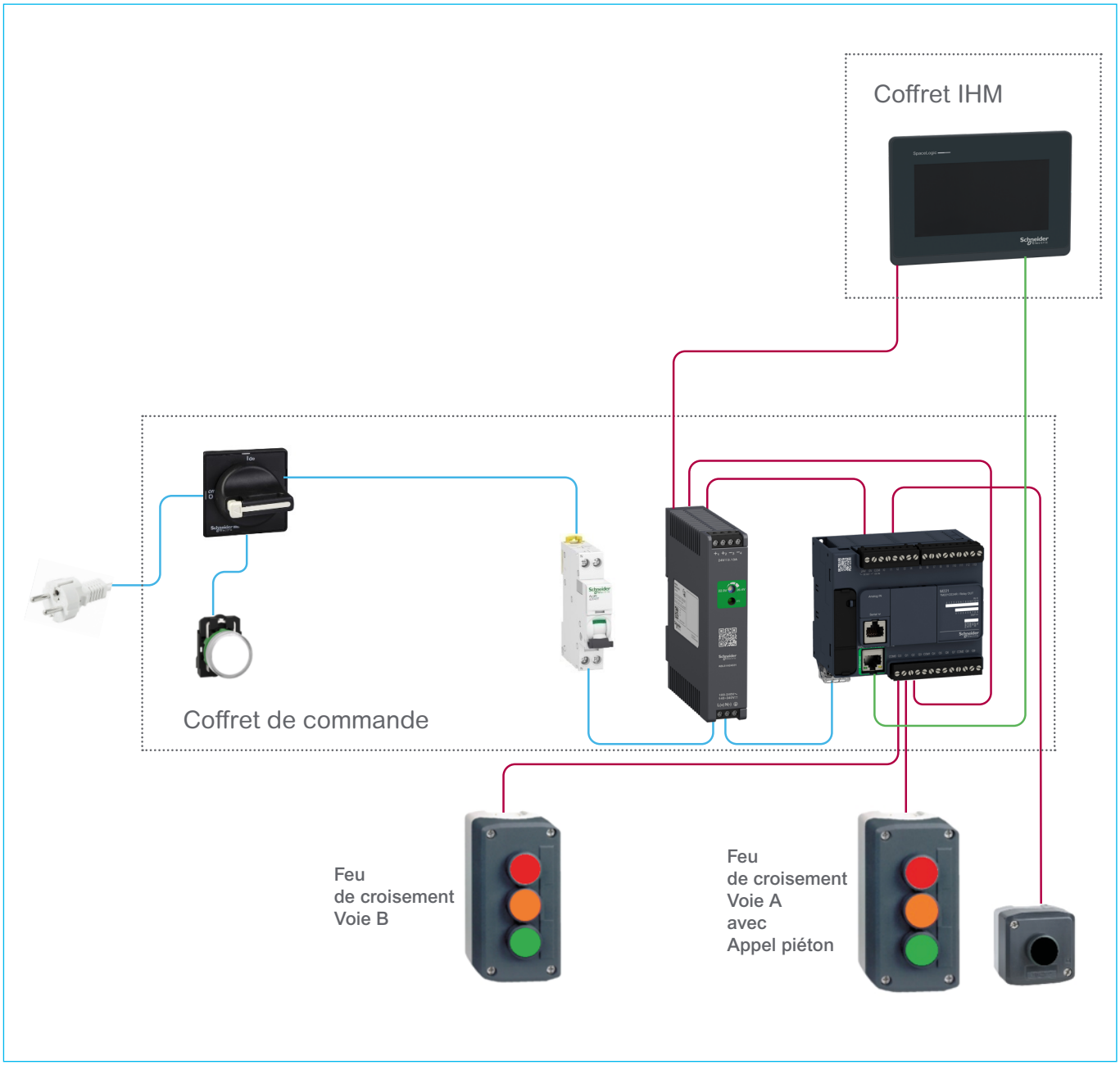
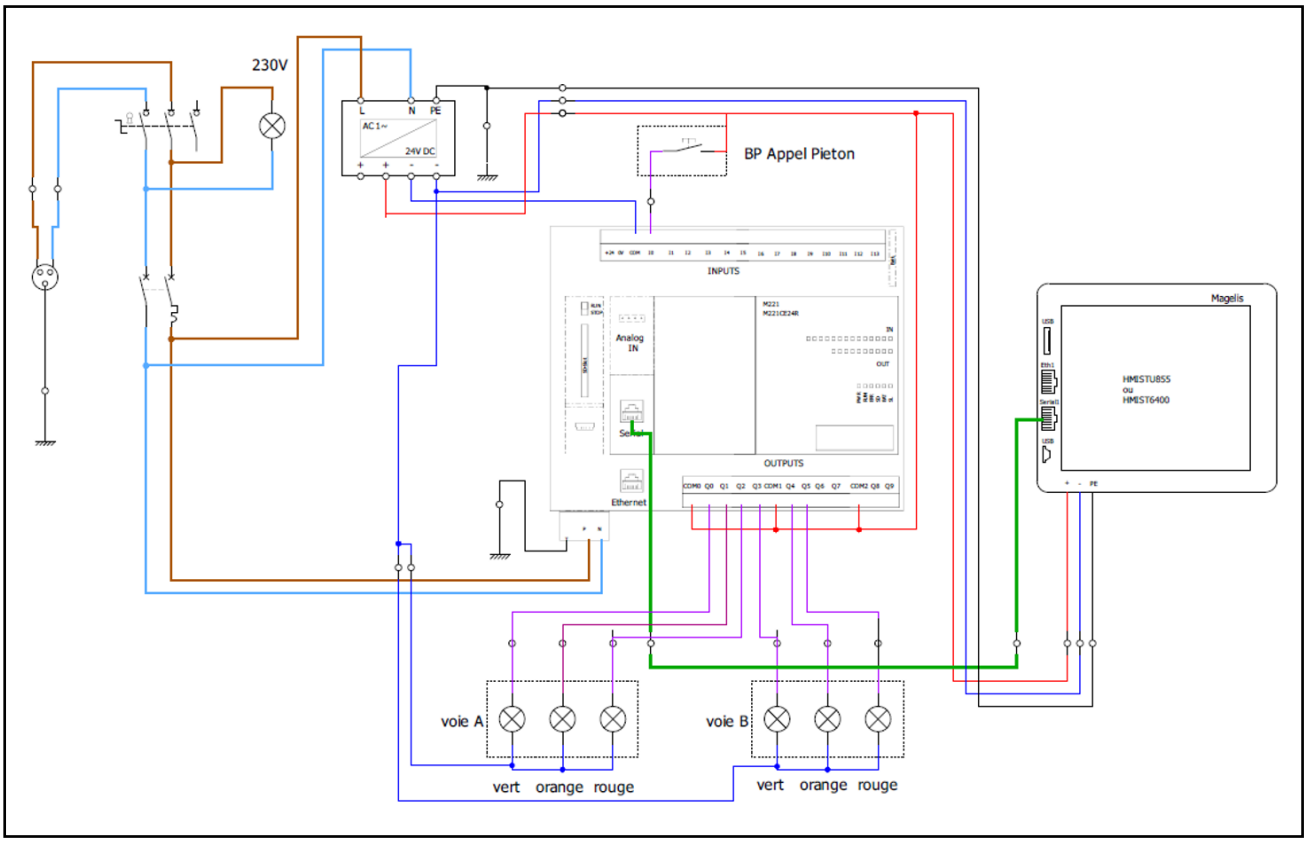
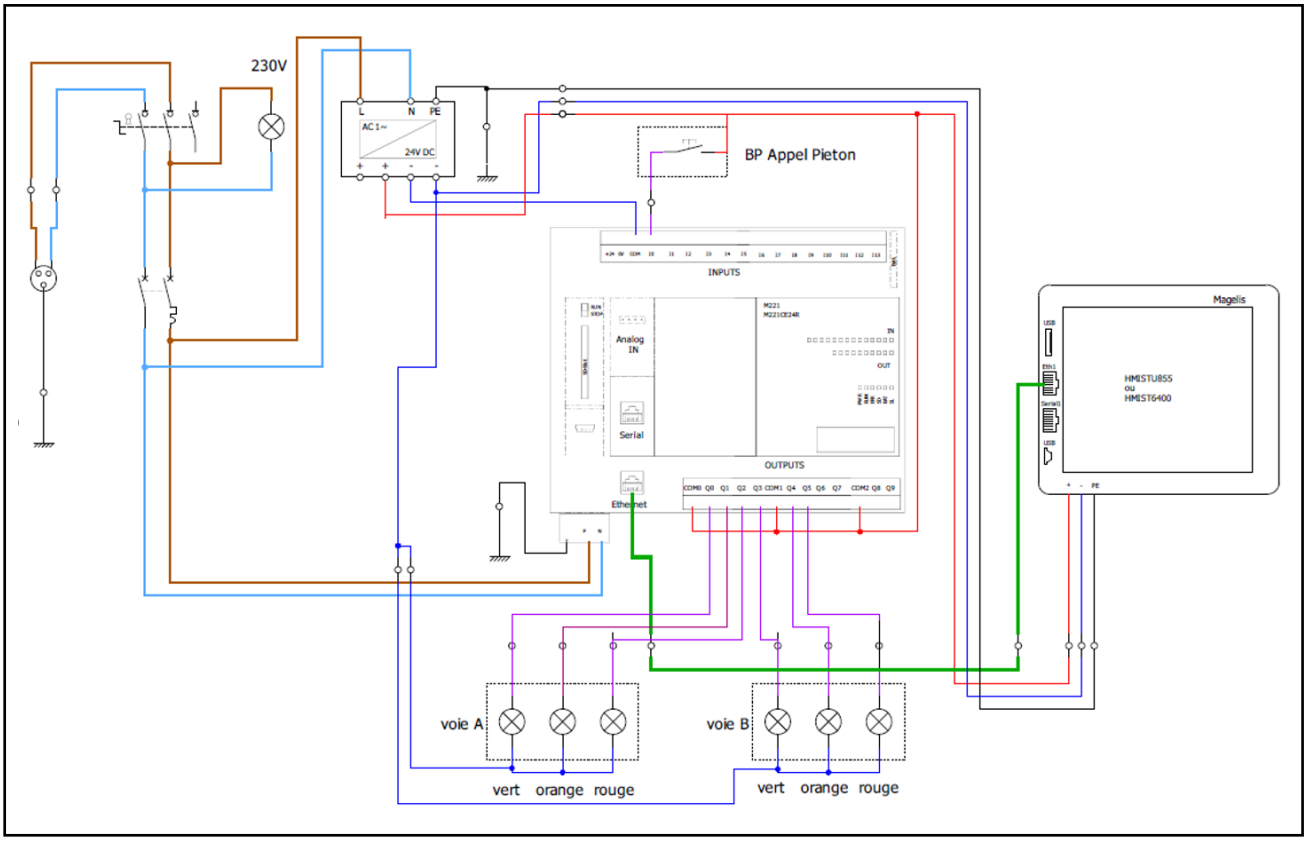


Schéma de câblage

Version Modbus Série



Version Modbus TCP



Réalisation du câblage

- Usiner le couvercle de la boîte industrielle pour fixer l'IHM en respectant la notice de montage du produit.
- Usiner le fond de la boîte industrielle pour fixer l'interface RJ45.
- Ajouter une interface RJ45 dans le coffret de l'automate.
- Raccorder le connecteur de l'alimentation de l'IHM dans le coffret de l'automate sur des bornes 24VDC.
- Raccorder les cordons RJ45 sur les ports Modbus Série ou Ethernet (Modbus TCP) en fonction de la configuration retenue.



Écriture du programme

Les étapes clés :

- définition des variables entre l'automate et l'IHM
- définition de la communication Modbus série / Modbus TCP
- paramétrage des liaisons de communication
- analyse du programme automate
- création des variables dans l'IHM
- création des écrans IHM
- transférer des applications et essais

Descriptif du programme applicatif

Etape 1 : définition des variables entre l'automate et l'IHM

• Etat des feux (couleurs)

- Voie A (feu Rouge, feu Orange, feu Vert)
- Voie B (feu Rouge, feu Orange, feu Vert)
- Total de 6 variables (bool)

• Bouton d'appel piéton

- 1 variable (bool) modifiable

• Durées restantes d'allumage des différentes couleurs

- Voie A (feu Rouge, feu Orange, feu Vert)
- Voie B (feu Rouge, feu Orange, feu Vert)
- Total de 6 variables (intégrer)

• Modification des temporisations d'allumage des différentes couleurs

- Temporisation Vert
- Temporisation Orange
- Temporisation Rouge
- Total de 3 variables (intégrer) modifiables

Etape 2 : communication Modbus série / Modbus TCP

Il existe deux types de communication entre un automate et un terminal graphique :

• Modbus Série

Le protocole Modbus RTU est un moyen de communication permettant l'échange de données entre un automate (API) et un terminal graphique (IHM). Les appareils électroniques peuvent échanger des informations via des lignes série à l'aide du protocole Modbus.

Le protocole a été développé et publié par Modicon® en 1979 pour être utilisé avec ses contrôleurs logiques programmables.
Il est basé sur une architecture maître/esclave et supporte les périphériques série utilisant les protocoles RS232/RS485/RS422. Modbus intervient souvent dans les cas où de nombreux appareils de mesure et de contrôle doivent transmettre des signaux vers un système ou un contrôleur central pour récupérer des données et les analyser.

L'automatisation industrielle et les systèmes de contrôle et d'acquisition de données en temps réel (SCADA) utilisent également le protocole Modbus de manière intensive.

Modbus RTU (Remote Terminal Unit) est l'un des deux modes de transmission définis dans la spécification Modbus originale. Ces deux modes, Modbus RTU et ASCII, sont conçus pour être utilisés avec les périphériques série supportant les protocoles RS232, RS485 et RS422. L'une des caractéristiques spécifiques de Modbus RTU est son utilisation du codage binaire et sa méthode de recherche d'erreurs CRC avancée.

Modbus RTU est l'implémentation du protocole Modbus la plus utilisée pour les applications industrielles et les systèmes de production automatisés.

Dans ce projet, l'automate est de type TM221, il dispose d'un port de communication série paramétrable en RS485 2 fils / Modbus RTU
Le terminal graphique de type HMISTU ou HMIST6 dispose également d'un port de communication série paramétrable en RS485 2 fils / Modbus RTU.

La communication Modbus Série sera donc définie comme suit :

Média : liaison série RS485 2 fils
Protocole : Modbus RTU
Paramétrage des ports :

- Vitesse : 19200 bauds
- Parité : Paire
- Bits de données : 8
- Bits d'arrêt : 1

> Maître / Esclave Modbus RTU

Maître Modbus RTU

Le maître Modbus RTU est le périphérique central demandant des informations aux périphériques esclaves qui y sont connectés. Un contrôleur central dans un système de production automatisé peut jouer le rôle d'un maître Modbus RTU. Une implémentation Modbus ne peut avoir qu'un seul maître. Les périphériques maîtres collectent les informations des esclaves et peuvent également écrire dans les registres des périphériques esclaves.

Esclave Modbus RTU

L'esclave Modbus RTU est le périphérique répondant aux requêtes envoyées par le maître. Il ne peut pas initier les transferts d'informations et est en phase d'attente tant qu'il n'a pas à répondre à une requête du maître.
Il y a un seul périphérique maître dans une implémentation Modbus RTU et il peut y avoir jusqu'à 247 périphériques esclaves. Chaque esclave est identifié par une adresse numérotée de 1 à 247.

Au cœur du protocole Modbus se trouve le composant appelé Protocol Data Unit (PDU). Le PDU se compose de données et d'un code de fonction créés toujours de la même manière quel que soit le mode de transmission Modbus utilisé.
Le code de fonction indique quelles données sont demandées par le maître.
Les codes de fonction indiquent comment le maître interagit avec le périphérique esclave spécifié par l'identifiant de ce dernier.
Selon le code de fonction envoyé, le périphérique maître peut lire l'un des registres de l'esclave ou y écrire.

Dans le projet, l'IHM est à l'origine des requêtes, il est défini comme maître sur le bus Modbus.
L'automate met à dispositions les variables (en lecture/écriture) il est esclave sur le bus Modbus.

