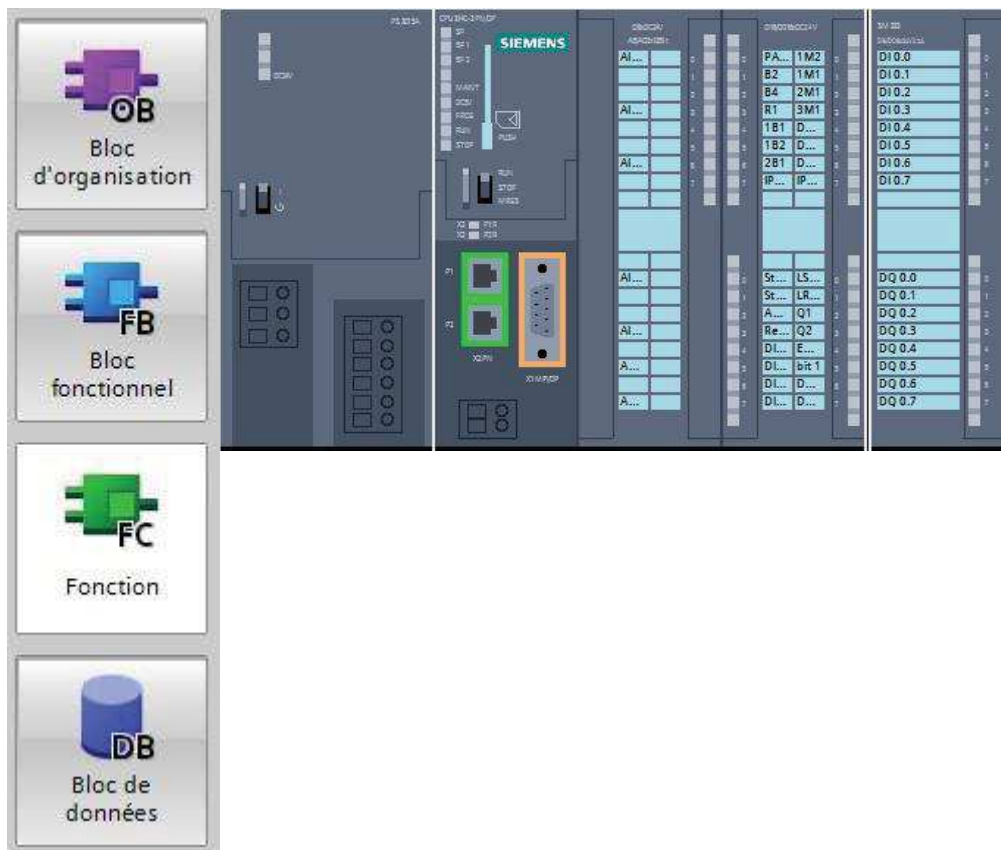


# Atelier C – TIA Portal

## CTIA01 : Programmation des automates S7-300 – Introduction au logiciel TIA Portal



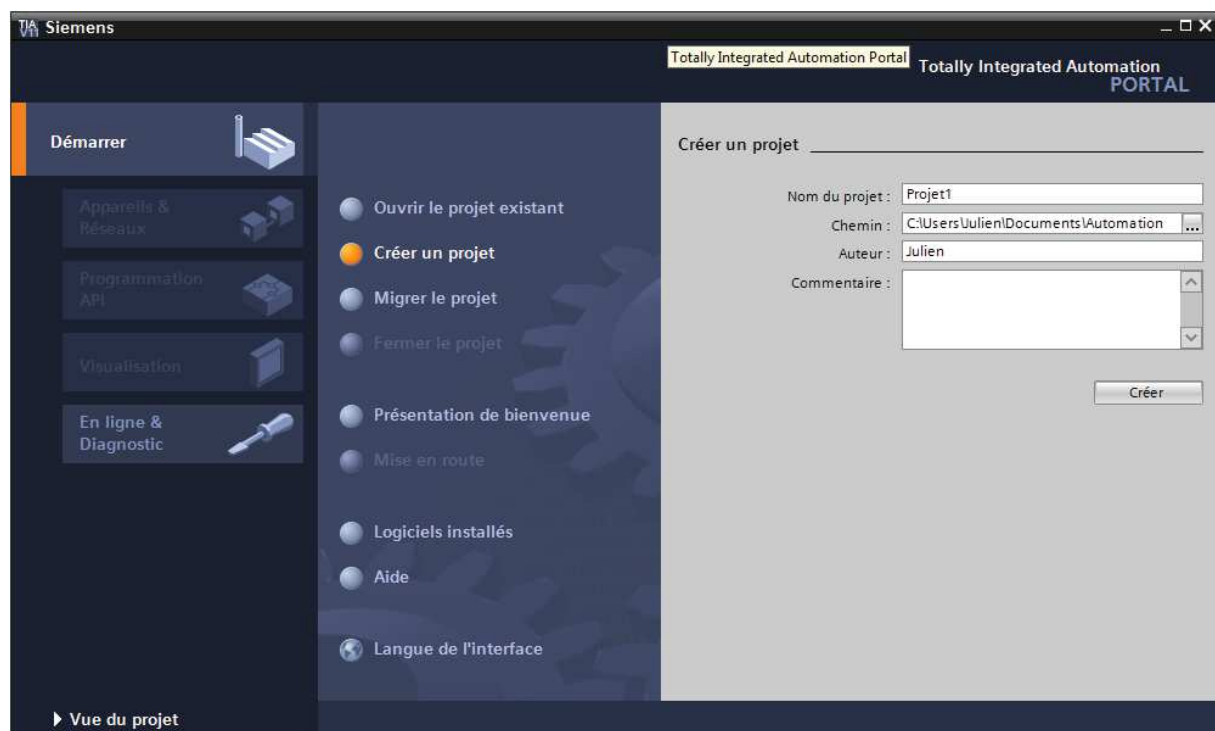
## 1. Introduction

La plateforme Totally Integrated Automation Portal est le nouvel environnement de travail Siemens qui permet de mettre en œuvre des solutions d'automatisation avec un système d'ingénierie intégré comprenant les logiciels SIMATIC STEP 7 V11 et SIMATIC WinCC V11 (dans la version du programme disponible au CTA de Virton).

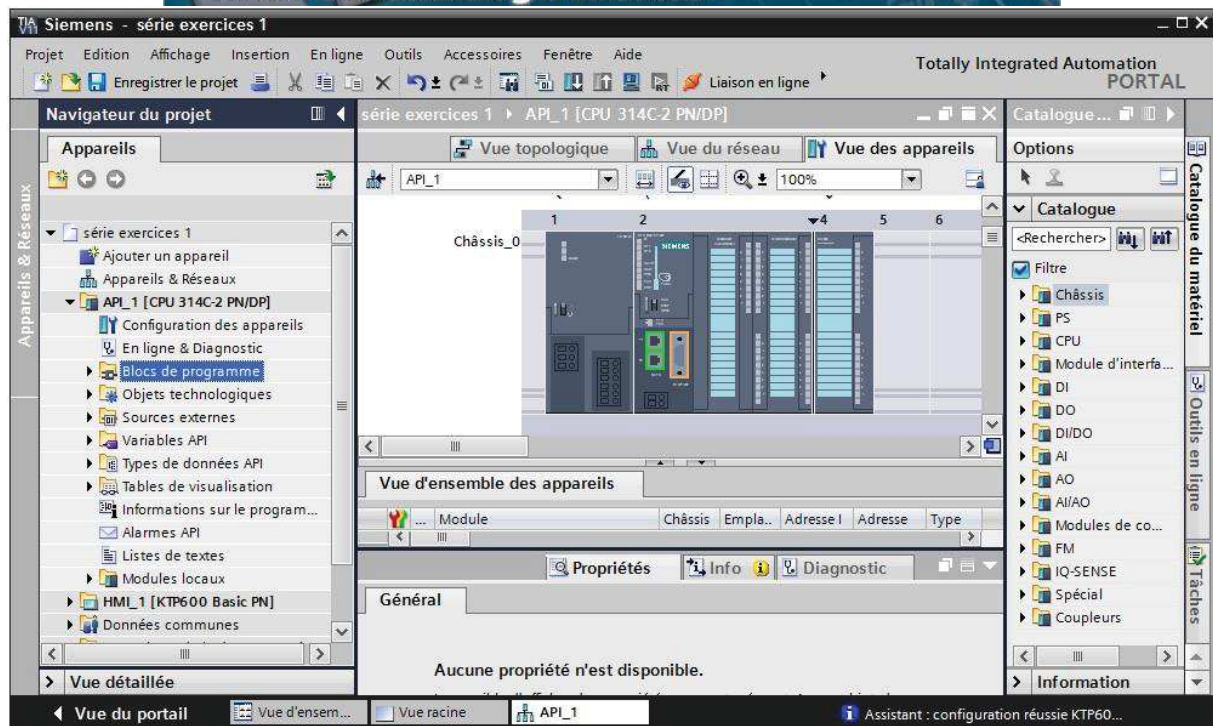
## 2. Vue du portail et vue du projet

Lorsque l'on lance TIA Portal, l'environnement de travail se décompose en deux types de vue :

- **La vue du portail** : elle est axée sur les tâches à exécuter et sa prise en main est très rapide.
- **La vue du projet** : elle comporte une arborescence avec les différents éléments du projet. Les éditeurs requis s'ouvrent en fonction des tâches à réaliser. Données, paramètres et éditeurs peuvent être visualisés dans une seule et même vue.



