

SIEMENS

SIMATIC

Portail TIA STEP 7 Basic V10.5

Mise en route

Introduction au Portail TIA

1

Exemple simple

2

Exemple avancé

3

Exemple "Régulation PID"

4


Exemple "Motion"


5


Mentions légales

Signalétique d'avertissement

Ce manuel donne des consignes que vous devez respecter pour votre propre sécurité et pour éviter des dommages matériels. Les avertissements servant à votre sécurité personnelle sont accompagnés d'un triangle de danger, les avertissements concernant uniquement des dommages matériels sont dépourvus de ce triangle. Les avertissements sont représentés ci-après par ordre décroissant de niveau de risque.

 DANGER
signifie que la non-application des mesures de sécurité appropriées entraîne la mort ou des blessures graves.

 ATTENTION
signifie que la non-application des mesures de sécurité appropriées peut entraîner la mort ou des blessures graves.

 PRUDENCE
accompagné d'un triangle de danger, signifie que la non-application des mesures de sécurité appropriées peut entraîner des blessures légères.

PRUDENCE
non accompagné d'un triangle de danger, signifie que la non-application des mesures de sécurité appropriées peut entraîner un dommage matériel.

IMPORTANT
signifie que le non-respect de l'avertissement correspondant peut entraîner l'apparition d'un événement ou d'un état indésirable.


En présence de plusieurs niveaux de risque, c'est toujours l'avertissement correspondant au niveau le plus élevé qui est reproduit. Si un avertissement avec triangle de danger prévient des risques de dommages corporels, le même avertissement peut aussi contenir un avis de mise en garde contre des dommages matériels.

Personnes qualifiées

L'appareil/le système décrit dans cette documentation ne doit être manipulé que par du **personnel qualifié** pour chaque tâche spécifique. La documentation relative à cette tâche doit être observée, en particulier les consignes de sécurité et avertissements. Les personnes qualifiées sont, en raison de leur formation et de leur expérience, en mesure de reconnaître les risques liés au maniement de ce produit / système et de les éviter.

Utilisation des produits Siemens conforme à leur destination

Tenez compte des points suivants:

 ATTENTION
Les produits Siemens ne doivent être utilisés que pour les cas d'application prévus dans le catalogue et dans la documentation technique correspondante. S'ils sont utilisés en liaison avec des produits et composants d'autres marques, ceux-ci doivent être recommandés ou agréés par Siemens. Le fonctionnement correct et sûr des produits suppose un transport, un entreposage, une mise en place, un montage, une mise en service, une utilisation et une maintenance dans les règles de l'art. Il faut respecter les conditions d'environnement admissibles ainsi que les indications dans les documentations afférentes.

Marques de fabrique

Toutes les désignations repérées par ® sont des marques déposées de Siemens AG. Les autres désignations dans ce document peuvent être des marques dont l'utilisation par des tiers à leurs propres fins peut enfreindre les droits de leurs propriétaires respectifs.

Exclusion de responsabilité

Nous avons vérifié la conformité du contenu du présent document avec le matériel et le logiciel qui y sont décrits. Ne pouvant toutefois exclure toute divergence, nous ne pouvons pas nous porter garants de la conformité intégrale. Si l'usage de ce manuel devait révéler des erreurs, nous en tiendrons compte et apporterons les corrections nécessaires dès la prochaine édition.

Sommaire

1	Introduction au Portail TIA	7
1.1	Présentation	7
1.2	Concepts fondamentaux du portail TIA	9
1.2.1	Présentation du portail TIA	9
1.2.2	Concept d'ingénierie	10
1.2.3	Gestion des données	11
1.3	Vues du portail TIA	12
1.3.1	Navigation dans le portail TIA	12
1.3.2	Vue du portail	13
1.3.3	Vue du projet	14
1.4	Projets-exemples	16
1.4.1	Présentation	16
1.4.2	Chargement de projets	18
2	Exemple simple	23
2.1	Introduction	23
2.2	Création du projet	23
2.3	Insertion et configuration d'un automate	25
2.3.1	Insertion d'un automate	25
2.3.2	Présentation de l'éditeur d'appareils et de réseaux	28
2.3.3	Configuration de l'automate	31
2.4	Création du programme	33
2.4.1	Qu'est-ce qu'un bloc d'organisation ?	33
2.4.2	Ouverture d'un bloc d'organisation	34
2.4.3	Présentation de l'éditeur de programmes	36
2.4.4	Qu'est-ce qu'un réseau ?	37
2.4.5	Insertion d'instructions CONT	39
2.4.6	Qu'est-ce qu'une variable ?	44
2.4.7	Définition et connexion de variables API	47
2.5	Test du programme	51
2.5.1	Chargement du programme dans le système cible	51
2.5.2	Test du programme avec la visualisation de l'état du programme	57
2.6	Création d'une vue IHM	61
2.6.1	Visualisation dans le portail TIA	61
2.6.2	Création d'un pupitre opérateur avec vue IHM	61
2.6.3	Qu'est-ce qu'un objet graphique ?	69
2.6.4	Création et configuration d'objets graphiques	70
2.6.4.1	Bouton "Marche/Arrêt installation"	70
2.6.4.2	Objets graphiques "DEL"	73
2.6.4.3	Objet graphique "Bande transporteuse"	78
2.6.4.4	Objet graphique "Bouteille" avec simulation de déplacement	81
2.6.4.5	Commande de la visibilité de l'animation de déplacement	86

2.7	Test de la vue IHM	89
2.7.1	Chargement de la vue IHM dans le pupitre opérateur	89
2.7.2	Simulation Runtime	91
3	Exemple avancé	95
3.1	Introduction	95
3.2	Compléter le programme	97
3.2.1	Définition de variables dans la table des variables API	97
3.2.2	Programmation des conditions pour démarrer la bande transporteuse	101
3.2.2.1	Interrogation de l'état de l'installation	101
3.2.2.2	Interrogation de la position de la bouteille et de l'état de la chambre de chauffe	103
3.2.2.3	Interrogation de l'avancement de la pasteurisation	105
3.2.2.4	Commande de la bande transporteuse.....	107
3.2.3	Programmation des conditions pour arrêter la bande transporteuse	109
3.2.4	Programmation de la commande du chauffage	114
3.2.5	Programmation de la durée de chauffage.....	116
3.2.6	Programmation du voyant d'état	120
3.3	Test du programme complété avec la visualisation de l'état du programme.....	124
3.4	Compléter la vue IHM	131
3.4.1	Objet graphique "Chambre de chauffe"	131
3.4.2	Objet graphique "DEL de la chambre de chauffe"	133
3.4.3	Objets graphiques "Barrières photoélectriques"	135
3.5	Simulation de la vue IHM	140
4	Exemple "Régulation PID"	143
4.1	Introduction	143
4.2	Création du bloc d'organisation pour le régulateur PID	146
4.3	Création de l'objet technologique pour le régulateur PID	149
4.4	Chargement du bloc de simulation	151
4.5	Configuration du régulateur PID	155
4.6	Modification de la commande de la chambre de chauffe	160
4.7	Intégration dans le programme de commande d'une comparaison de température comme condition	162
4.8	Adaptation de la vue IHM.....	166
4.9	Activation du régulateur PID en mode en ligne.....	169

5	Exemple "Motion"	173
5.1	Introduction	173
5.2	Insertion de l'objet technologique "Axe"	177
5.3	Configuration de l'objet technologique "Axe"	179
5.4	Déblocage de l'axe.....	182
5.5	Positionnement relatif de l'axe	185
5.6	Compléter la vue IHM	189
5.6.1	Modification de l'objet graphique Bande transporteuse.....	189
5.6.2	Création d'un deuxième objet graphique Bouteille	191
5.6.3	Association d'objets IHM avec une instruction Motion.....	195
5.7	Simulation de la vue IHM	198
5.8	Démarrage de la vue de diagnostic	200
	Glossaire	205

Introduction au Portail TIA

1.1 Présentation

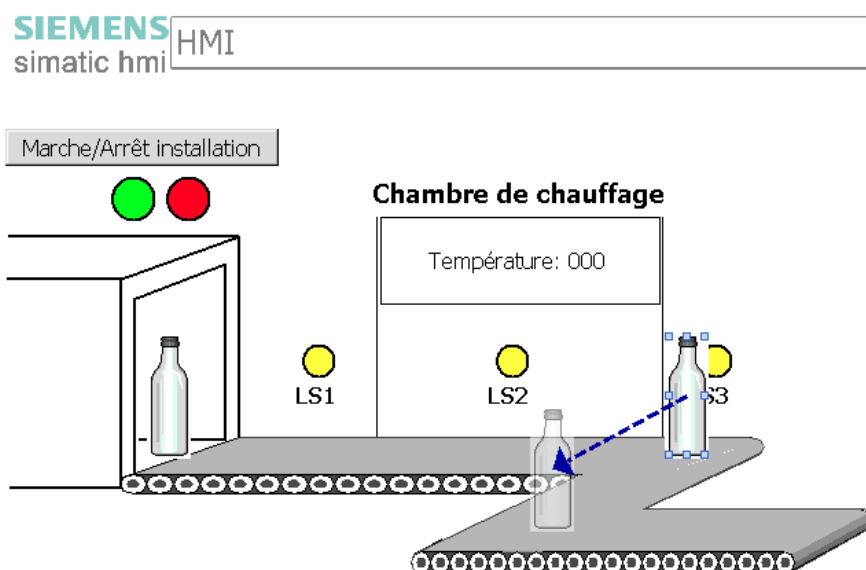
Introduction

Bienvenue dans "STEP 7 Basic V10.5 Getting Started".

Ce guide de mise en route Getting Started montre la simplicité d'utilisation du portail TIA à l'aide d'un projet-exemple.

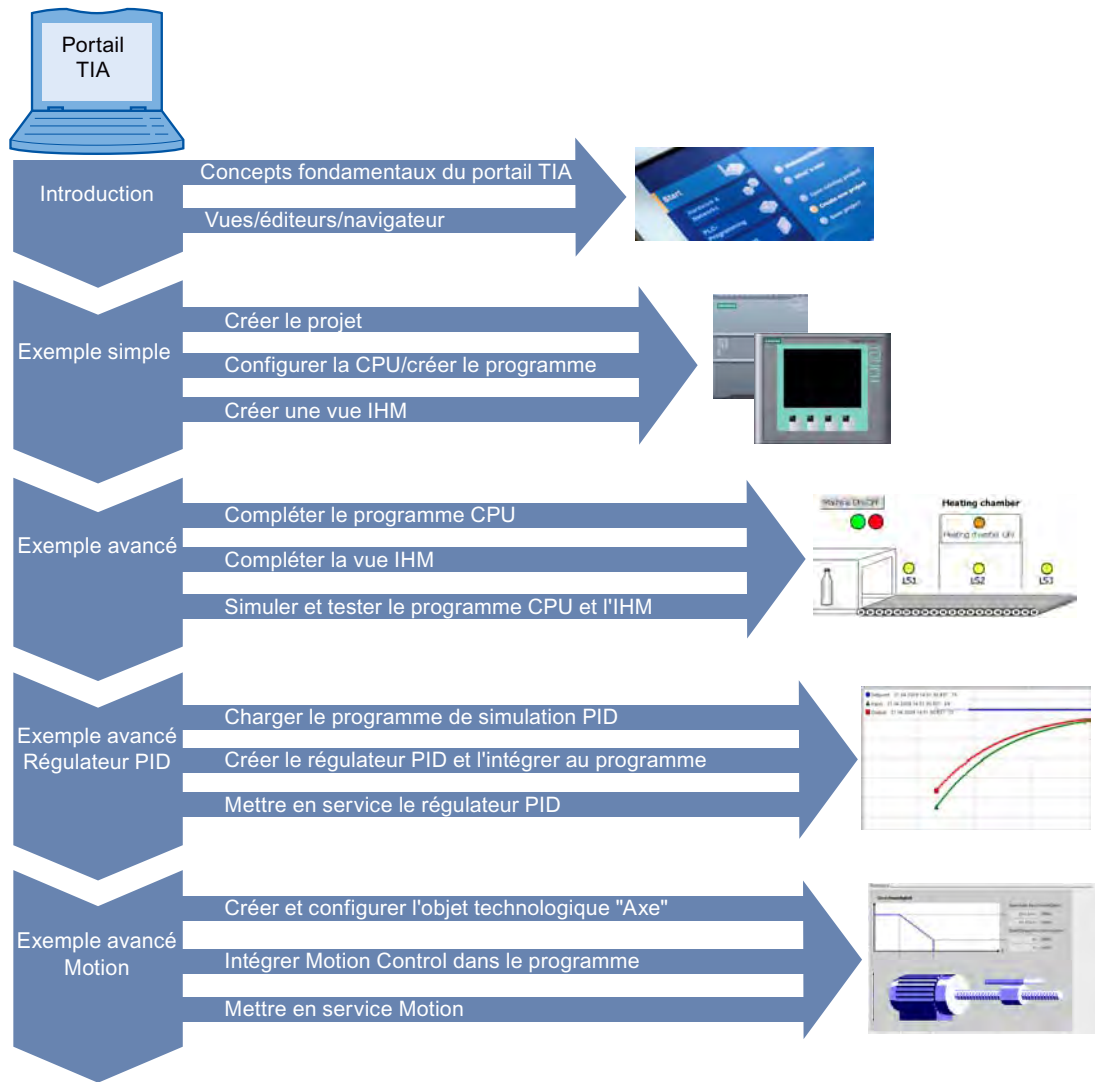
Projet-exemple

Le projet-exemple créé dans ce guide de mise en route est une station pour la pasteurisation de lait dans une chambre de chauffe. Dans la représentation simplifiée de ce procédé, des bouteilles individuelles sont transportées dans la chambre de chauffe sur une bande transporteuse, puis en sont retirées une fois l'opération de chauffage achevée.



Organisation du guide de mise en route Getting Started

Le projet-exemple est complété dans chaque chapitre. Vous partez d'un projet simple dans lequel vous utilisez uniquement les fonctions de base du portail TIA et vous le complétez étape par étape en utilisant des fonctions toujours plus complexes du portail TIA. Les utilisateurs expérimentés peuvent sauter des chapitres et les débutants trouveront en outre dans le chapitre "Exemple simple" des informations générales sur la programmation et la visualisation.



1.2 Concepts fondamentaux du portail TIA

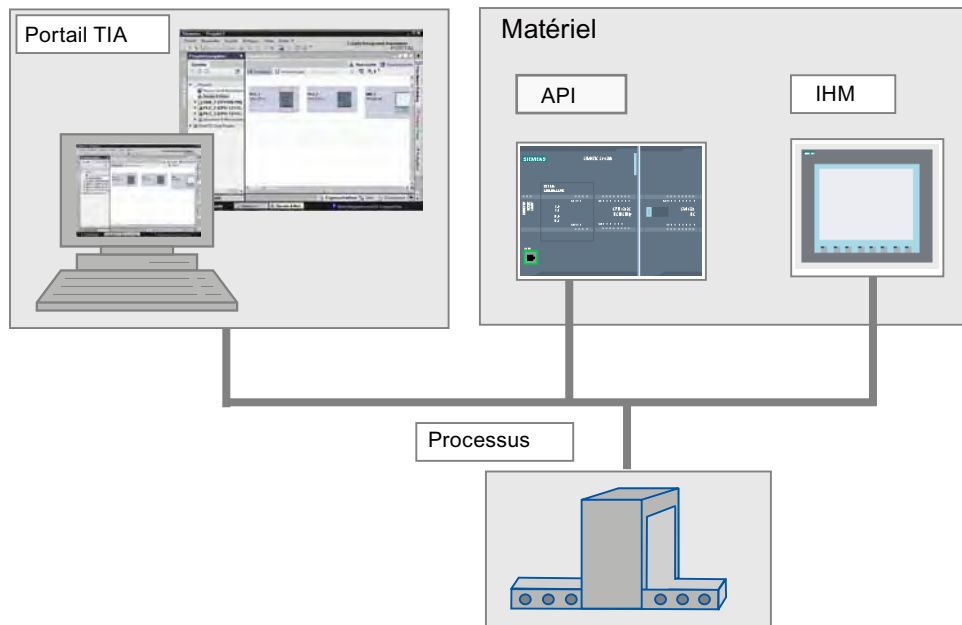
1.2.1 Présentation du portail TIA

Introduction

Le portail Totally Integrated Automation (portail TIA) intègre différents produits SIMATIC dans une application logicielle qui vous permet d'accroître votre productivité et votre efficacité. Les produits TIA fonctionnent ensemble dans le portail TIA pour vous assister dans toutes vos tâches de création de solutions d'automatisation.

Dans une tâche d'automatisation typique :

- un automate commande le processus à l'aide du programme,
- un pupitre opérateur permet de conduire et de visualiser le processus.



Tâches

Le portail TIA vous assiste lors de la création d'une solution d'automatisation. Les principales étapes de configuration sont les suivantes :

- Création du projet
- Configuration du matériel
- Mise en réseau des appareils
- Programmation de l'automate
- Configuration de la visualisation
- Chargement des données de configuration
- Utilisation des fonctions en ligne et des fonctions de diagnostic

Avantages

Les avantages du portail TIA sont les suivants :

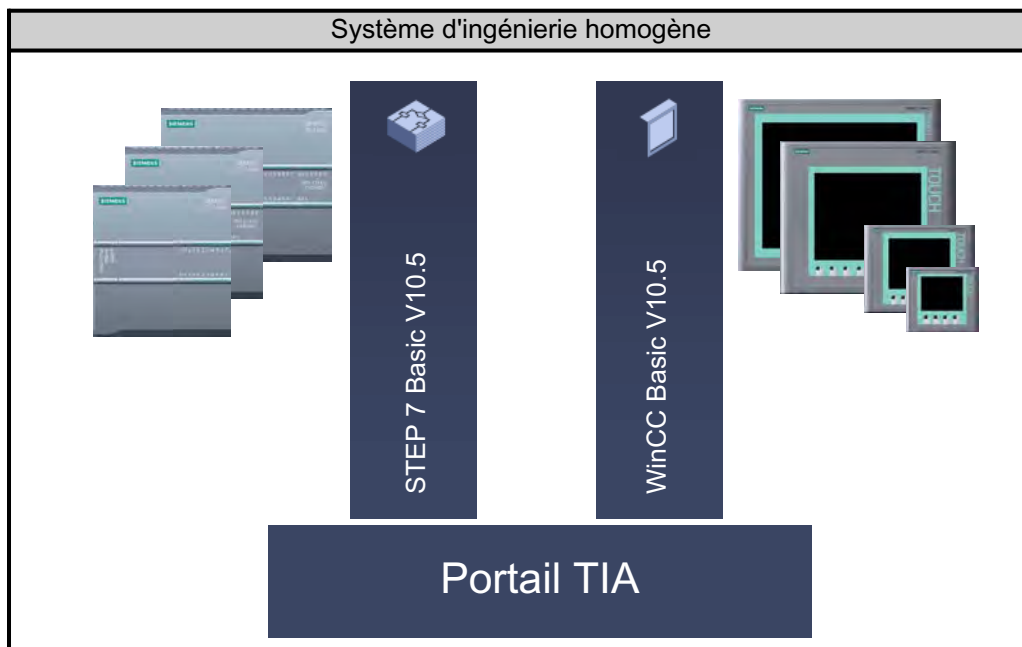
- Gestion commune des données
- Manipulation homogène des programmes, données de configuration et données de visualisation
- Edition simple par glisser-déplacer
- Chargement aisé des données dans les appareils
- Commande homogène
- Configuration et diagnostic graphiques

1.2.2 Concept d'ingénierie

Système d'ingénierie

Grâce au portail TIA, vous configurez aussi bien la commande que la visualisation dans un système d'ingénierie homogène. L'ensemble des données est stocké dans un projet. Les composants pour la programmation (STEP 7) et la visualisation (WinCC) ne sont pas des programmes autonomes mais les éditeurs d'un système qui accède à une base de données commune. Toutes les données sont enregistrées dans un fichier de projet commun.

Vous utilisez pour toutes les tâches une interface utilisateur commune vous permettant d'accéder à tout moment à toutes les fonctions de programmation et de visualisation.



1.2.3 Gestion des données

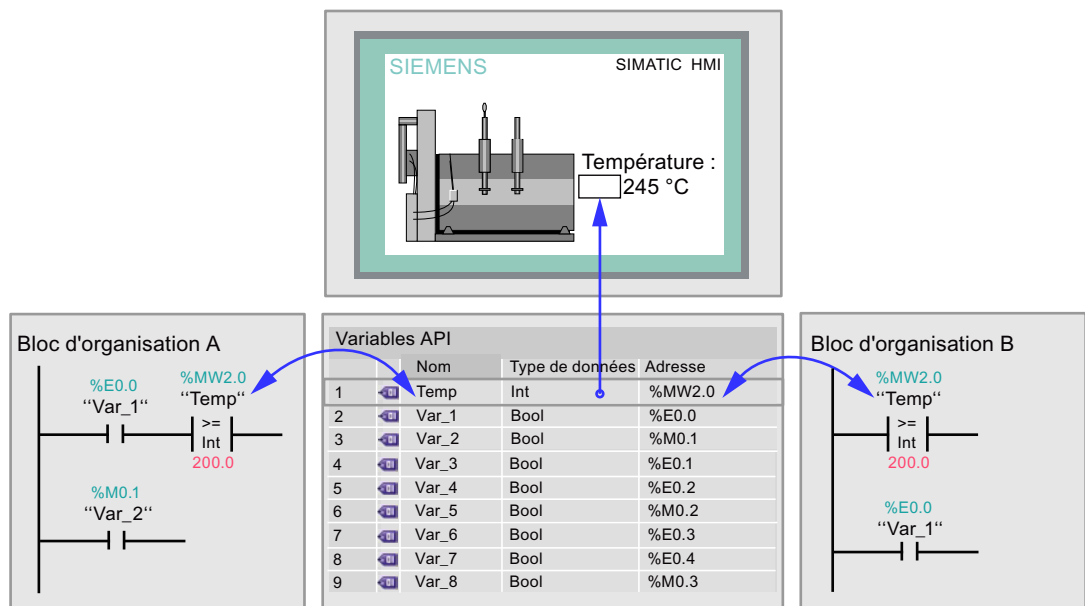
Gestion centralisée des données

Dans le portail TIA, l'ensemble des données est stocké dans un projet. Les données d'application modifiées, par exemple les variables, sont actualisées automatiquement dans tout le projet, sur plusieurs appareils si nécessaire.

Adressage symbolique global

Lorsque vous utilisez une variable de process dans plusieurs blocs de différents automates et dans des vues IHM, vous pouvez la créer ou la modifier à n'importe quel endroit dans le programme. Le bloc et l'appareil dans lesquels vous procédez à la modification sont indifférents. Le portail TIA offre les possibilités suivantes pour la définition de variables API :

- Définition dans la table des variables API
- Définition dans l'éditeur de programmes
- Définition par association avec des entrées et sorties de l'automate



Toutes les variables API définies figurent dans la table des variables API et peuvent y être éditées. Les modifications sont effectuées de manière centralisée et l'actualisation est continue. Grâce à la gestion cohérente des données, la synchronisation entre les participants à un même projet, le programmeur et le concepteur IHM par exemple, devient inutile.