L'adresse Modbubs de l'automate est : @1

• Ethernet (Modbus TCP)

Modbus TCP est une extension du protocole permettant aux protocoles Modbus de pouvoir fonctionner sur des réseaux TCP/IP.

La latence inhérente et les autres aspects de la communication sur un réseau ont nécessité quelques modifications sur le mode de synchronisation des requêtes et des réponses entre elles afin d'assurer que des données non souhaitées ne soient pas envoyées par un périphérique esclave.

Modbus TCP met en évidence une différence dans la manière avec laquelle le PDU est empaqueté par rapport à Modbus RTU.

Le cadre TCP contenant le PDU débute avec un identifiant de transaction de protocole d'application MODBUS (MBAP) de 2 octets au lieu d'utiliser l'identifiant du périphérique esclave.

Les équipements doivent disposer d'une adresse IP et masque de sous réseau et appartenir à la même classe de réseau.
Le réseau Ethernet relie les différents ports Ethernet avec un switch (ou routeur) mais celui-ci n'est pas indispensable s'il n'y a que 2 équipements.

Dans ce projet, la communication Modbus TCP sera définie comme suit :
La classe du réseau est C.

Les adresses IP seront :

Equipement 1 (automate) :

- Adresse IP : 192.168.1.10
- Masque de sous réseau : 255.255.255.0

Equipement 2 (IHM) :

- Adresse IP : 192.168.1.20
- Masque de sous réseau : 255.255.255.0

> Client / Serveur Modbus RTU

Dans le projet, l'IHM est à l'origine des requêtes, il est défini comme client Modbus TCP.

L'automate met à disposition les variables (en lecture/écriture), il est défini comme serveur Modbus TCP.

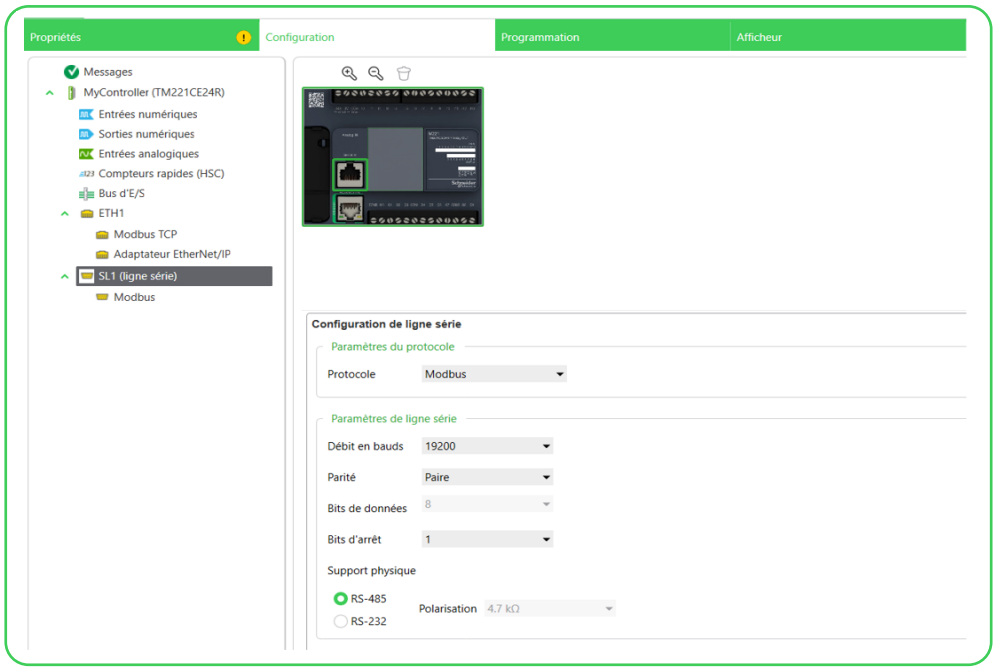
Etape 3 : paramétrage des liaisons de communication

Liaison Modbus Série

> Paramétrage de l'automate TM221 :

Utilisation du logiciel Machine Expert Basic.

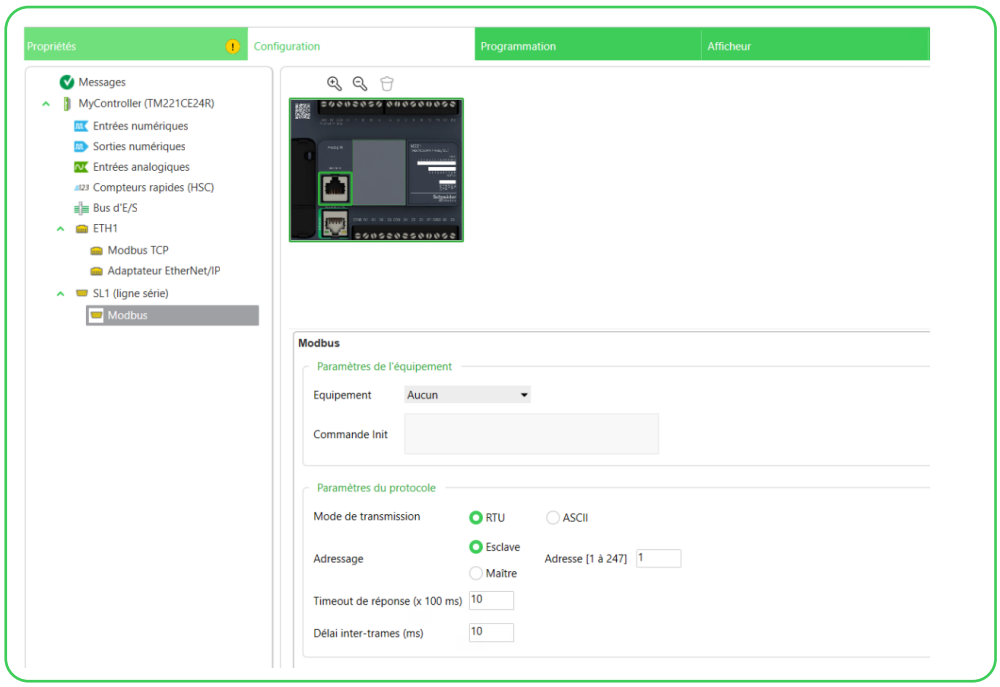
Liaison Série



> Dans l'onglet **« Configuration »,** cliquer sur la section **« SL1 (ligne série) »**.

- Définir le protocole sur « Modbus »
- Débit en bauds : 19200
- Parité : Paire
- Bits de données : 8
- Bits d'arrêt : 1
- Support physique : RS-485

Modbus :



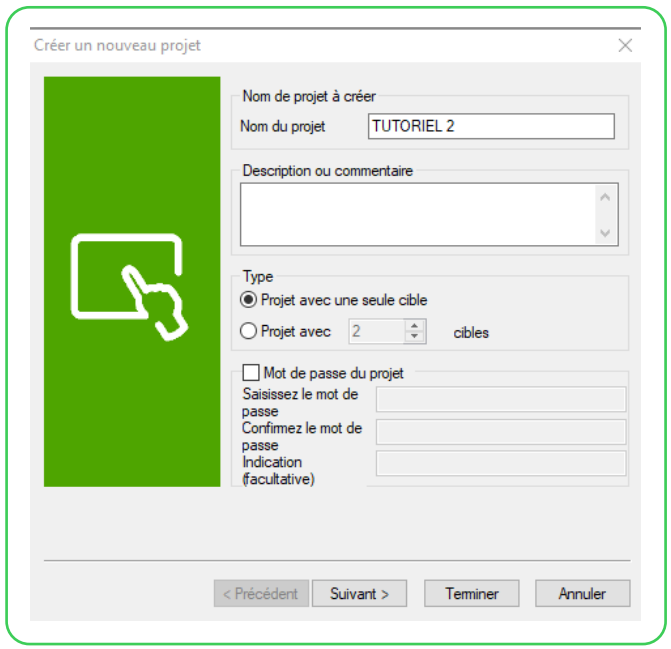
Dans l'onglet **« Configuration »**, cliquer sur la section **« Modbus ».**

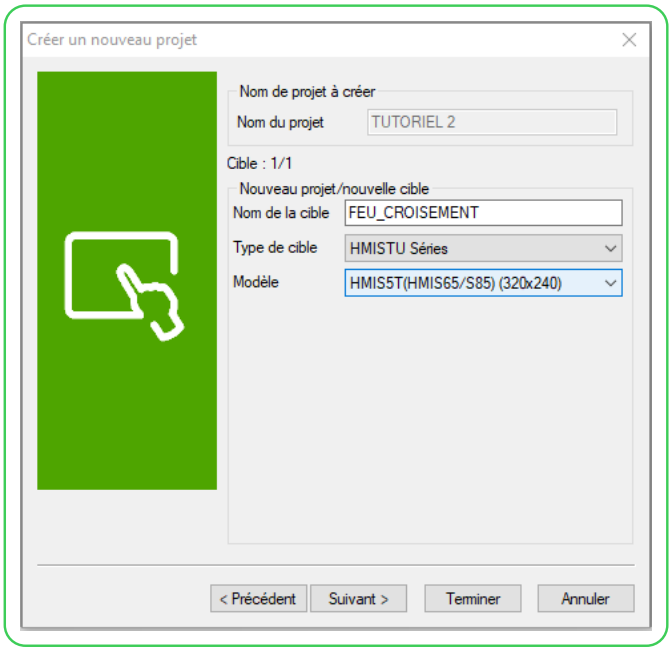
Paramètres du protocole :

- Mode de transmission : RTU
- Adressage : Esclave
- Adresse : 1
- Timeout de réponse (x100ms) : 10 (par défaut)
- Délai inter-trames (ms) : 10 (par défaut)

> Paramétrage de l'IHM HMISTU855:

Utilisation du logiciel Vijeo Designer.
Créer un nouveau projet.



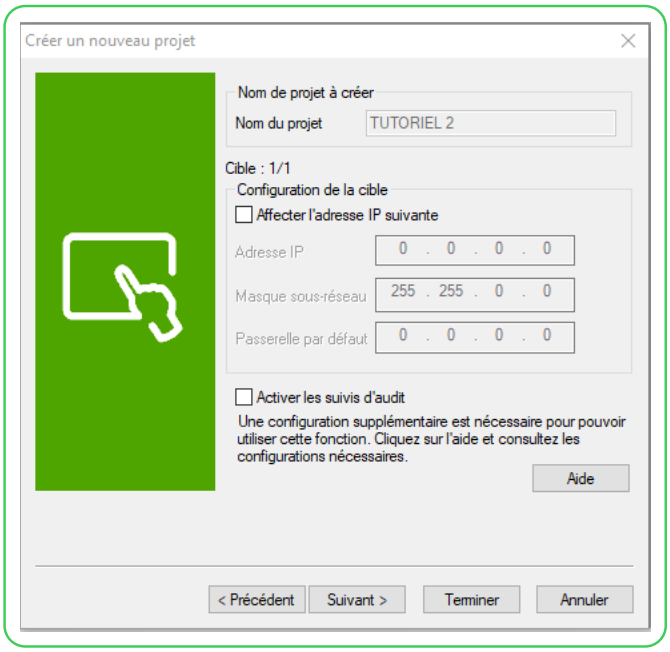


Le nom de la cible est défini : **FEU_CROISEMENT**.
Le type de la cible est sélectionné en fonction de la référence du matériel.
Dans ce projet, l'IHM porte la référence HMISTU855.

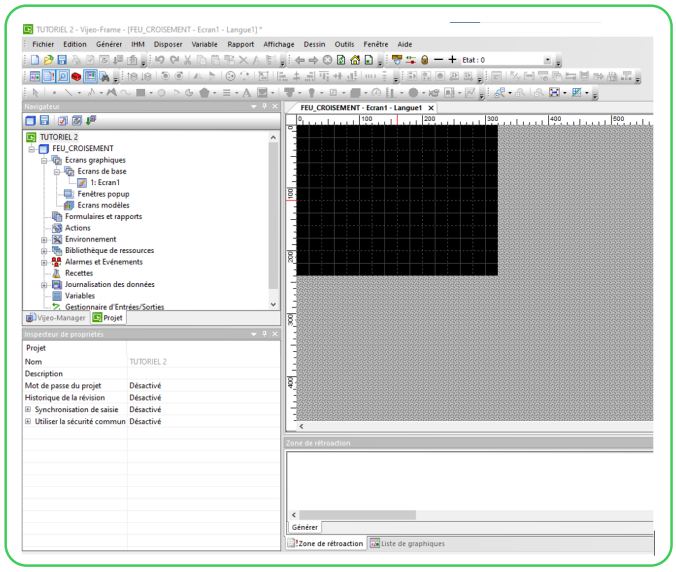
Cette référence est composée de :

- HMIS85 (l'écran)
- HMIS5T (le contrôleur)

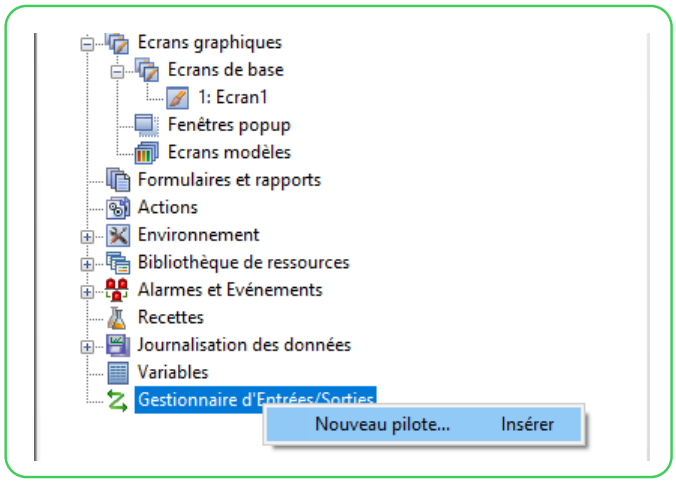
Le type de cible est : **HMISTU Séries**.
Le modèle est : **HMIS5T (HMIS65/S85) (320x240)**.
Cliquer ensuite sur **« Suivant »**.



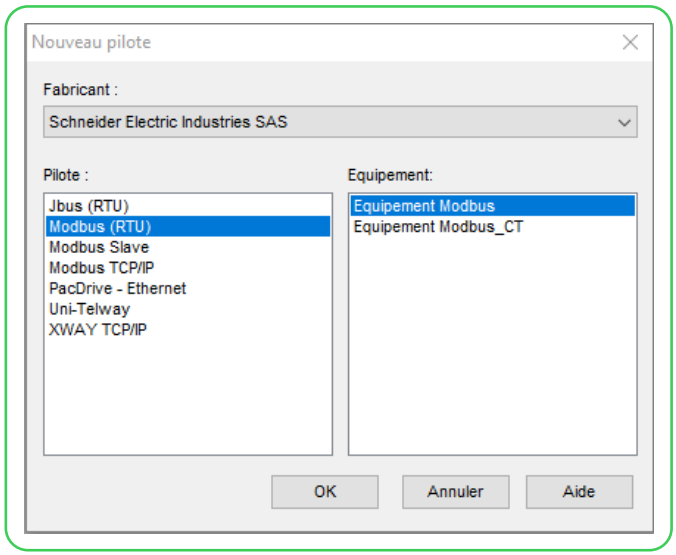
La configuration de la cible sera effectuée directement dans le projet.
Cliquer sur **« Terminer »**.



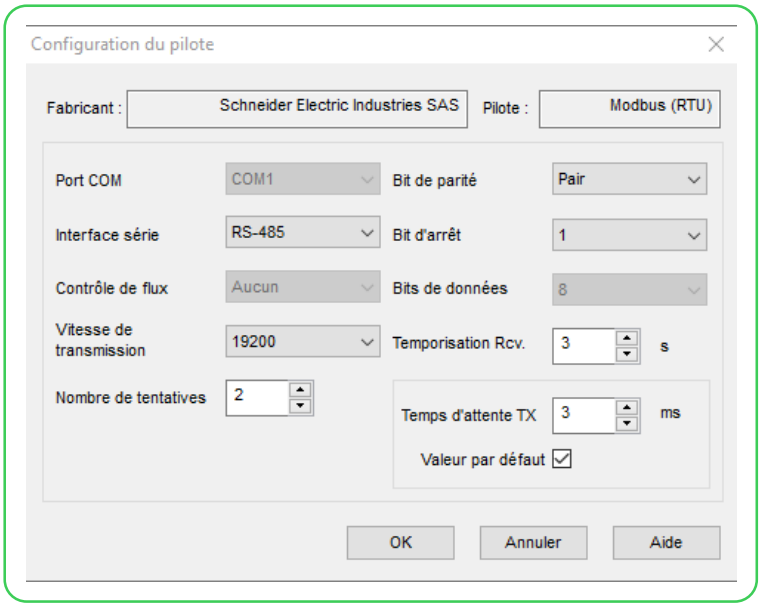
Le projet Vijeo Designer s'ouvre avec un écran vide par défaut.
Dans le menu de gauche, **cliquer droit sur « Gestionnaire d'Entrées/Sorties ».**



Une fenêtre s'ouvre pour définir le nouveau pilote de communication.