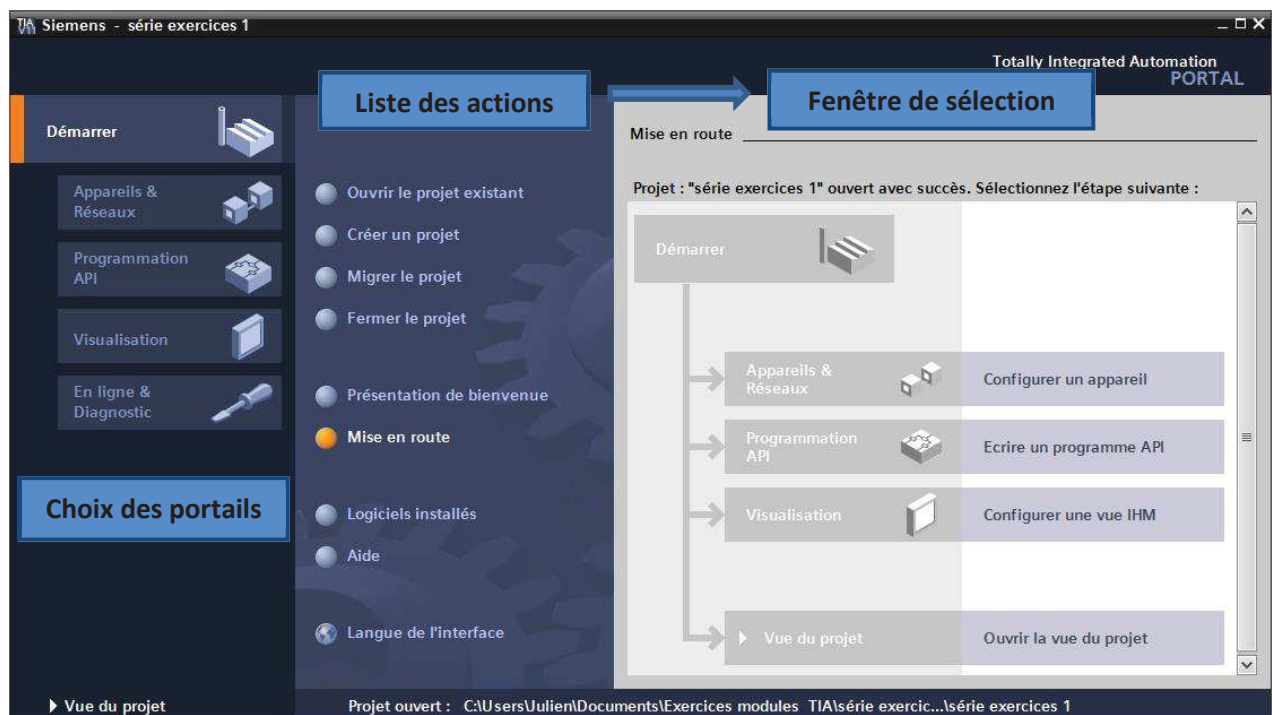
*Vue du portail*



*Vue du projet*

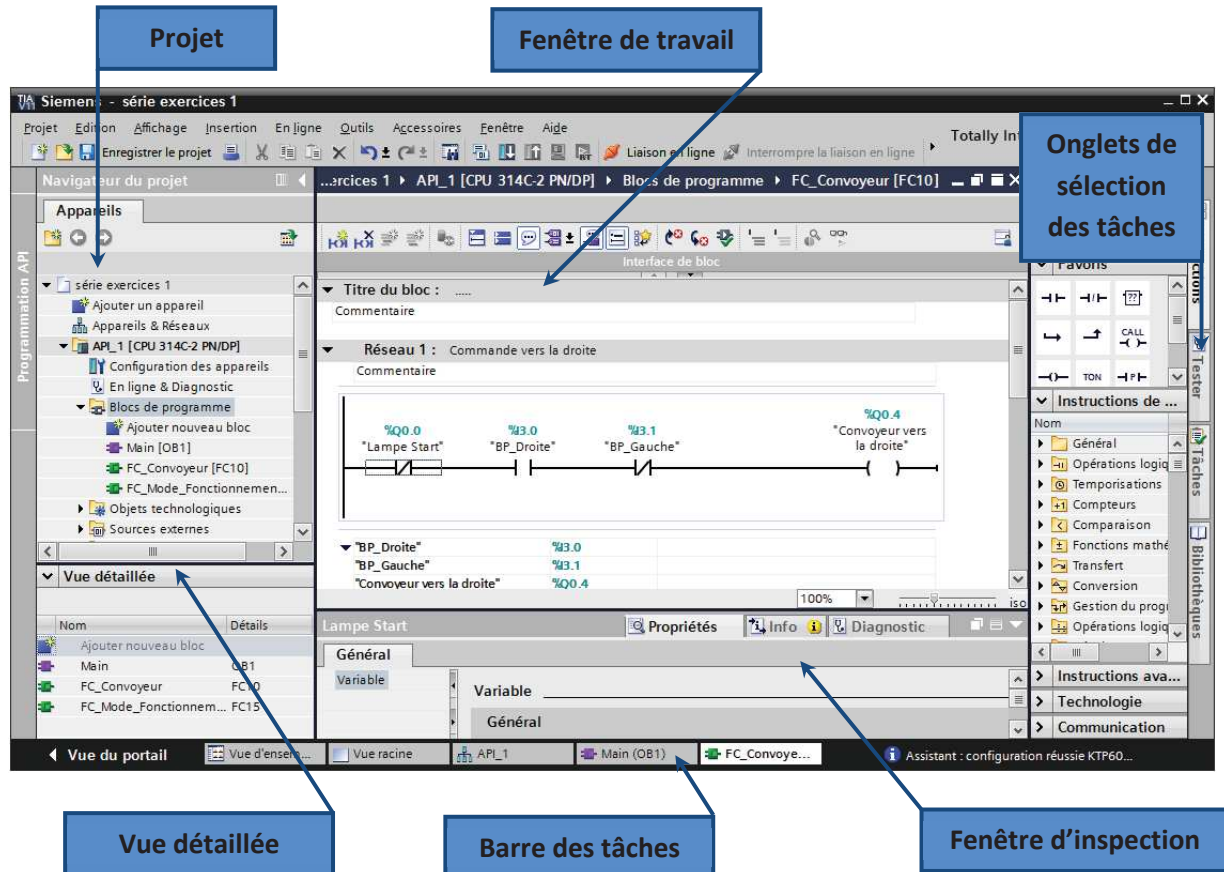
## 2.1 Vue du portail

Chaque portail permet de traiter une catégorie de tâche (actions). La fenêtre affiche la liste des actions pouvant être réalisées pour la tâche sélectionnée.



## 2.2 Vue du projet

L'élément « Projet » contient l'ensemble des éléments et des données nécessaires pour mettre en œuvre la solution d'automatisation souhaitée.



La **fenêtre de travail** permet de visualiser les objets sélectionnés dans le projet pour être traités. Il peut s'agir des composants matériels, des blocs de programme, des tables des variables, des HMI,...

La **fenêtre d'inspection** permet de visualiser des informations complémentaires sur un objet sélectionné ou sur les actions en cours d'exécution (propriété du matériel sélectionné, messages d'erreurs lors de la compilation des blocs de programme,...).

Les **onglets de sélection de tâches** ont un contenu qui varie en fonction de l'objet sélectionné (configuration matérielle → bibliothèques des composants, bloc de programme → instructions de programmation).

Cet environnement de travail contient énormément de données. Il est possible de masquer ou réduire certaines de ces fenêtres lorsque l'on ne les utilise pas.

Il est également possible de redimensionner, réorganiser, désancrer les différentes fenêtres.



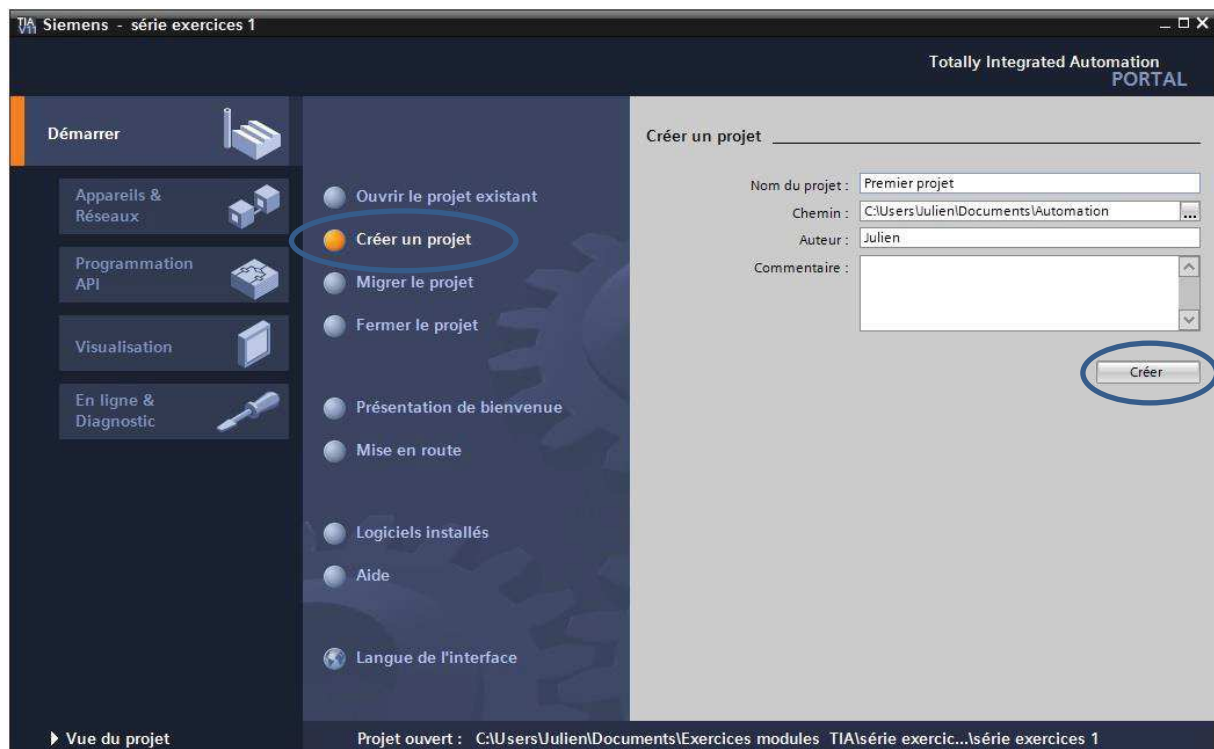
### 3. Création d'un projet et configuration d'une station de travail

#### 3.1 Création d'un projet

Pour créer un projet dans la vue du portail, il faut sélectionner l'action « **Créer un projet** ».