Concept de bibliothèque

Des parties du projet peuvent être réutilisées aussi bien à l'intérieur du projet que dans d'autres projets grâce à la bibliothèque.

- Les éléments, tels que blocs, variables API, tables de variables, alarmes, vues IHM, modules individuels ou stations complètes, peuvent être stockés aussi bien dans des bibliothèques locales que dans des bibliothèques globales.
- Appareils et fonctions définies peuvent être réutilisés.
- La bibliothèque globale permet d'échanger facilement des données entre projets.

1.3 Vues du portail TIA

1.3.1 Navigation dans le portail TIA

Introduction

Vous travaillerez avec différentes vues lors de la création des projets. Aussi le paragraphe suivant vous donne-t-il un aperçu des vues disponibles dans le portail TIA.

Vues du portail TIA

Le portail TIA offre pour vos projets d'automatisation deux vues de travail différentes qui permettent un accès rapide aux outils et aux composants individuels du projet :

- Vue du portail : La vue du portail permet la configuration orientée tâche.
- Vue du projet : La vue du projet permet la configuration orientée objet.

Navigation

Vous pouvez commuter à tout moment entre la vue du portail et la vue du projet grâce au lien situé dans le coin inférieur gauche de l'interface utilisateur. Pendant la configuration, la vue change automatiquement en fonction du type de tâche exécutée. Par exemple, si vous souhaitez éditer un objet présenté dans la vue du portail, l'application bascule automatiquement dans l'éditeur correspondant dans la vue du projet. Une fois l'objet édité, vous pouvez repasser à la vue du portail et y poursuivre votre travail avec l'objet suivant ou l'activité suivante.

Enregistrement global des données du projet

Lors de l'enregistrement, c'est toujours l'ensemble du projet qui est sauvegardé, quelle que soit la vue ou quel que soit l'éditeur ouvert.

1.3.2 Vue du portail

Vue du portail

La vue du portail offre une vue orientée tâche sur les outils. La vue du portail a pour but de vous faciliter la navigation dans les tâches et données du projet. A cet effet, les fonctions de l'application sont accessibles via des portails distincts, en fonction des principales tâches à réaliser. La figure suivante montre la structure de la vue du portail :



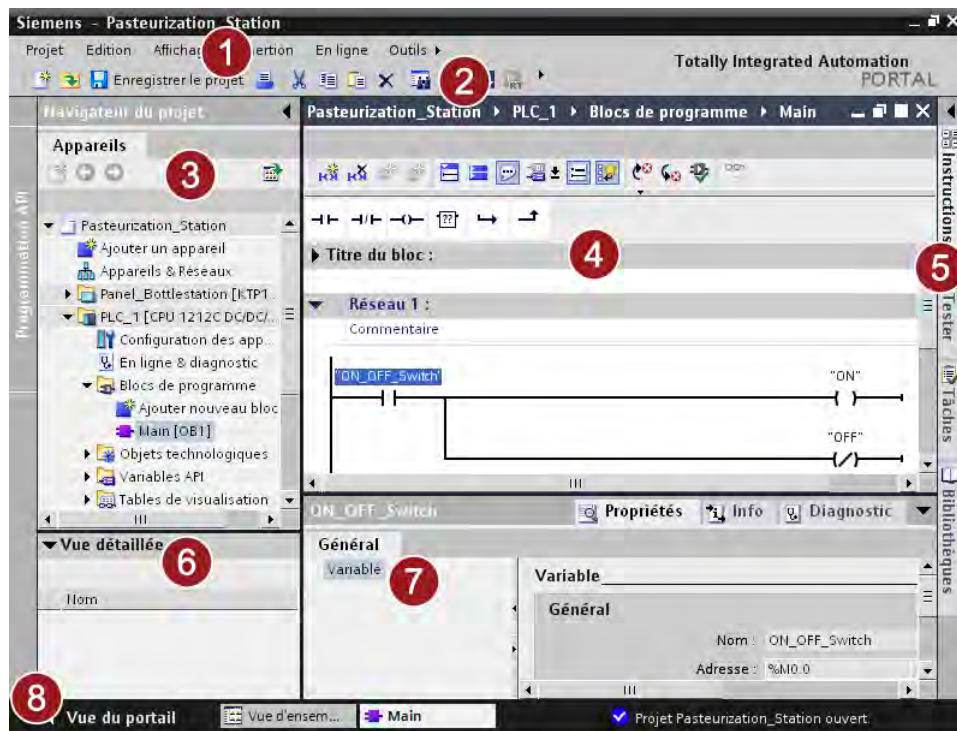
- ① Portails pour les différentes tâches :
Les portails mettent à disposition les fonctions élémentaires requises par chaque type de tâche. Les portails proposés dans la vue du portail dépendent des produits installés.
- ② Actions correspondant au portail sélectionné :
En fonction du portail sélectionné, les actions que vous pouvez exécuter dans ce portail vous sont proposées ici. L'appel d'une aide contextuelle vous est proposée dans chaque portail.
- ③ Fenêtre de sélection correspondant à l'action sélectionnée :
La fenêtre de sélection est disponible dans chaque portail. Son contenu s'adapte à la sélection en cours.
- ④ Basculement dans la vue du projet :
Le lien "Vue du projet" vous permet de basculer dans la vue du projet.
- ⑤ Affichage du projet actuellement ouvert :
Indique quel est le projet actuellement ouvert.

1.3.3 Vue du projet

Vue du projet

La vue du projet correspond à une vue structurée de l'ensemble des composants d'un projet. Dans la vue du projet, vous disposez de différents éditeurs à l'aide desquels vous créez et éditez les composants de projet correspondants.

La figure suivante montre la structure de la vue du projet :



- ① Barre des menus :
La barre des menus contient toutes les commandes dont vous avez besoin pour réaliser votre tâche.
- ② Barre d'outils :
La barre d'outils met à votre disposition des boutons vous permettant d'exécuter les commandes les plus fréquemment utilisées. L'accès à ces commandes est ainsi plus rapide que via des menus.
- ③ Navigateur du projet :
Le navigateur du projet vous permet d'accéder à tous les composants et données du projet. Vous pouvez par exemple réaliser les actions suivantes dans le navigateur du projet :
 - Ajouter de nouveaux composants
 - Editer des composants existants
 - Interroger et modifier les propriétés de composants existants
- ④ Zone de travail :
La zone de travail affiche les objets que vous ouvrez afin de les éditer.

- ⑤ **Task Cards :**
Vous disposez de Task Cards en fonction de l'objet édité ou sélectionné. Les Task Cards disponibles figurent dans une barre sur le bord droit de l'écran. Vous pouvez les ouvrir ou les fermer à tout moment.
- ⑥ **Vue détaillée :**
La vue détaillée affiche certains contenus d'un objet sélectionné. Il peut s'agir par exemple de listes de textes ou de variables.
- ⑦ **Fenêtre d'inspection :**
La fenêtre d'inspection affiche des informations supplémentaires sur un objet sélectionné ou sur des actions exécutées.
- ⑧ **Basculement dans la vue du portail :**
Le lien "Vue du portail" vous permet de basculer dans la vue du portail.

Remarque

Vous pouvez ouvrir et fermer les différentes fenêtres de la vue du projet à l'aide de la combinaison de touches "<Ctrl> + 1-5". Vous trouverez une présentation de toutes les combinaisons de touches dans le système d'information du portail TIA.

1.4 Projets-exemples

1.4.1 Présentation

Introduction

Le guide de mise en route est subdivisé en quatre chapitres représentant une progression. L'état actuel à la fin d'un chapitre est sauvegardé dans un fichier de projet. Si vous avez sauté un chapitre, vous pouvez charger l'état respectif de ce chapitre.

Contenu des projets-exemples

La présentation suivante décrit à quel groupe cible chaque chapitre s'adresse. Vous pouvez charger les projets qui sont enregistrés sous forme de fichiers ZIP à l'adresse suivante:

<http://support.automation.siemens.com/WW/view/fr/40263542>

Cliquez sur l'icône "Infos" pour afficher les fichiers ZIP.

- Exemple simple

Ce chapitre s'adresse aux débutants sans connaissances préalables. Dans l'exemple simple, vous configurez l'automate et un pupitre opérateur et vous créez un court programme et une vue IHM pour sa visualisation. A la fin du chapitre, il est possible de mettre l'installation en marche et de l'arrêter par le biais du pupitre opérateur.

- Exemple avancé

Si vous avez déjà des connaissances, vous pouvez charger l'exemple simple et poursuivre à partir d'ici. Dans le prolongement de l'exemple simple, vous créez un programme complet en CONT ainsi que les éléments IHM pour la visualisation du processus.

L'état du projet à la fin du chapitre "Exemple simple" est sauvegardé dans le fichier "Simple_Example.ZIP".

- Exemple avancé Régulateur PID

Dans ce chapitre, vous complétez l'exemple avancé avec l'objet technologique "Régulateur PID". Le régulateur PID permet de réguler automatiquement la température dans la chambre de chauffe.

Si vous voulez uniquement utiliser la fonction de l'objet technologique "PID", vous pouvez charger l'état du projet à la fin du chapitre "Exemple avancé" qui se trouve dans le fichier "Extended_Example.ZIP".

- Exemple avancé Motion Control

Dans ce chapitre, vous complétez l'exemple avancé avec l'objet technologique "Axe". L'objet technologique "Axe" pilote automatiquement une deuxième bande transporteuse. Vous commandez le positionnement de la bouteille sur la bande transporteuse par le biais d'une instruction Motion Control. L'état du projet avant ce chapitre est sauvegardé dans le fichier "Extended_Example_PID.ZIP".

L'état du projet à la fin du guide de mise en route est sauvegardé dans le fichier "Extended_Example_Motion.ZIP".



ATTENTION

N'utilisez les projets-exemples qu'à des fins de test

Les exemples et descriptions fournis servent uniquement à vous donner un aperçu des fonctions de base du portail TIA.

- Utilisez ces exemples uniquement dans un environnement de test et non dans une installation qui fonctionne.
- Le chargement des programmes-exemples fournis durant le fonctionnement de l'installation risque de provoquer des dysfonctionnements, des erreurs de programme, des dégâts matériels importants et des blessures graves !

1.4.2 Chargement de projets

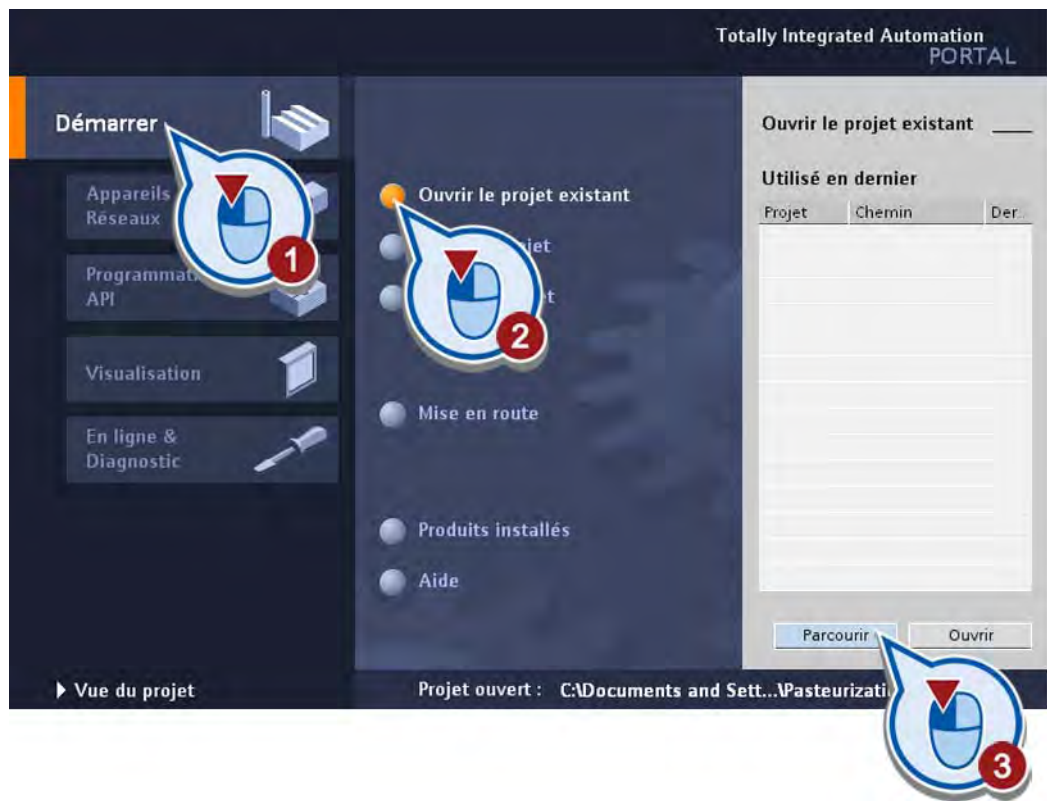
Introduction

Le guide de mise en route Getting Started commence avec le chapitre "Exemple simple (Page 23)". Si vous ne voulez pas étudier le guide de mise en route dans sa totalité, vous pouvez charger un état du projet de la manière décrite ci-après.

Marche à suivre

Procédez de la manière suivante pour charger un projet :

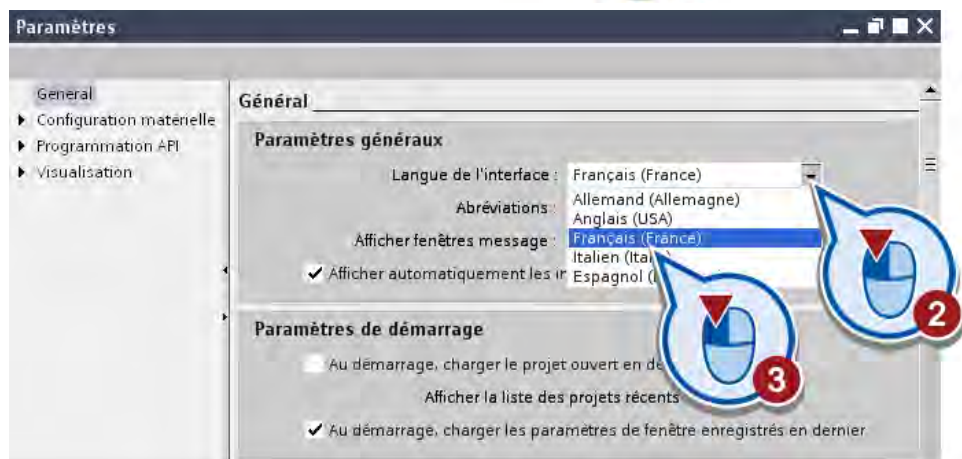
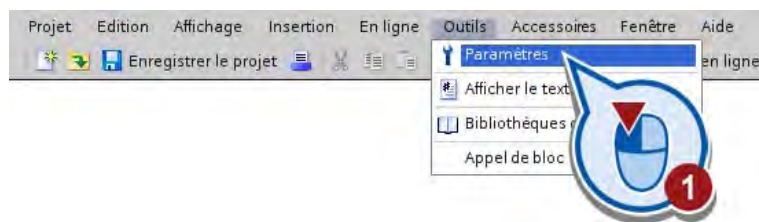
1. Ouvrez le projet correspondant par biais de la vue du portail du portail TIA.



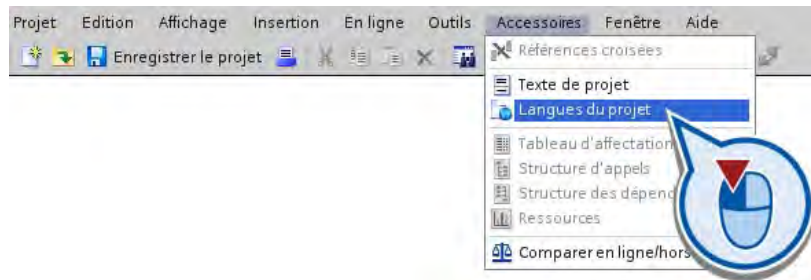


Le projet se charge.

2. Ouvrez la vue du projet.
3. Sélectionnez la langue de l'interface utilisateur.

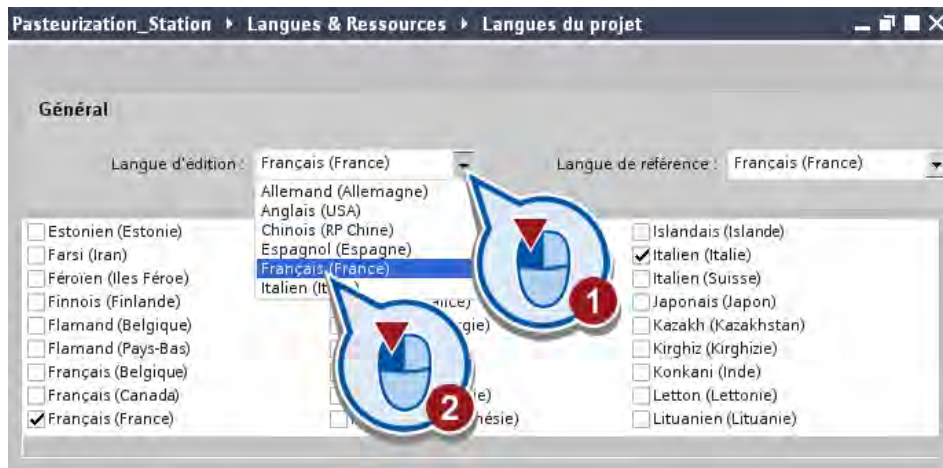


4. Ouvrez la fenêtre pour le paramétrage de la langue du projet.



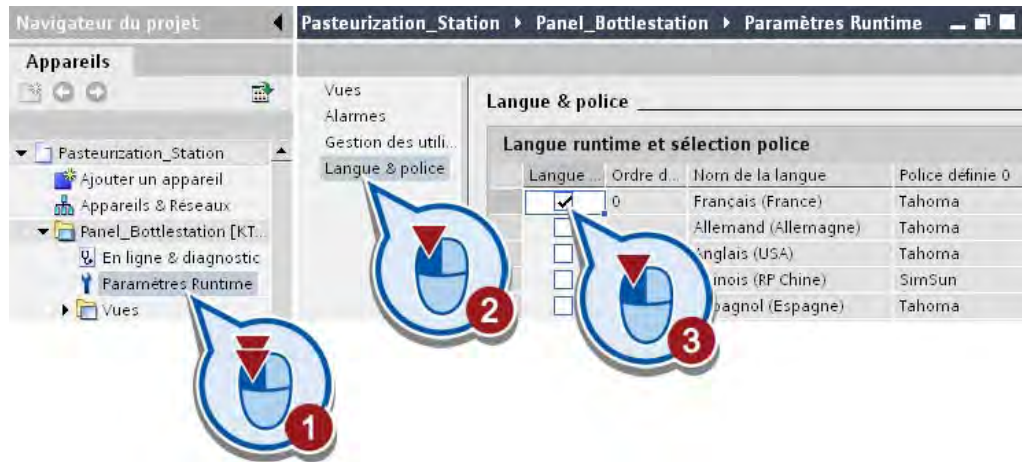
La fonction Langues du projet vous permet de sélectionner dans quelle langue les textes du projet ainsi que les zones de texte et les commentaires doivent s'afficher.

5. Sélectionnez la langue pour les textes du projet.



La fonction Langues du projet vous permet de sélectionner dans quelle langue les textes du projet ainsi que les zones de texte et les commentaires doivent s'afficher.

6. Dans la vue du projet, sélectionnez la langue souhaitée dans les paramètres Runtime.



Le paramétrage de la langue vous permet d'indiquer quelles langues doivent être chargées dans le pupitre opérateur. Le nombre de langues que vous pouvez charger dans le pupitre opérateur dépend du pupitre opérateur utilisé. La langue ayant le numéro d'ordre "0" est affichée en premier au démarrage.

Remarque

Changement de langue

Les noms des variables restent inchangés après un changement de langue de projet, puisque la désignation des variables doit être univoque.

Exemple simple

2.1 Introduction

Etapas de travail

Dans la première partie du guide de mise en route, vous allez programmer un interrupteur poussoir électrique pour la mise en marche et l'arrêt d'une installation. L'installation se met en marche si l'on appuie une fois sur l'interrupteur. Si l'on appuie à nouveau sur l'interrupteur, l'amenée de courant est interrompue et l'installation s'arrête.

Les étapes de travail suivantes sont prévues :

- Création du projet
- Configuration de l'automate
- Création du programme
- Chargement du programme dans l'automate
- Test du programme
- Création d'une vue IHM

2.2 Création du projet

Introduction

Les étapes suivantes décrivent comment créer un nouveau projet. Les données et les programmes qui sont générés lors de la création d'une tâche d'automatisation sont stockés de manière ordonnée dans le projet. Pour cet exemple, vous ouvrez le portail Totally Integrated Automation dans la vue du portail. Le portail de démarrage contient les commandes permettant de créer un projet ou d'ouvrir un projet existant.

Marche à suivre

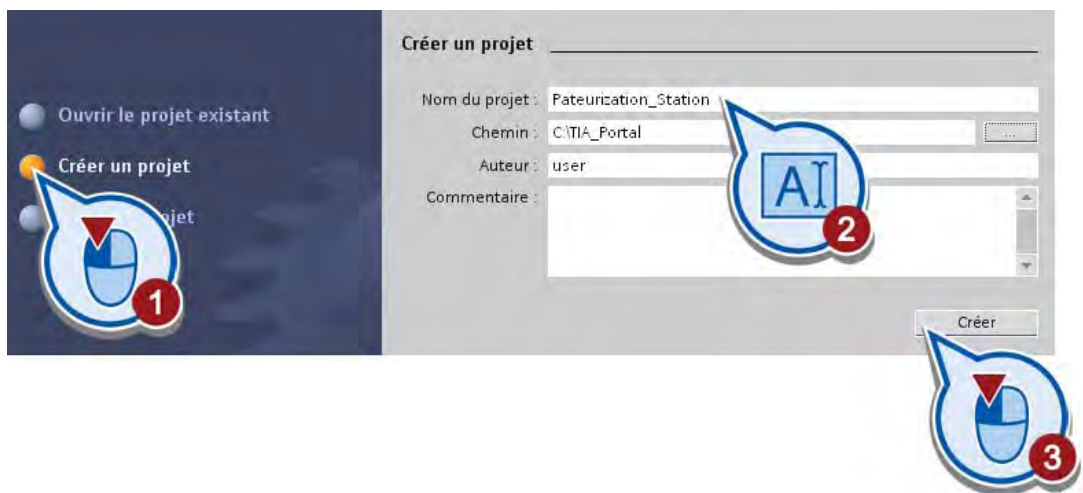
Procédez de la manière suivante pour créer un projet :

1. Démarrez le portail Totally Integrated Automation.



Le portail Totally Integrated Automation s'ouvre dans la vue du portail.