Dans la liste Pilote, sélectionner **« Modbus (RTU) »** puis dans la liste Equipement, sélectionner **« Equipement Modbus ».**
Cliquer sur **« OK »** pour valider.

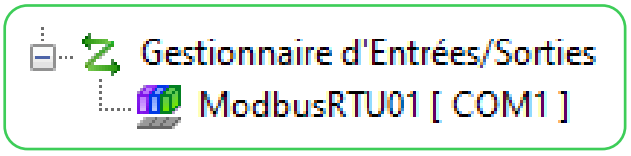


Vérifier que les paramètres de communication RS485 correspondent au cahier des charges :

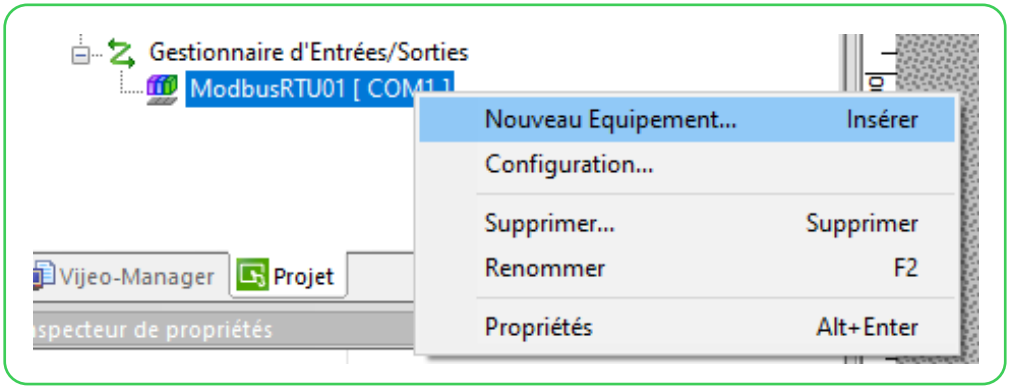
- Vitesse : 19200 bauds
- Parité : Paire
- Bits de données : 8
- Bit d'arrêt : 1

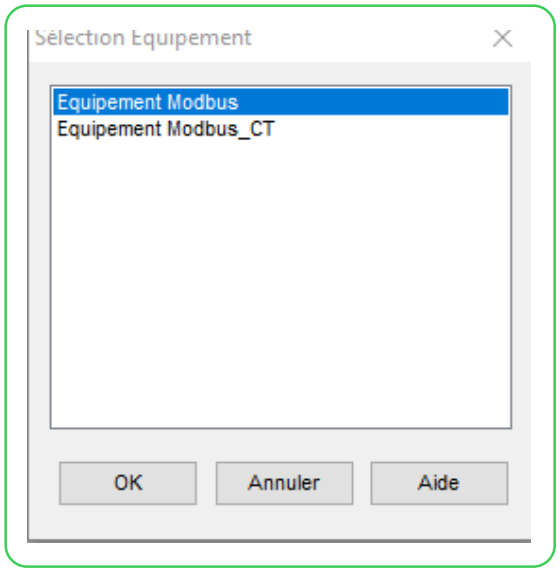
Cliquer sur **OK** pour valider.

Un nouvel pilote apparaît sous le Gestionnaire d'Entrées/Sorties.

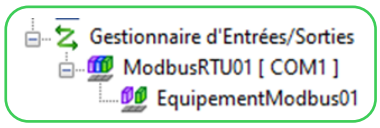


Cliquer droit sur le pilote ModbusRTU01 et sélectionner « Nouvel Equipement ».

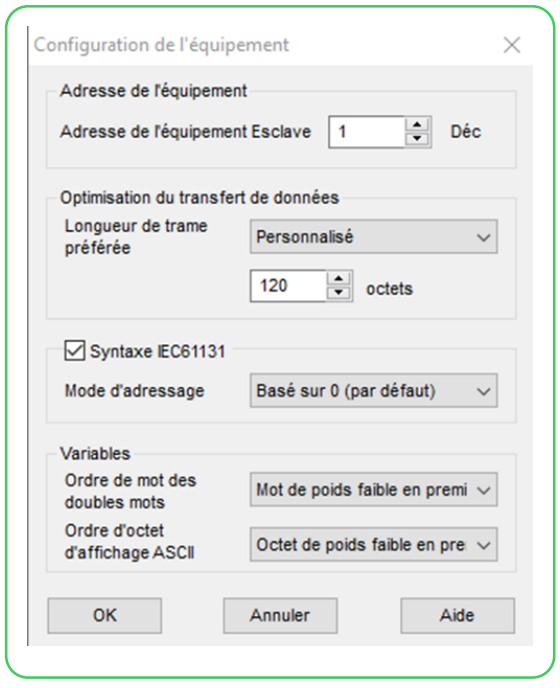




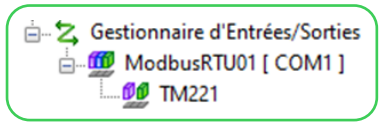
Sélectionner **« Equipement Modbus ».**
Un nouvel équipement apparaît sous le pilote Modbus.



Une fenêtre de paramétrage s'ouvre.



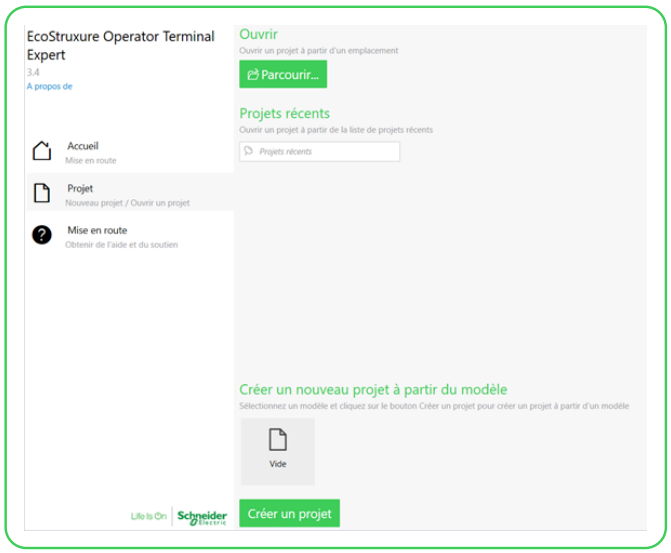
L'adresse de l'équipement Esclave est définie sur 1. Cette adresse correspond à l'adresse Modbus de l'automate.
Cocher la case **« Syntaxe IEC61131 »** pour définir le format des variables en %M et %MW.
Cliquer sur **OK** pour valider.
Cliquer sur l'équipement pour le **renommer « TM221 ».**



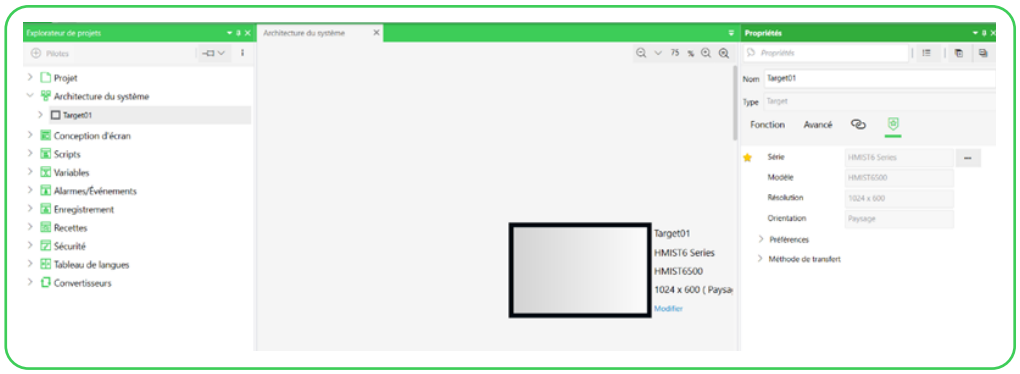
Le projet est paramétré pour une communication Modbus RTU (RS485) avec l'automate.
Enregistrer le projet.

> Paramétrage de l'IHM HMIST6400:

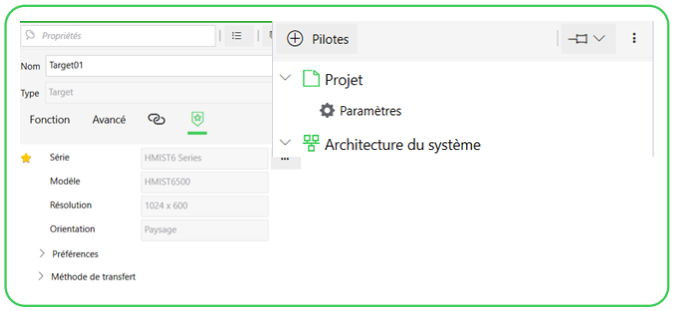
Utilisation du logiciel Operator Terminal Expert.
Créer un nouveau projet.



Le projet est enregistré et renommé **« TUTORIEL 2 »**.



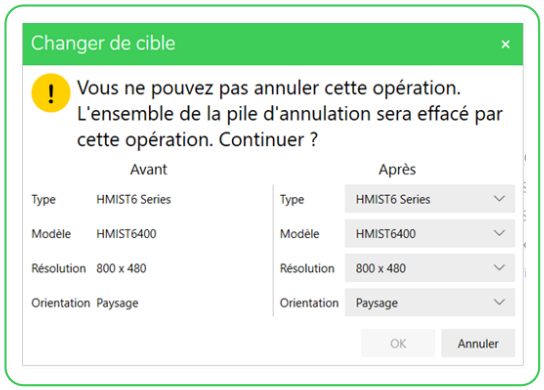
Cliquer sur l'onglet **« Architecture système ».**



Le nom de la cible est défini : FEU_CROISEMENT

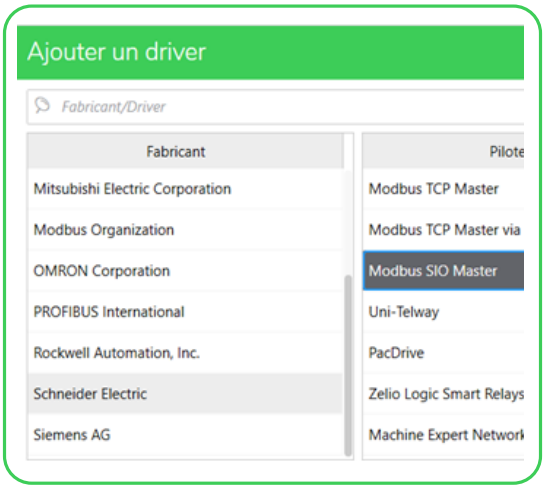
Cliquer sur les trois petits points pour modifier la référence du matériel.
Le type de la cible est sélectionné en fonction de la référence du matériel.
Dans ce projet, l'IHM porte la référence HMIST6400.

- Le type de cible est : HMIST6 Séries
- Le modèle est : HMIST6400 (800x480)

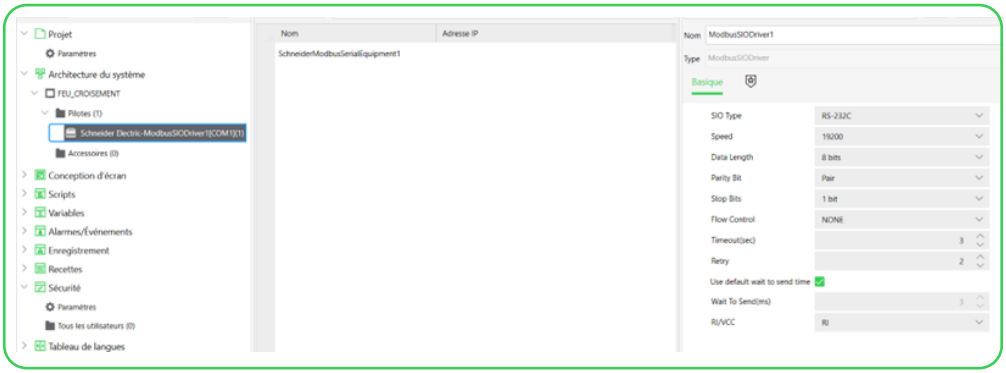


Cliquer ensuite sur **« OK ».**

Pour paramétrer l'application en Modbus serie, cliquer sur le bouton **« + Pilotes »** en haut à gauche de l'écran.
Dans l'onglet **« fabricant »,** choisir **« Schneider Electric ».**
Dans l'onglet **« pilotes »,** choisir **« Modbus SIO Master ».**



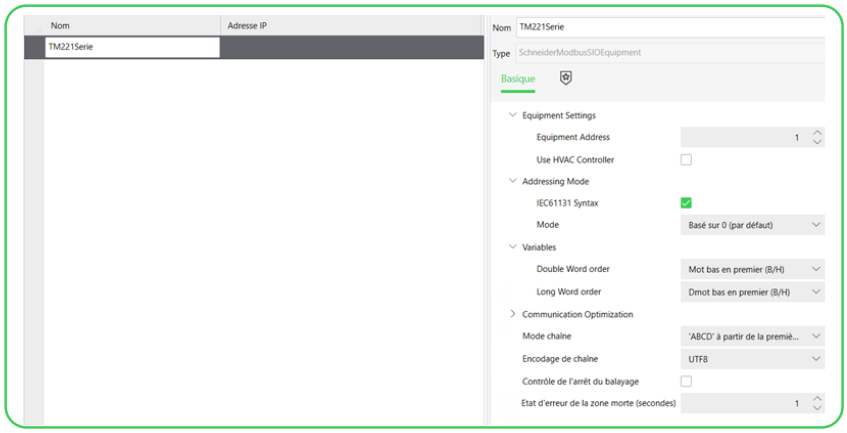
Cliquer sur **« OK »** pour valider l'ajout du pilote.
Ensuite, cliquer sur le **nouveau pilote dans l'onglet « Architecture système ».**



Vérifier que les paramètres de communication RS485 correspondent au cahier des charges :

- Vitesse : 19200 bauds
- Parité : Paire
- Bits de données : 8
- Bit d'arrêt : 1

Renommer ce pilote « TM221Serie ».
Cliquer sur l'équipement pour accéder à l'autre section de paramètres.



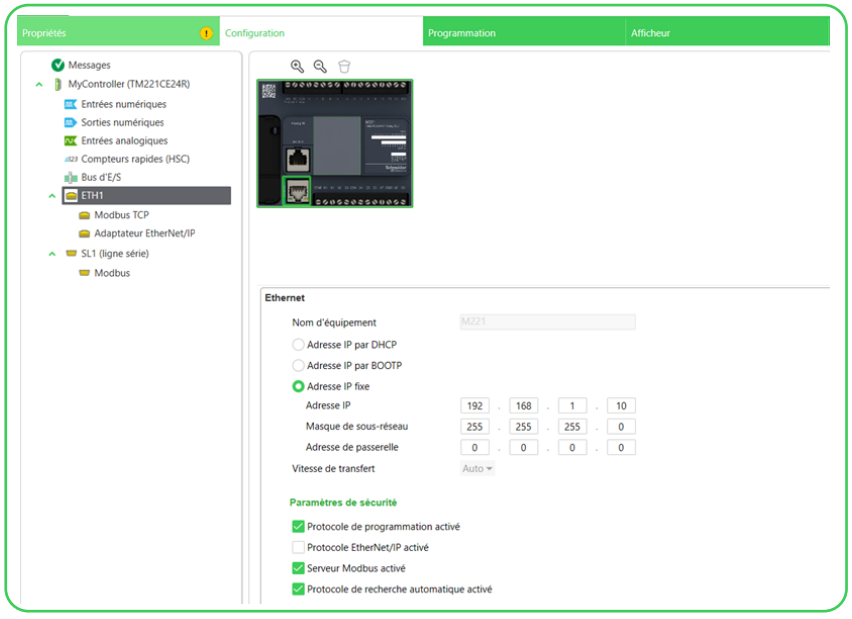
Définir l'adresse de l'équipement : @1
Cocher la case **« Syntaxe IEC61131 »** pour définir le format des variables en %M et %MW.