On peut donner un nom au projet, choisir un chemin où il sera enregistré, indiquer un commentaire ou encore définir l'auteur du projet.

Une fois que ces informations sont entrées, il suffit de cliquer sur le bouton « **créer** »

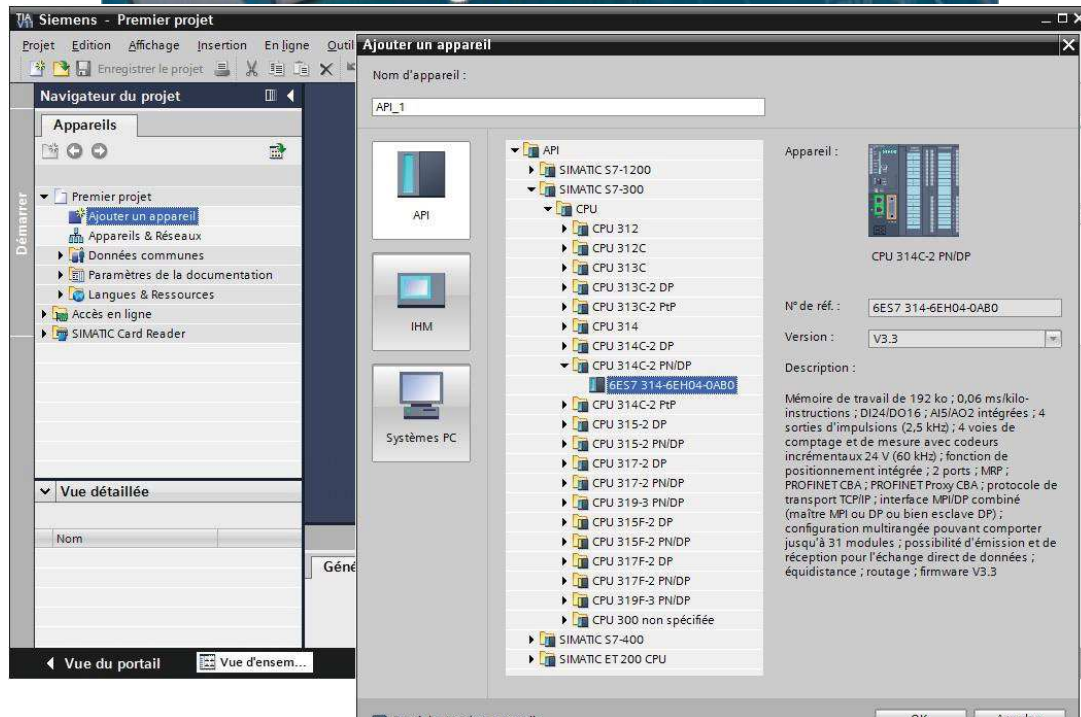


#### 3.2 Configuration et paramétrage du matériel

Une fois votre projet créé, on peut configurer la station de travail.

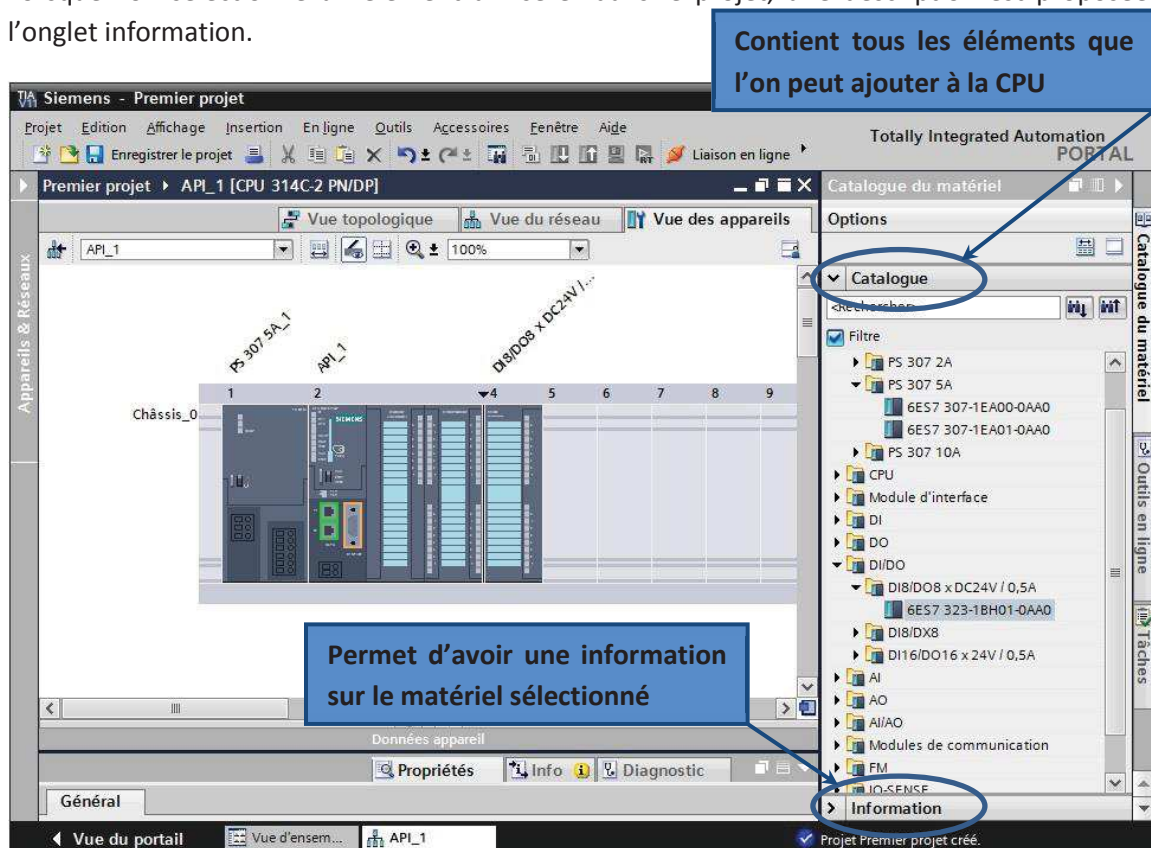
La première étape consiste à définir le matériel existant. Pour cela, on peut passer par la **vue du projet** et cliquer sur « **ajouter un appareil** » dans le navigateur du projet.

La liste des éléments que l'on peut ajouter apparaît (API, HMI, système PC). On commencera par faire le choix de notre CPU pour ensuite venir ajouter les modules complémentaires (alimentation, E/S TOR ou analogiques, module de communication AS-i,...).



Les modules complémentaires de l'API peuvent être ajoutés en utilisant le catalogue. Si l'on veut ajouter un écran ou un autre API, il faut repasser par la commande « ajouter un appareil » dans le navigateur du projet.

Lorsque l'on sélectionne un élément à insérer dans le projet, une description est proposée dans l'onglet information.



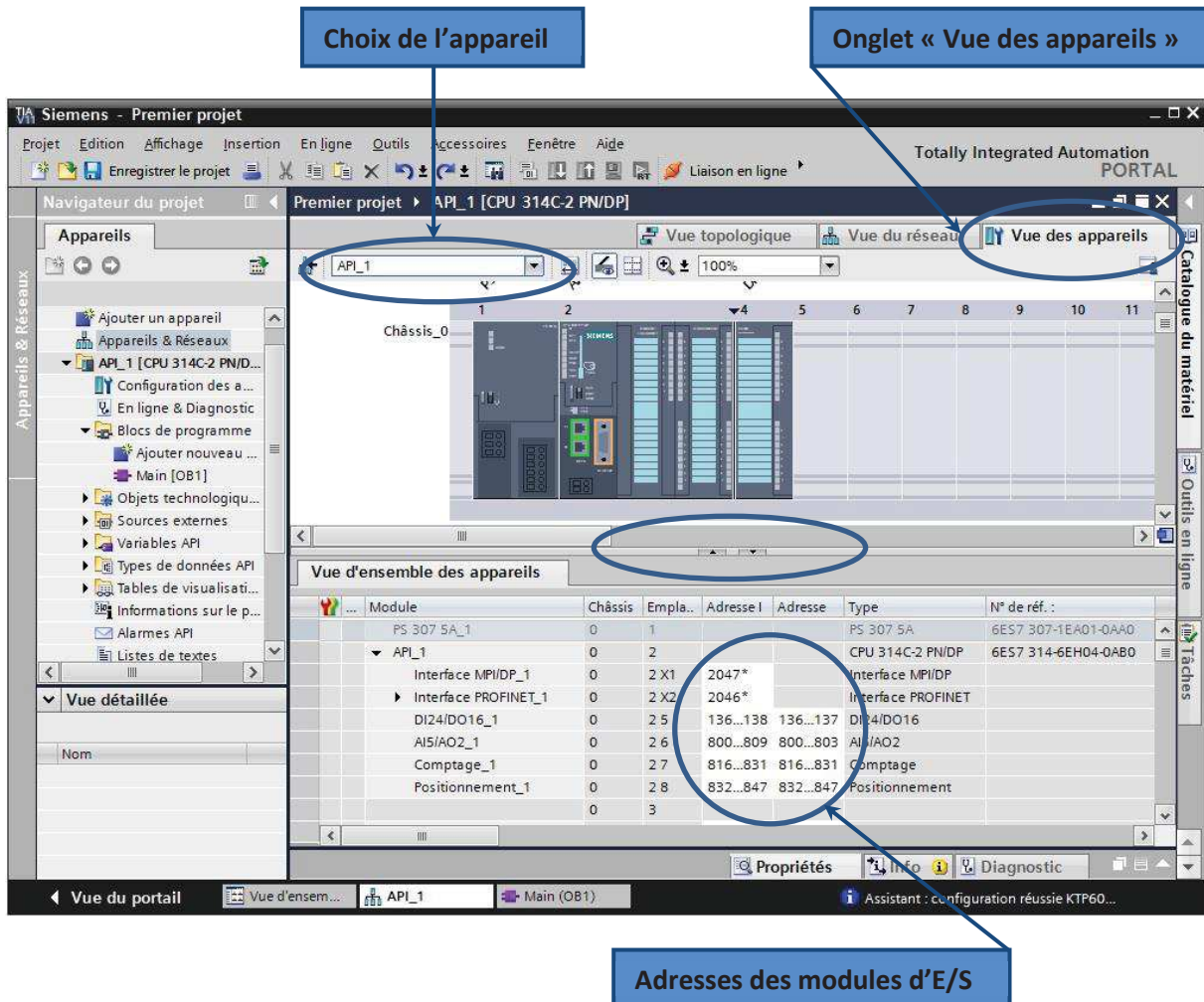
### 3.3 Adressage des E/S

Pour connaître l'adressage des entrées et sorties présentes dans la configuration matériel, il faut aller dans « **appareil et réseau** » dans le navigateur du projet.