2. Créez le projet "Pasteurization_Station" dans un chemin au choix.



Résultat

Vous avez créé un projet. Dans le paragraphe suivant, vous allez insérer un nouvel automate dans le projet et configurer ses propriétés.

2.3 Insertion et configuration d'un automate

2.3.1 Insertion d'un automate

Introduction

Les étapes suivantes décrivent comment insérer un automate par le biais de la vue du portail et comment ouvrir sa configuration dans la vue du projet. Le type d'automate que vous créez dans le projet doit correspondre au matériel disponible.

Condition requise

Le projet a été créé.

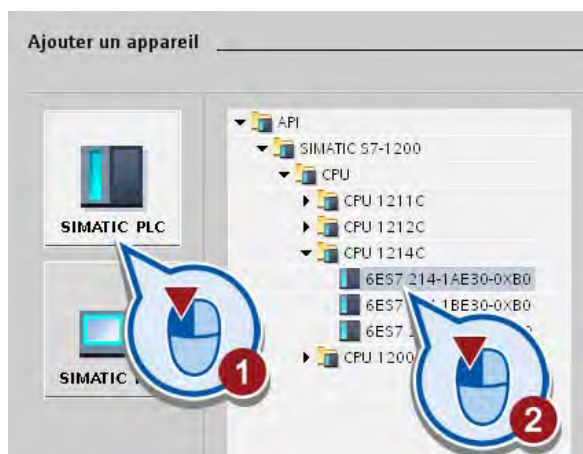
Marche à suivre

Procédez de la manière suivante pour ajouter un nouvel appareil dans le projet :

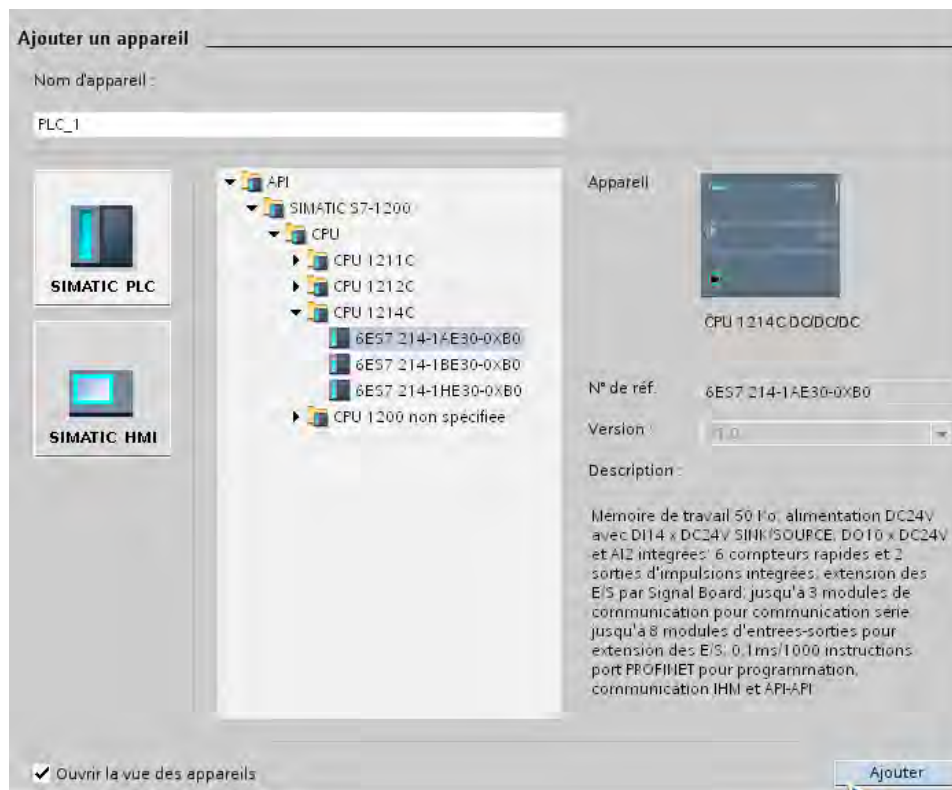
1. Insérez un nouvel appareil par le biais du portail.



2. Sélectionnez l'automate désiré.

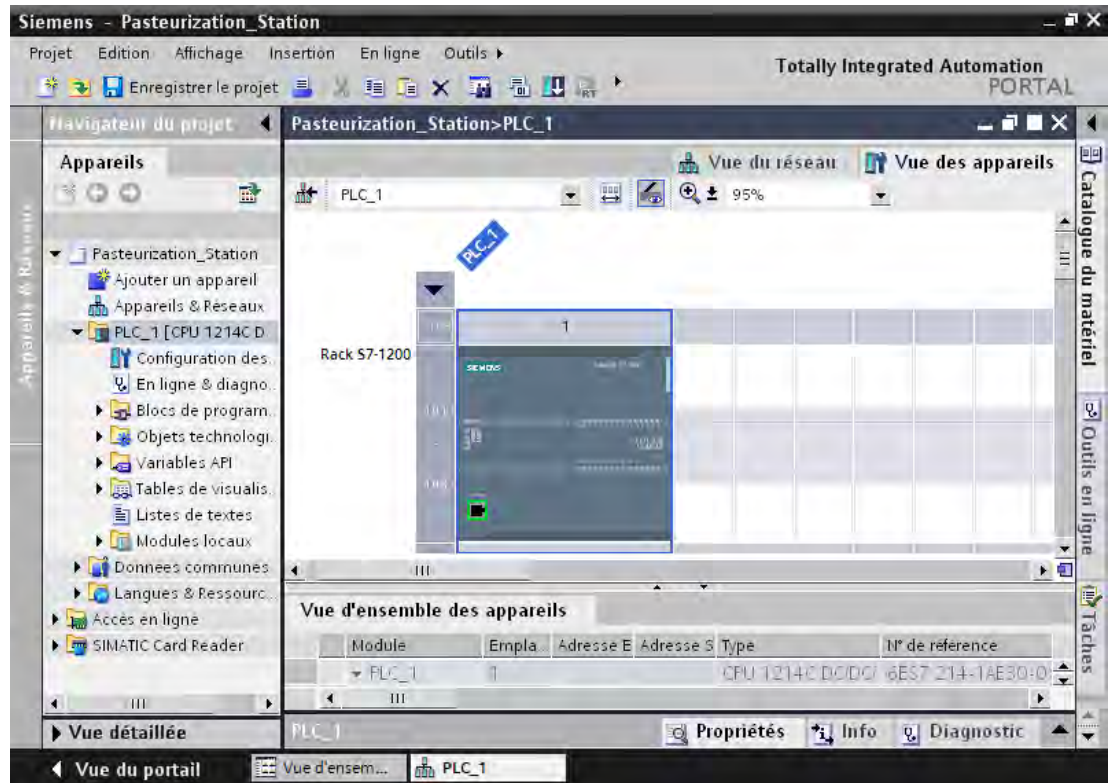


3. Vérifiez que l'option "Ouvrir la vue des appareils" est activée.
Si elle est désactivée, cliquez avec le bouton gauche de la souris sur l'option pour l'activer.
4. Cliquez sur le bouton "Ajouter".



Résultat

Vous avez créé un nouvel automate dans le projet et l'avez ouvert dans la vue des appareils de l'éditeur Appareils et réseaux.



2.3.2 Présentation de l'éditeur d'appareils et de réseaux

Fonction de l'éditeur d'appareils et de réseaux

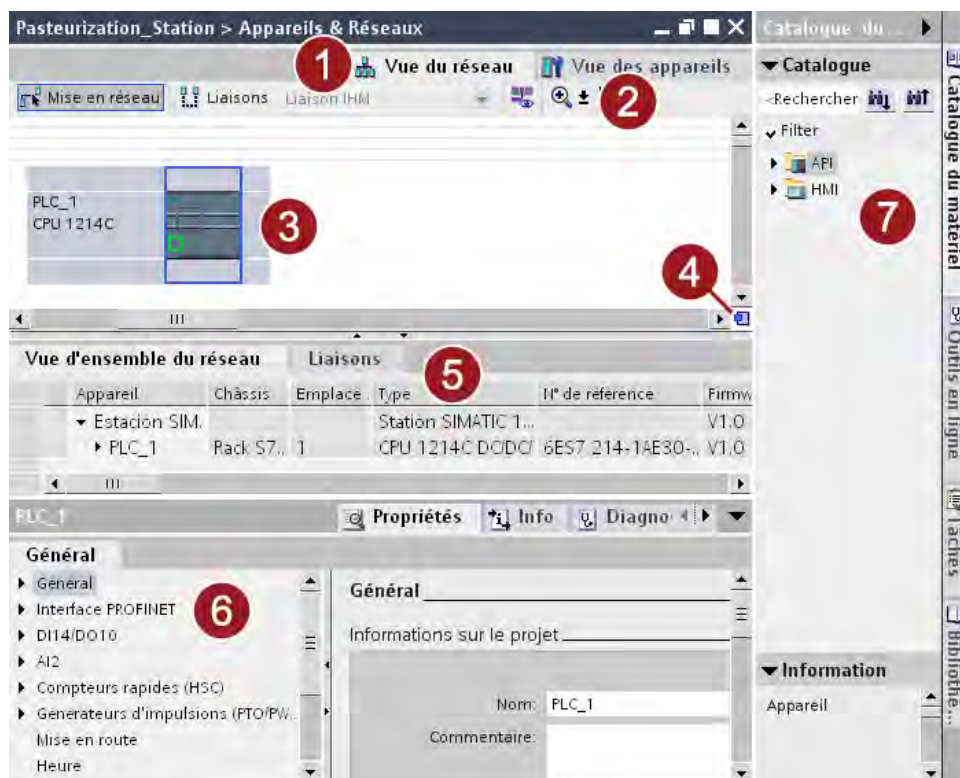
L'éditeur d'appareils et de réseaux est l'environnement de développement intégré pour la configuration, le paramétrage et la mise en réseau des appareils et des modules. Il est composé d'une vue du réseau et d'une vue des appareils. Vous pouvez à tout moment basculer entre ces deux éditeurs.

Vue du réseau

La vue du réseau est la zone de travail de l'éditeur d'appareils et de réseaux dans laquelle vous pouvez exécuter les tâches suivantes :

- Configurer et paramétrer les appareils
- Interconnecter les appareils

La figure suivante montre la structure de la vue du réseau :



① Onglets pour commuter entre la vue des appareils et la vue du réseau

② Barre d'outils :

La barre d'outils contient les outils pour la mise en réseau graphique d'appareils, la configuration de liaisons et l'affichage des informations d'adresse. La fonction zoom vous permet de modifier la représentation dans la zone graphique.

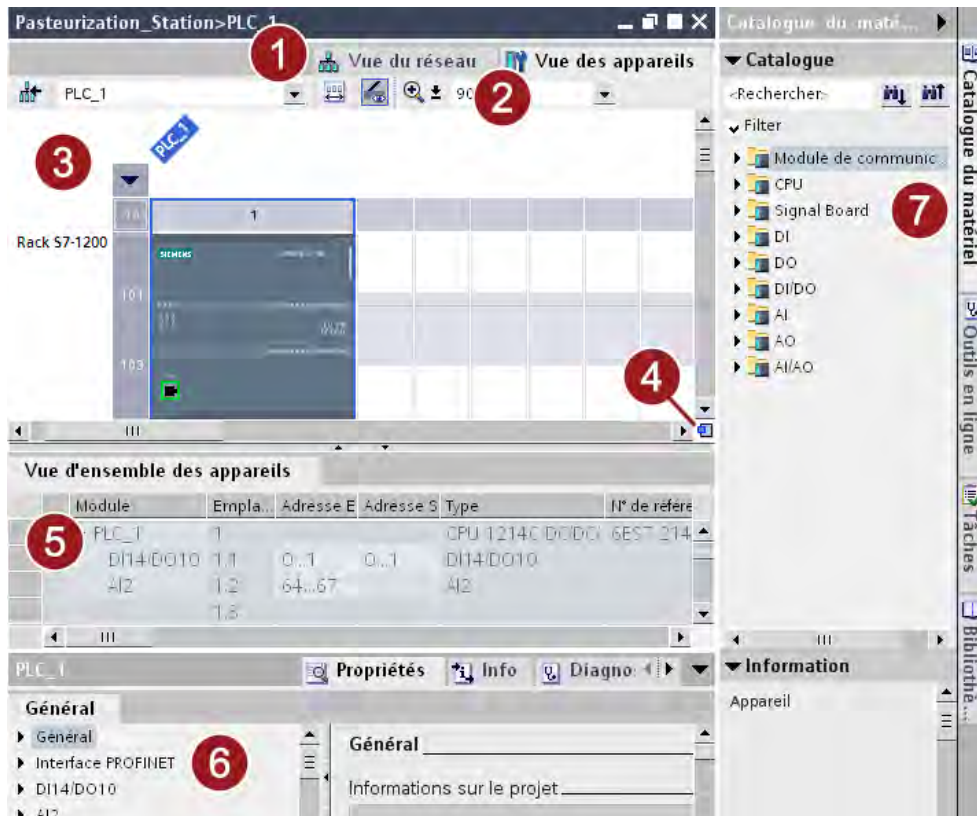
- ③ Zone graphique :
La zone graphique permet d'afficher les appareils mis en réseau, les réseaux, les liaisons et relations. Vous pouvez insérer des appareils du catalogue du matériel (7) dans la zone graphique et les interconnecter par le biais de leurs interfaces.
- ④ Navigation générale :
La navigation générale donne une vue d'ensemble des objets créés dans la zone graphique. En maintenant le bouton de la souris enfoncé, vous pouvez naviguer rapidement dans la navigation générale vers les objets que vous souhaitez afficher dans la zone graphique.
- ⑤ Zone tabellaire :
La zone tabellaire présente une vue d'ensemble des appareils, connexions et liaisons de communication utilisés.
- ⑥ Fenêtre d'inspection :
La fenêtre d'inspection affiche les informations sur les objets actuellement sélectionnés. Dans l'onglet "Propriétés" de la fenêtre d'inspection, vous pouvez éditer les paramètres des objets sélectionnés.
- ⑦ Task Card "Catalogue du matériel" :
Le catalogue du matériel permet d'accéder rapidement aux différents composants matériels. A partir du catalogue du matériel, vous pouvez faire glisser les appareils et modules requis pour votre tâche d'automatisation dans la zone graphique de la vue du réseau.

Vue des appareils

La vue du réseau est la zone de travail de l'éditeur d'appareils et de réseaux dans laquelle vous pouvez exécuter les tâches suivantes :

- Configurer et paramétrer les appareils
- Configurer et paramétrer les modules

La figure suivante montre la structure de la vue des appareils :



- ① Onglets pour commuter entre la vue des appareils et la vue du réseau
- ② Barre d'outils :
La barre d'outils permet de commuter entre les différents appareils et d'afficher ou de masquer certaines informations. La fonction zoom vous permet de modifier la représentation dans la zone graphique.
- ③ Zone graphique :
La zone graphique de la vue des appareils affiche les appareils et les modules associés qui sont affectés les uns aux autres via un ou plusieurs châssis. Dans la zone graphique, vous pouvez faire glisser d'autres objets matériels du catalogue du matériel (7) dans les emplacements des châssis et les configurer.
- ④ Navigation générale :
La navigation générale donne une vue d'ensemble des objets créés dans la zone graphique. En maintenant le bouton de la souris enfoncé, vous pouvez naviguer rapidement dans la navigation générale vers les objets que vous souhaitez afficher dans la zone graphique.

- ⑤ **Zone tabellaire :**
La zone tabellaire présente une vue d'ensemble des modules utilisés avec leurs caractéristiques techniques et organisationnelles.
- ⑥ **Fenêtre d'inspection :**
La fenêtre d'inspection affiche les informations sur les objets actuellement sélectionnés. Dans l'onglet "Propriétés" de la fenêtre d'inspection, vous pouvez éditer les paramètres des objets sélectionnés.
- ⑦ **Task Card "Catalogue du matériel" :**
Le catalogue du matériel permet d'accéder rapidement aux différents composants matériels. A partir du catalogue du matériel, vous pouvez faire glisser les appareils et modules requis pour votre tâche d'automatisation dans la zone graphique de la vue des appareils.

2.3.3 Configuration de l'automate

Introduction

Les étapes suivantes décrivent comment configurer l'interface PROFINET de l'automate inséré.

Condition requise

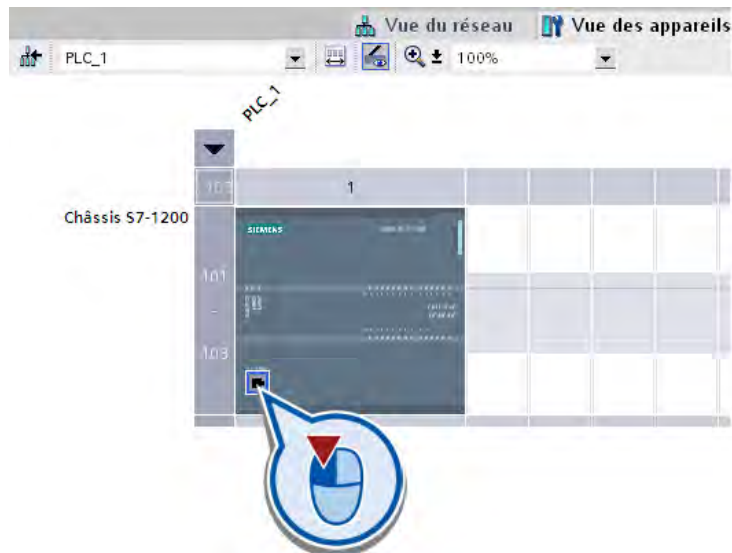
- Le projet a été créé.
- L'automate est ouvert dans la vue des appareils de l'éditeur de matériels et de réseaux.

Marche à suivre

Procédez de la manière suivante pour configurer l'automate :

1. Sélectionnez l'interface PROFINET dans la représentation graphique.

La fenêtre d'inspection affiche les propriétés de l'interface PROFINET.



2. Dans la fenêtre d'inspection, saisissez l'adresse IP de votre automate sous "Adresses Ethernet".



3. Enregistrez le projet en cliquant sur l'icône "Enregistrer le projet" dans la barre d'outils.
4. Fermez l'éditeur de matériels et de réseaux.

Résultat

Vous avez configuré l'automate en paramétrant les propriétés de l'interface PROFINET.

2.4 Création du programme

Introduction

Avec l'automate, le bloc d'organisation "Main [OB1]" est automatiquement créé dans le projet. Dans la suite, vous allez créer le programme utilisateur dans le bloc d'organisation.

2.4.1 Qu'est-ce qu'un bloc d'organisation ?

Programme utilisateur

Un programme utilisateur peut être composé d'un ou de plusieurs blocs. Il faut utiliser au moins un bloc d'organisation. Les blocs contiennent toutes les fonctions requises pour le traitement de vos tâches d'automatisation spécifiques.

Font partie des fonctions du programme :

- Traitement des données du processus, par exemple combinaison de signaux binaires, lecture et évaluation de valeurs analogiques, détermination des signaux binaires de sortie, sortie des valeurs analogiques
- Réaction aux alarmes, par exemple alarme de diagnostic en cas de dépassement de la plage de mesure d'un module d'extension analogique
- Traitement des perturbations dans l'exécution normale du programme

Blocs d'organisation

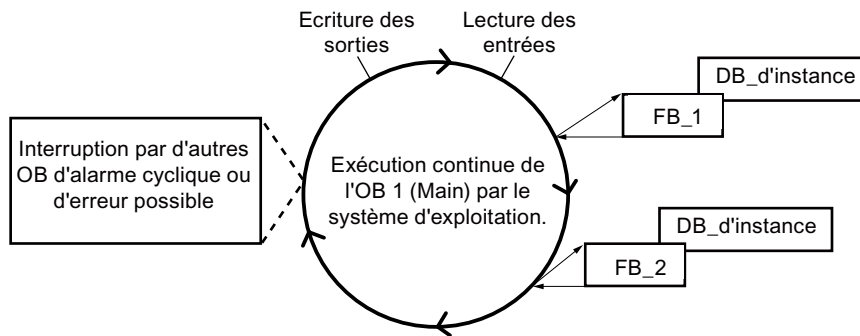
Les blocs d'organisation (OB) constituent l'interface entre le système d'exploitation de l'automate et le programme utilisateur. Ils sont appelés par le système d'exploitation et commandent les opérations suivantes :

- Comportement de démarrage du système d'automatisation
- Traitement cyclique du programme
- Traitement du programme déclenché par alarme
- Traitement des erreurs

Un projet d'automatisation doit comporter au moins un OB de cycle de programme. On écrit dans cet OB de cycle de programme le programme qui détermine le comportement de l'automate. Le système d'exploitation appelle cet OB une fois par cycle et lance ainsi le traitement du programme qu'il contient. A la fin de chaque traitement du programme, le cycle recommence.

Le traitement du programme d'un bloc d'organisation peut être interrompu par l'appel d'autres blocs d'organisation. Pour les tâches d'automatisation complexes, on structure le programme et on le subdivise en plusieurs blocs qui sont appelés dans l'OB de cycle de programme et qui sont traités les uns après les autres.

La figure suivante montre l'exécution d'un OB de cycle de programme :



L'insertion d'un automate dans le projet crée automatiquement un OB de cycle de programme avec le nom "Main [OB1]". C'est dans ce bloc d'organisation que vous créez le programme du projet Mise en route.

2.4.2 Ouverture d'un bloc d'organisation

Introduction

Les étapes suivantes décrivent comment ouvrir le bloc d'organisation dans l'éditeur de programmes. L'éditeur de programmes est l'environnement de développement intégré pour la création du programme.

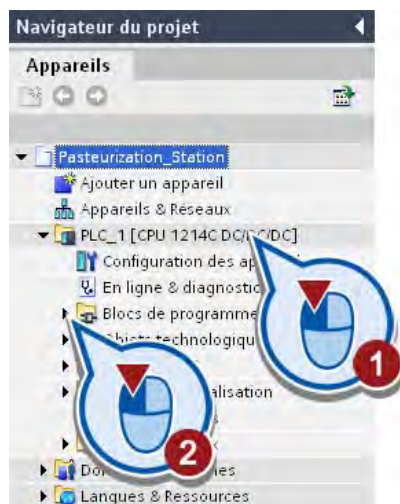
Condition requise

Un automate existe dans le projet.