Enregistrer le projet

> Liaison Modbus TCP :

Paramétrage de l'automate M221.

Utilisation du logiciel Machine Expert Basic.



Dans l'onglet **« Configuration »**, cliquez sur la section **« ETH1 »**.
Cocher la case **« Adresse IP fixe »** et **saisir les informations de l'adresse Ip et du masque de sous réseau.**

- Adresse IP : 192.168.1.10
- Masque de sous réseau : 255.255.255.0
- Adresse de la passerelle : 0.0.0.0 (par défaut si non utilisée)

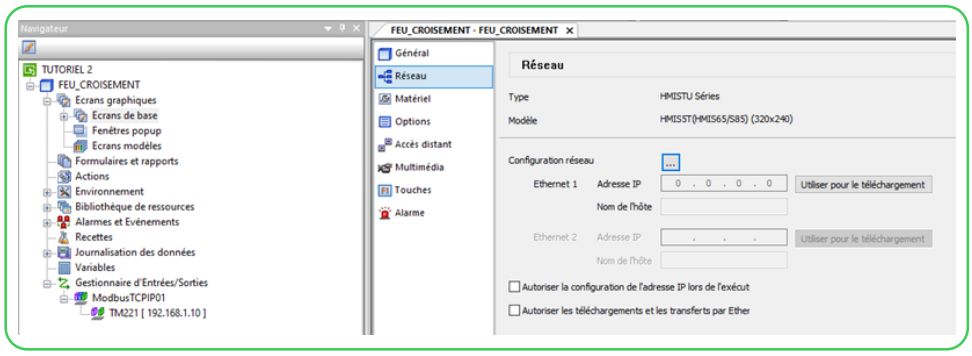
Dans les paramètres de sécurité, **cocher les cases :**

- Protocole de programmation active (autorise la programmation de l'automate via la liaison Ethernet)
- Serveur Modbus activé (autorise les échanges en Modbus TCP avec un client)
- Protocole de recherche automatique activé (autorise la découverte automatique des équipements sur le réseau)
- La liaison Ethernet Modbus TCP est paramétrée pour une communication avec l'IHM.

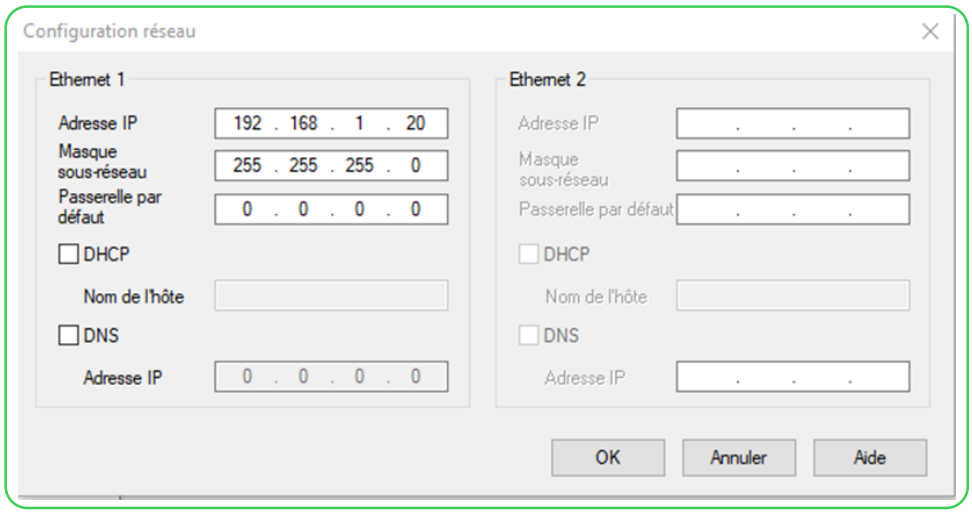
> Paramétrage de l'IHM HMISTU855 :

Utilisation du logiciel Vijeo Designer.

Reprendre le projet créé pour Modbus RTU.
Dans le navigateur du projet (sur la gauche) cliquer sur la cible **(FEU_CROISEMENT)**.
Dans la fenêtre principale, sélectionner l'onglet **« Réseau »**.



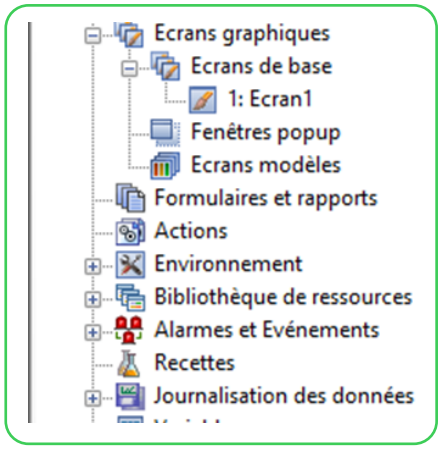
Cliquer sur le bouton **« … »** de la **« Configuration réseau »**.
Une fenêtre s'ouvre pour configurer le port Ethernet de l'IHM.



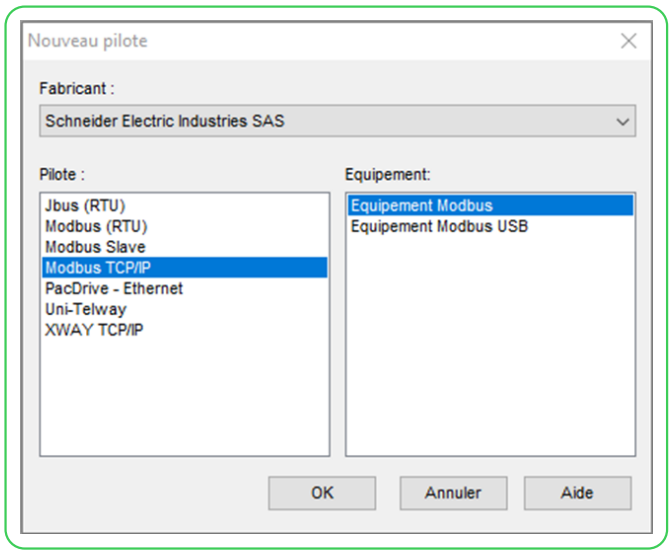
Saisir les informations souhaitées :

- Adresse IP : 192.168.1.20
- Masque de sous réseau : 255.255.255.0
- Cliquer sur OK pour valider.

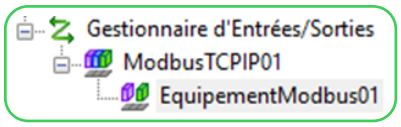
Dans le Gestionnaire d'Entrées/Sorties, **supprimer le pilote ModbusRTU01.**
Puis **cliquer droit pour ajouter un nouveau pilote.**



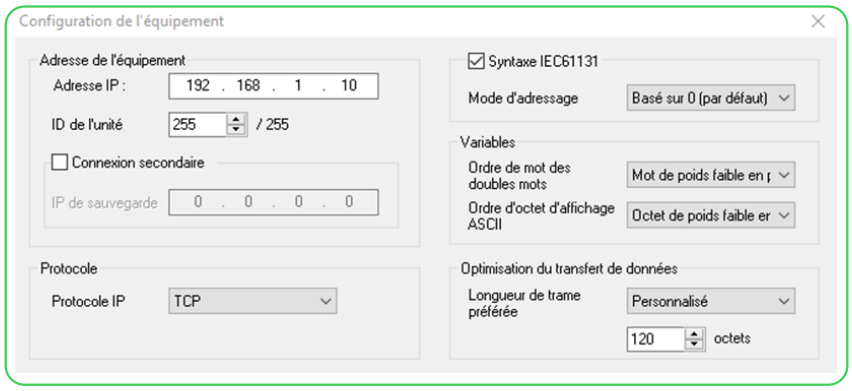
Une fenêtre s'ouvre pour définir le nouveau pilote de communication.



Sélectionner le Pilote **« Modbus TCP/IP »** et l'Equipement **« Equipement Modbus ».**
Cliquer sur OK pour valider.
Un nouveau pilote et un nouvel équipement apparaissent sous le gestionnaire d'Entrées/Sorties.



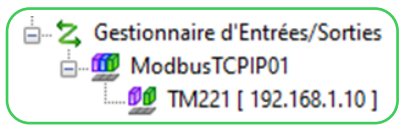
Une fenêtre s'ouvre pour paramétrer ce nouvel équipement.



Saisir les informations souhaitées :

- Adresse IP : 192.168.1.10 (correspondant à l'adresse IP de l'automate).
- ID de l'unité : 255 correspondant à l'adresse Modbus de l'automate en ModbusTCP.
- Cocher la case « Syntaxe IEC61131 » pour définir le format des variables %M et %MW.
- Sélectionner l'ordre des mots pour la variables doubles mots : Mot de poids faible en premier.

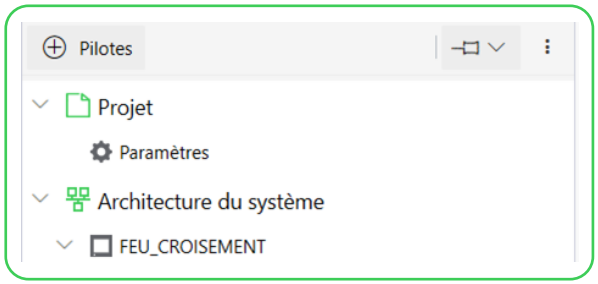
Cliquer sur **OK** pour valider.
Cliquer sur le **nom de l'équipement** pour le renommer : **TM221.**



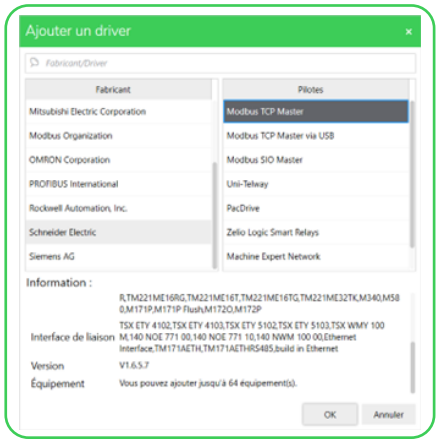
Il est noté avec son adresse IP.
Le projet est paramétré pour une communication Modbus RTU (RS485) avec l'automate.
Enregistrer le projet.

> Paramétrage de l'IHM HMIST6400 :

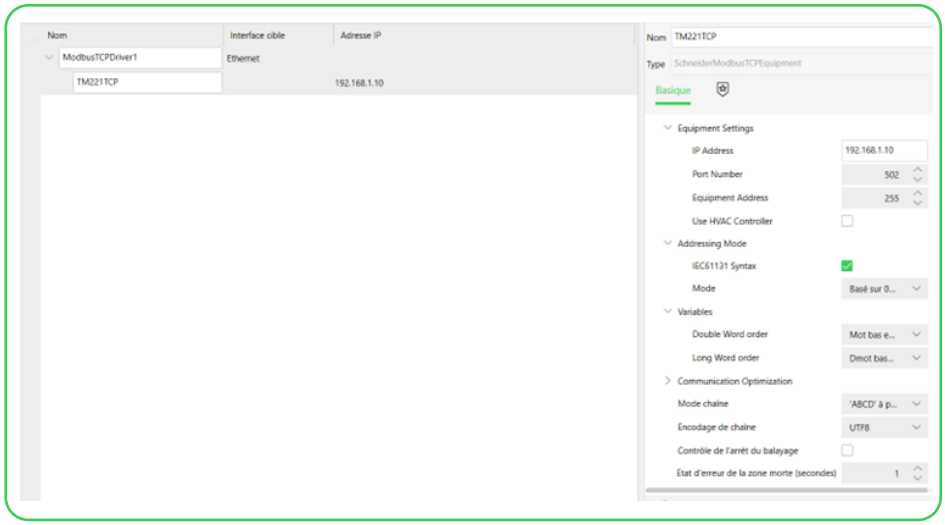
Utilisation du logiciel Operator Terminal Expert.
Reprendre le projet créé pour Modbus RTU.
Retourner dans l'architecture du système, **supprimer le pilote Modbus RTU et ajouter un nouveau pilote.**



Dans l'onglet **« fabricant »,** choisir **« Schneider Electric ».**
Dans l'onglet **« pilotes »**, choisir **« Modbus TCP Master ».**

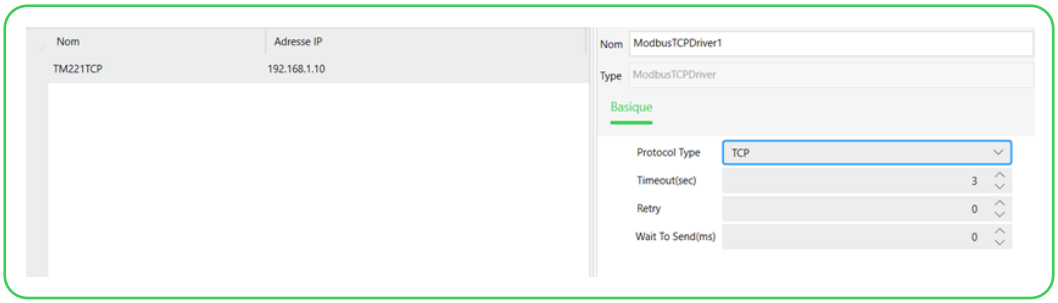


Dans l'onglet **« équipement » :**

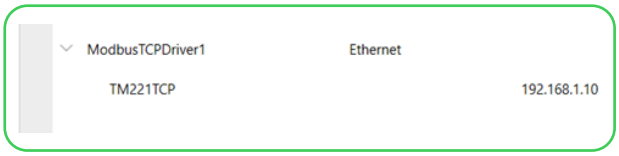


Saisir les informations souhaitées :

- Adresse IP : 192.168.1.10 (correspondant à l'adresse IP de l'automate)
- Port Number : 502 (correspond au numéro de port utilisé pour Modbus)
- ID de l'unité : 255 (correspond à l'adresse Modbus de l'automate en ModbusTCP)
- Cocher la case « Syntaxe IEC61131 » pour définir le format des variables %M et %MW.
- Sélectionner l'ordre des mots pour les variables double mots : Mot bas en premier.



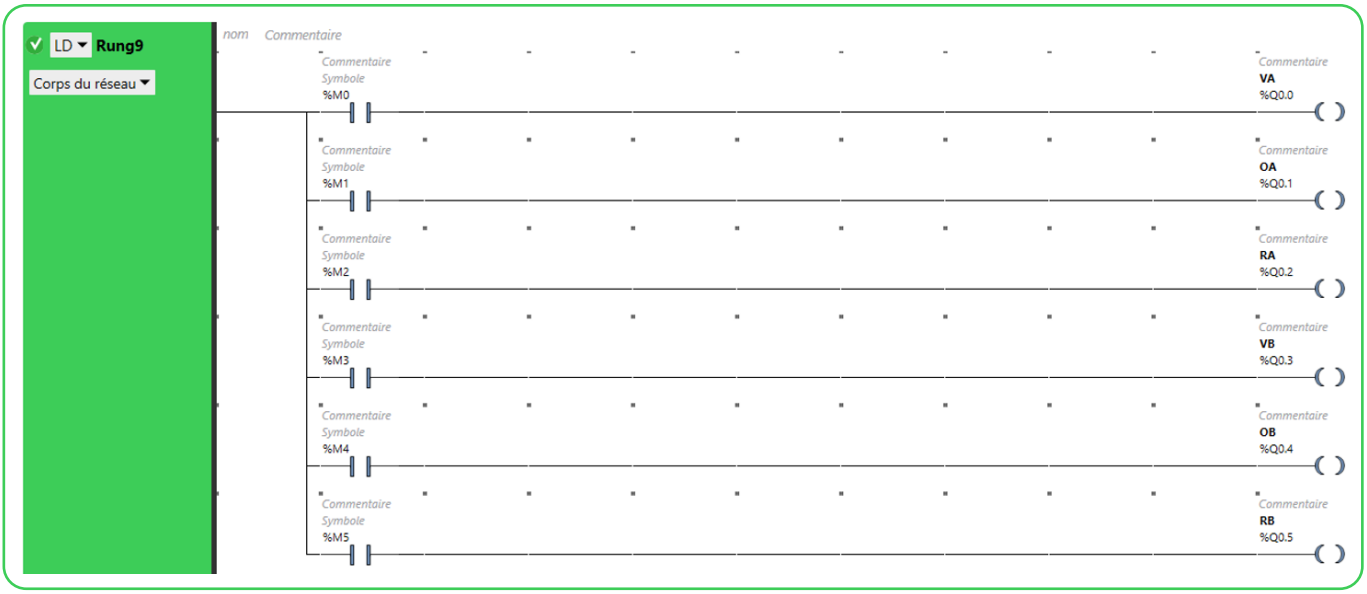
Vérifier en cliquant sur le pilote que le protocole est bien TCP.
Vérifier les préréglages des timeouts et du retry.