Dans la fenêtre de travail, on doit s'assurer d'être dans l'onglet « **Vue des appareils** » et de sélectionner l'appareil voulu.

Choix de l'appareil

Onglet « Vue des appareils »



Module	Châssis	Empla..	Adresse I	Adresse	Type	N° de réf. :
PS 307 5A_1	0	1			PS 307 5A	6ES7 307-1EA01-0AA0
API_1	0	2			CPU 314C-2 PN/DP	6ES7 314-6EH04-0AB0
Interface MPI/DP_1	0	2 X1	2047*		Interface MPI/DP	
Interface PROFINET_1	0	2 X2	2046*		Interface PROFINET	
DI24/DO16_1	0	2 5	136...138	136...137	DI24/DO16	
AI5/AO2_1	0	2 6	800...809	800...803	AI5/AO2	
Comptage_1	0	2 7	816...831	816...831	Comptage	
Positionnement_1	0	2 8	832...847	832...847	Positionnement	
	0	3				

Adresses des modules d'E/S

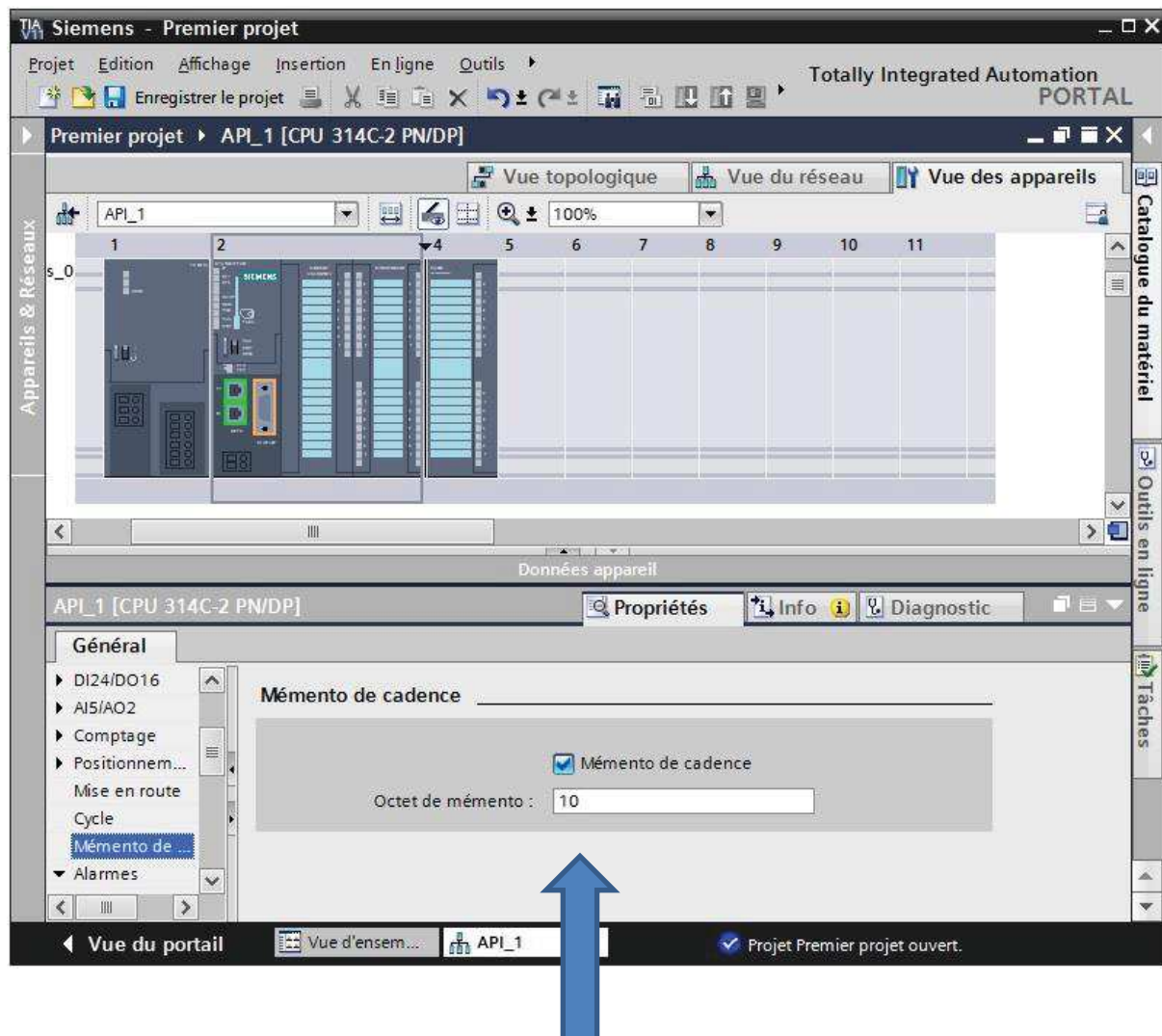
On sélectionne la CPU puis à l'aide des deux petites flèches (voir figure), on fait apparaître l'onglet « **Vue d'ensemble des appareils** » (n'hésitez pas à masquer certaines fenêtres et à en réorganiser d'autres). Les adresses des entrées et sorties apparaissent. Vous pouvez les modifier en entrant une nouvelle valeur dans la case correspondante.

### 3.4 Memento de cadence

Une fois la CPU déterminée, on peut définir le memento de cadence. Pour cela, on sélectionne la CPU dans la fenêtre « **Vue des appareils** » et l'onglet « **propriété** » dans la fenêtre d'inspection.



Dans le menu « **Général** », choisir l'option « **Memento de cadence** », cocher la case « **Memento de cadence** » et choisir l'octet du memento de cadence que l'on va utiliser.



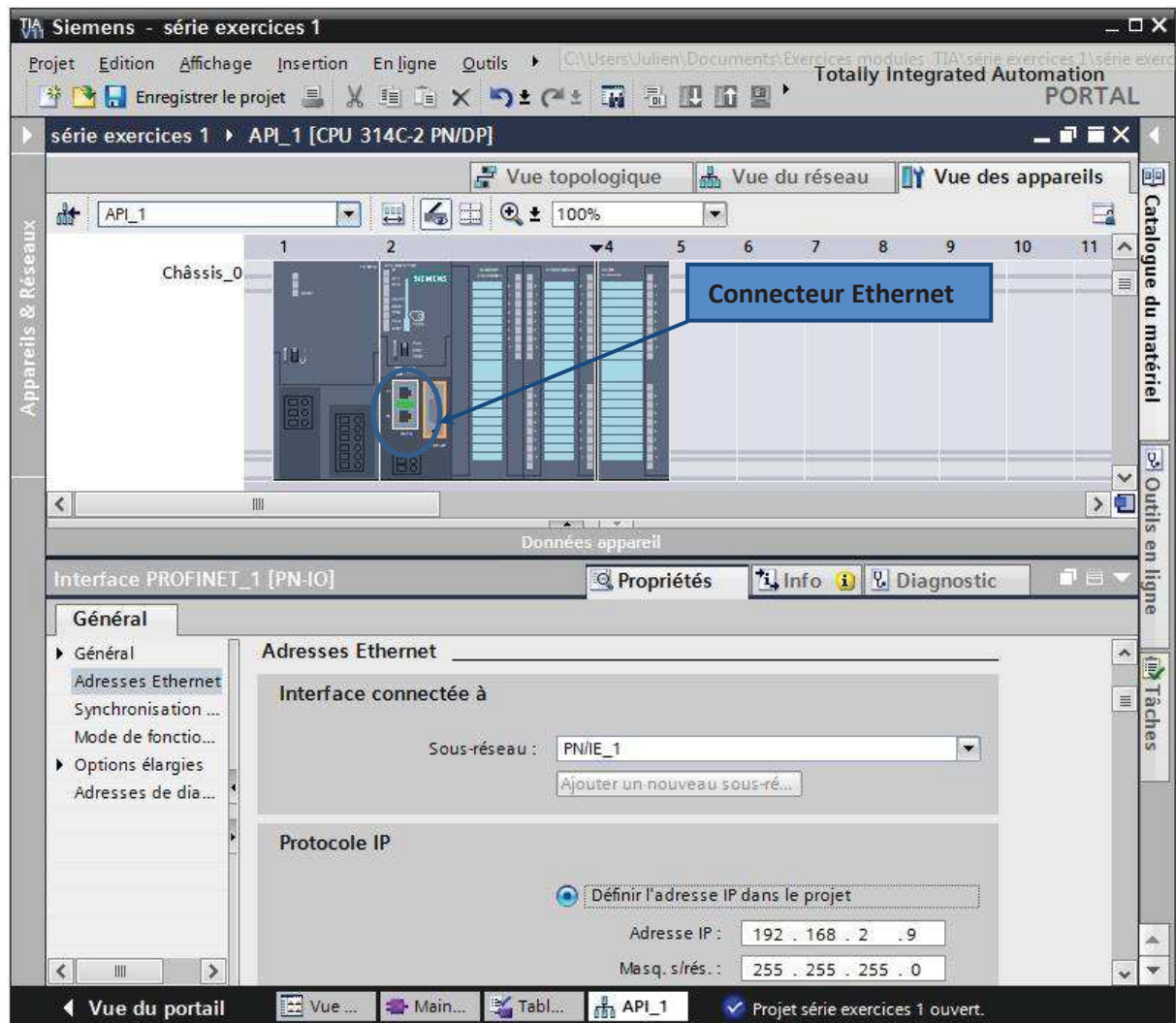
Bit de l'octet de memento de cadence	7	6	5	4	3	2	1	0
Période [s]	2	1.6	1	0.8	0.5	0.4	0.2	0.1
Fréquence [Hz]	0.5	0.625	1	1.25	2	2.5	5	10

### 3.5 Adresse Ethernet de la CPU

Toujours dans les propriétés de la CPU, il est possible de définir son adresse Ethernet. Un double clic sur le connecteur Ethernet de la station fait apparaître la fenêtre d'inspection permettant de définir ses propriétés.