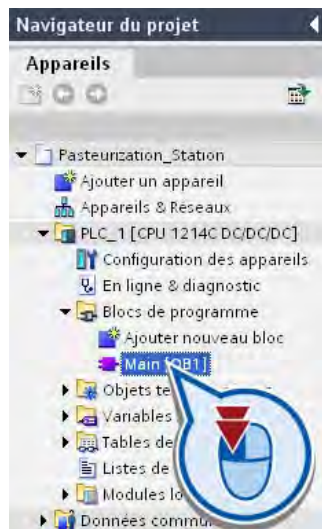
Marche à suivre

Procédez de la manière suivante pour ouvrir le bloc d'organisation "Main [OB1]" :

1. Ouvrez le dossier "Blocs de programme" dans le navigateur du projet.

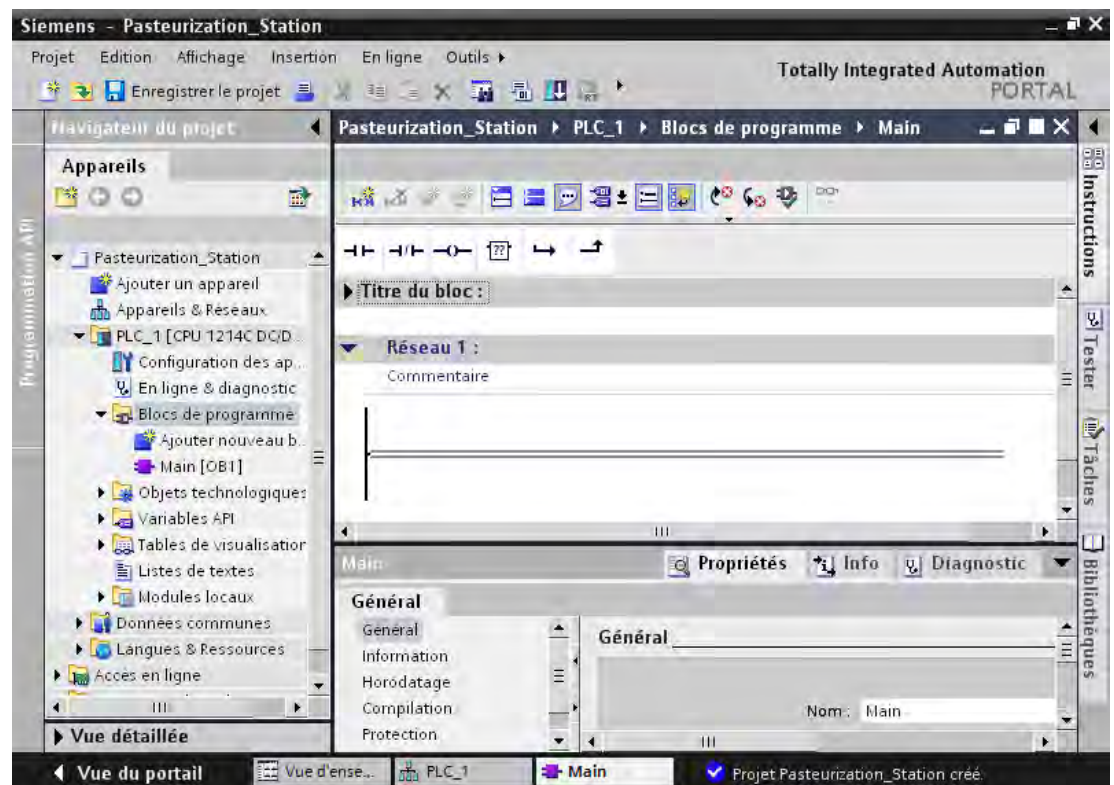


2. Ouvrez le bloc d'organisation "Main [OB1]".



Résultat

Vous avez ouvert le bloc d'organisation "Main [OB1]" dans l'éditeur de programmes et pouvez y créer votre programme.



Remarque

Adaptation de la zone de travail

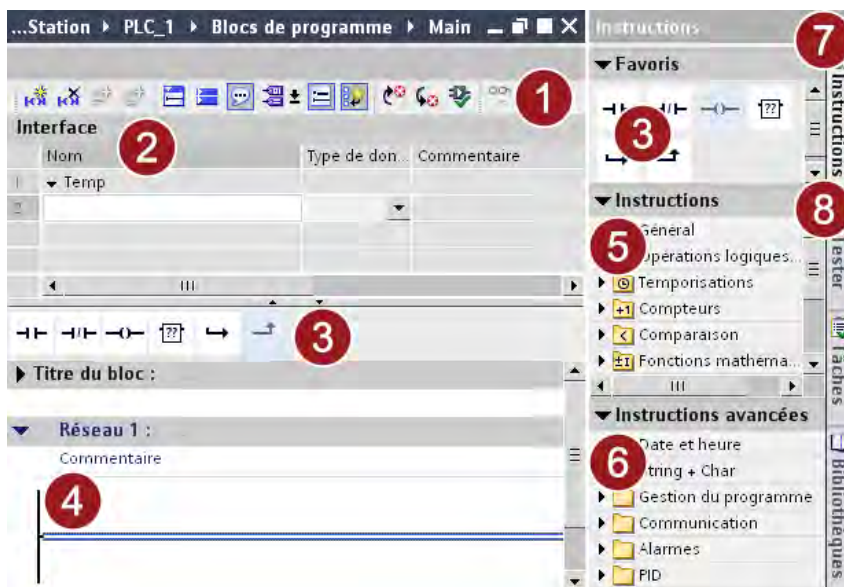
Vous pouvez à volonté déplacer, désancrer et fractionner horizontalement ou verticalement les fenêtres de la zone de travail. Vous trouverez plus d'informations à ce sujet dans le système d'information du portail TIA.

2.4.3 Présentation de l'éditeur de programmes

Fonction de l'éditeur de programmes

L'éditeur de programmes vous permet de créer les blocs du programme. Il est constitué de plusieurs zones qui, selon leur fonction, vous assistent dans l'exécution des différentes tâches de programmation.

La figure suivante montre la structure de l'éditeur de programmes :



- ① **Barre d'outils :**
La barre d'outils permet d'accéder rapidement aux principales fonctions de l'éditeur de programmes, par exemple :
 - Ajouter, supprimer, agrandir et réduire des réseaux
 - Afficher et masquer les opérandes absolus
 - Afficher et masquer les commentaires de réseau
 - Afficher et masquer les favoris
 - Afficher et masquer l'état du programme
- ② **Interface de bloc :**
L'interface de bloc sert à créer et à gérer les variables locales.

- ③ Palette "Favoris" dans la Task Card "Instructions" et Favoris dans l'éditeur de programmes :
Les favoris permettent un accès rapide aux instructions souvent utilisées. Vous pouvez compléter la palette "Favoris" de manière individualisée avec des instructions supplémentaires.
- ④ Fenêtre d'instructions :
La fenêtre d'instructions est la zone de travail de l'éditeur de programmes. Vous pouvez y exécuter les tâches suivantes :
 - Créer et gérer des réseaux (Page 37)
 - Saisir les titres et commentaires du bloc et des réseaux
 - Insérer des instructions et leur affecter des variables
- ⑤ Palette "Instructions" dans la Task Card "Instructions"
- ⑥ Palette "Instructions avancées" dans la Task Card "Instructions"
- ⑦ Task Card "Instructions" :
La Task Card "Instructions" contient les instructions avec lesquelles vous créez le contenu du programme.
- ⑧ Task Card "Test"

2.4.4 Qu'est-ce qu'un réseau ?

Introduction

Le programme d'un bloc d'organisation est subdivisé en réseaux. Les réseaux servent à structurer les programmes. Chaque bloc peut contenir jusqu'à 999 réseaux.

Un réseau est automatiquement créé dans le bloc d'organisation "Main [OB1]".

Réseaux en langage de programmation CONT

Le programme d'un bloc d'organisation peut être écrit dans différents langages de programmation. Pour le projet-exemple, le bloc d'organisation "Main [OB1]" est édité dans le langage de programmation graphique CONT.

La représentation de ce langage de programmation imite les schémas de circuit. Ainsi, chaque programme CONT d'un bloc est subdivisé en réseaux qui sont respectivement constitués d'une barre conductrice et d'au moins un circuit.

Il est possible d'étendre un réseau en ajoutant d'autres circuits. Vous pouvez également, à l'aide de branches, programmer des montages en parallèle dans chaque circuit. Le traitement des circuits et des réseaux s'effectue de haut en bas et de gauche à droite.

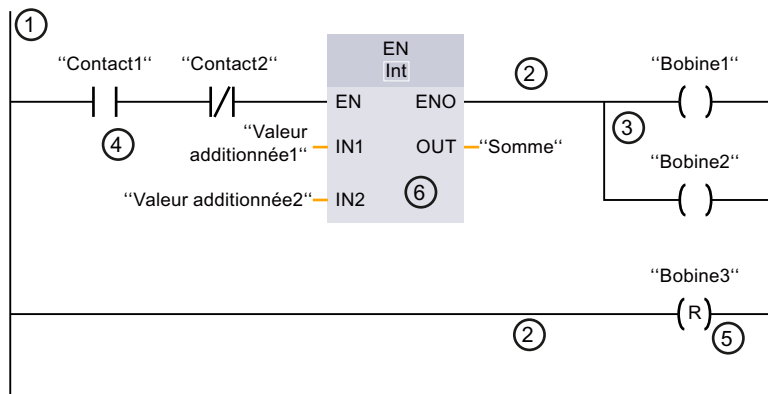
Instructions CONT

Vous écrivez le contenu effectif du programme à l'aide des instructions CONT disponibles dans la Task Card "Instructions" dans l'interface utilisateur. On distingue les trois types suivants d'instructions CONT :

- **Contacts** : Les contacts permettent d'établir ou d'interrompre une liaison de conduction de courant entre deux éléments. Dans ce contexte, les éléments sont soit des éléments de programme CONT, soit les bords de la barre conductrice. La conduction du courant s'effectue de gauche à droite. Les contacts vous permettent d'interroger l'état logique ou la valeur d'un opérande et de contrôler le flux de courant en fonction du résultat.
- **Bobines** : Les bobines permettent de commander des opérandes binaires. Elles peuvent mettre à "1" ou à "0" un opérande binaire en fonction de l'état du résultat logique.
- **Boîtes** : Les boîtes sont des éléments CONT avec des fonctions complexes. La boîte vide constitue une exception. Il s'agit d'une boîte générique à laquelle l'opération souhaitée sera substituée.

Dans la Task Card "Instructions", vous trouverez différentes variations des contacts, bobines et boîtes qui sont classées dans différents dossiers selon leur fonction. La plupart des instructions CONT doivent recevoir des variables.

La figure suivante montre un exemple de réseau CONT programmé :



- ① Barre conductrice
- ② Circuit
- ③ Branche
- ④ Contact
- ⑤ Bobine
- ⑥ Boîte

2.4.5 Insertion d'instructions CONT

Introduction

Les étapes suivantes décrivent comment créer le programme pour la mise en marche et l'arrêt de l'installation fournie en exemple dans le premier réseau du bloc d'organisation "Main [OB1]". Pour ce faire, vous insérez une branche et les instructions suivantes :

- Contact à fermeture
- Bobine de relais, sortie
- Négation bobine de relais

Vous utilisez en outre une branche pour programmer un montage parallèle dans le langage de programmation CONT.

Contact à fermeture

La figure suivante montre le symbole du contact à fermeture dans le programme :

<opérande>

---|---

L'activation d'un contact à fermeture dépend de l'état logique de la variable correspondante (<opérande>).

- Lorsque la variable fournit l'état logique "1", le contact à fermeture se ferme. Le courant circule à travers le contact à fermeture, de la barre conductrice gauche à la barre conductrice droite, et l'état logique à la sortie de l'instruction est mis à "1".
- Lorsque la variable fournit l'état logique "0", le contact à fermeture n'est pas activé. Le flux de courant vers la barre conductrice droite est interrompu et l'état logique à la sortie de l'instruction est mis à "0".

Dans la suite, vous utiliserez également un contact à ouverture (Page 101) dont le principe de fonctionnement est identique à celui du contact à fermeture si ce n'est qu'il réagit de manière inversée à l'état logique de la variable.

Bobine de relais, sortie

La figure suivante montre le symbole de l'instruction "Bobine de relais, sortie" dans le programme :

<opérande>

---()---

L'instruction "Bobine de relais, sortie" vous permet de mettre à "1" le bit d'une variable spécifiée (<opérande>). Lorsque l'état logique à l'entrée de la bobine est "1", le bit de la variable est mis à "1". Lorsque l'état logique à l'entrée de la bobine est "0", le bit de la variable fournit également l'état logique "0".

Négation bobine de relais

La figure suivante montre le symbole de l'instruction "Négation bobine de relais" dans le programme :

<opérande>
---(/)---

L'instruction "Négation bobine de relais" inverse l'état logique et affecte le bit correspondant à la variable spécifiée (<opérande>). Lorsque l'état logique à l'entrée de la bobine est "1", le bit est mis à "0". Lorsque l'état logique à l'entrée de la bobine est "0", le bit de la variable est mis à "1".

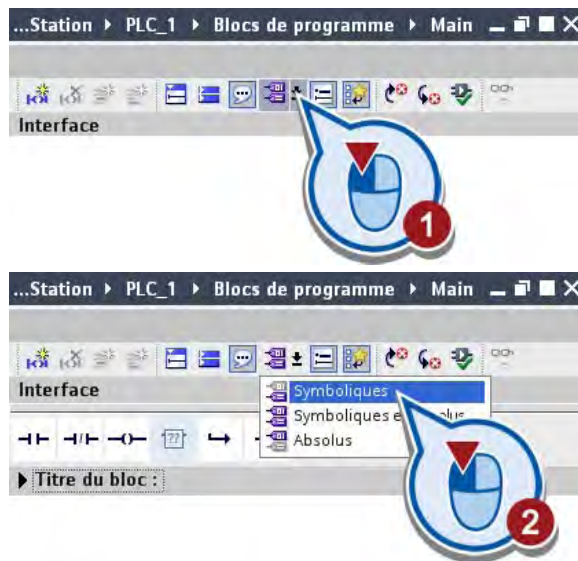
Condition requise

- Le projet a été créé.
- L'automate est configuré.
- Le bloc d'organisation "Main [OB1]" est ouvert.

Marche à suivre

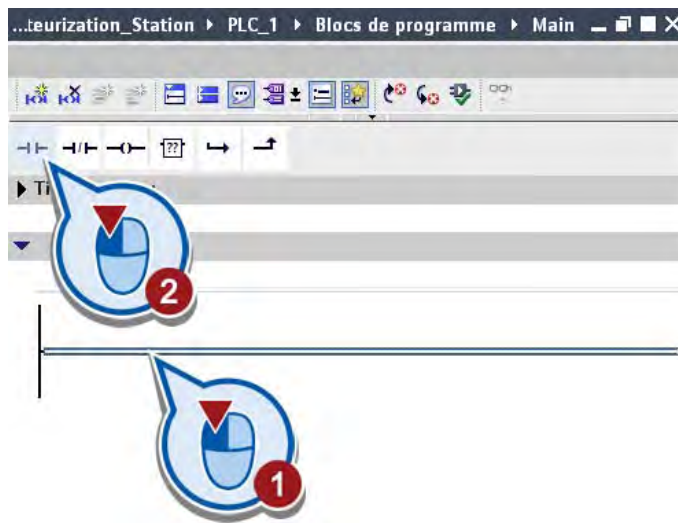
Procédez de la manière suivante pour créer le programme de mise en marche et d'arrêt de l'installation fournie en exemple :

1. Activez la représentation symbolique des variables.

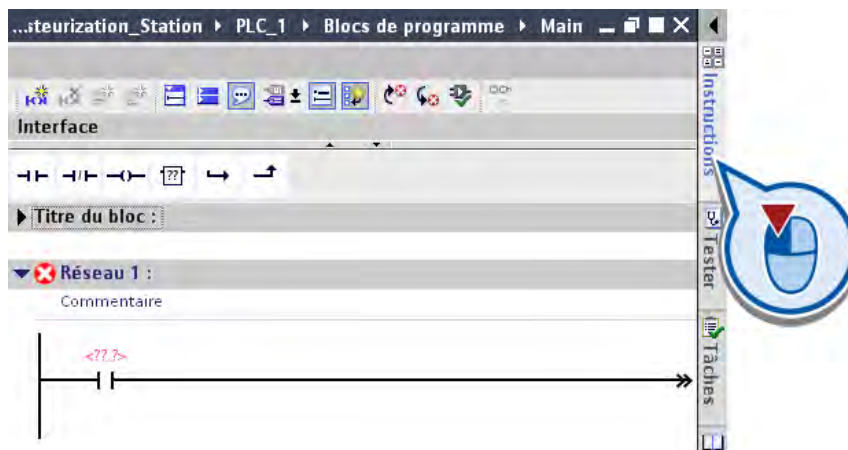


Lorsque la représentation symbolique des variables est activée, les adresses de ces dernières ne sont pas affichées dans le réseau.

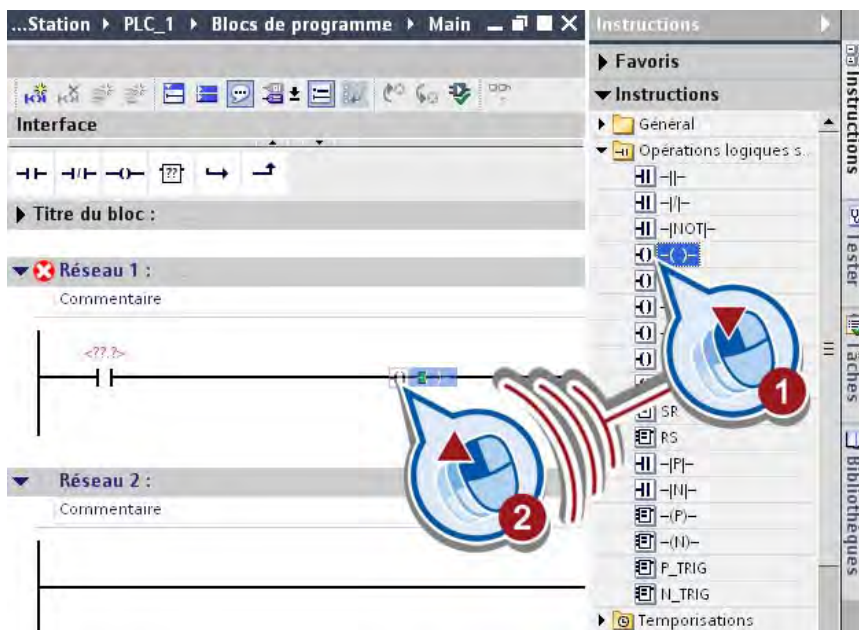
2. Insérez un contact à fermeture dans le premier réseau du bloc.



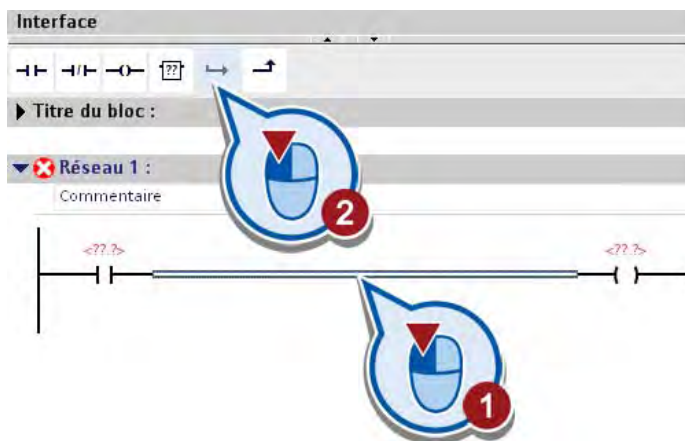
3. Ouvrez la Task Card "Instructions".



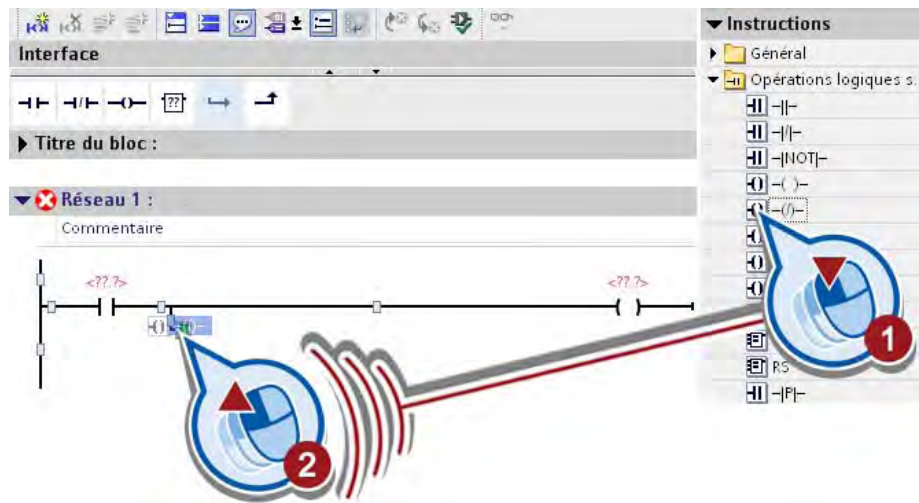
- 4. Insérez l'instruction "Bobine de relais, sortie" à la fin du circuit principal.



- 5. Insérez une branche.



6. Insérez l'instruction "Négation bobine de relais" à la fin de la branche.



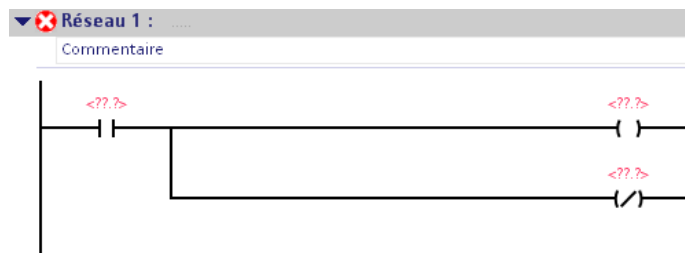
Remarque

Changement d'instruction

Si vous avez sélectionné le mauvais type d'instruction, vous pouvez le modifier directement dans le réseau par le biais du symbole de l'instruction. Pour ouvrir la liste déroulante de sélection d'un type d'instruction, déplacez le pointeur de la souris au-dessus du triangle jaune dans le coin supérieur droit de l'instruction.

Résultat

Vous avez créé la première partie du programme pour la mise en marche et l'arrêt de l'installation fournie en exemple grâce à l'insertion des instructions CONT dans le premier réseau du bloc d'organisation "Main [OB1]".



Dans le paragraphe suivant, vous allez connecter les instructions CONT aux variables API et achever ainsi le programme.

2.4.6 Qu'est-ce qu'une variable ?

Introduction

Dans l'étape précédente, vous avez inséré trois instructions CONT dans le premier réseau du bloc d'organisation "Main [OB1]". Dans le prochain paragraphe, vous allez connecter ces instructions à des variables API. Les variables API fournissent des valeurs aux instructions qui sont alors exécutées en fonction de ces valeurs.

Dans la suite, vous allez en apprendre plus sur la fonction générale des variables dans les programmes utilisateur.