Vérifier que l'adresse IP automate dans Modbus TCP est correcte.

> Variables existantes

Dans le programme développé dans le tutoriel automatisme séquentiel, les variables correspondant à l'état des feux Vert, Orange et Rouge sont déjà définies dans le POU « Sorties ».

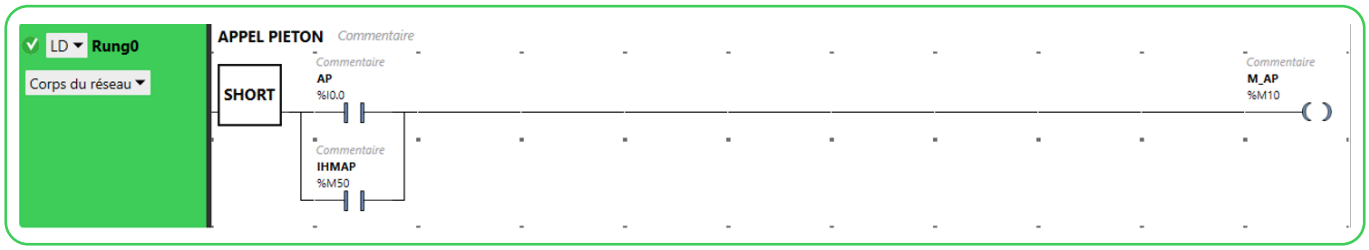


- La variable %M0 est l'image de la sortie %Q0.0 du Feu Vert de la voie A
- La variable %M1 est l'image de la sortie %Q0.1 du Feu Orange de la voie A
- La variable %M2 est l'image de la sortie %Q0.2 du Feu Rouge de la voie A
- La variable %M3 est l'image de la sortie %Q0.3 du Feu Vert de la voie B
- La variable %M4 est l'image de la sortie %Q0.4 du Feu Orange de la voie B
- La variable %M5 est l'image de la sortie %Q0.5 du Feu Rouge de la voie B

> Variables à créer

• Bouton Poussoir pour l'Appel Piéton

Une nouvelle variable de type booléen doit être créée. Cette variable aura le même rôle que l'entrée physique %I0.0 qui correspond au bouton poussoir câblé.
Dans le POU « Entrées », ajouter un contact Normalement Ouvert en parallèle de l'entrée %I0.0.



Cette nouvelle variable est %M50 et son symbole est HMIAP. Lorsque l'IHM changera l'état de cette variable, l'ordre Appel Piéton sera reçu et le cycle des feux sera changé.

• Durées restantes d'allumage des feux

Pour plus de clarté dans le programme, les modifications suivantes seront intégrées dans un nouveau POU nommé IHM.

Dans le cycle du programme, des temporisateurs sont lancés. Pendant la temporisation, le feu correspondant au cycle est allumé. La valeur de cette temporisation est donnée par une variable spécifique au temporisateur.

Pour le temporisateur %TM0 qui correspond au feu vert, la valeur renvoyée est %TM0.V. Cette variable évolue entre 0 et le préréglage du temporisateur (%TM0.P).

Pour la durée d'allumage des feux Orange, c'est similaire. Le temporisateur est %TM1, la valeur est %TM1.V et le préréglage est %TM1.P

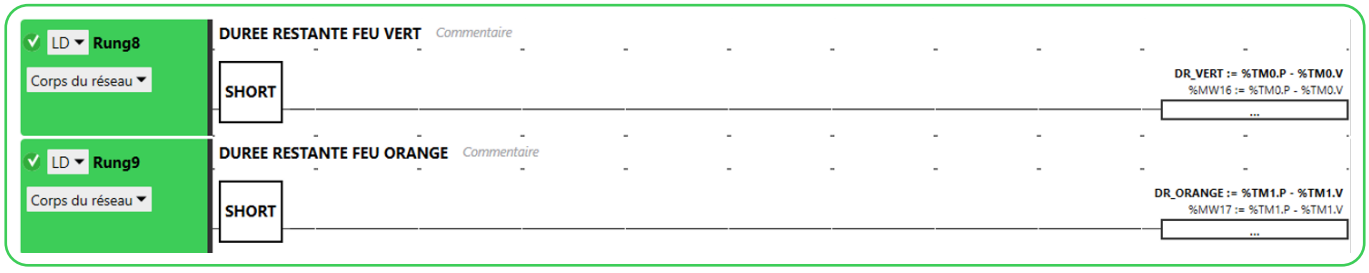
Si on souhaite connaître la durée restante pour l'allumage du feu vert, il faut **faire l'opération suivante :**

✅ Durée restante = préréglage – valeur

Une nouvelle variable de type INTEGER est créée %MW16 son symbole est DR_VERT
%MW16 = %TM0.P - %TM0.V

De la même façon pour le feu orange :

Une nouvelle variable de type INTEGER est créée %MW17 son symbole est DR_ORANGE
%MW17 = %TM1.P - %TM1.V



Pour le feu rouge, il y a deux cas : le feu rouge de la voie A et le feu rouge de la voie B.
Comme indique dans le tutoriel automatisme séquentiel, il existe deux phases ou les feux Rouge des deux voies sont allumés en même temps.

De ce fait, pour la voie A, la durée d'allumage du feu rouge est :
Durée rouge A = durée du temporisateur rouge + durée du temporisateur vert + durée du temporisateur orange + durée du temporisateur rouge.



Ce calcule est conditionné par les étapes du cycle.

Pour la voie B :

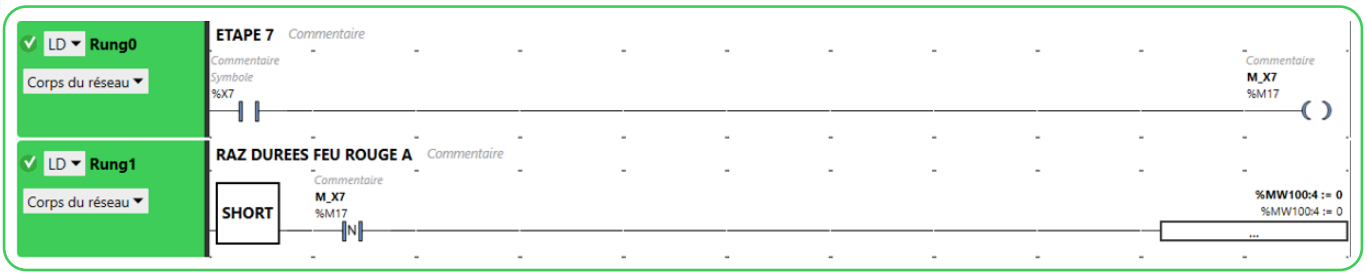
Durée rouge B = durée du temporisateur vert + durée du temporisateur orange + durée du temporisateur rouge + durée du temporisateur rouge.



Ce calcule est conditionné par les étapes du cycle.

Pour que le calcul soit juste il faut **remettre les variables à zéro à la fin de chaque cycle.**
La dernière étape du cycle est l'étape 7 (%X7) elle est mémorisée dans une variable %M17. Lorsque l'étape 7 se désactive, %M17 passe de 1 à 0.

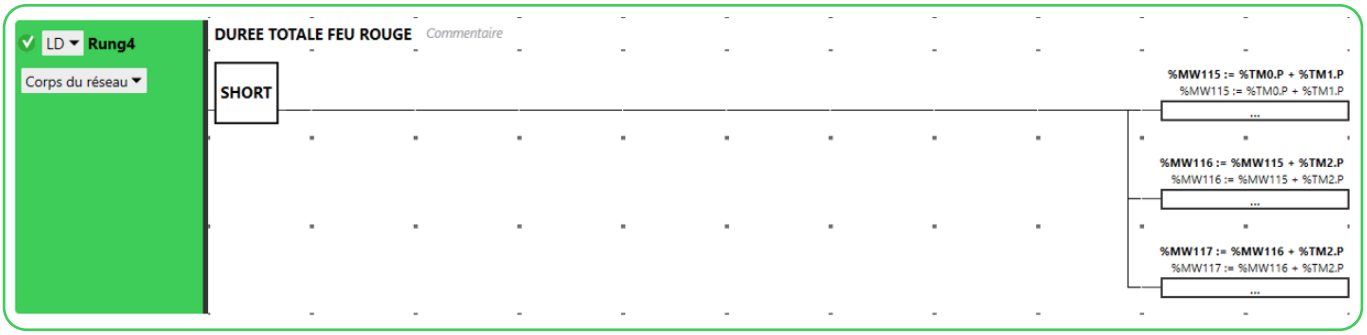
Pour la voie A :



Et pour la voie B :



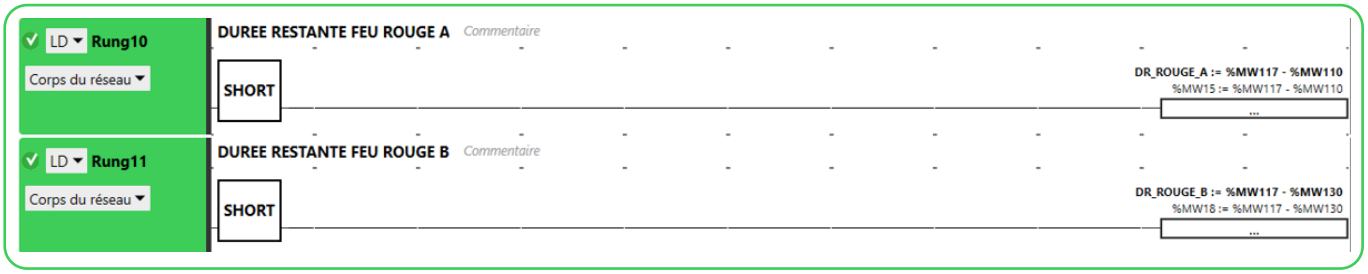
Pour calculer les durées restantes, il faut connaitre la durée totale qui est la somme des préréglages :



La variable %MW117 correspond à la durée totale pour les feux rouges.

Les durées restantes d'allumage des feux Rouge de la voie A et de la voie B sont :

- Voie A : %MW15 = %MW117 - %MW110
- Voie B : %MW18 = %MW117 - %MW130

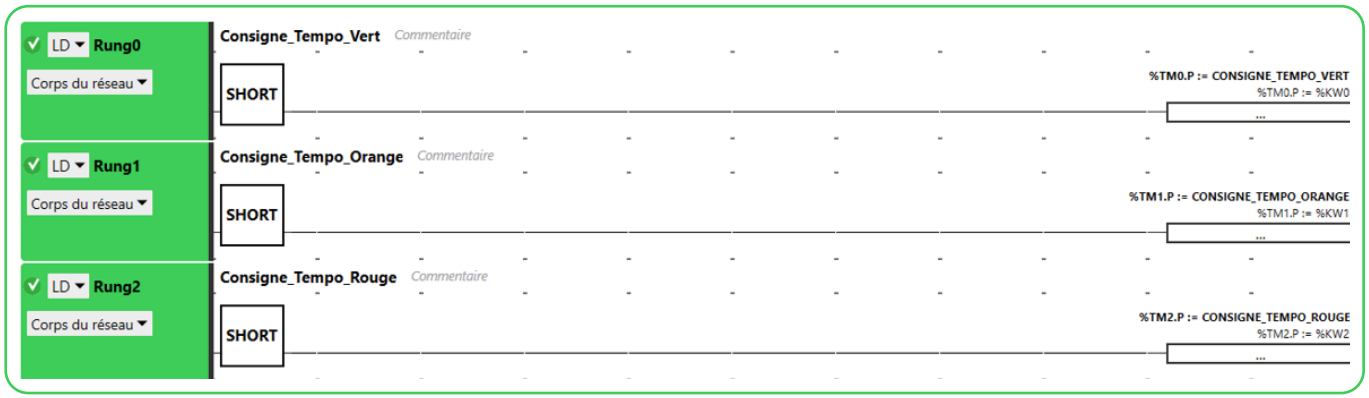


Les 4 variables nécessaires à l'affichage des durées restantes d'allumage des feux sont :

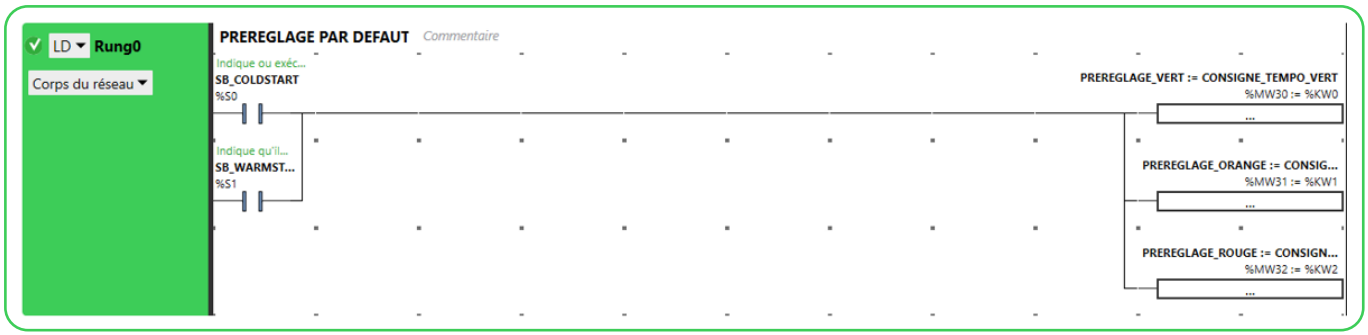
- %MW15 : durée restante feu Rouge voie A
- %MW16 : durée restante feu Vert voie A et voie B
- %MW17 : durée restante feu Orange voie A et voie B
- %MW18 : durée restante feu Rouge voie B

• Préréglage des temporisateurs

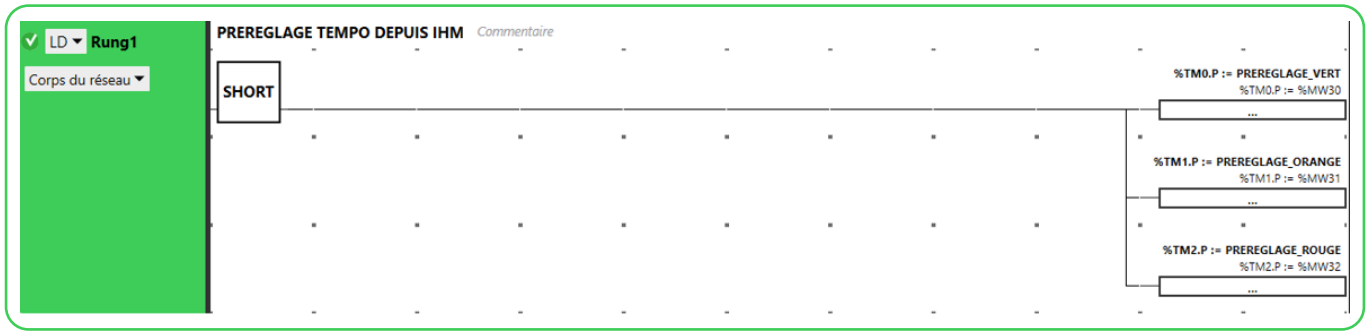
Chaque temporisateur dispose d'un préréglage :
- Pour les feux Vert : %TM0.P
- Pour les feux Orange : %TM1.P
- Pour les feux Rouge : %TM2.P
- Dans le tutoriel automatisme séquentiel ces variables étaient égales à des constantes %KW



Pour pouvoir modifier ces préréglages depuis l'IHM, il faut modifier ces lignes.
Lors du démarrage à chaud ou à froid de l'automate, les constantes sont transférées dans des variables (%MW30, %MW31 et %MW32). Ce transfert ne se fait qu'une seule fois.