Pour établir une liaison entre la CPU et la console de programmation, il faut affecter aux deux appareils des adresses appartenant au même réseau. On utilisera comme adresse pour l'automate 192.168.2.n° de l'automate.





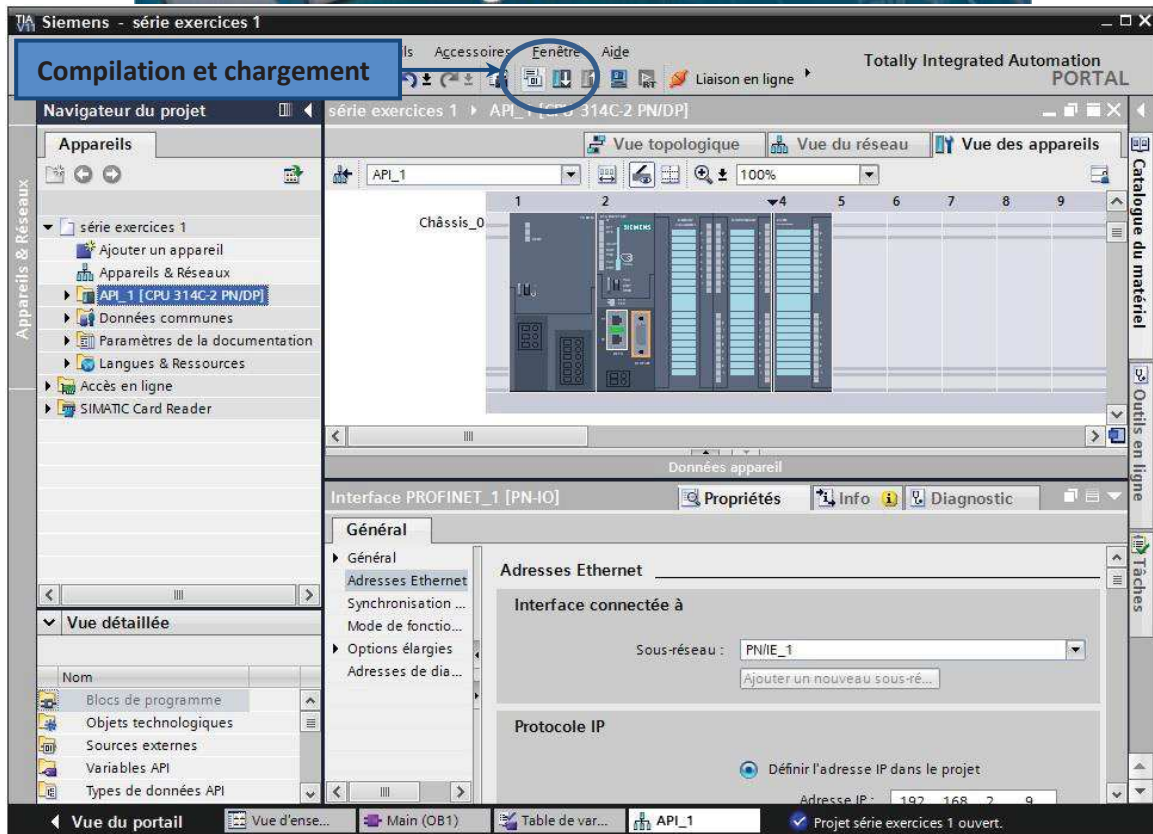
### 3.6 Compilation et chargement de la configuration matérielle

Une fois la configuration matérielle réalisée, il faut la compiler et la charger dans l'automate.

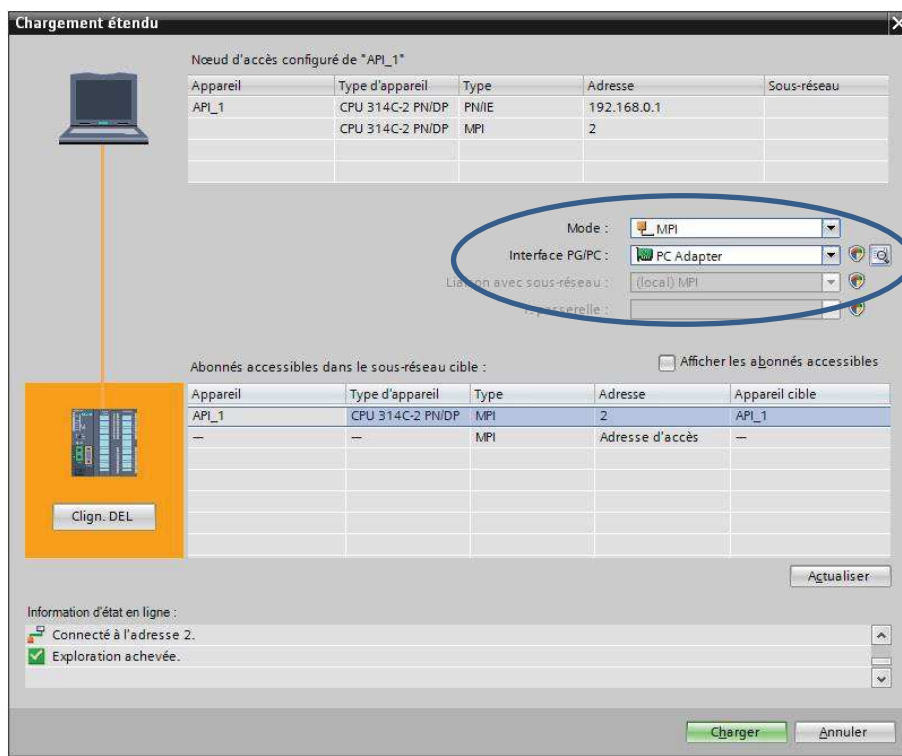
La compilation se fait à l'aide de l'icône « **compiler** » de la barre de tâche. On sélectionne l'API dans le projet puis cliquer sur l'icône « **compiler** ».

En utilisant cette manière, on effectue une compilation matérielle et logicielle.

Une autre solution pour compiler est de faire un clic droit sur l'API dans la fenêtre du projet et de choisir l'option « Compiler → Configuration matérielle ».

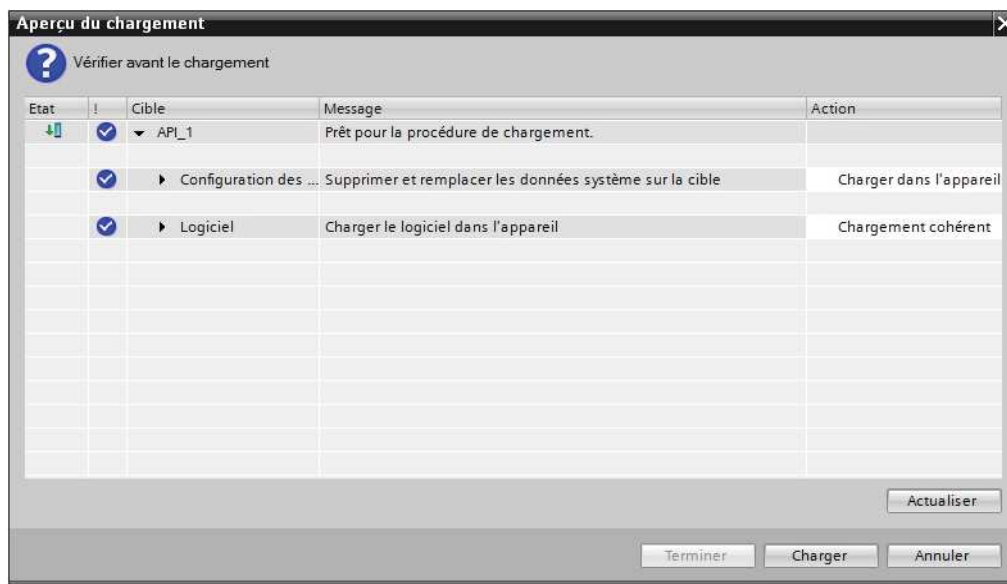


Pour charger la configuration dans l'automate, on effectue un clic sur l'icône « **charger dans l'appareil** ». La fenêtre ci-dessous s'ouvre et vous devez faire le choix du mode de connexion (PN/IE, Profibus, MPI). Si vous choisissez le mode PN/IE, l'API doit posséder une adresse IP.