Variables

Une variable est une grandeur utilisée dans le programme et pouvant prendre différentes valeurs. Les variables sont classées dans les catégories suivantes en fonction de leur domaine de validité :

- Variables locales : Les variables locales sont valables uniquement dans le bloc dans lequel elles ont été définies.
- Variables API : Les variables API sont valables dans tout l'automate.

La plupart des instructions dans le programme utilisent des variables. Si vous affectez une variable à une instruction, l'instruction est exécutée avec les valeurs de la variable indiquée.

Les variables sont gérées de manière centrale dans le portail TIA. Vous pouvez créer une variable API indifféremment dans l'éditeur de programmes ou dans la table des variables. Si la variable est utilisée à plusieurs endroits dans le programme ou dans la vue IHM, les modifications apportées à la variable sont immédiatement appliquées dans tous les éditeurs.

Utilité des variables

Les variables présentent l'avantage de permettre la modification centralisée d'un adressage utilisé dans le programme. Sans l'adressage symbolique des variables, il faudrait, en cas de modification de l'affectation des entrées et sorties de l'automate, modifier un adressage utilisé plusieurs fois dans le programme à chaque fois à plusieurs endroits dans le programme.

Variables API

Une variable API est composée des éléments suivants :

- Nom (par exemple, MARCHE_BANDE) : Le nom d'une variable vaut pour un automate et doit être univoque à l'intérieur de l'ensemble du programme et de cet automate.
- Type de données (par exemple, BOOL) : Le type de données détermine la représentation de la valeur et la plage de valeurs autorisée. En choisissant le type de données BOOL par exemple, vous décidez qu'une variable ne peut prendre que les valeurs binaires "0" et "1".
- Adresse (par exemple, M 3.1) : L'adresse d'une variable est absolue et définit la zone de mémoire dans laquelle la variable lit ou écrit une valeur. Les entrées, les sorties et les mémentos sont des exemples de zones de mémoire possibles.

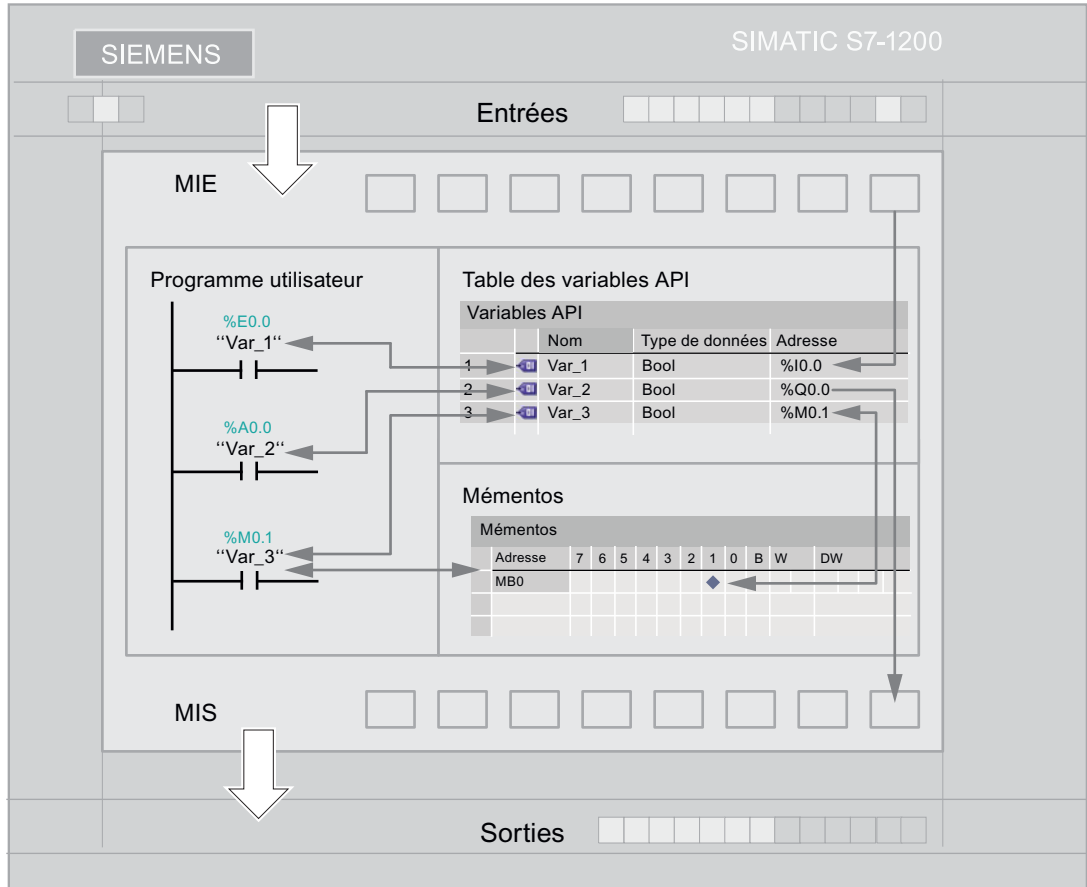
Vous pouvez accéder à la mémoire image par le biais des zones de mémoire I (entrées) et Q (sorties).

La mémoire image contient une image des entrées et sorties d'un module d'automatisation :

- Les états logiques des entrées dans la mémoire image sont actualisés à chaque cycle du programme (voir Qu'est-ce qu'un bloc d'organisation ? (Page 33)) par le système d'exploitation de l'automate.
- A chaque cycle du programme, les états logiques des sorties dans la mémoire image sont écrits par le système d'exploitation de l'automate dans les sorties du module de sorties correspondant.

Une machine ou un processus est en général commandé par les états logiques des sorties.

La figure suivante montre comment la table des variables, les variables dans le programme utilisateur ainsi que les mémentos et les entrées et sorties de l'automate sont liés les uns aux autres.




La zone des mémentos sert avant tout à la sauvegarde de résultats intermédiaires. Les valeurs des variables adressées dans la zone des mémentos sont stockées dans la mémoire système sans être transmises aux modules. L'espace occupé en mémoire par une variable dépend du type de données de la variable. Une variable de type BOOL occupe, par exemple, un bit seulement en mémoire. Une variable de type INT occupe 16 bits en mémoire. Les variables ne doivent pas se chevaucher dans une zone de mémoire. L'adresse d'une variable doit être univoque.

Table des variables API

La table des variables API contient la définition des variables et des constantes valables pour un automate. Une table des variables API est automatiquement créée pour chaque automate figurant dans le projet.

Le tableau suivant donne la signification des différentes colonnes de l'onglet "Variables".

Colonne	Explication
	Symbole sur lequel vous pouvez cliquer pour faire glisser une variable dans un réseau et l'y utiliser comme opérande.
Nom	Nom univoque à l'intérieur de l'automate, que vous définissez pour la variable.
Type de données	Type de données que vous définissez pour la variable.
Adresse	Adresse de la variable
Rémanence	Les valeurs des variables rémanentes sont conservées même après coupure de la tension d'alimentation.
Valeur de visualisation	Valeur de données en cours dans l'automate. Cette colonne n'apparaît que s'il existe une liaison en ligne et que vous sélectionnez le bouton "Visualiser".
Commentaire	Commentaire pour la documentation de la variable

2.4.7 Définition et connexion de variables API

Introduction

Dans le portail TIA, vous avez la possibilité de créer des variables dans les réseaux directement lors de la création du programme utilisateur. Les étapes suivantes décrivent comment créer des variables API et connecter les instructions CONT insérées à des variables API. Les instructions CONT sont exécutées en fonction des valeurs des variables, ce qui commande la mise en marche et l'arrêt de l'installation.

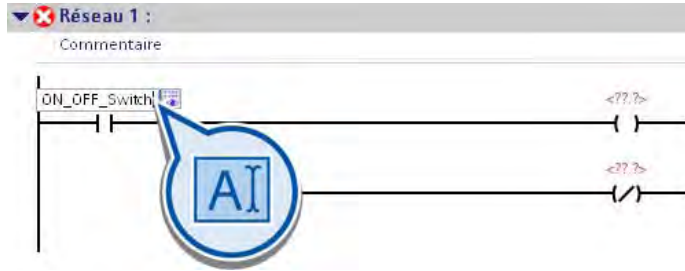
Condition requise

- Le projet a été créé.
- L'automate est configuré.
- Le bloc "Main [OB1]" est ouvert.
- Les instructions CONT "Contact à fermeture", "Bobine de relais, sortie" et "Négation bobine de relais" ont été insérées dans le premier réseau du bloc d'organisation "Main [OB1]".

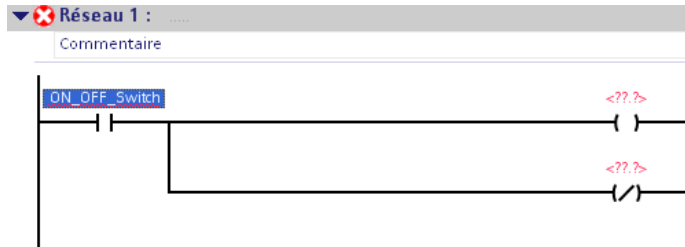
Marche à suivre

Procédez de la manière suivante pour connecter les instructions CONT à des variables API :

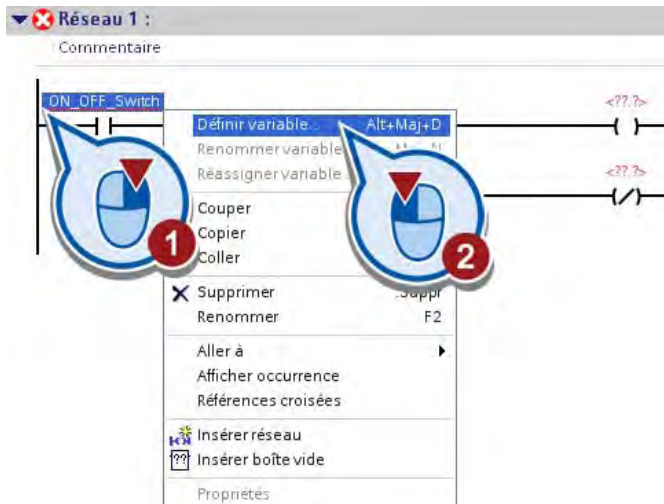
1. Ouvrez le premier réseau du bloc d'organisation "Main [OB1]".
2. Saisissez le nom "ON_OFF_Switch" dans l'emplacement réservé à l'opérande du contact à fermeture.



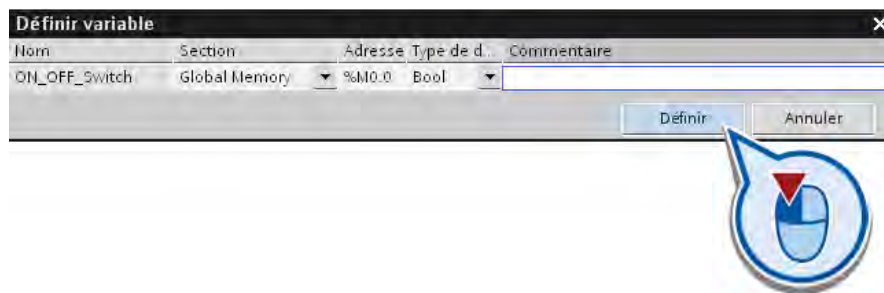
3. Confirmez la saisie avec la touche Entrée.



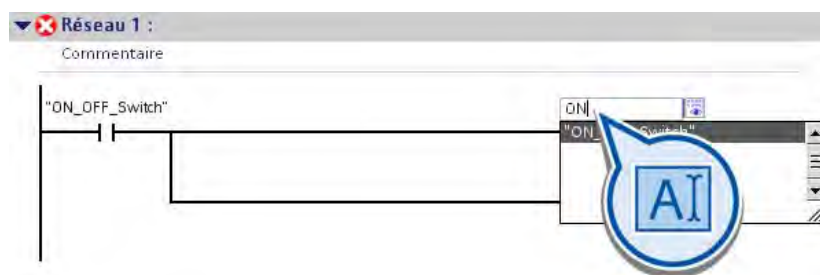
4. Ouvrez la boîte de dialogue "Définir variable".



5. Définissez la variable "ON_OFF_Switch".

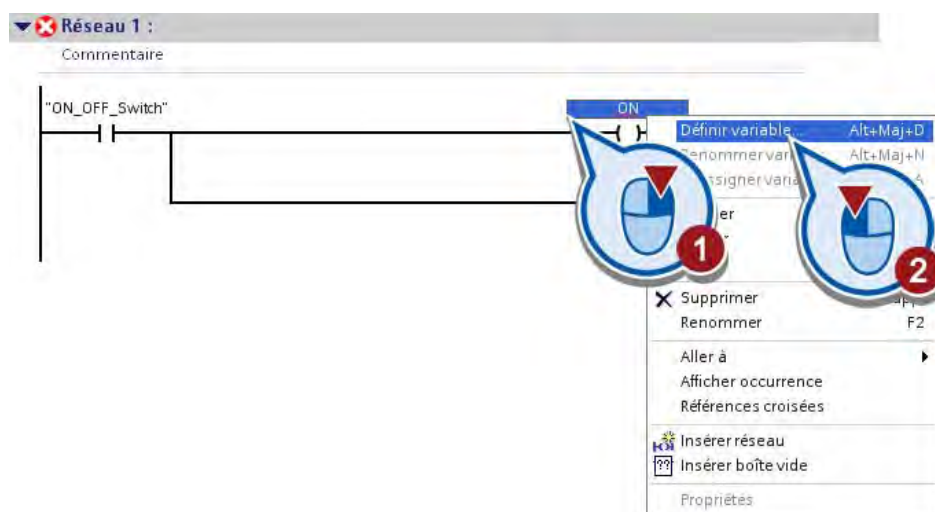


6. Saisissez le nom "ON" dans l'emplacement réservé à l'opérande au-dessus de l'instruction "Bobine de relais, sortie".



7. Confirmez la saisie avec la touche Entrée.

8. Ouvrez la boîte de dialogue "Définir variable".



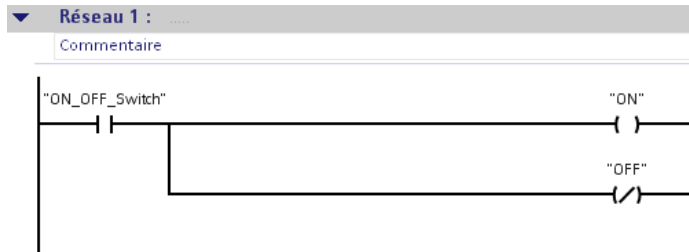
9. Définissez la variable "ON".



10. Saisissez le nom "OFF" dans l'emplacement réservé à l'opérande au-dessus de l'instruction "Négation bobine de relais" et définissez la variable correspondante.
11. Enregistrez le projet à l'aide du bouton "Enregistrer le projet" de la barre d'outils.

Résultat

Vous avez programmé un interrupteur poussoir pour la mise en marche et l'arrêt de l'installation fournie en exemple.



Actionner l'interrupteur poussoir a l'effet suivant :

- Lorsqu'on appuie une fois sur l'interrupteur poussoir, la variable "ON_OFF_Switch" est mise à l'état logique "1".
 - Le courant est conduit et la variable ""ON"" est mise à l'état logique "1" par l'instruction "Bobine de relais, sortie".
 - L'installation se met en marche.
 - La variable "OFF" fournit l'état logique "0" et reste donc sans effet.
- Lorsqu'on appuie une deuxième fois sur l'interrupteur poussoir, la variable "ON_OFF_Switch" est mise à l'état logique "0".
 - Le flux de courant est interrompu et la variable ""OFF"" est mise à l'état logique "1" par l'instruction "Négation bobine de relais".
 - L'installation s'arrête.
 - La variable "ON" fournit l'état logique "0" et reste donc sans effet.

2.5 Test du programme

2.5.1 Chargement du programme dans le système cible

Introduction

Les étapes suivantes décrivent comment charger le programme dans l'automate. Pendant le chargement, une liaison en ligne est établie entre votre console de programmation (PG) ou PC de programmation et l'automate. Lors du chargement, le programme stocké sur le disque dur de votre PG ou PC est écrit dans la mémoire de l'automate. Si nécessaire, les blocs contenus dans le programme sont compilés pendant le chargement afin qu'ils puissent être traités par l'automate. Une fois le programme compilé et chargé, il peut être traité par l'automate.

Remarque

Comparaison en ligne/hors ligne

Les modifications apportées au programme dans la PG ou le PC après le chargement ne sont pas enregistrées par l'automate. Vous avez la possibilité dans le portail TIA d'effectuer une comparaison en ligne/hors ligne des données de votre projet et d'afficher les différences. En mode en ligne, les symboles dans le navigateur du projet vous permettent de reconnaître si les composants du programme "hors ligne" sur votre PG ou PC sont identiques aux éléments du programme "en ligne" dans l'automate. Pour actualiser l'état du programme dans l'automate, vous devez recharger le programme.

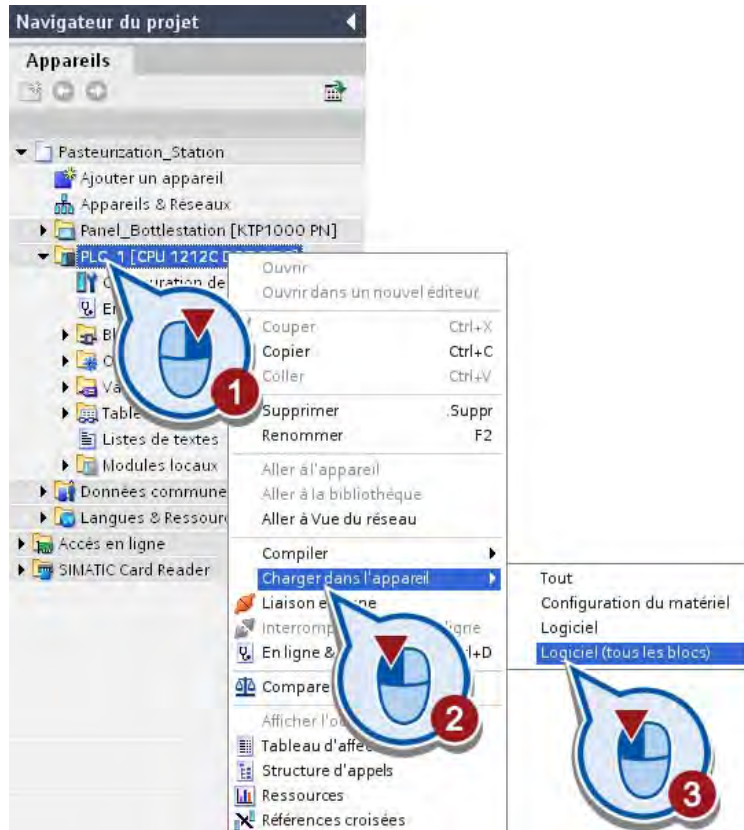
Condition requise

Une liaison est établie entre la PG ou le PC et l'automate (voir "Pour plus d'informations ...").

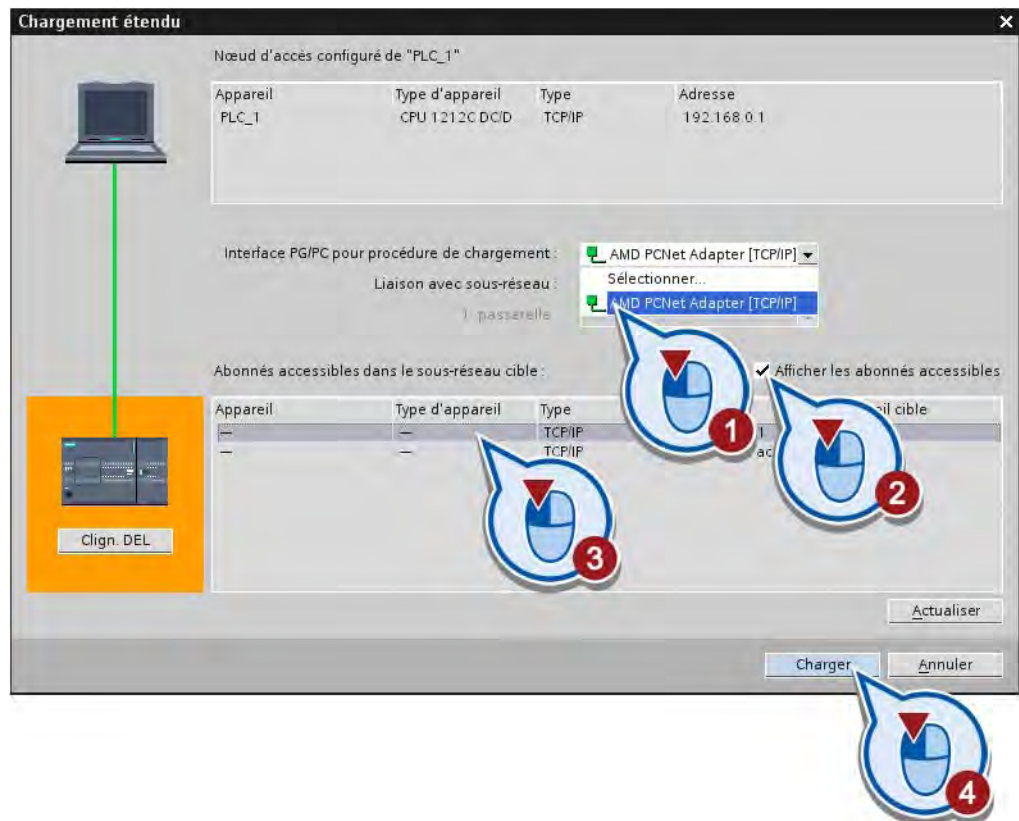
Marche à suivre

Procédez de la manière suivante pour charger le programme dans l'automate :

1. Démarrez la procédure de chargement.



2. Sélectionnez l'interface par l'intermédiaire de laquelle vous désirez connecter l'appareil. Cochez la case d'option "Afficher les abonnés accessibles". Tous les appareils accessibles via l'interface sélectionnée s'affichent alors sous "Abonnés accessibles dans le sous-réseau cible". Sélectionnez l'automate et chargez le programme utilisateur.