Ces variables sont ensuite transférées dans les préréglages des temporisateurs.



Lorsque la variables %MW30 sera modifiée dans l'IHM, le préréglage du temporisateur Vert sera modifié.
Lorsque la variables %MW31 sera modifiée dans l'IHM, le préréglage du temporisateur Orange sera modifié.
Lorsque la variables %MW32 sera modifiée dans l'IHM, le préréglage du temporisateur Rouge sera modifié.

Sauvegarder le programme.

Un tableau permet de récapituler les variables pour les échanges avec l'IHM.

Adresse	Type	Symbole	Lecture Ecriture	Description
%M0	BOOL	VA	R	Etat Feu Vert A
%M1	BOOL	OA	R	Etat Feu Orange A
%M2	BOOL	RA	R	Etat Feu Rouge A
%M3	BOOL	VB	R	Etat Feu Vert B
%M4	BOOL	OB	R	Etat Feu Orange B
%M5	BOOL	RB	R	Etat Feu Rouge B
%M50	BOOL	HMIAP	R/W	Appel Piéton
%MW15	INT	DR_RA	R	Durée restante Feu Rouge Voie A
%MW16	INT	DR_V	R	Durée restante Feu Vert Voie A et Voie B
%MW17	INT	DR_O	R	Durée restante Feu Orange Voie A et Voie B
%MW18	INT	DR_RB	R	Durée restante Feu Rouge Voie B
%MW30	INT	PV	R/W	Préréglage temporisateur Vert
%MW31	INT	PO	R/W	Préréglage temporisateur Orange
%MW32	INT	PR	R/W	Préréglage temporisateur Rouge

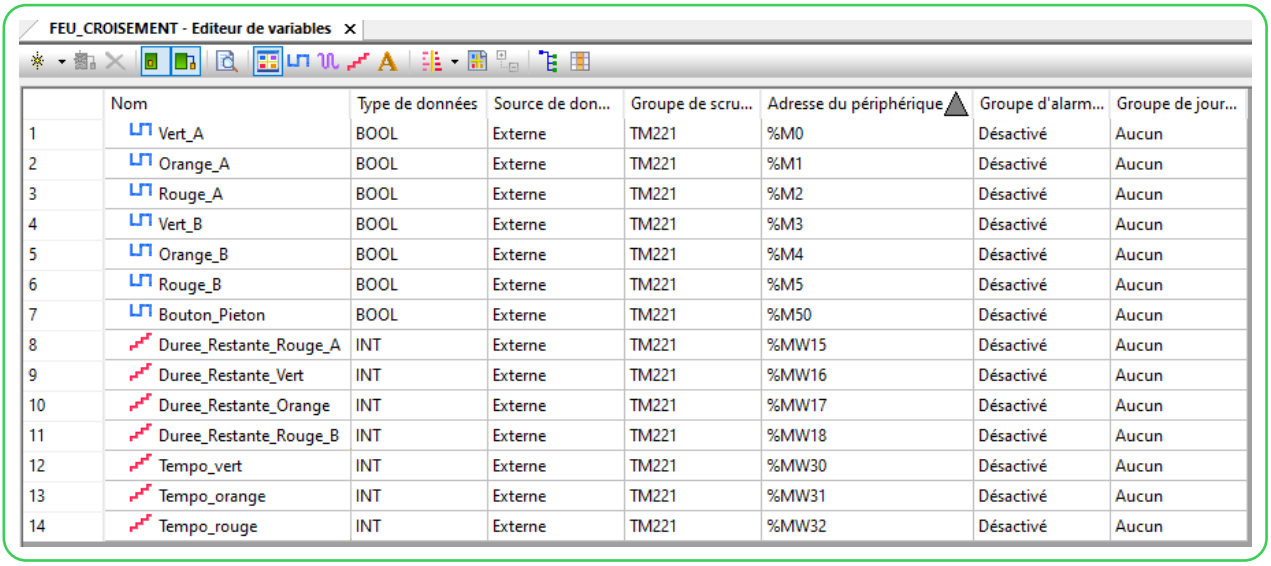
Etape 5 : création des variables dans l'IHM

> Vijeo Designer (HMISTU855)

Dans le navigateur, **ouvrir la rubrique « Variables ».**

Créer les variables correspondant aux variables définies dans l'automate.
Chaque variable est définie par :

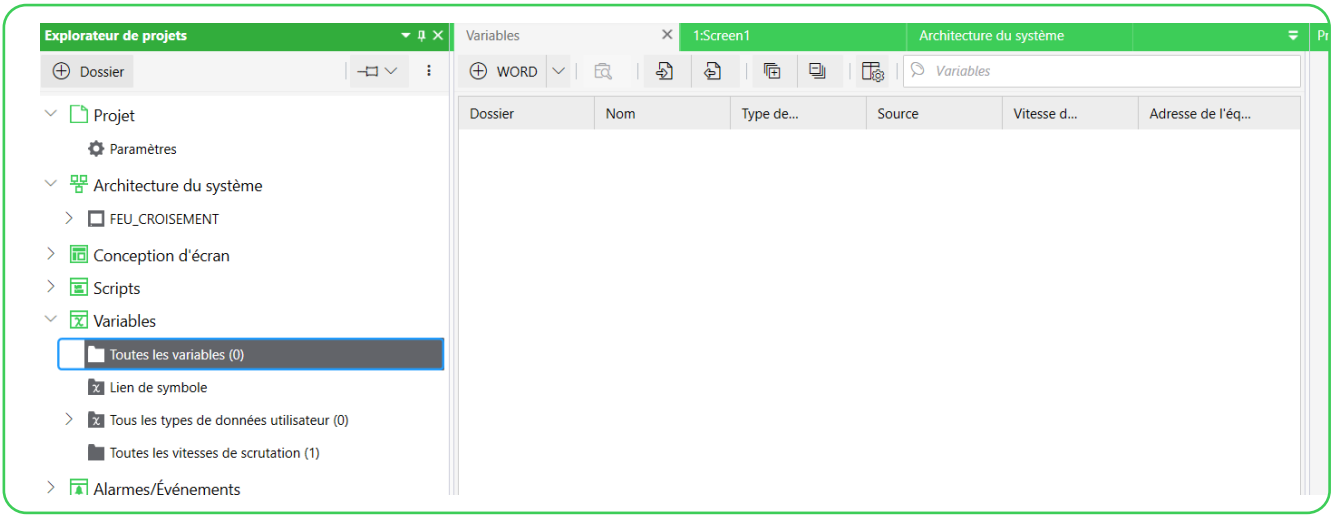
- Le nom : le symbole ou un nom explicite
- Le type de données : booléen ou entier
- La source de données : externe ou interne
- Le groupe de scrutation : pour les données externes, c'est l'équipement destinataire de la requête (l'automate TM221)
- Adresse du périphérique : adresse de la variable dans l'automate



	Nom	Type de données	Source de don...	Groupe de scru...	Adresse du périphérique	Groupe d'alarm...	Groupe de jour...
1	Vert_A	BOOL	Externe	TM221	%M0	Désactivé	Aucun
2	Orange_A	BOOL	Externe	TM221	%M1	Désactivé	Aucun
3	Rouge_A	BOOL	Externe	TM221	%M2	Désactivé	Aucun
4	Vert_B	BOOL	Externe	TM221	%M3	Désactivé	Aucun
5	Orange_B	BOOL	Externe	TM221	%M4	Désactivé	Aucun
6	Rouge_B	BOOL	Externe	TM221	%M5	Désactivé	Aucun
7	Bouton_Pieton	BOOL	Externe	TM221	%M50	Désactivé	Aucun
8	Duree_Restante_Rouge_A	INT	Externe	TM221	%MW15	Désactivé	Aucun
9	Duree_Restante_Vert	INT	Externe	TM221	%MW16	Désactivé	Aucun
10	Duree_Restante_Orange	INT	Externe	TM221	%MW17	Désactivé	Aucun
11	Duree_Restante_Rouge_B	INT	Externe	TM221	%MW18	Désactivé	Aucun
12	Tempo_vert	INT	Externe	TM221	%MW30	Désactivé	Aucun
13	Tempo_orange	INT	Externe	TM221	%MW31	Désactivé	Aucun
14	Tempo_rouge	INT	Externe	TM221	%MW32	Désactivé	Aucun

> Operator Terminal Expert (HMIST6400)

Dans l'explorateur de projets, cliquez sur l'onglet **« Variables »**, puis **« Toutes les variables ».**



A partir de cette interface, **créer toutes les variables nécessaires.**

Chaque variable est définie par :
- Le nom : le symbole ou un nom explicite
- Le type de données : booléen ou entier
- La source de données : externe

Dossier	Nom	Type de...	Source	Vitesse d...	Adresse de l'éq...
	VA	BOOL	TM221TCP	DefaultScanRate (1...	%M0
	OA	BOOL	TM221TCP	DefaultScanRate (1...	%M1
	RA	BOOL	TM221TCP	DefaultScanRate (1...	%M2
	VB	BOOL	TM221TCP	DefaultScanRate (1...	%M3
	OB	BOOL	TM221TCP	DefaultScanRate (1...	%M4
	RB	BOOL	TM221TCP	DefaultScanRate (1...	%M5
	HMIAP	BOOL	TM221TCP	DefaultScanRate (1...	%M50
	DR_RA	WORD	TM221TCP	DefaultScanRate (1...	%MW15
	DR_V	WORD	TM221TCP	DefaultScanRate (1...	%MW16
	DR_O	WORD	TM221TCP	DefaultScanRate (1...	%MW17
	DR_RB	WORD	TM221TCP	DefaultScanRate (1...	%MW18
	PV	WORD	TM221TCP	DefaultScanRate (1...	%MW30
	PO	WORD	TM221TCP	DefaultScanRate (1...	%MW31
	PR	WORD	TM221TCP	DefaultScanRate (1...	%MW32

Toutes ces variables pourront être utilisées dans les écrans.

Enregistrer le projet.

Etape 6 : création des écrans

> Vijeo Designer (HMISTU855)

Dans le navigateur, cliquer droit sur la rubrique **« Ecrans de base »** pour créer un second écran.
Dans l'écran 1.

En utilisant les objets **« dessin »**  **, placer des objets pour représenter le croisement de deux routes et les cadres des feux de croisement.**

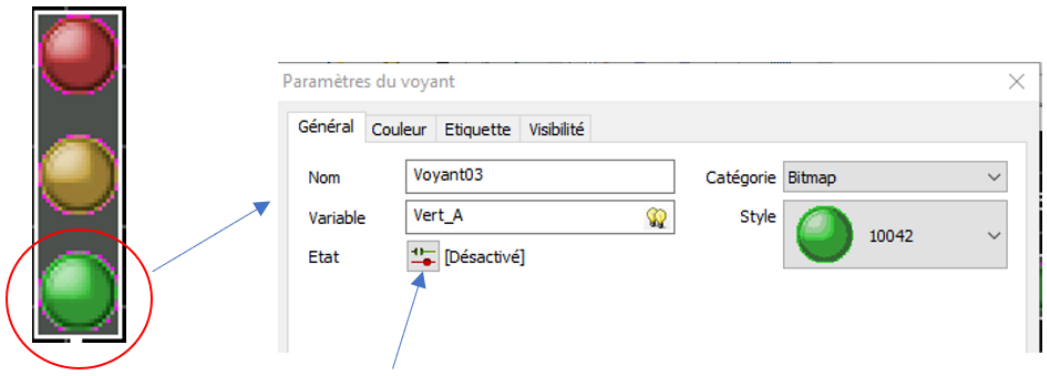
En utilisant l'objet **« texte »**  , **ajouter les textes pour le nom des voies et le titre.**



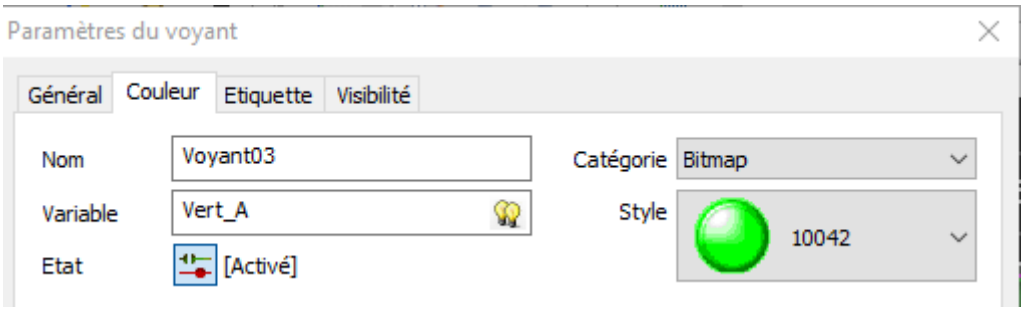
En utilisant l'objet **« voyant »**  **, placer des voyants vert, orange et rouge de chaque feu.**

Une variable est affectée à chaque voyant pour l'animation « éteint / allumé ».

Le voyant vert de la voie A est animé par la variable « Vert_A ».



En cliquant sur le bouton **Etat,** il est possible de visualiser le résultat

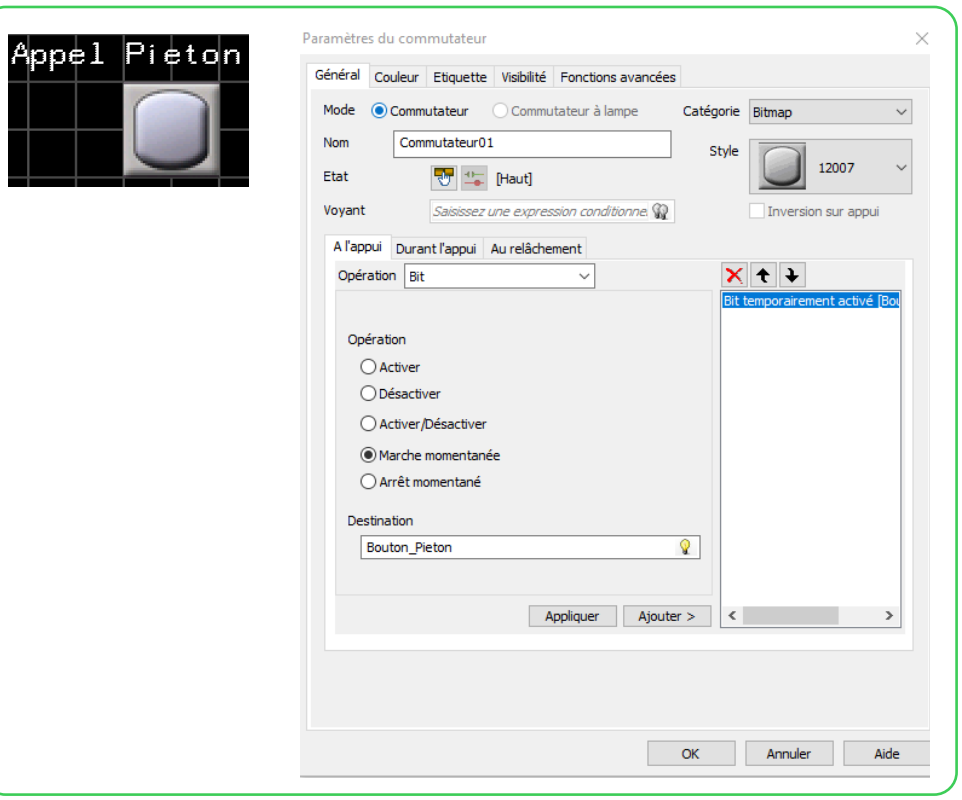


Si la variable « Vert_A » est égale à 1 (Activé) le voyant sera « allumé ».

Procéder de la même façon pour tous les autres voyants en affectant la bonne variable d'animation.