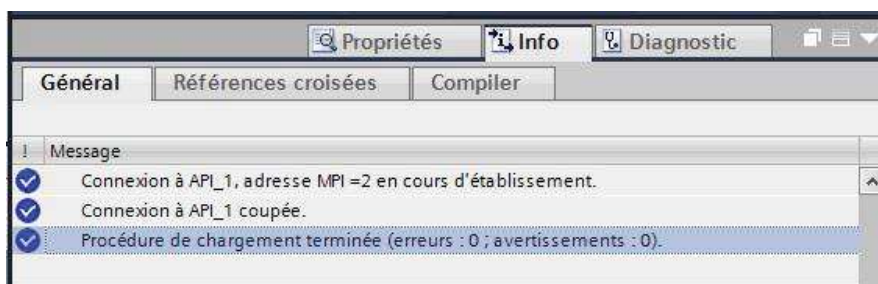


Pour une première connexion ou pour charger l'adresse IP désirée dans la CPU, il est plus facile de choisir le mode de connexion MPI et de relier le PC à la CPU via le « **PC Adapter** ».

Si le programme trouve un appareil, il figure dans la liste en bas de la fenêtre. La touche « **Clign. DEL** » permet de faire clignoter une LED sur la face avant de l'appareil afin de s'assurer que l'on est connecté à l'appareil désiré.



Une fois la configuration terminée, on peut charger le tout dans l'appareil. Des avertissements / confirmations peuvent être demandés lors de cette opération. Si des erreurs sont détectées, elles seront visibles via cette fenêtre. Le programme ne pourra pas être chargé tant que les erreurs persistent.



### 3.7 Exercice : configuration d'une station de travail

Lancer TIA Portal et créer un projet nommé « Configuration station ». Réaliser la configuration matérielle de l'automate mis à votre disposition.

En plus de la configuration matérielle, on demande :

- De nommer votre station « API\_X » ou X est le numéro de votre automate.
- De définir un memento de cadence (octet 10)
- De faire débiter les adresses des E/S liées à la CPU à 0
- De faire débiter les adresses des E/S du module DI/DO supplémentaire à 3



## 4. Les variables API

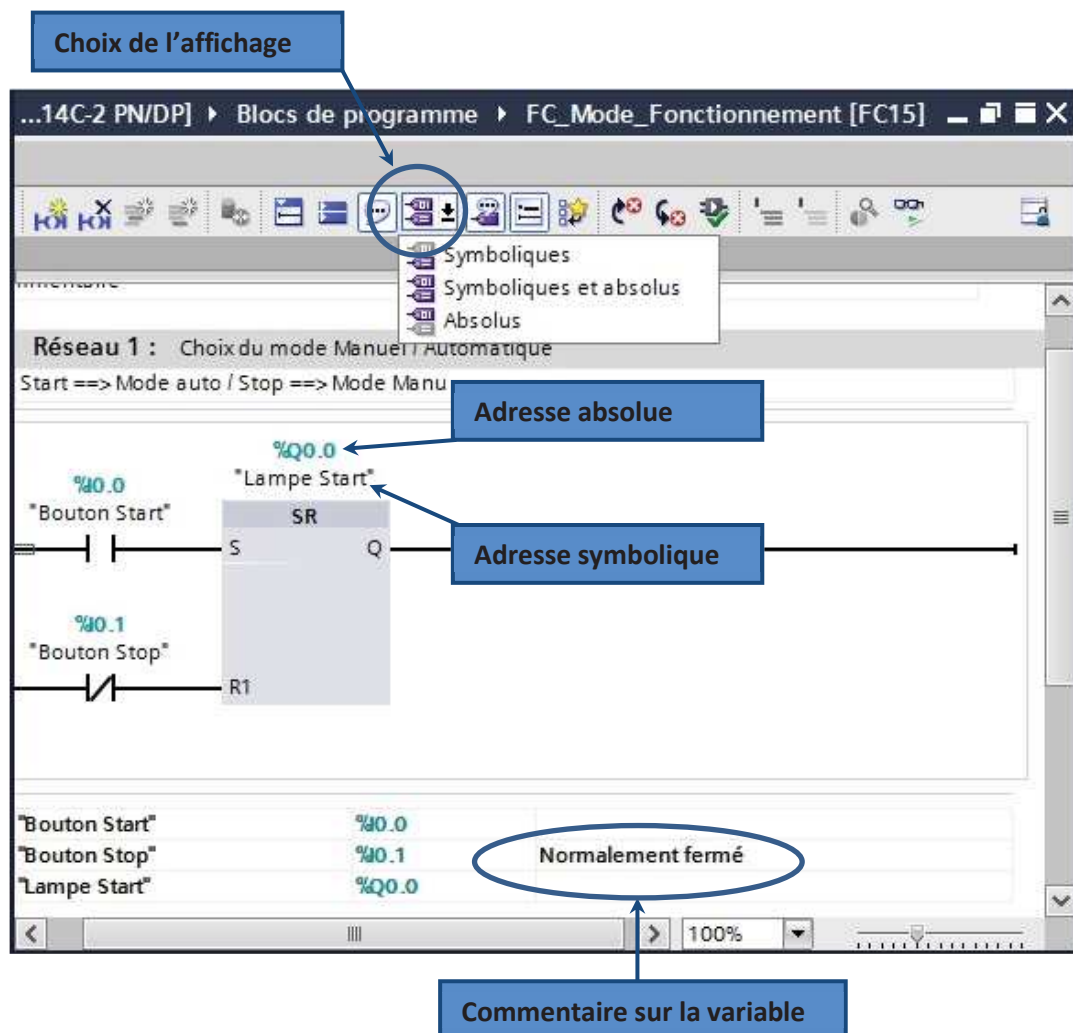
### 4.1 Adresses symbolique et absolue

Dans TIA Portal, toutes les variables globales (entrées, sorties, mémentos,...) possède une **adresse symbolique** et une **adresse absolue**.

- **L'adresse absolue** représente l'identificateur d'opérande (I, Q, M,...) et son adresse et numéro de bit.
- **L'adresse symbolique** correspond au nom que l'utilisateur a donné à la variable (ex : Bouton\_Marche).

Le lien entre les adresses symbolique et absolue se fait dans **la table des variables API**.

Lors de la programmation, on peut choisir d'afficher les adresses absolues, symboliques ou encore les deux simultanément.



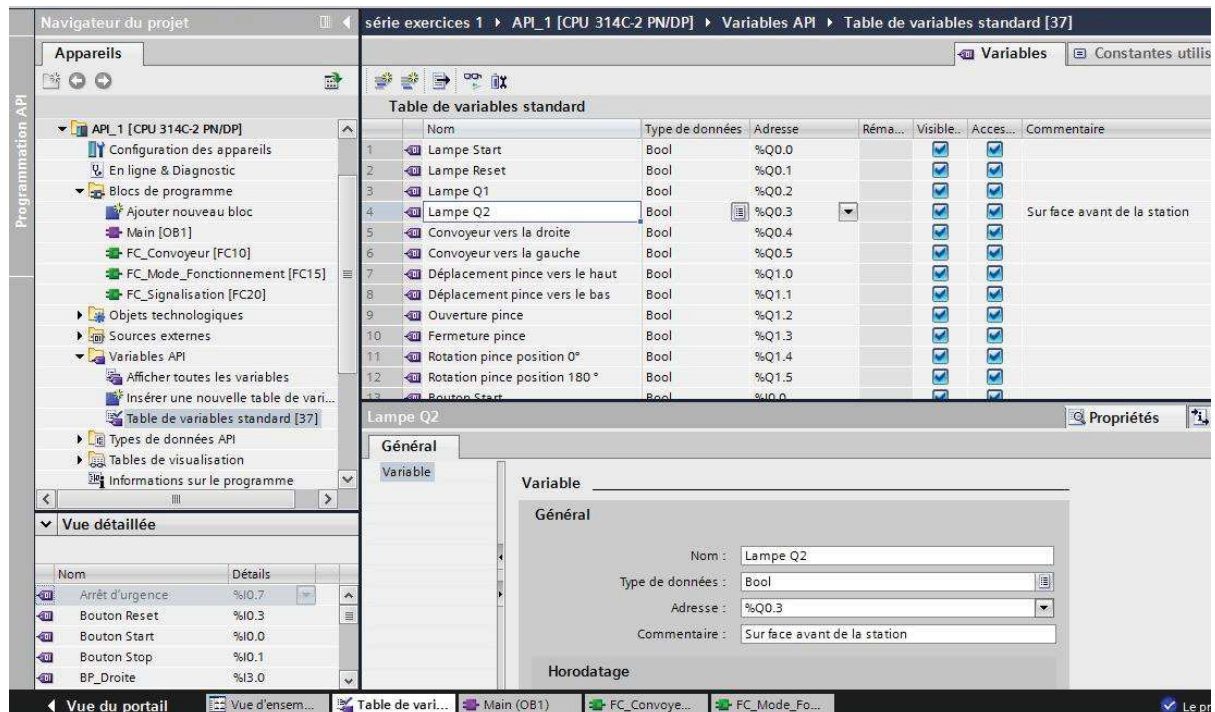
## 4.2 Table des variables API

C'est dans la table des variables API que l'on va pouvoir déclarer toutes les variables et les constantes utilisées le programme.

Lorsque l'on définit une variable API, il faut définir :

- Un nom : c'est l'adressage symbolique de la variable.
- Le type de donnée : BOOL, INT,...
- L'adresse absolue : par exemple Q1.5

On peut également insérer un commentaire qui nous renseigne sur cette variable. Le commentaire peut être visible dans chaque réseau utilisant cette variable.



The screenshot shows the 'Table de variables standard' (Standard Variables Table) in the software. The table lists variables with their names, data types, addresses, and visibility settings. The 'Lampe Q2' variable is highlighted, and its configuration is shown in the 'Général' (General) tab of the 'Propriétés' (Properties) dialog.