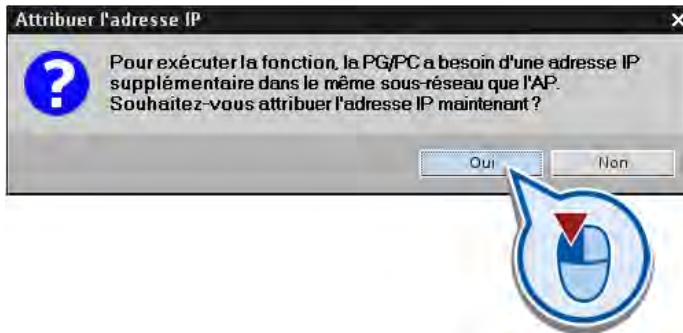
Remarque

Test de la liaison en ligne

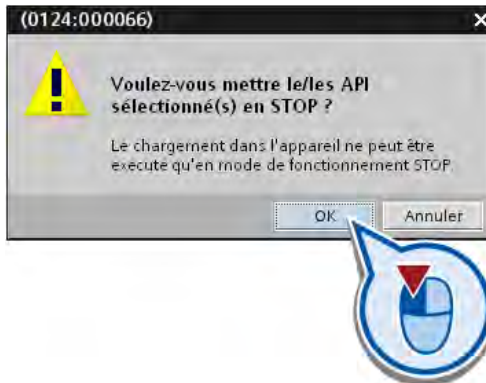
Si aucun appareil ne s'affiche dans la liste "Abonnés accessibles dans le sous-réseau cible", cela peut être dû à l'une des raisons suivantes :

- Il existe un problème dans la connexion matérielle de l'automate.
- Il existe un problème dans l'interface Ethernet de votre PG/PC.
- L'adresse IP de l'automate ne se trouve pas dans le même sous-réseau que l'adresse IP de votre PG/PC.

- 3. Confirmez l'affectation de l'adresse IP correcte si cette adresse n'a pas encore été affectée.

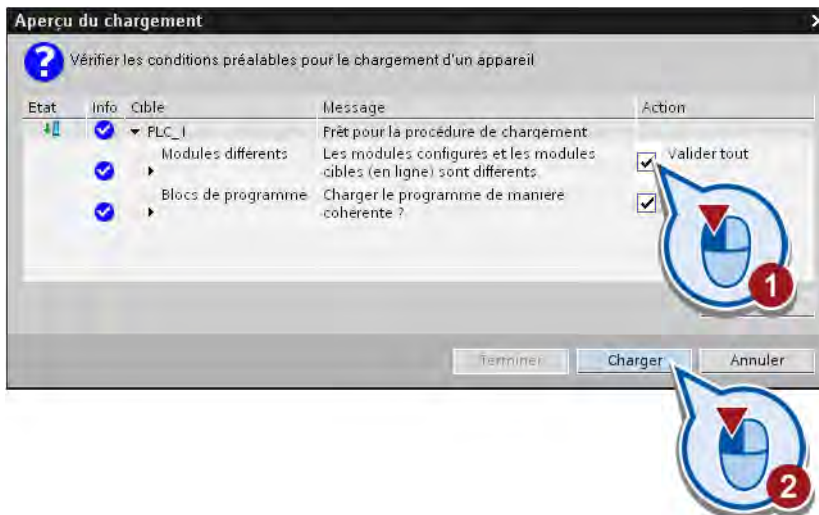


- 4. Si l'automate est à l'état de fonctionnement MARCHE, faites-le passer à l'état ARRÊT dans la fenêtre suivante.



La boîte de dialogue "Aperçu du chargement" s'ouvre.

- 5. S'il existe des différences entre les modules configurés et les modules cibles, cochez la case d'option correspondante pour appliquer les différences. Cliquez sur le bouton "Charger". Veillez à ce que la case d'option "Continuer" soit cochée.

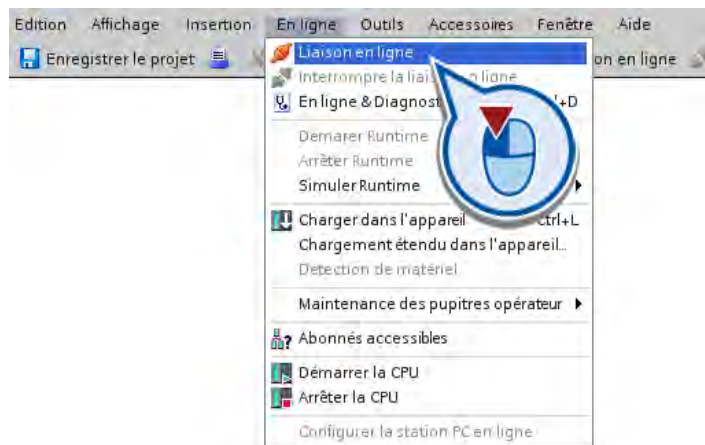


Le programme est chargé et les actions ayant lieu pendant la procédure de chargement sont affichées. Une fois le chargement terminé, la boîte de dialogue "Résultats du chargement" s'ouvre.

6. Démarrez le module.



7. Etablissez la liaison en ligne.



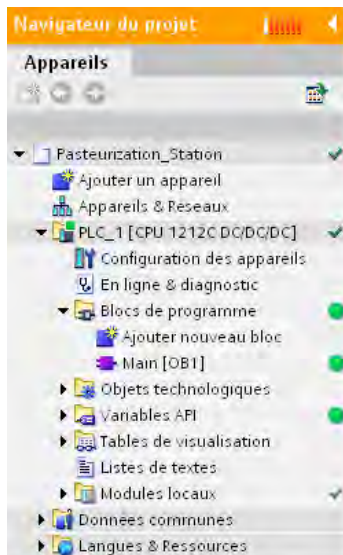
Remarque

Correction des erreurs de compilation

Si des erreurs ont été détectées dans le programme lors de la compilation, vous pouvez afficher les erreurs et la manière d'y remédier dans la fenêtre d'inspection sous "Info > Compiler".

Résultat

Vous avez chargé le programme dans l'automate. L'état des composants du programme est affiché dans le navigateur du projet. Les symboles verts montrent que les éléments de programme "hors ligne" et "en ligne" sont identiques. La signification des autres icônes d'état figure dans l'info-bulle correspondante.



Remarque

Chargement des blocs

Le chargement des blocs par le biais du navigateur du projet n'est pas l'unique façon de transférer les blocs dans l'automate. Vous pouvez également faire glisser les blocs dans la liste des abonnés accessibles dans le navigateur du projet. Si vous utilisez le bouton "Charger dans l'appareil" de la barre d'outils, les blocs qui sont ouverts dans l'éditeur ou sélectionnés dans le navigateur du projet sont chargés.

Pour plus d'informations...

Vous trouverez des informations sur la création de la liaison Ethernet et sur la configuration dans les documentations suivantes :

- Manuel système "Automate programmable SIMATIC S7-1200"
- Système d'information du portail TIA, chapitre "Paramétrage de l'interface Ethernet"
- Système d'information du portail TIA, chapitre "Configuration de réseau Industrial Ethernet"

Vous trouverez des informations sur la configuration de l'interface Ethernet sur la PG/le PC dans la documentation du système d'exploitation que vous utilisez ou dans la documentation de la carte réseau.

2.5.2 Test du programme avec la visualisation de l'état du programme

Introduction

Les étapes suivantes décrivent comment tester le programme créé en visualisant l'état du programme. L'affichage de l'état du programme permet d'en visualiser l'exécution. Vous pouvez activer la visualisation d'état à partir d'un endroit donné du programme et obtenir ainsi une vue d'ensemble des valeurs des différentes variables et des résultats logiques. Vous pouvez de cette manière vérifier si les composants du système d'automatisation sont pilotés correctement.

L'affichage de l'état du programme est actualisé de façon cyclique. Il commence au réseau sélectionné.

Dans la visualisation d'état du programme, vous pouvez affecter des valeurs aux variables en exécutant l'une des actions suivantes au moyen de la commande "Forcer" du menu contextuel :

- Forcer à 1 : Cette commande vous permet de donner à des variables de type BOOL l'état logique "1".
- Forcer à 0 : Cette commande vous permet de donner à des variables de type BOOL l'état logique "0".
- Valeur de forçage : Pour les variables qui ne sont pas de type BOOL, vous pouvez saisir une valeur de forçage.

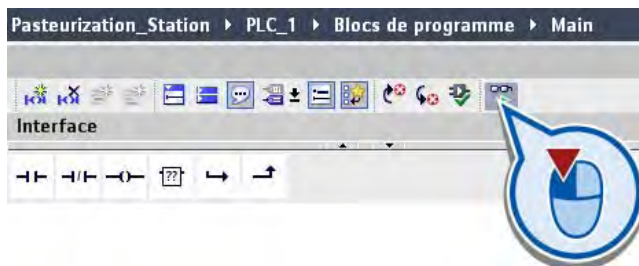
Condition requise

- L'automate est configuré.
- Aucune tension n'est appliquée aux entrées et sorties de l'automate, car les valeurs forcées sont écrasées par le module en mode en ligne.
- Le bloc d'organisation "Main [OB1]" est ouvert dans l'éditeur de programmes.

Marche à suivre

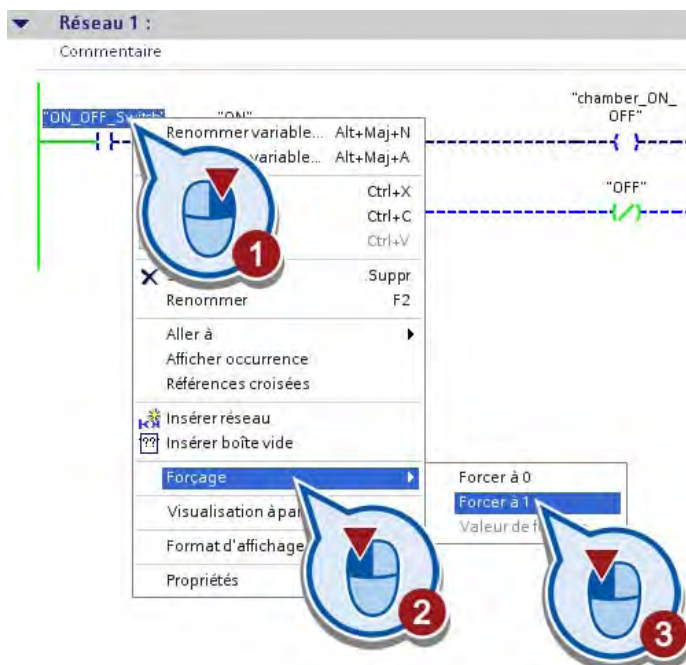
Procédez de la manière suivante pour tester le programme créé avec la visualisation d'état du programme :

1. Activez la visualisation de l'état du programme.

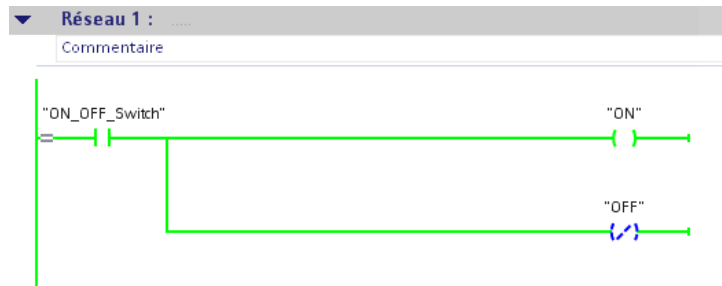


L'état du programme s'affiche.

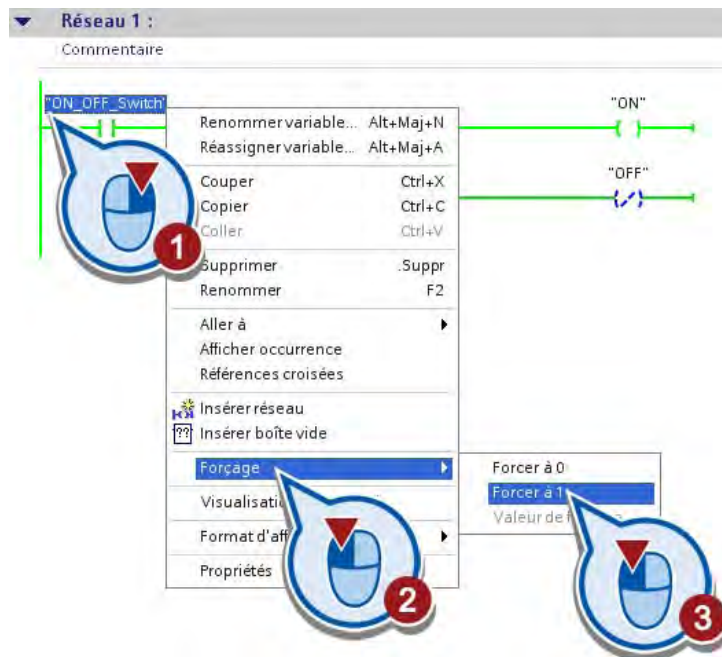
2. Forcez la variable "ON_OFF_Switch" à "1".



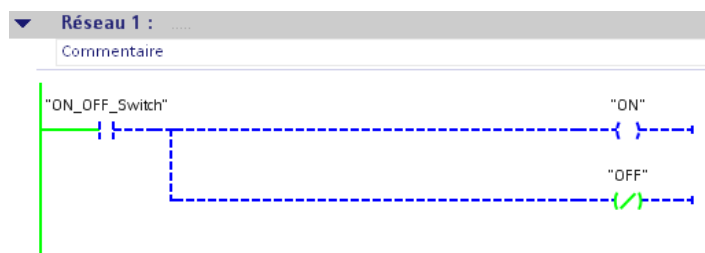
La variable "ON_OFF_Switch" est mise à "1" et le contact à fermeture se ferme. Le courant traverse le contact à fermeture et atteint les bobines à la fin du réseau. Vous reconnaissez le flux de courant à la couleur verte du circuit. La variable "ON" est mise à 1, ce qui met en marche l'installation. La variable "OFF" a l'état logique "0" et reste donc sans effet. C'est ce que signifient les pointillés bleus.



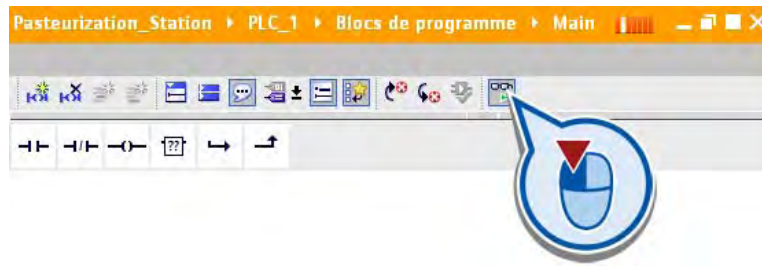
3. Forcez la variable "ON_OFF_Switch" à "0".



La variable "ON_OFF_Switch" est mise à l'état logique "0". Le flux de courant jusqu'aux bobines à la fin du circuit s'interrompt. La variable "OFF" est mise à "1". La variable "ON" est mise à "0".



4. Désactivez la visualisation de l'état du programme.



5. Coupez la liaison en ligne.

Résultat

Vous avez testé le programme et vérifié qu'il s'exécute correctement.

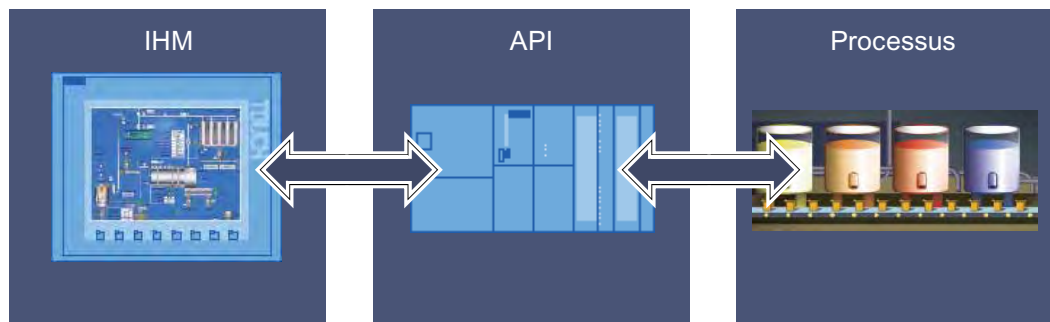
Dans le paragraphe suivant, vous allez programmer une vue IHM pour visualiser l'exécution du programme.

2.6 Création d'une vue IHM

2.6.1 Visualisation dans le portail TIA

IHM - Interface Homme-Machine

Un système IHM constitue l'interface entre l'utilisateur et le processus. Le processus est pour l'essentiel piloté par l'automate. L'utilisateur peut visualiser le processus ou intervenir dans le processus en cours par le biais d'un pupitre opérateur.



Les possibilités suivantes vous sont offertes pour le contrôle-commande de machines et d'installations :

- Représenter les processus
- Commander les processus
- Emettre des alarmes
- Gérer les paramètres du processus et les recettes

2.6.2 Création d'un pupitre opérateur avec vue IHM

Introduction

Les étapes suivantes décrivent comment créer un nouveau pupitre opérateur et un modèle pour la vue IHM.

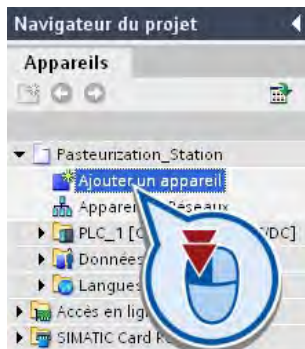
Condition requise

- Le programme a été créé.
- La vue du projet est ouverte.

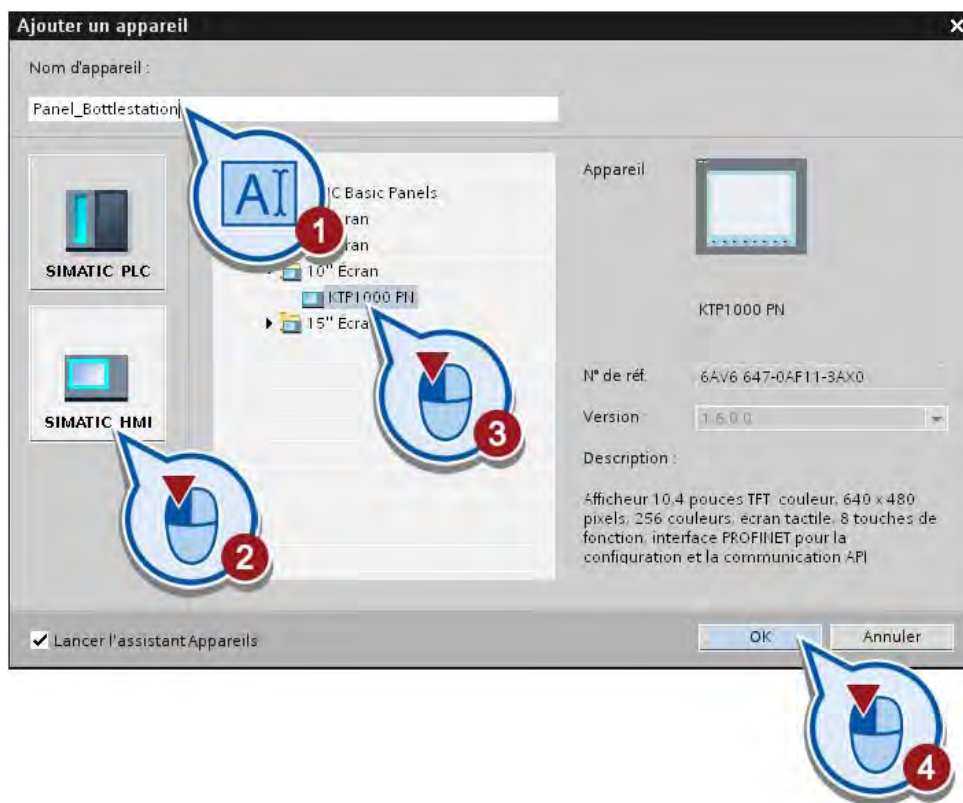
Ajout d'un nouveau pupitre opérateur

Procédez comme suit pour ajouter un pupitre opérateur :

1. Insérez un nouvel appareil par le biais du navigateur du projet.



2. Donnez-lui un nom et sélectionnez un pupitre opérateur. Ne désactivez pas la case d'option "Lancer l'assistant Appareils".

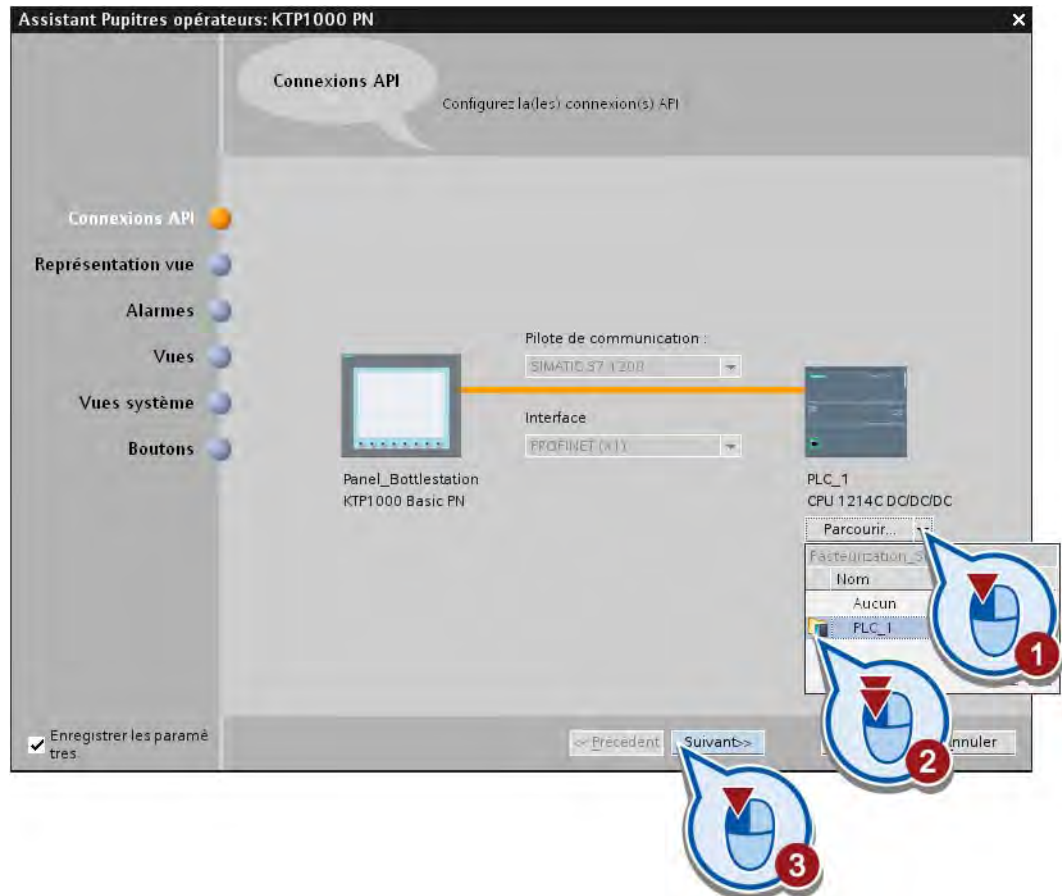


Création d'un modèle pour une vue IHM

L'assistant des pupitres opérateur s'ouvre une fois que vous avez créé un pupitre opérateur. Cet assistant démarre en affichant le dialogue "Connexions API".