En utilisant l'objet **« commutateur »** , **placer un objet pour l'appel piéton.**

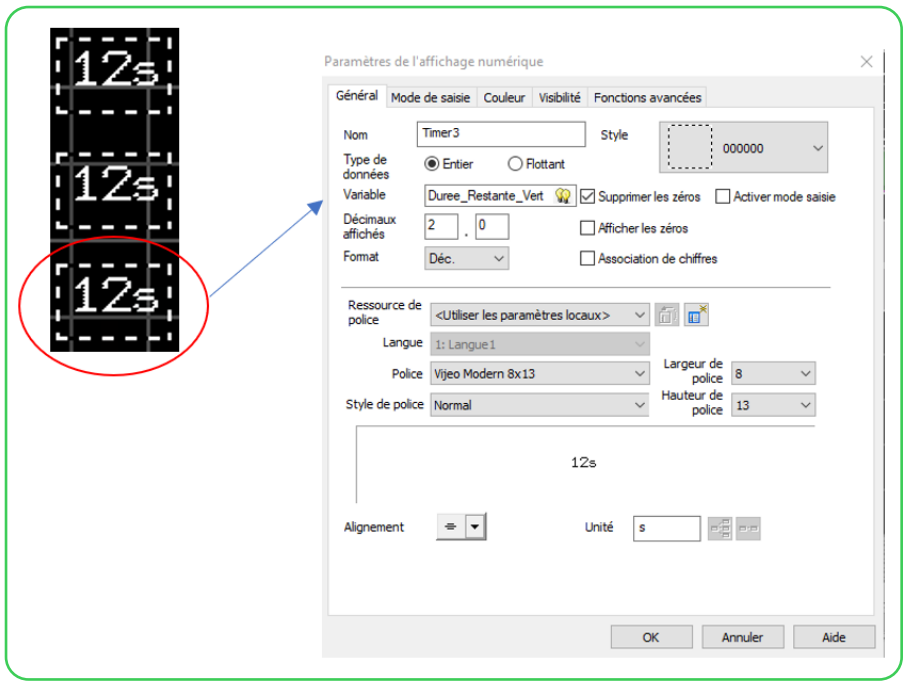
Une variable est affectée au commutateur.



La variable « Bouton_Pieton » est affectée à ce commutateur et l'opération utilisée est « Marche momentanée » : lorsque on appuie sur le bouton, la variable passe à 1 sinon la variable est à 0.

En utilisant l'objet « **affichage numérique »** , **placer des afficheurs au près des voyants pour l'affichage des durées restantes.**

Une variable est affectée à chaque affichage numérique.



La variable affectée à cet afficheur est « Duree_Restante_Vert ».
Le type de donnée est « Entier » et on affiche 2 chiffres. L'unité est ajouté « s ».

La variable affectée à cet afficheur est « Duree_Restante_Vert ».

Le type de donnée est « Entier » et on affiche 2 chiffres. L'unité est ajouté « s ».

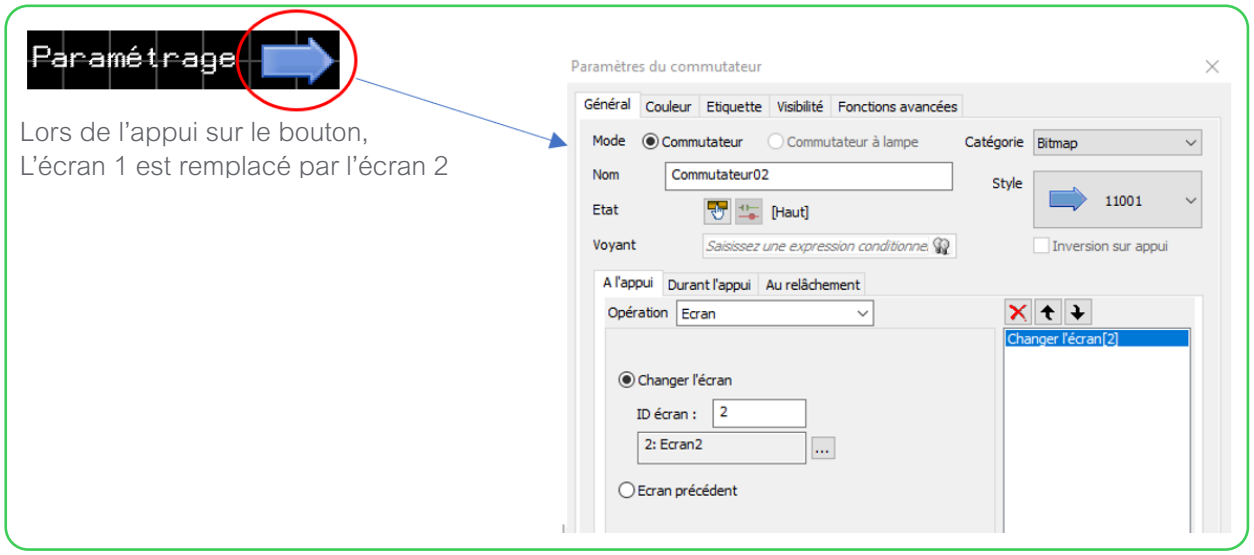
Dans l'onglet visibilité, activer la visibilité lorsque la variable Vert_A est vraie.



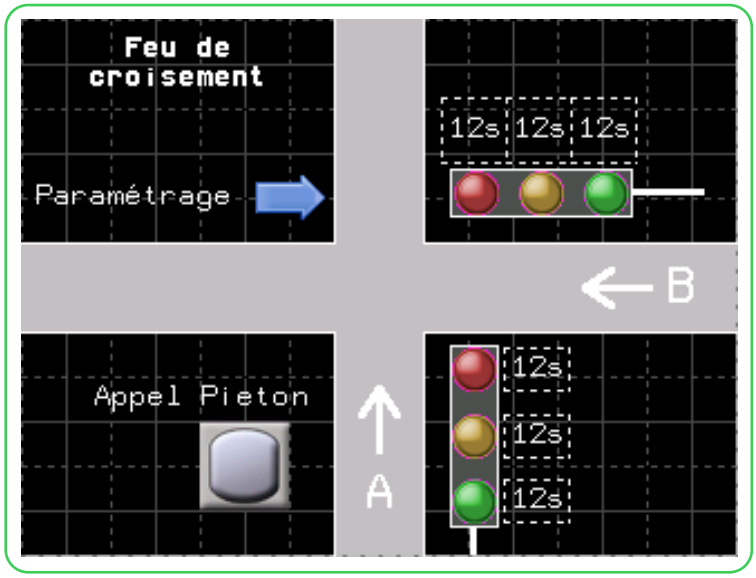
Procéder de la même façon pour tous les autres afficheurs en affectant la bonne variable.

En utilisant l'objet **« commutateur »**  **, placer un objet pour le changement d'écran.**

Une fonction interne est affectée au commutateur.



Le résultat final est le suivant :

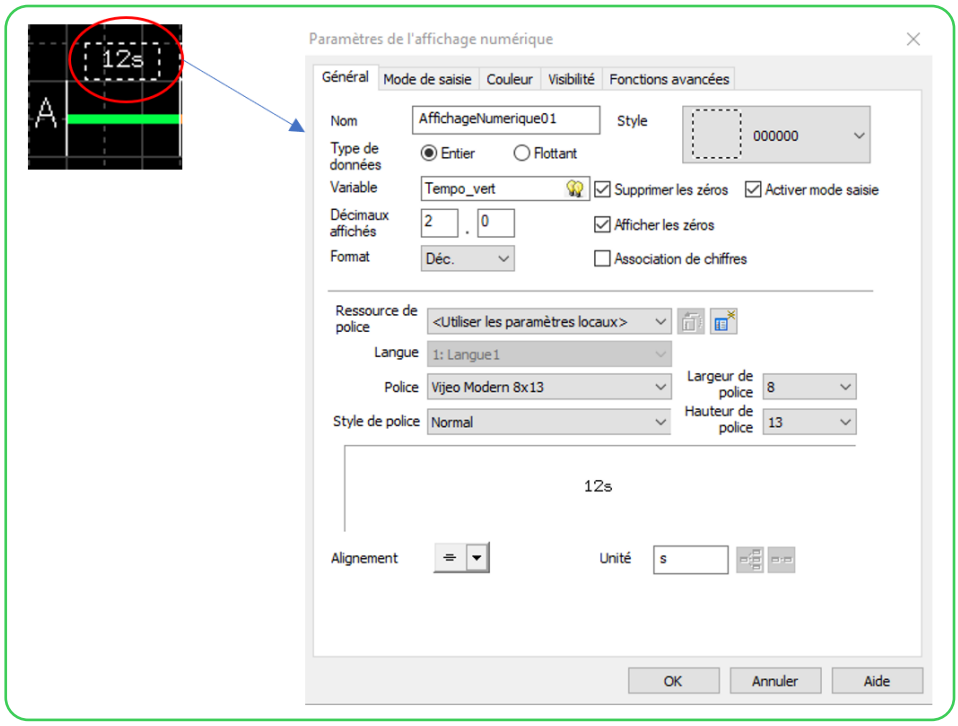


Dans l'écran 2

En utilisant les objets **« dessin »**  **, placer des objets pour représenter les 2 cycles d'allumage des feux de la voie A et de la voie B.**

En utilisant l'objet **« affichage numérique »**  **, placer des afficheurs auprès de la zone Vert voie A, orange voie A et Rouge voie B pour afficher et modifier les préréglages des temporisateurs Vert, Orange et Rouge.**

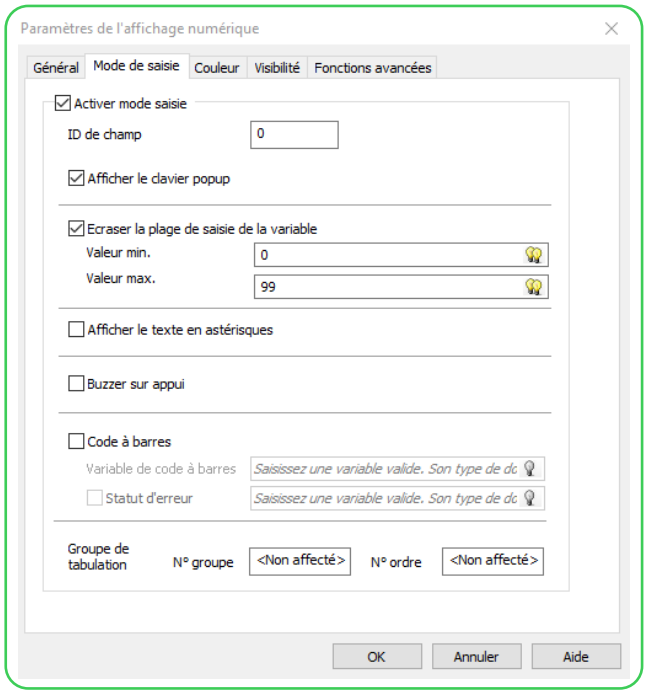
Une variable est affectée à chaque affichage numérique.



La variable affectée est « Tempo_vert ».

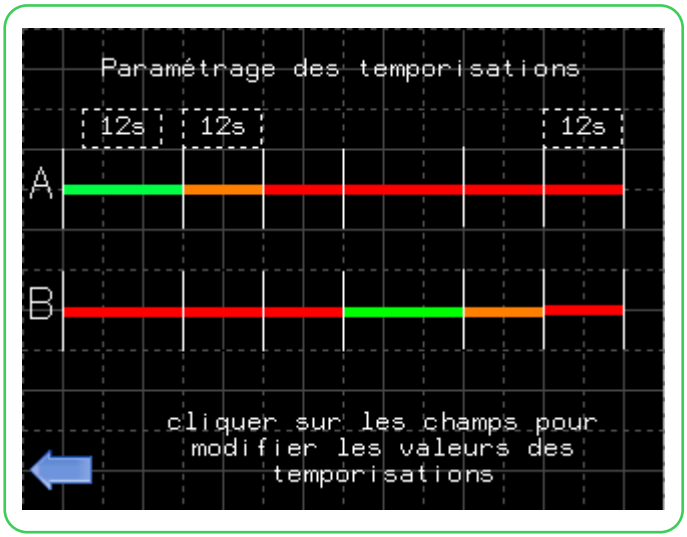
Le type de données est « Entier » on affiche 2 chiffres. L'unité est ajoutée « s ».

Ce champ de données est modifiable, dans l'onglet « Mode de saisie », les paramètres d'écriture sont renseignés.



Une limite basse « 0 » et haute « 99 » interdit de saisir une valeur hors de cette plage.

Le résultat final est le suivant :



Générer et Enregistrer le projet.

> Operator Terminal Expert (HMIST6400)

Dans l'explorateur de projets, **cliquer sur « Conception d'écran » et créer un second écran. Sélectionner l'écran 1 (S00001).**

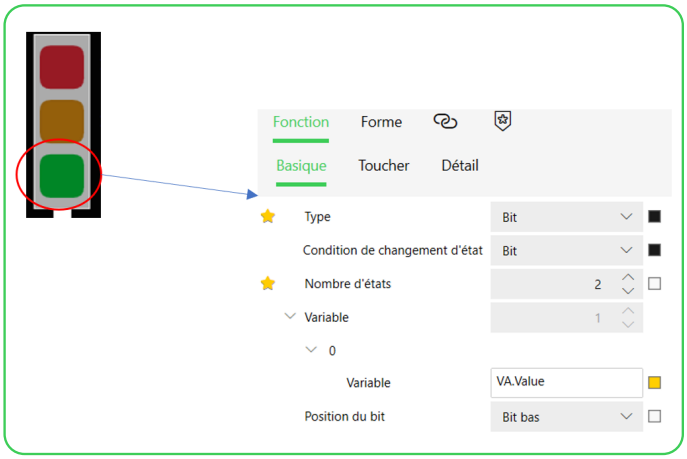
La création d'objets via OTE se fait à partir d'une bibliothèque d'objets disponibles en bas à gauche de l'explorateur de projets.

Des objets visant à modifier l'arrière-plan de l'écran, des composants permettant l'affichage de certaines données mais aussi des boutons, des formes et des images prédéfinies sont disponibles.
Le but étant de créer un visuel représentatif de la réalité (deux routes et deux feux).

En utilisant l'objet **« Zone de texte »** Zone de texte **, ajouter des textes pour les noms des voies et des titres.**

En utilisant l'objet **« voyant d'état N»** N-StateLamp Voyant d'état N **, placer des voyants vert, orange et rouge de chaque feu.**

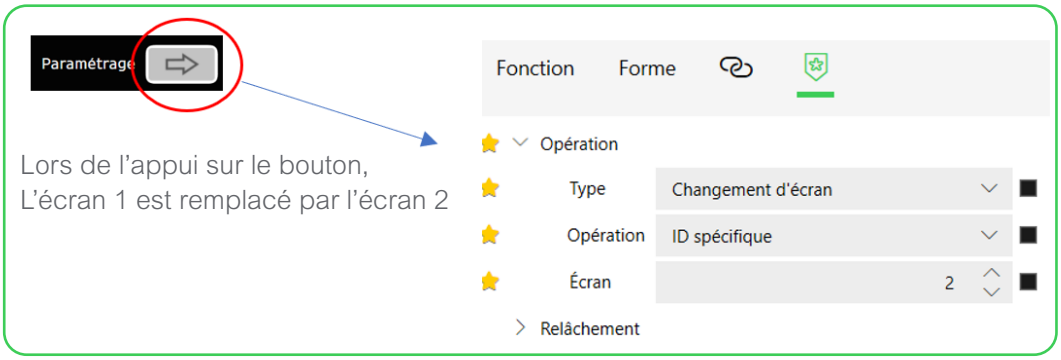
Une variable est affectée à chaque voyant pour l'animation « éteint / allumé ».
Le voyant vert de la voie A est animé par la variable « VA ».



En utilisant l'objet **« commutateur »** 

, placer un objet pour le changement d'écran.