Nom	Type de données	Adresse	Réma...	Visible..	Acces...	Commentaire
1	Bool	%Q0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2	Bool	%Q0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	Bool	%Q0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	Bool	%Q0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Sur face avant de la station
5	Bool	%Q0.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6	Bool	%Q0.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	Bool	%Q1.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	Bool	%Q1.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	Bool	%Q1.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10	Bool	%Q1.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
11	Bool	%Q1.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
12	Bool	%Q1.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
13	Bool	%Q1.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

**Lampe Q2**

**Général**

Variable

Nom : Lampe Q2

Type de données : Bool

Adresse : %Q0.3

Commentaire : Sur face avant de la station

Horodatage

Remarques :

- La commande copier / coller permet de copier une table des variable sur Excel facilement.
- En sélectionnant le coin inférieur droit d'une cellule d'adresse ou d'un nom et en le faisant glisser vers le bas, on peut créer des variables automatiquement (comme dans Excel).
- Visible dans IHM : seules les variables cochées peuvent être visibles lors de la configuration HMI.
- Accessible dans HMI (uniquement S7 – 1200) : permet au pupitre HMI l'accès en ligne aux variables API sélectionnées.

### 4.3 Signalisation des erreurs dans la table des variables

Lorsqu'il y a une erreur de syntaxe dans la table des variables API, celle-ci est signalée en rouge ou en orange. Lorsque l'on sélectionne la case colorée, un message signalant le type d'erreur apparaît.

Une table des variables contenant des erreurs peut être enregistrée mais ne pourra pas être compilée et chargée dans l'automate.

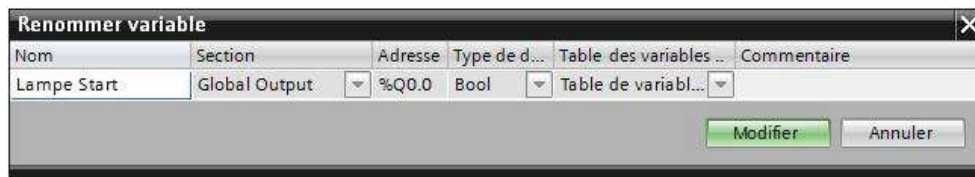
Nom	Type de données	Adresse	Réma...	Visible dans IHM	Accessible depuis IHM	Commentaire
1 Lampe Start	Bool	%Q0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2 Lampe Reset	Bool	%Q0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3 Lampe Q1	Bool	%Q0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4 Lampe Q2	Bool	%Q0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Sur face avant de la station
5 Convoyeur vers la droite	Bool	%Q0.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6 Convoyeur vers la gauche	Bool	%Q0.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7 Déplacement pince vers le haut	Int	%Q1.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8 Déplacement pince vers le bas	Bool	%Q1.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9 Ouverture pince	Bool	%Q0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10 Fermeture pince	Bool	%Q1.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
11 Rotation pince position 0°	Bool	%Q1.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
12 Rotation pince position 180°	Bool			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
13 Lampe Q2(1)	Bool			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
14 Bouton Start	Bool			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
15 Bouton Stop	Bool			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
16 Interrupteur Auto / Manu	Bool	%I0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Normalement fermé
17 Bouton Reset	Bool	%I0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

### 4.4 Renommer / réassigner des variables

Il est possible de renommer et réassigner les variables API dans la fenêtre de programmation. Pour cela, il faut faire un clic droit sur la variable en question et choisir l'option renommer ou réassigner.



La fenêtre ci dessous s'ouvre.

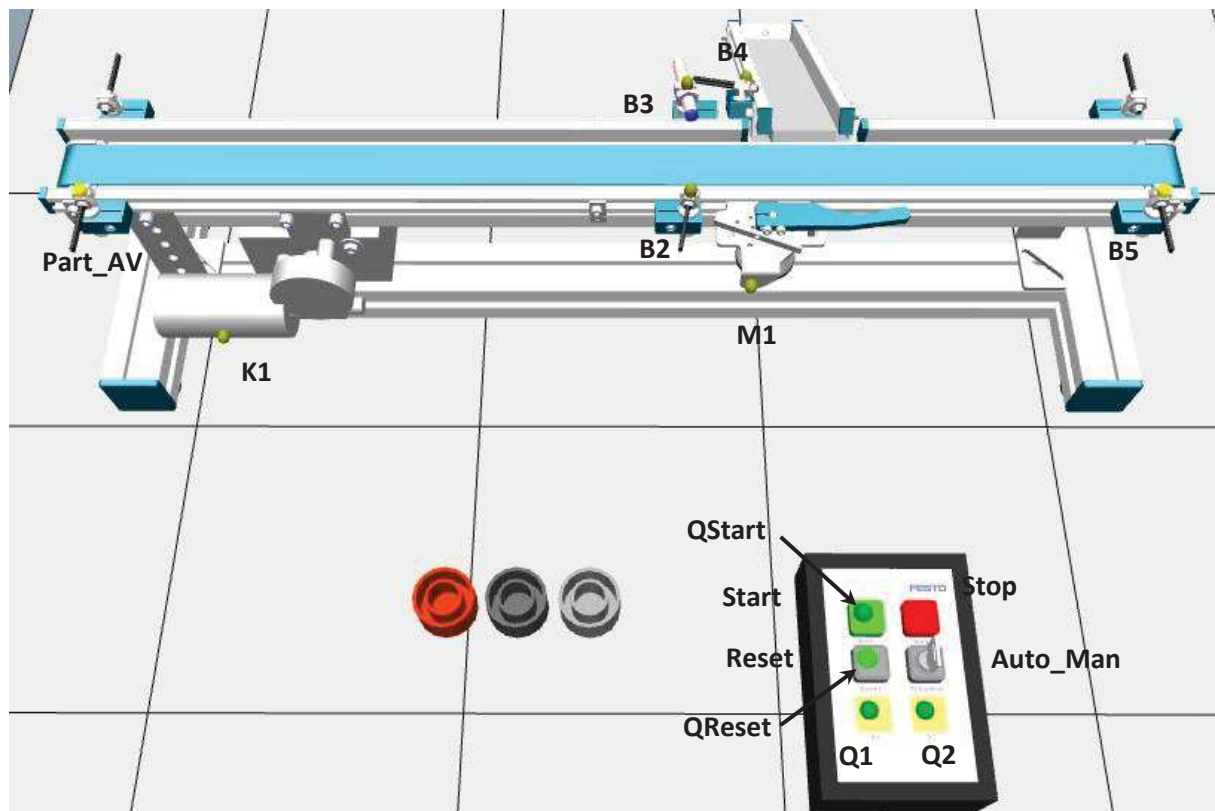


- **Renommer la variable** donne l'accès aux paramètres Nom et Commentaire
- **Réassigner la variable** donne l'accès aux paramètres Section, Adresse, Type de données, Table des variables API et Commentaire.

#### 4.5 Exercice : réalisation d'une table des variables API

Les exercices qui vont suivre seront réalisés à l'aide du logiciel CIROS Mechatronics. Ce logiciel permet de simuler le fonctionnement de stations modulaires FESTO.

Le modèle utilisé se trouve dans la catégorie « Système de transfert » et se nomme « Module de projet Ligne de convoyage 4 »



Symbole	Adresse	Type	Commentaire
K1	Q 0.0	Bool	Marche moteur convoyeur
M1	Q 0.1	Bool	Vérin rotatif
QStart	Q 1.0	Bool	Voyant bouton Start
QReset	Q 1.1	Bool	Voyant bouton Reset
Q1	Q 1.2	Bool	Voyant Q1
Q2	Q 1.3	Bool	Voyant Q2
Part_AV	I 0.0	Bool	Présence pièce entrée convoyeur (NF)
B2	I 0.1	Bool	Présence pièce vérin rotatif
B3	I 0.2	Bool	Détecteur pièce métallique
B4	I 0.3	Bool	Magasin plein
B5	I 0.4	Bool	Présence pièce sortie convoyeur (NF)
Start	I 1.0	Bool	Bouton Start
Stop	I 1.1	Bool	Bouton Stop (NF)
Reset	I 1.3	Bool	Bouton Reset
Auto_Man	I 3.0	Bool	Commutateur Auto / Manu / Vient de PLC Sim

La simulation se fera par l'intermédiaire de PLC Sim. Il n'est donc pas nécessaire de faire une configuration matérielle de toutes les E/S, il faut juste faire le choix d'une CPU.

On vous demande de créer un nouveau projet intitulé « première série d'exercices ». La CPU utilisée dans ce projet sera une CPU 314C-2 PN/DP.

Réaliser une table des variable API reprenant les éléments du tableau ci-dessus en ajoutant à cette table les données suivantes :

- Un bit de cadence à 1 Hz
- Un bit de cadence à 2 Hz
- Un bit de cadence à 5 Hz



## 5. Les opérations binaires

Les opérations binaires sont les opérations réalisées sur les variables de type « Bool ». On retrouve dans ces opérations les fonctions logiques (ET, OU, OU Exclusif), les bascule RS, les détections de fronts,...









Plusieurs langages de programmation sont disponibles : le schéma à contact (LADDER), les logigrammes (LOG) ou le langage LIST (instructions). Seul le langage Ladder sera utilisé dans ce cours.

## 5.1 Interrogation à 1 et à 0

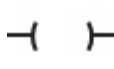
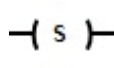
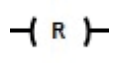
Les symboles d'interrogation à 1 et à 0 sont très utilisés en Ladder. C'est à partir de ces symboles que l'on va pouvoir réaliser des fonctions logiques, lancer des temporisations, des compteurs,...

- 
 ➔ Interrogation à 1 ou Contact NO (à fermeture). Il fournit le résultat d'interrogation 1 lorsque l'opérande interrogé est à l'état 1.
- 
 ➔ Interrogation à 0 ou Contact NF (à ouverture). Il fournit le résultat d'interrogation 1 lorsque l'opérande interrogé est à l'état 0.

Le résultat de l'interrogation va être différent en fonction du type de détecteur interrogé. Le tableau ci-dessous montre les différents cas qui peuvent être rencontrés.