Procédez de la manière suivante pour créer un modèle pour la vue IHM :

1. Configurez la connexion à l'automate.

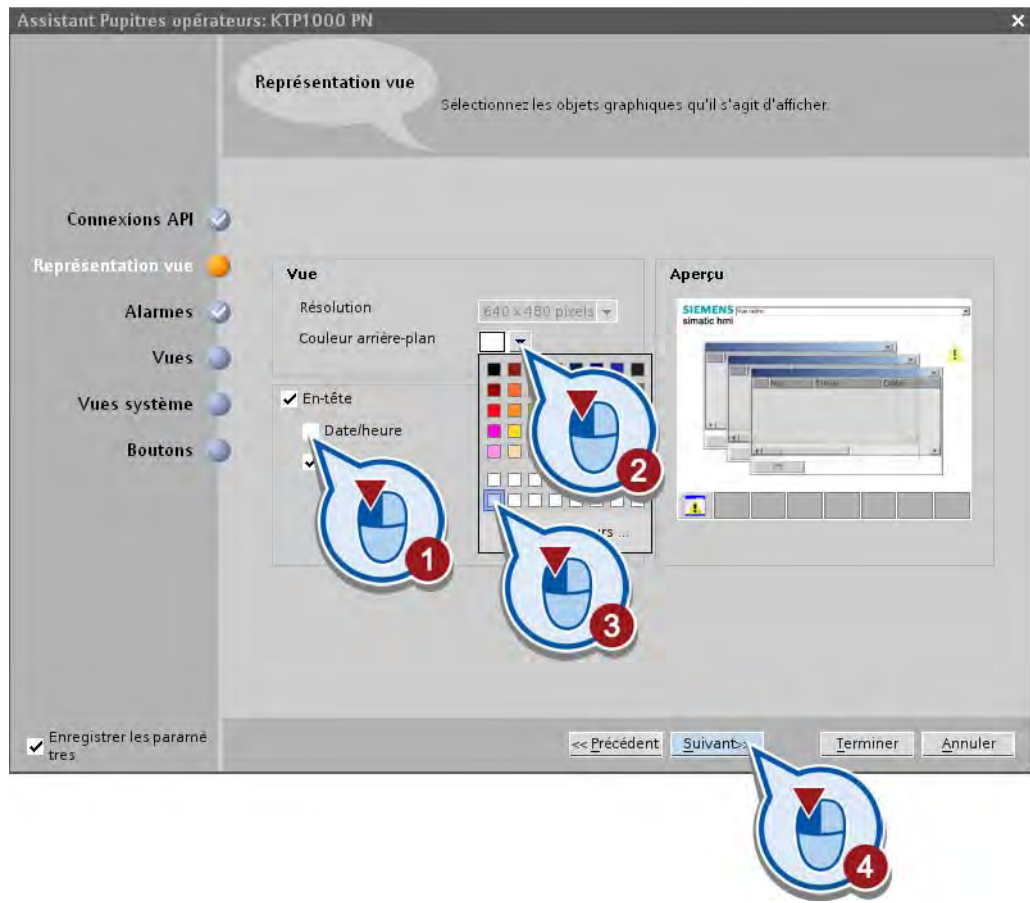


Remarque

Configuration a posteriori de la connexion à l'automate

Vous pouvez également établir la connexion entre le pupitre opérateur et l'automate sous Appareils et réseaux. Lorsque vous configurez la connexion dans ce dialogue, elle est automatiquement créée.

2. Choisissez la couleur d'arrière-plan pour le modèle et les éléments de la ligne d'en-tête.

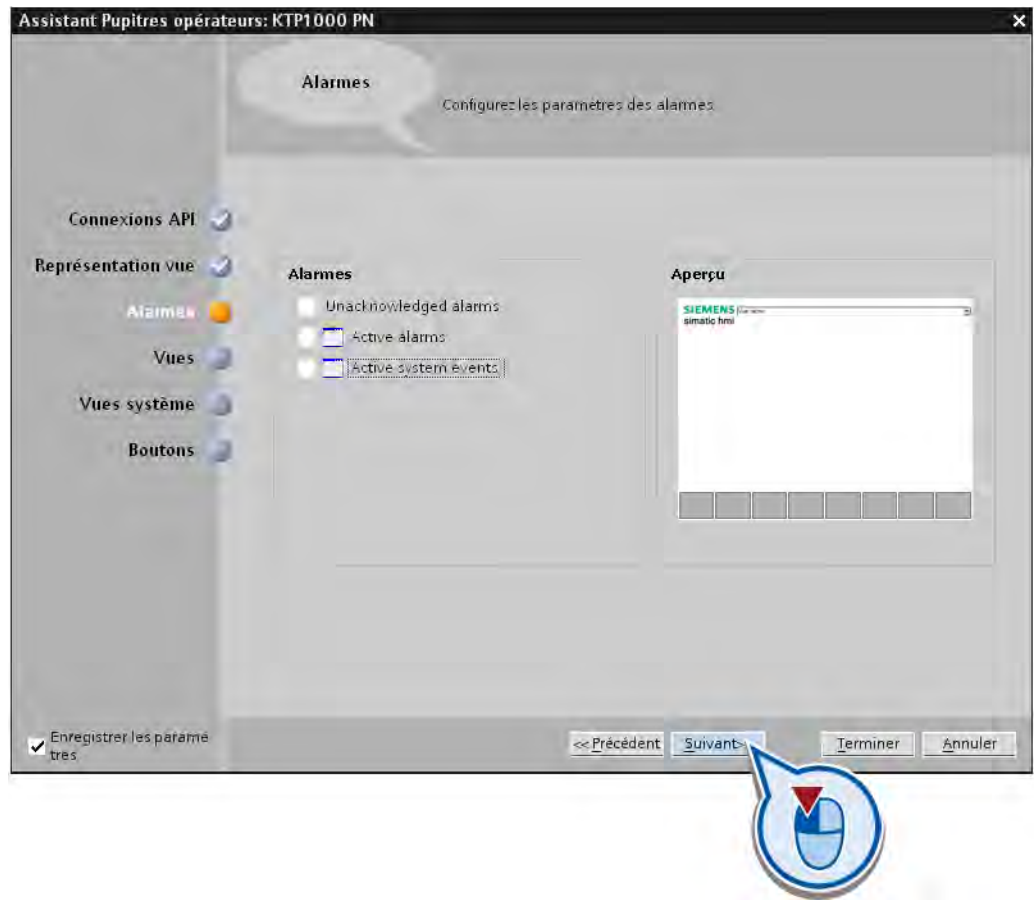


Remarque

Modification a posteriori de la représentation de la vue

Vous pouvez modifier les valeurs paramétrées ici pour la représentation de la vue a posteriori dans le modèle de la vue IHM.

3. Désactivez les alarmes. Elles ne sont pas nécessaires pour le projet-exemple.



Remarque

Alarmes

Si vous activez les alarmes dans l'assistant des pupitres opérateur, vous pourrez sortir les messages par le biais du pupitre opérateur. Les fenêtres de message créées ici sont placées sous "Gestion des vues" dans la vue globale. Vous utiliserez des messages, par exemple pour émettre via le pupitre opérateur des avertissements en cas de dépassement d'une valeur limite. Il est possible de compléter les messages à volonté avec des informations supplémentaires, par exemple pour localiser plus facilement des perturbations dans le système. On distingue principalement entre messages utilisateur et messages système :

- Les messages utilisateur servent à surveiller le processus de l'installation.
- Les messages système sont importés dans le projet et contiennent des informations d'état du pupitre opérateur utilisé.

Vous trouverez plus d'informations sur les messages dans le système d'information du portail TIA.

4. Donnez le nom "HMI" à la vue dans laquelle les éléments graphiques seront créés ultérieurement.



Remarque

Navigation intervues

Ce dialogue vous permet de créer plusieurs vues dans des projets volumineux et d'organiser une navigation intervues. Des boutons sont automatiquement créés pour naviguer entre les vues.

5. Désactivez les vues système. Elles ne sont pas nécessaires pour le projet-exemple.

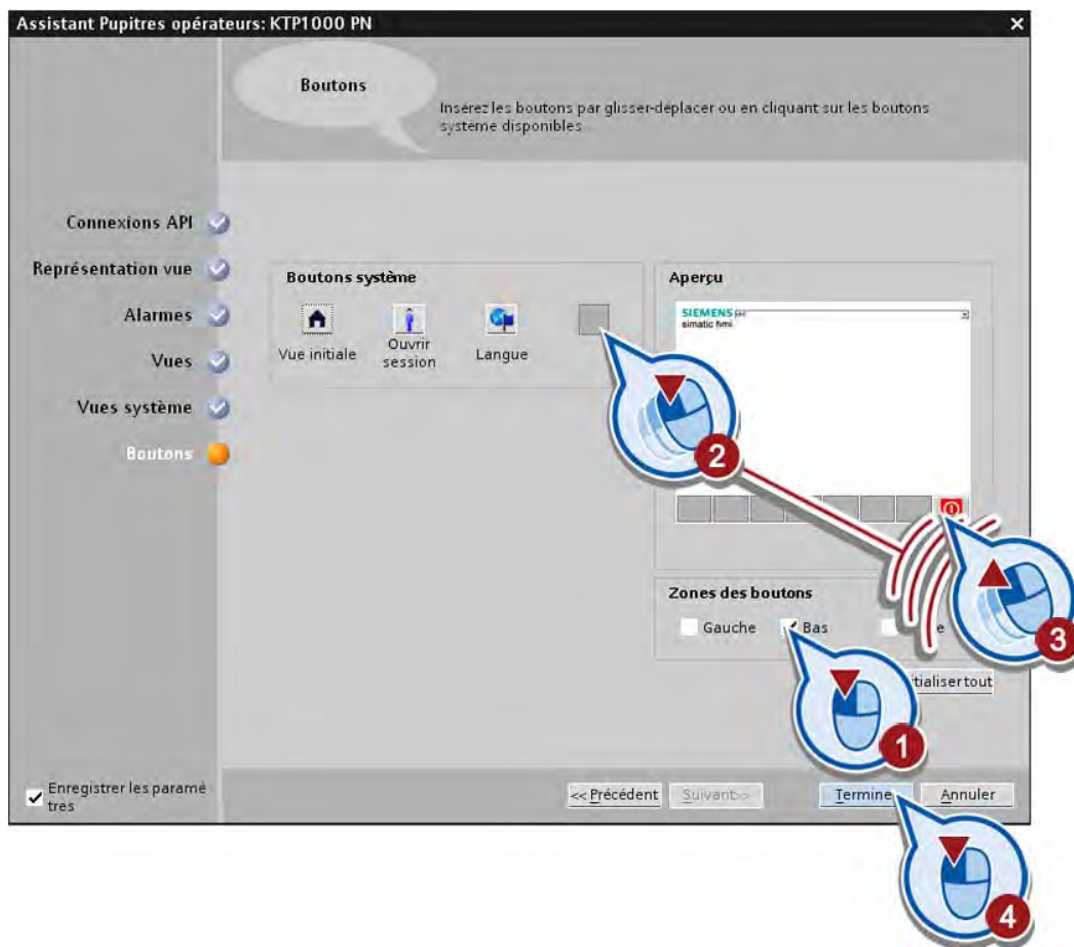


Remarque

Vues système

Les vues système sont des vues IHM contenant des informations sur le projet, le système et le fonctionnement ou permettant la gestion des utilisateurs. Comme pour la navigation intervues, les boutons pour la navigation entre la vue racine et les vues système sont créés automatiquement.

6. Activez la zone de boutons inférieure et insérez le bouton système "Quitter". Ce bouton système permet de quitter Runtime.



7. Enregistrez le projet à l'aide du bouton "Enregistrer le projet" de la barre d'outils.

Résultat

Vous avez créé un pupitre opérateur dans le projet ainsi qu'un modèle pour la vue IHM. Dans la vue du projet, la vue IHM créée s'affiche dans l'éditeur.

Dans le paragraphe suivant, vous allez créer des objets graphiques dans la vue IHM afin de visualiser les processus programmés.

2.6.3 Qu'est-ce qu'un objet graphique ?

Introduction

Dans le portail TIA, vous créez des vues pour le contrôle-commande de machines et d'installations. Pour créer des vues, vous disposez d'objets prédéfinis permettant de représenter votre installation, d'afficher des processus et de prédéfinir des valeurs de process. Les fonctions du pupitre opérateur déterminent la représentation du projet dans l'IHM ainsi que la fonctionnalité des objets graphiques.

Objets graphiques

On appelle objets graphiques tous les éléments que vous pouvez utiliser pour la représentation du projet dans l'IHM. Il s'agit, par exemple, de textes, boutons, schémas ou graphiques pour la représentation de parties de l'installation.

Utilisation d'objets graphiques

Il est possible de représenter les objets graphiques statiquement ou de les utiliser comme objets dynamiques à l'aide de variables :

- Les objets statiques ne changent pas en Runtime. Dans le projet Mise en route, nous allons créer une bande transporteuse en tant qu'objet statique.
- Les objets dynamiques varient en fonction du processus. Vous visualisez les valeurs de process actuelles par le biais :
 - de variables API dans la mémoire de l'automate,
 - de variables internes dans la mémoire du pupitre opérateur, au moyen d'affichages alphanumériques, de courbes et de bargraphes.

Font également partie des objets dynamiques les champs de saisie sur le pupitre opérateur qui permettent l'échange de valeurs de process et d'entrées de l'opérateur entre l'automate et le pupitre opérateur au moyen de variables.

2.6.4 Création et configuration d'objets graphiques

2.6.4.1 Bouton "Marche/Arrêt installation"

Introduction

Les étapes suivantes décrivent comment créer le bouton "Marche/Arrêt installation" et le relier à la variable API "ON_OFF_Switch" par le biais d'une variable IHM externe. Cela vous permet de commander les valeurs de process de la variable API par le biais de la vue IHM.

Variables IHM externes

Une variable IHM externe vous permet d'accéder à une adresse dans l'automate. Vous pouvez ainsi, par exemple, entrer une valeur de process par le biais d'un pupitre opérateur ou modifier directement des valeurs de process du programme de commande par le biais d'un bouton. L'adressage se fait au moyen de la table des variables API de l'automate connecté au pupitre opérateur. La variable API est associée à la variable IHM par le biais du nom symbolique. Aucune adaptation dans IHM n'est donc nécessaire en cas de modification de l'adresse dans la table des variables API.

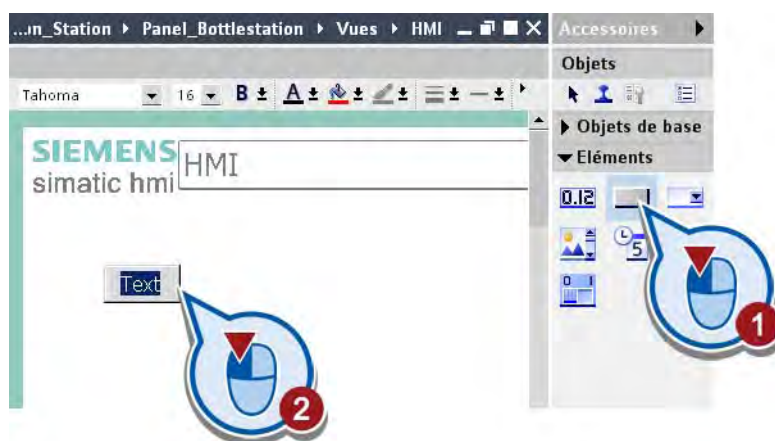
Condition requise

La vue IHM est ouverte.

Marche à suivre

Procédez de la manière suivante pour associer le bouton "Marche/Arrêt installation" à la variable API "ON_OFF_Switch" :

1. Effacez la zone de texte standard "Bienvenue..." de la vue IHM.
2. Créez un bouton.



3. Dans la fenêtre d'inspection, activez l'option "Dimensionner objet au contenu" pour adapter automatiquement la taille du bouton à la longueur du texte.

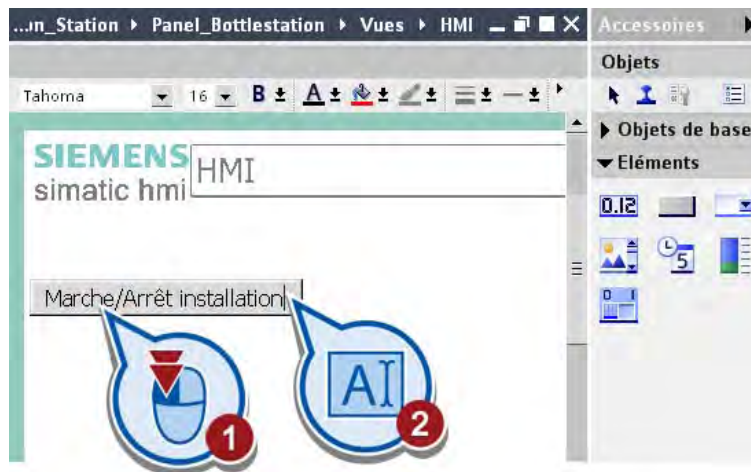


Remarque

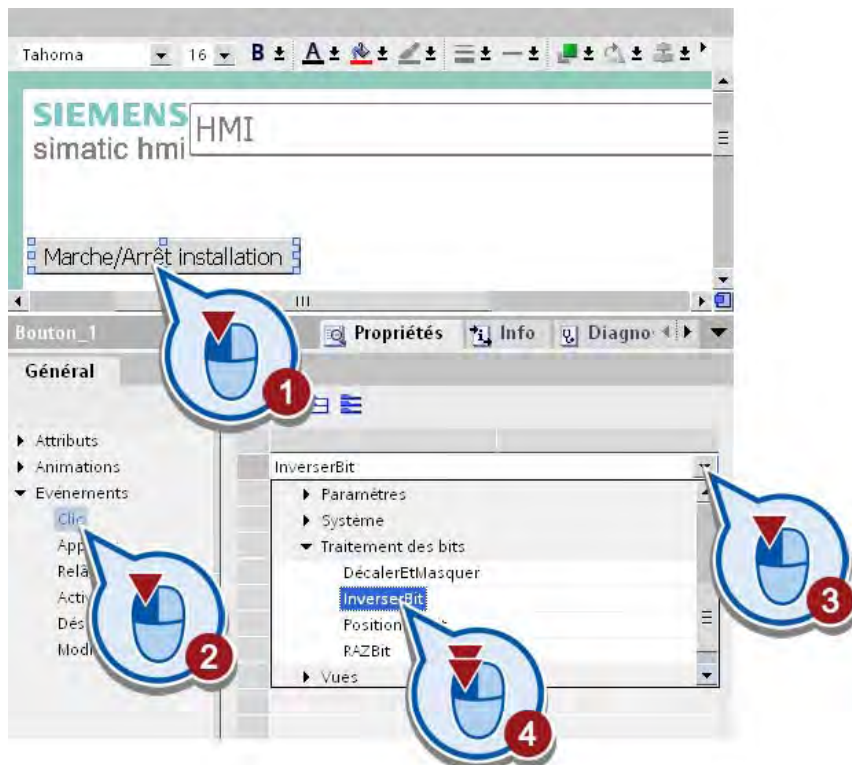
Fonction "Dimensionner objet au contenu"

Vous pouvez en particulier utiliser cette fonction en vue de la traduction des vues IHM dans des projets futurs. En effet, un texte traduit peut être plus ou moins long que l'original selon la langue. Grâce à cette fonction, vous êtes sûr que la dénomination des boutons ne sera pas tronquée. En outre, la taille du bouton s'adapte automatiquement en cas de modification du texte dans l'original.

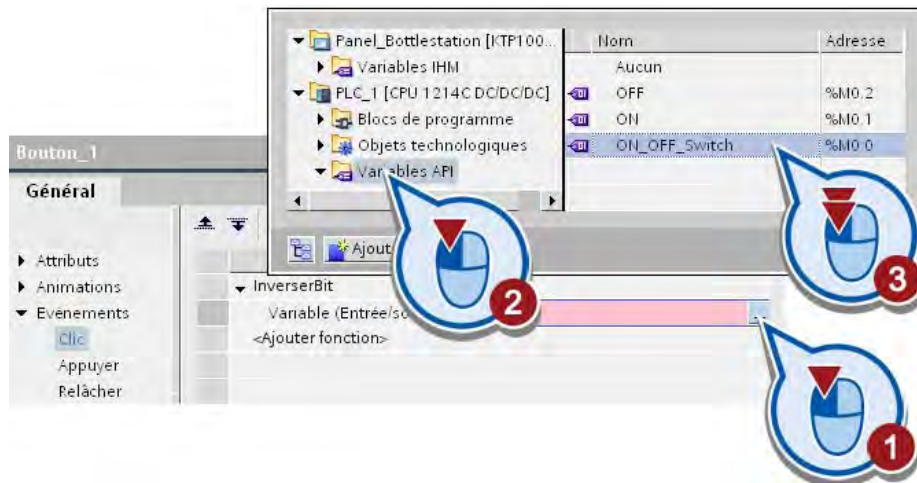
4. Ecrivez le texte "Marche/Arrêt installation" dans le bouton.



5. Affectez la fonction "InverserBit" à l'événement "Presser" du bouton.



6. Associez la fonction "InverserBit" à la variable API ""ON_OFF_Switch"".



Remarque

Création automatique des connexions IHM dans le portail TIA

Si vous n'avez pas configuré au préalable de connexion entre le pupitre opérateur et l'automate, la connexion est automatiquement créée dès que vous associez une variable API à un objet IHM.

Résultat

Vous avez associé le bouton "Marche/Arrêt installation" à la variable API "ON_OFF_Switch". Lorsque vous appuyerez sur le bouton sur le pupitre opérateur, le bit de la variable API sera mis à la valeur "1" (installation mise en marche). Lorsque vous appuyerez une deuxième fois sur le bouton, le bit de la variable API sera mis à la valeur "0" (arrêt de l'installation).

2.6.4.2 Objets graphiques "DEL"

Introduction

Les étapes suivantes décrivent comment créer deux DEL d'état (rouge/verte) avec l'objet cercle et comment les animer en fonction de la valeur de la variable API "ON_OFF_Switch".

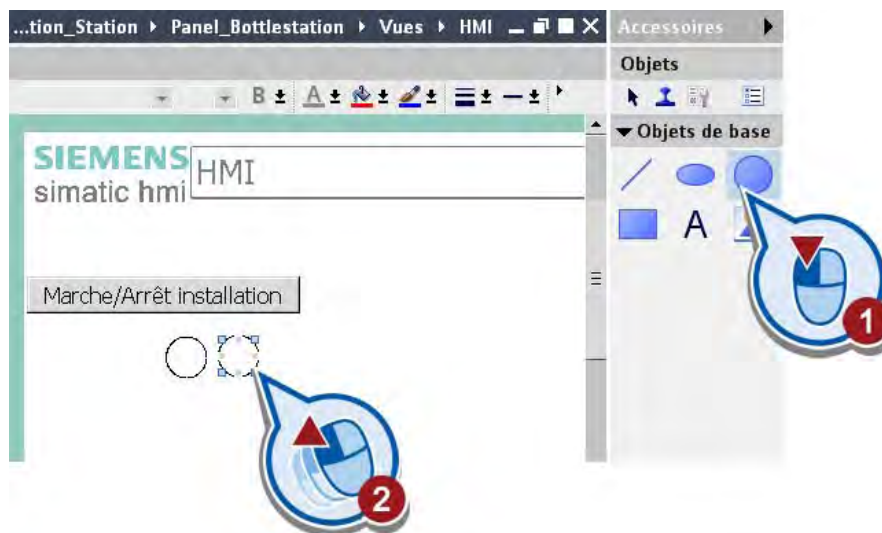
Condition requise

La vue IHM est ouverte.