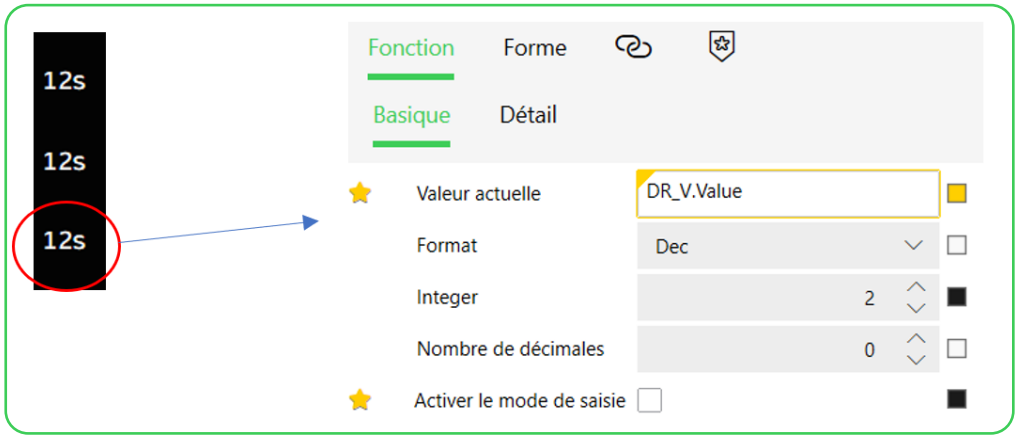
Une fonction interne est affectée au commutateur.



En utilisant l'objet **« affichage numérique »** 

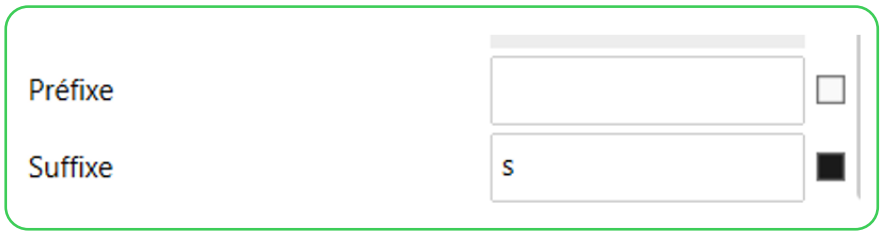
, placer des afficheurs auprès des voyants pour l'affichage des durées restantes.

Une variable est affectée à chaque affichage numérique.



La variable affectée à cet afficheur est « DR_V ».
Le format est « Décimal » et on affiche 2 chiffres.

Décocher le mode de saisie pour interdire la modification de l'afficheur.
Dans l'onglet « Détail », ajouter le suffixe « s » pour un affichage en secondes.



Procéder de la même façon pour tous les autres afficheurs en affectant la bonne variable.
La visibilité d'un objet peut être définie en fonction d'une variable.

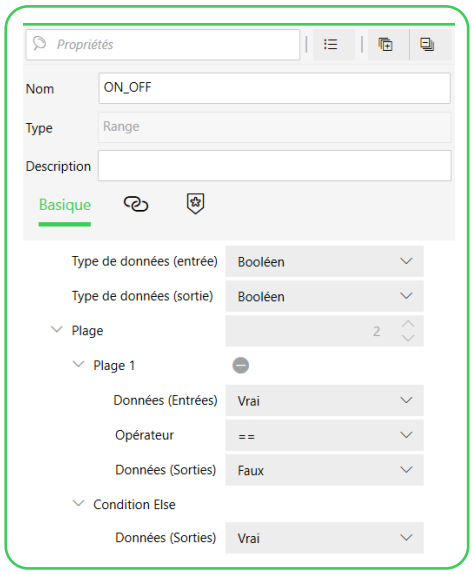
Le logiciel possède la logique suivante : Si la variable est à l'état 0, alors l'objet sera visible. Si la variable est à l'état 1, alors l'objet sera invisible.

L'afficheur de durée restante doit être visible lorsque le voyant correspondant est allumé (variable d'état = 1).
Cette variable d'état ne peut donc pas être utilisée en l'état pour gérer la visibilité de l'objet afficheur.

Pour résoudre ce problème, il est nécessaire de c**réer un convertisseur qui permettra d'inverser les conditions de visibilité.**

Pour cela, aller dans l'onglet **« explorateur de projets »** et cliquer sur la page **« convertisseurs ». Créer une nouvelle unité.**

Nommer le convertisseur et commencer le réglage.

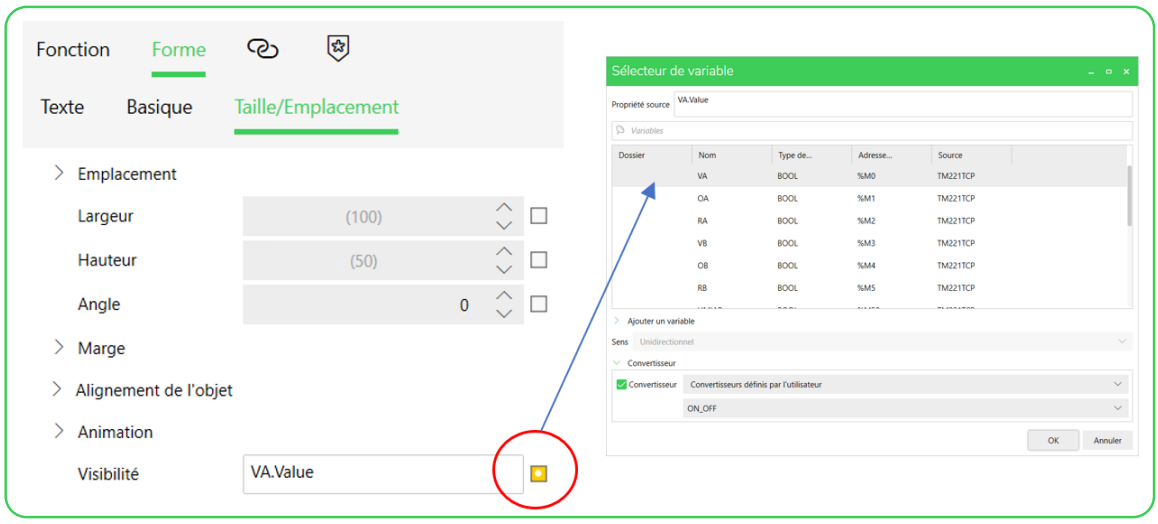


Le type d'entrée concerné est un booléen et le type de sortie également.

Dans l'onglet « Plage 1 » :
Établir la donnée d'entrée sur « Vrai ».
L'opérateur est « = = ».
Établir la donnée de sortie sur « faux ».
Dans l'autre cas, établir la donnée de sortie sur « Vrai ».

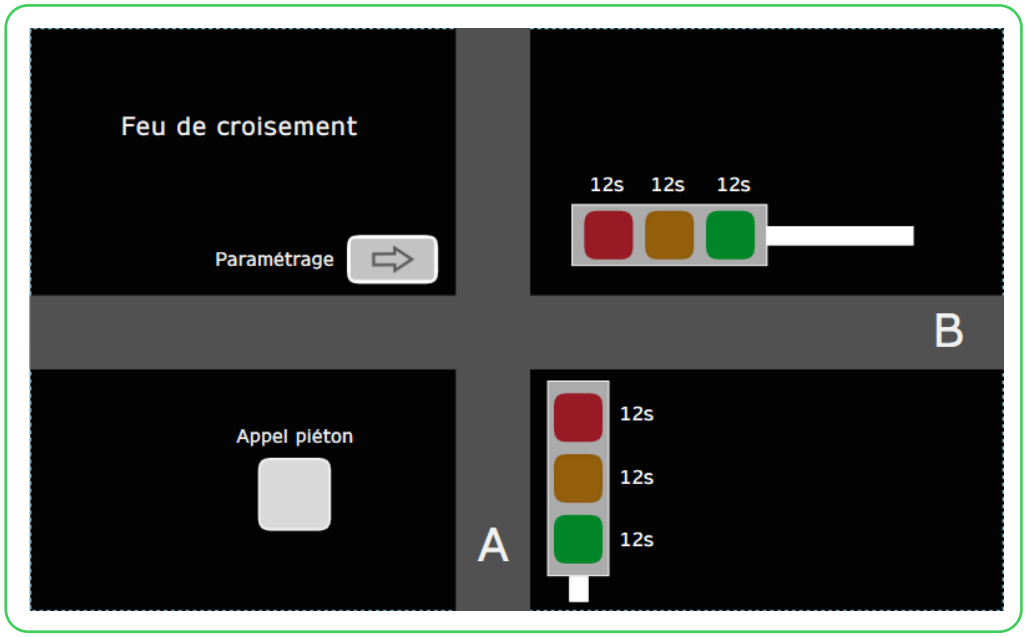
Le convertisseur ci-dessus permet d'inverser l'état d'une variable booléenne.

Quitter la page « convertisseurs » et revenir sur les réglages des afficheurs numériques.



Dans l'onglet visibilité, affecter à la variable correspondante (durée restante feu vert voie A état du feu vert voie A).
Une page « sélecteur de variable » s'ouvre. **Choisir la variable correspondante.**
Dans l'onglet **« Convertisseur »**, **sélectionner le convertisseur créé précédemment.**
Cliquer sur **« OK ».**

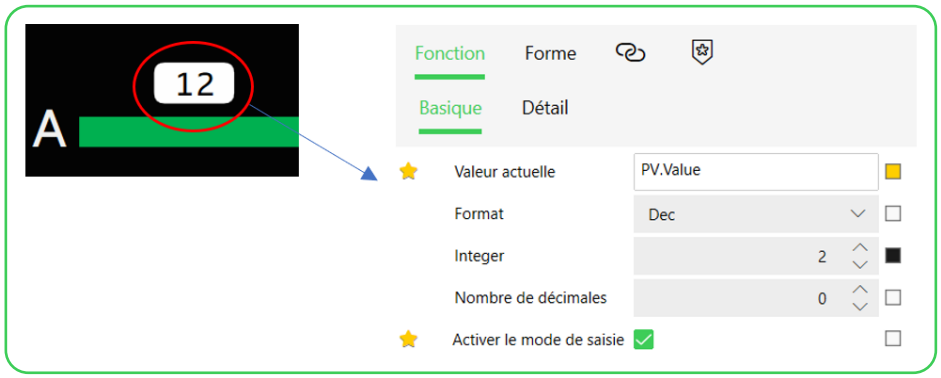
Le résultat final est le suivant :



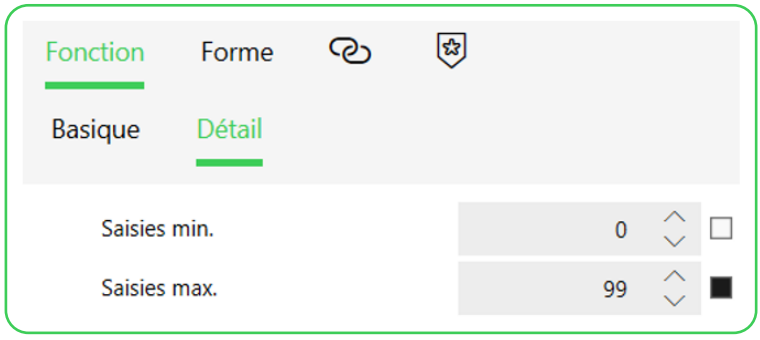
Dans l'écran 2, En utilisant les objets **« formes », placer des objets pour représenter les 2 cycles d'allumage des feux de la voie A et de la voie B.**

En utilisant l'objet **« affichage numérique »** 

, placer des afficheurs auprès de la zone Vert voie A, Orange voie A et Rouge voie B pour afficher et modifier les préréglages des temporisateurs Vert, Orange et Rouge.
Une variable est affectée à chaque affichage numérique.

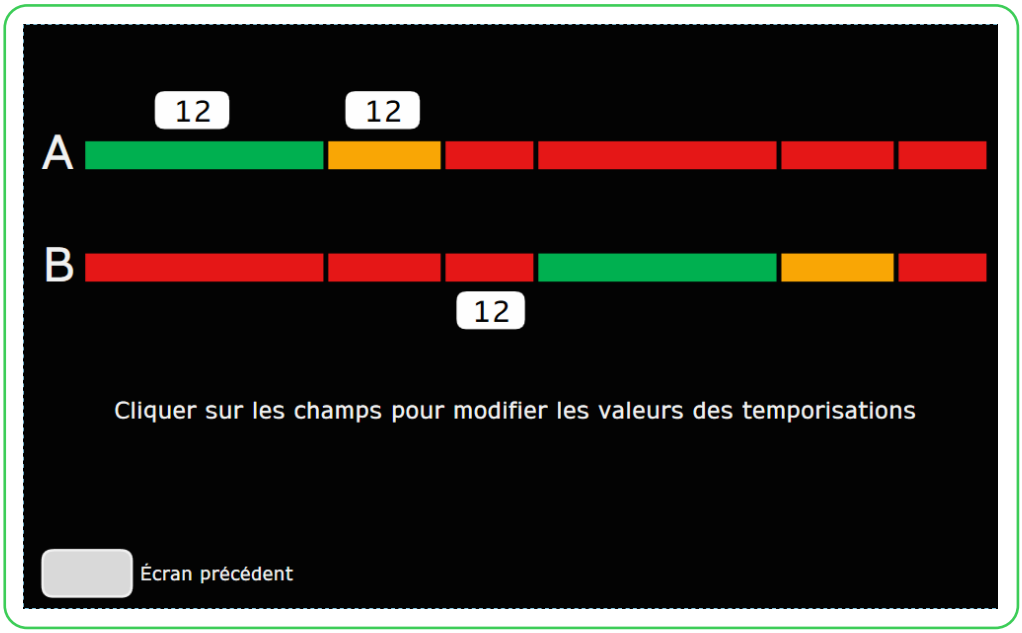


La variable affectée est « PV ».
Le format est « Décimal » on affiche 2 chiffres. L'unité « s » est ajoutée dans la section « détail » puis « suffixe ».
Ce champ de données est modifiable, dans l'onglet « Détail », les paramètres d'écriture sont renseignés.



Une limite basse « 0 » et haute « 99 » interdit de saisir une valeur hors de cette plage.

Le résultat final est le suivant :



Générer et enregistrer le projet.

Etape 7 : transfert de l'application et essais

Mise en service

1 Vérification du câblage

Le câblage du coffret et des boîtes feux de croisement doit être vérifié.

2 Mise sous tension

La mise sous tension du coffret se fait « porte fermée » en basculant le sectionneur général sur la position 1. Vérifier que le voyant blanc « sous tension » est allumé.

3 Chargement du programme dans l'automate

Cette opération ne peut être réalisée que par une personne habilitée.

Machine Expert Basic (TM221)

Pour le transfert dans l'automate, connecter un cordon USB – MiniUSB puis, dans l'onglet **« Mise en service »**, cliquer sur le bouton **« Connection »** puis choisir **« PC vers contrôleur »**. Démarrer le contrôleur.

Vijeo Designer (HMISTU855)

Pour le transfert dans l'IHM, connecter un cordon USB – MiniUSB puis cliquer sur le bouton **« Transférer la cible ».**

Operator Terminal Expert (HMIST6400)

Pour le transfert dans l'IHM, connecter un cordon USB – MicroUSB puis cliquer sur le bouton **« Transférer ».**

HMISTU855 :



HMIST6400 :



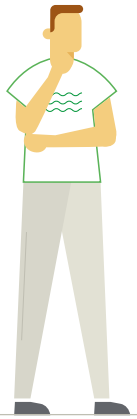
En synthèse

Où trouver les applications et les logiciels ?

Récupérer
les logiciels gratuits :
EcoStruxure Machine
Expert Basic



Que je dois retenir ?



- Réaliser l'analyse fonctionnelle à partir du cahier des charge.
- Structurer l'application à partir de l'analyse fonctionnelle.
- Écrire l'application en fonction du schéma électrique.
- Charger l'application.
- Tester le fonctionnement en sécurité.

Pour aller plus loin

Apprendre les bases de la communication avec
les objets connectés IOT via une passerelle ZBRN

Life Is On | Schneider Electric

se.com/fr

Schneider Electric France
Direction Marketing Communication France
Le Hive
35, rue Joseph Monier
92500 Rueil-Malmaison, France

Conseils et services
se.com/fr/contact

© 2023 Schneider Electric. Tous droits réservés. Life Is On Schneider Electric est une marque commerciale appartenant à Schneider Electric SAS, ses filiales et ses sociétés affiliées.
En raison de l'évolution des normes et du matériel, les caractéristiques indiquées par les textes et les images de ce document ne nous engagent qu'après confirmation par nos services.
Life Is On : la vie s'illumine - Conception, réalisation : Collectif OKA/FRV100

07/2023 - ZZ7880

LE TRI FACILE BAC DE TRI