Processus			Evaluation du programme dans l'automate				
Le capteur est un ...	Le capteur est ...	La tension à l'entrée est ...	Etat du signal à l'entrée	Interrogation à "1"		Interrogation à "0"	
				Symbole / instruction	Résultat interrogation	Symbole / instruction	Résultat interrogation
 contact NO	activé 	présente	1	CONT :  contact dit « NO »	« oui » 1	CONT :  contact dit « NF »	« non » 0
	inactivé 	absente	0		« non » 0		« oui » 1
 contact NF	activé 	absente	0		« non » 0		« oui » 1
	inactivé 	présente	1		« oui » 1		« non » 0

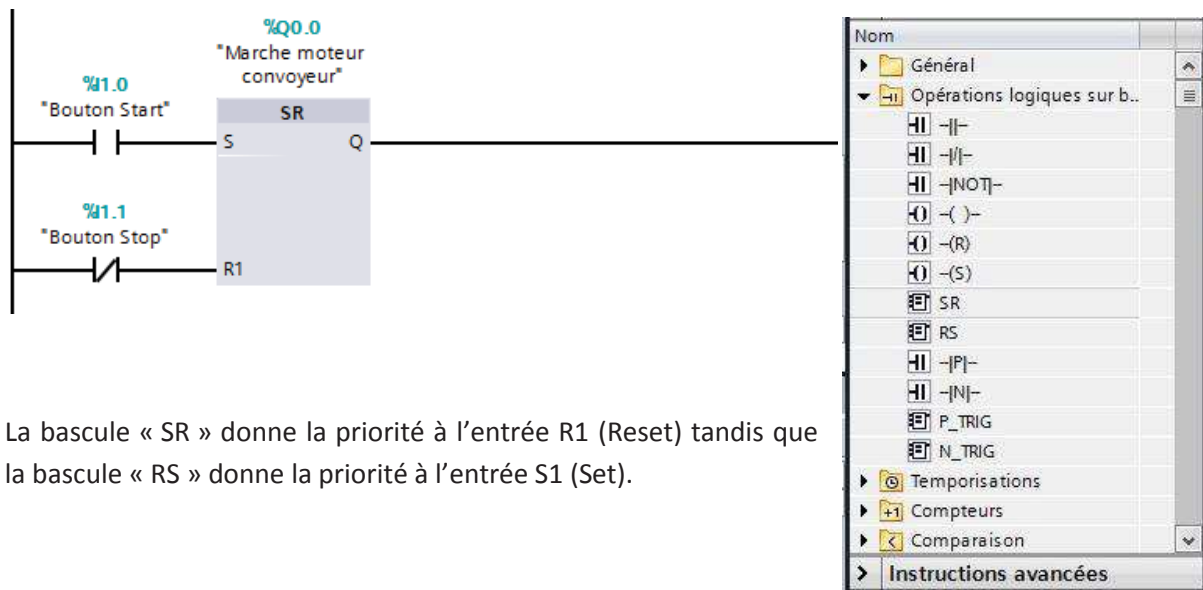
## 5.2 Affectation / Mise à 1 et mise à 0

- 
 L'affectation transmet le résultat logique à l'opérande indiquée. Ce résultat affecté reste disponible après l'affectation et peut être affecté à un autre opérande ou relié à d'autres opérations
- 
 Mise à 1 ➔ Si le résultat logique = « 1 », l'opérande indiquée est mise à « 1 » (SET). Si le résultat = 0, l'opérande reste inchangé.
- 
 Mise à 0 ➔ Si le résultat logique = « 1 », l'opérande indiquée est mise à « 0 » (RESET). Si le résultat = 0, l'opérande reste inchangé.





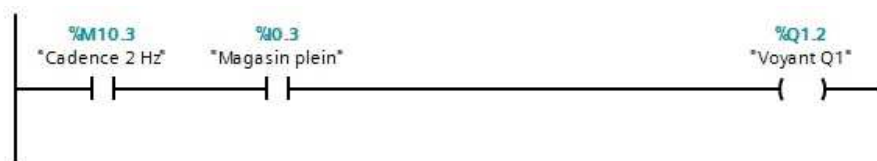
Les instructions SET et RESET peuvent être combinées en utilisant les bascules « SR » et « RS ».



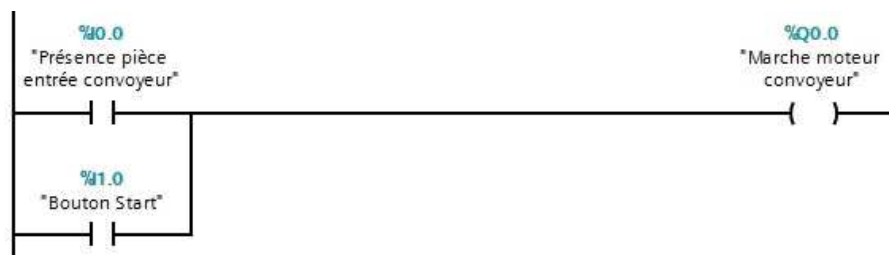
La bascule « SR » donne la priorité à l'entrée R1 (Reset) tandis que la bascule « RS » donne la priorité à l'entrée S1 (Set).

### 5.3 Fonctions logiques ET, OU, OU Exclusif

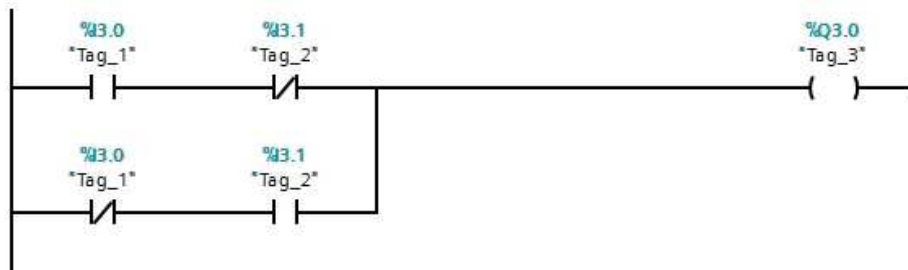
**Fonction ET :** le résultat logique est égal à « 1 » lorsque toutes les entrées sont à l'état « 1 ». En langage Ladder, cela revient à mettre les contacts en série.



**Fonction OU :** le résultat logique est égal à « 1 » lorsqu'au moins une des entrées est à l'état « 1 ». En langage Ladder, cela revient à mettre les contacts en parallèle.



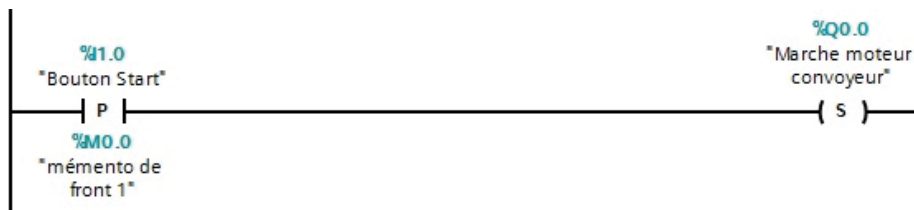
**Fonction OU Exclusif :** le résultat logique est égal à « 1 » lorsqu'une seule des deux entrées est à l'état 1 (cas de deux entrées). En langage Ladder, l'instruction se programme comme indiqué dans la figure ci-dessous.



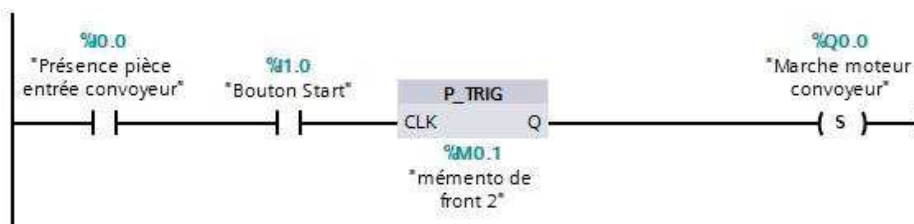
### 5.4 Détection de front

Il est possible d'évaluer le front d'un signal. Cela permet de détecter le passage de « 0 » à « 1 » (front montant) ou le passage de « 1 » à « 0 » (front descendant) d'un opérande.

Les symboles d'interrogation des fronts montant (P) ou descendant (N) d'un opérande doivent toujours être associées à un memento. Pour chaque détection de front, le memento doit être différent (même pour le même opérande).



La réponse au front d'un résultat logique, comme par exemple les fonctions ET ou OU, se fait à l'aide des instructions « P\_TRIG » et « N\_TRIG ».



### 5.5 Exercices : commande de la ligne de convoyage

**Exercice 1 :** lorsque l'on appuie sur le bouton « Start », le convoyeur se met en marche et lorsque l'on appuie sur le bouton « Reset », le vérin rotatif sort.

**Exercice 2 :** une impulsion sur le bouton « Start » démarre le convoyeur et une impulsion sur le bouton « Stop » l'arrête. Le convoyeur ne peut démarrer que si une pièce est présente à l'entrée de celui-ci.

**Exercice 3 :** Le convoyeur démarre lorsqu'une pièce est présente à l'entrée et que l'on appuie sur le bouton Start. Il s'arrête lorsque la pièce franchi la cellule B5 (présence pièce sortie convoyeur).

Lorsque la pièce est présente à l'entrée du convoyeur, le voyant Q1 est allumé. Lorsque le convoyeur est en marche, il clignote à la fréquence de 2 Hz.

**Exercice 4 :** on va distinguer le mode manuel et le mode automatique.

Lorsque le commutateur « Auto\_Man » est sur Manuel (0), le convoyeur et le vérin sont commandés à l'aide des boutons « Start » et « Reset » (impulsions).

Lorsque le commutateur est sur Auto, le convoyeur :

- Démarre si une pièce est présente à l'entrée du convoyeur, que le magasin n'est pas plein et que l'on appuie sur le bouton « Start »
- S'arrête lorsque la pièce a franchi la barrière B5 ou que le mode automatique est désactivé.

De plus, il faut prévoir une signalisation lumineuse :

- Lorsque l'on est en mode automatique, le voyant du bouton Start s'allume.
- Lorsque la pièce est présente à l'entrée du convoyeur, le voyant Q1 est allumé. Lorsque le convoyeur est en marche, il clignote à la fréquence de 2 Hz.
- Lorsque le magasin est plein, Q1 et Q2 clignotent à 1 Hz.