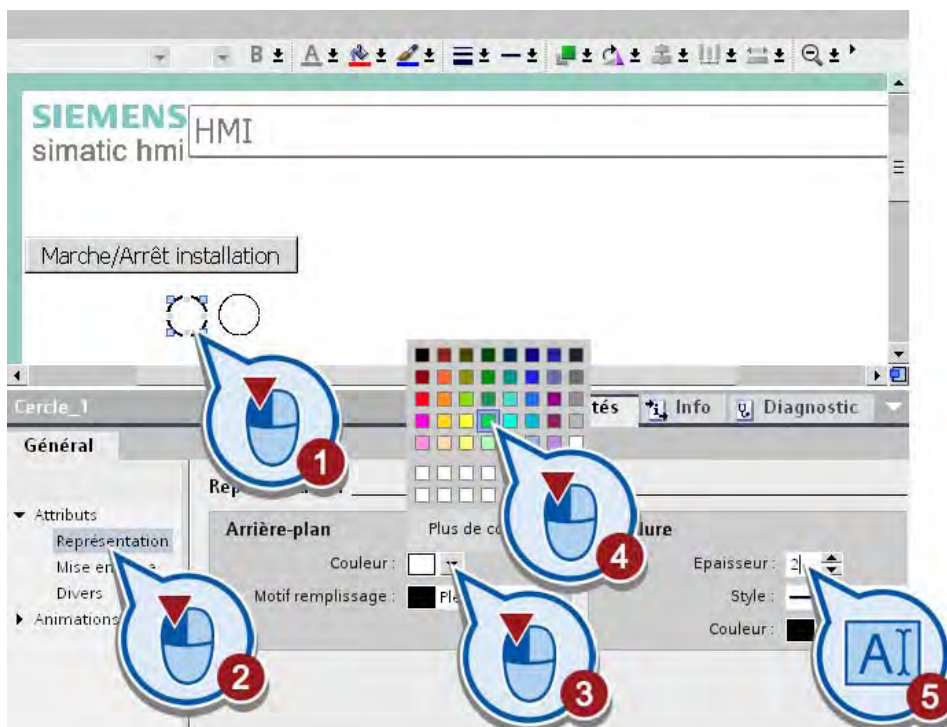
Marche à suivre

Procédez de la manière suivante pour créer et animer les DEL :

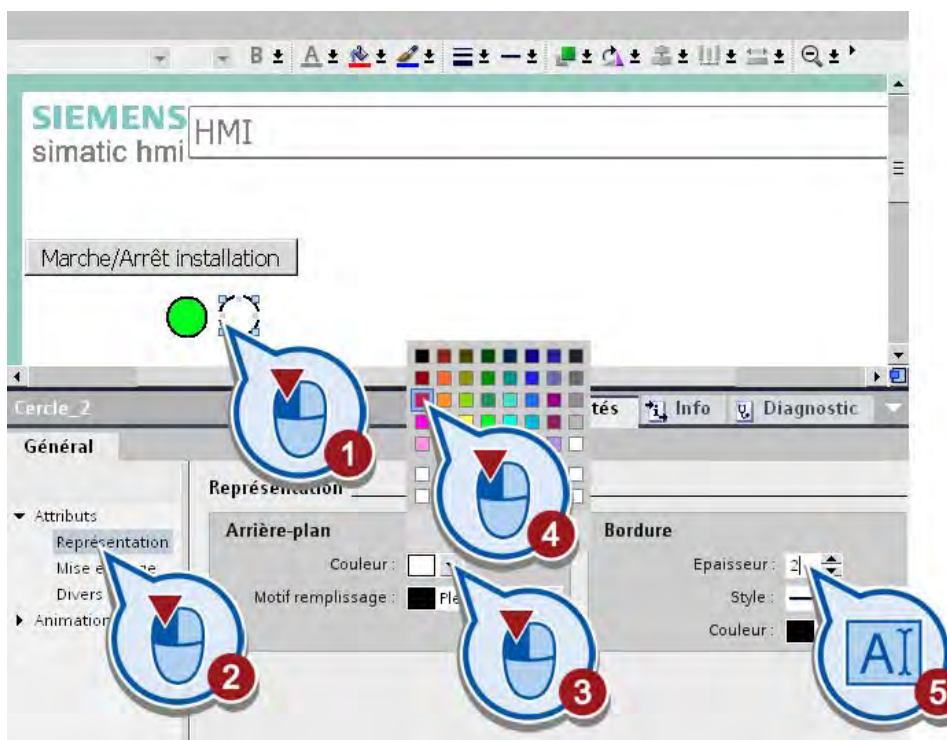
1. Tracez deux cercles sous le bouton "Marche/Arrêt installation" en maintenant la touche Maj enfoncée.



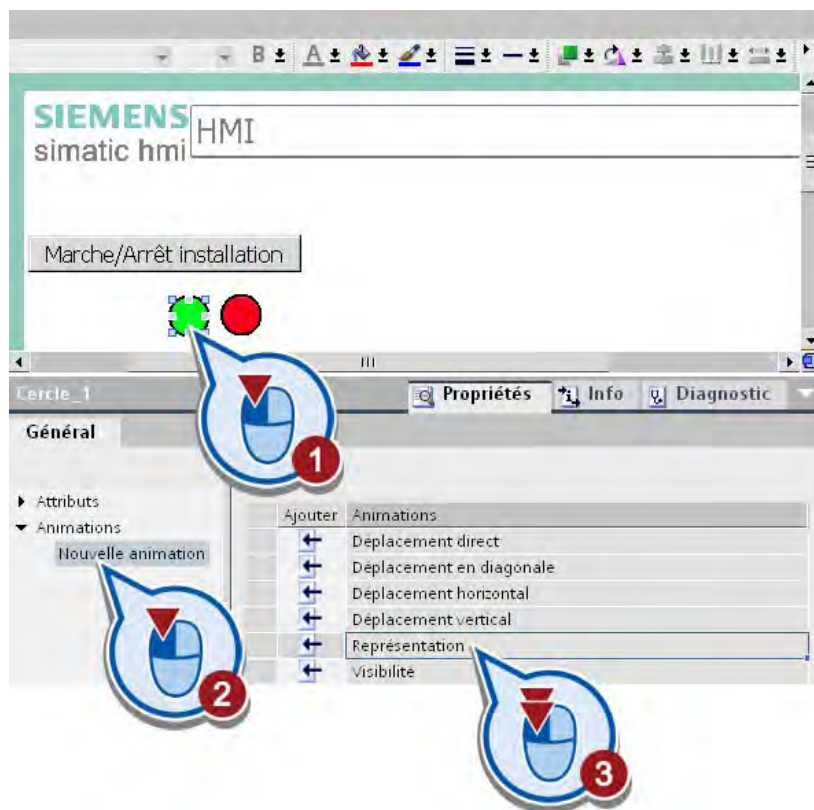
- Définissez la couleur d'arrière-plan à vert et la largeur de bord à "2" pour le premier cercle.



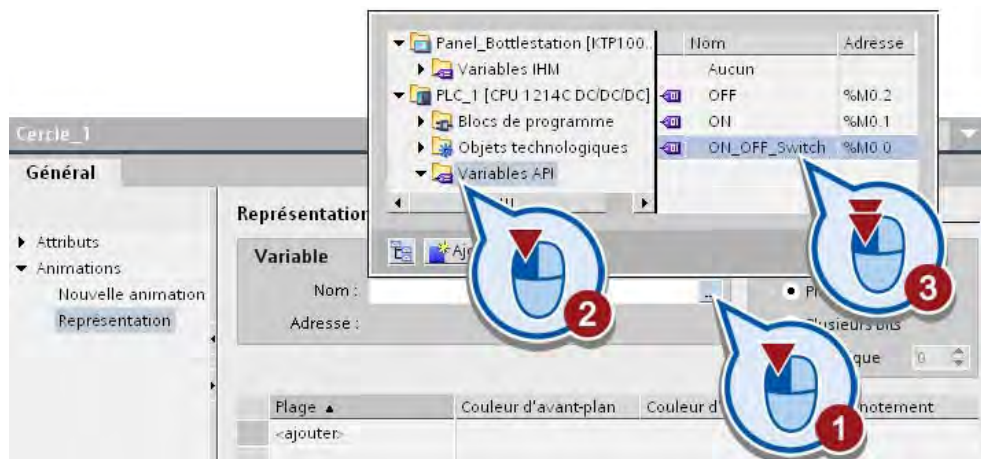
- Pour l'autre cercle, définissez la couleur d'arrière-plan à rouge et augmentez également la largeur de bord à "2".



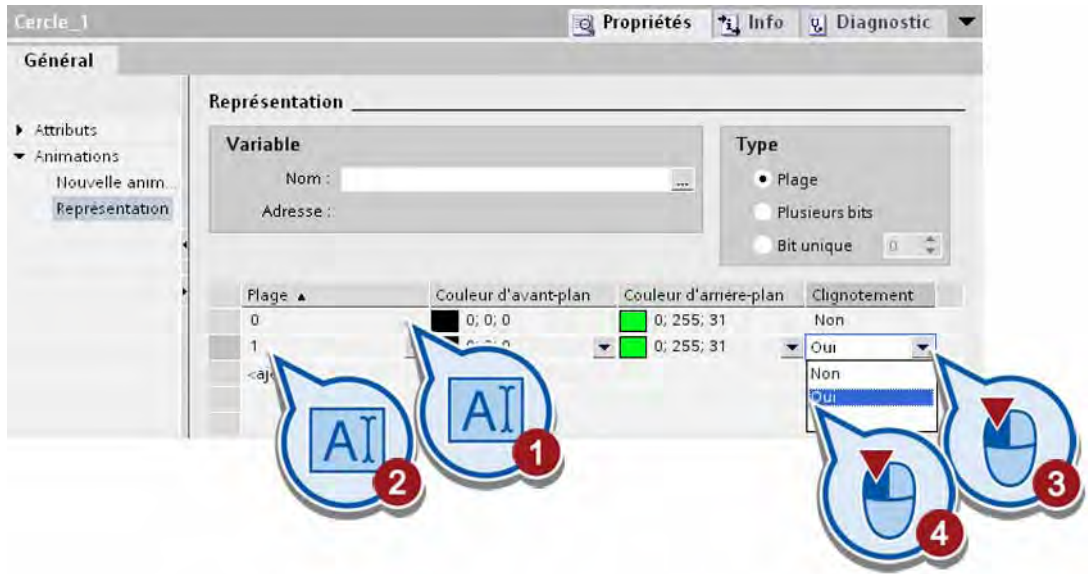
4. Créez une nouvelle animation de type "Représentation" pour la DEL verte.



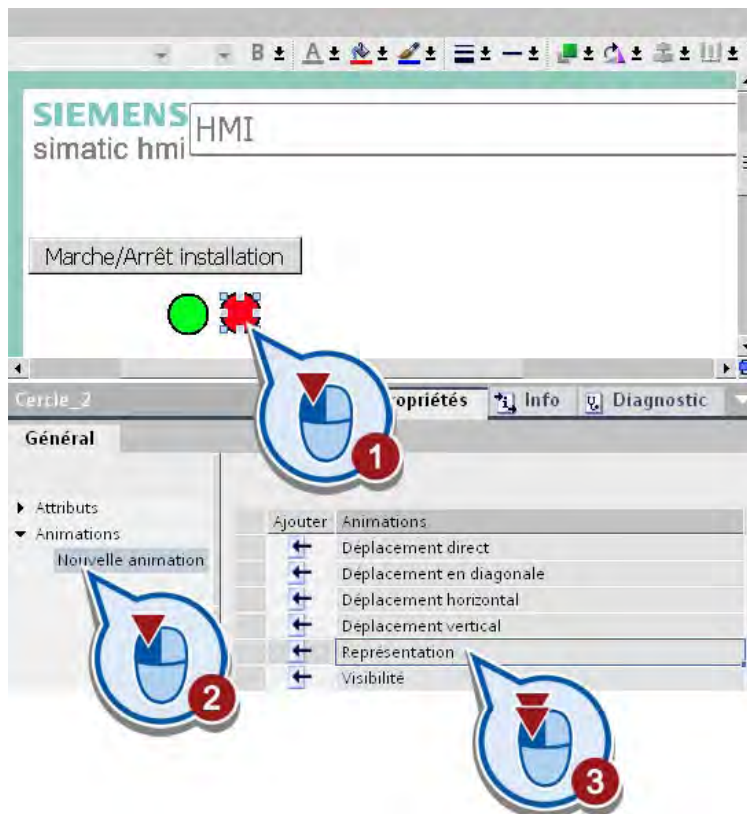
5. Associez l'animation à la variable API "ON_OFF_Switch".



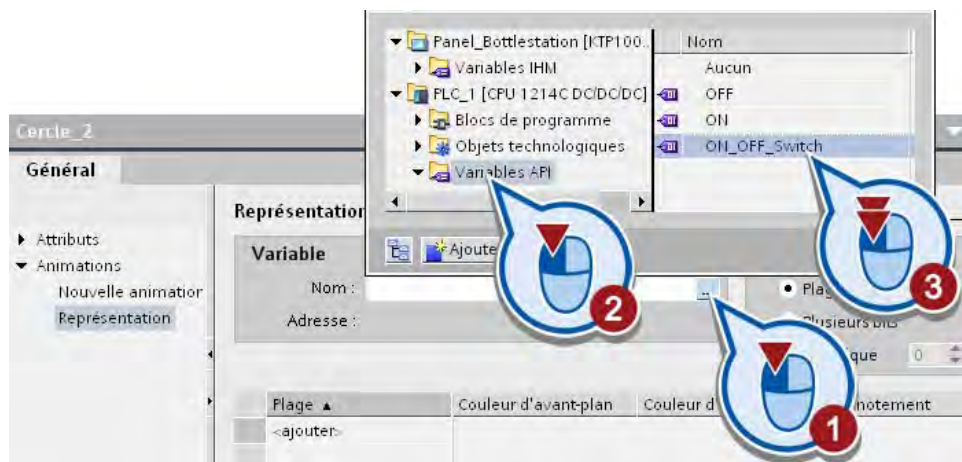
- 6. Modifiez la représentation de la DEL en fonction de l'état de la variable API. La DEL doit clignoter dès que la variable API est mise à la valeur binaire "1" par le programme de commande.



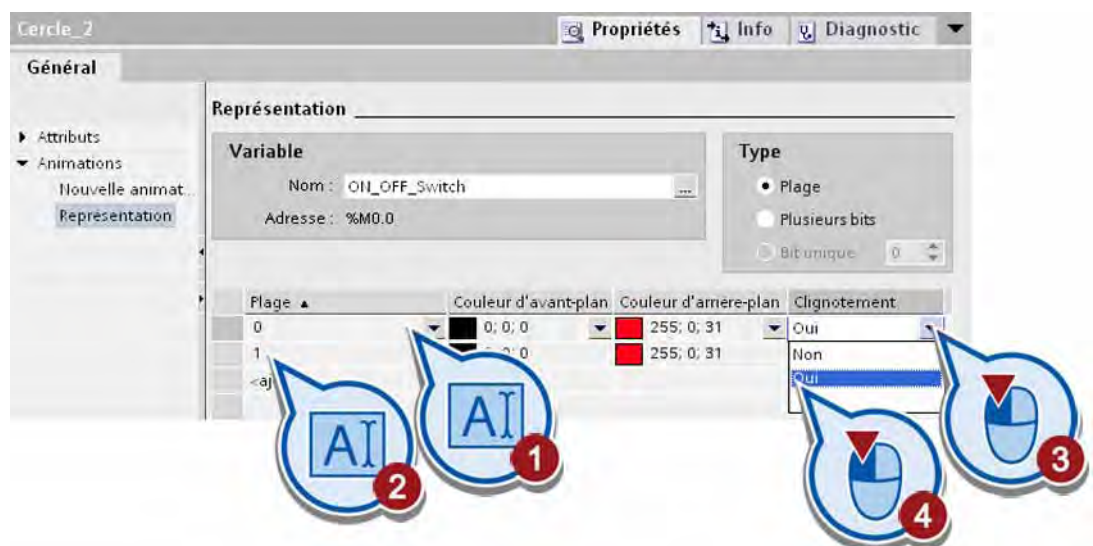
- 7. Créez une nouvelle animation de type "Représentation" pour la DEL rouge.



8. Associez également cette animation à la variable API "ON_OFF_Switch".



9. Modifiez la représentation de la DEL en fonction de l'état de la variable API. La DEL doit clignoter dès que la variable API est mise à la valeur binaire "0" par le programme de commande.



Résultat

Vous avez créé les DEL d'état avec l'objet graphique "Cercle" et les avez animées. A l'état normal, la DEL rouge clignote.

- Si le programme de commande est lancé via le bouton "Marche/Arrêt installation", la valeur binaire de la variable ON_OFF_Switch est mise à "1" et la DEL verte clignote.
- Si le programme de commande est arrêté par une nouvelle pression sur le bouton "Marche/Arrêt installation", la valeur binaire de la variable "ON_OFF_Switch" est mise à "0" et la DEL rouge clignote.

Dans le paragraphe suivant, vous allez créer l'objet graphique "Bande transporteuse".

2.6.4.3 Objet graphique "Bande transporteuse"

Introduction

Les étapes suivantes décrivent comment créer un raccourci à un dossier graphique afin d'importer des objets graphiques. Vous importerez l'objet graphique "Bande transporteuse" (Conveyor.Simple.wmf) par le biais du raccourci.

Condition requise

La vue IHM est ouverte.

Marche à suivre

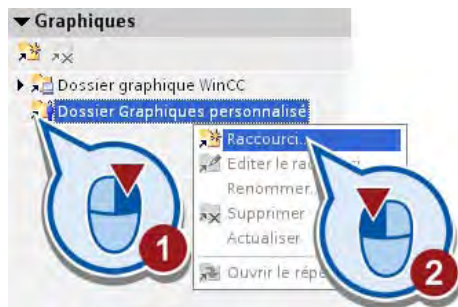
Procédez de la manière suivante pour importer un objet graphique :

1. Copiez le fichier ZIP "WinCC Graphics" de l'adresse Internet suivante sur votre disque dur local et extrayez le fichier :

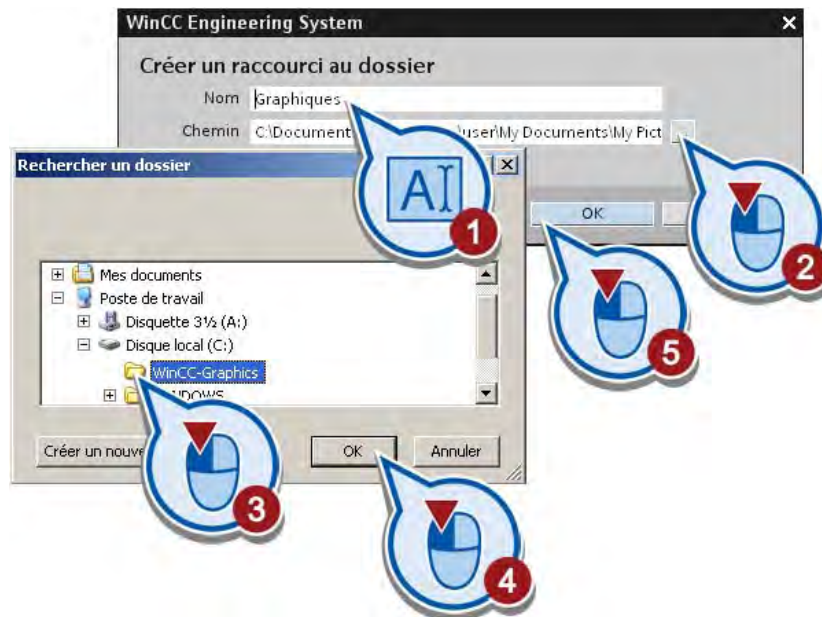
<http://support.automation.siemens.com/WW/view/fr/40263542>

Cliquez sur l'icône "Infos" pour afficher les fichiers ZIP.

2. Ouvrez la palette "Graphiques" dans la Task Card "Outils" et créez un nouveau raccourci.

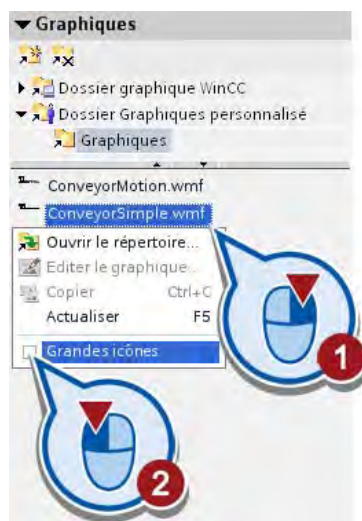


3. Donnez un nom au raccourci et sélectionnez le dossier précédemment extrait "WinCC Graphics".

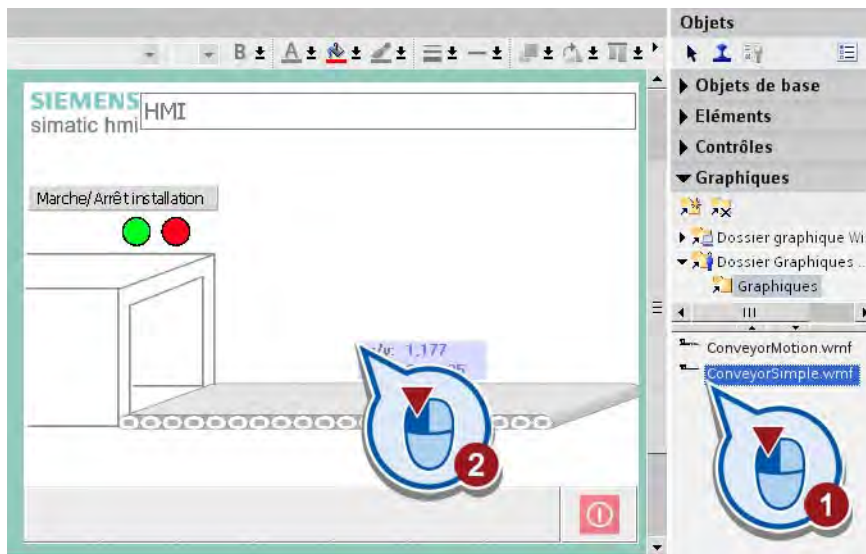


Deux graphiques sont affichés sous le raccourci nouvellement créé.

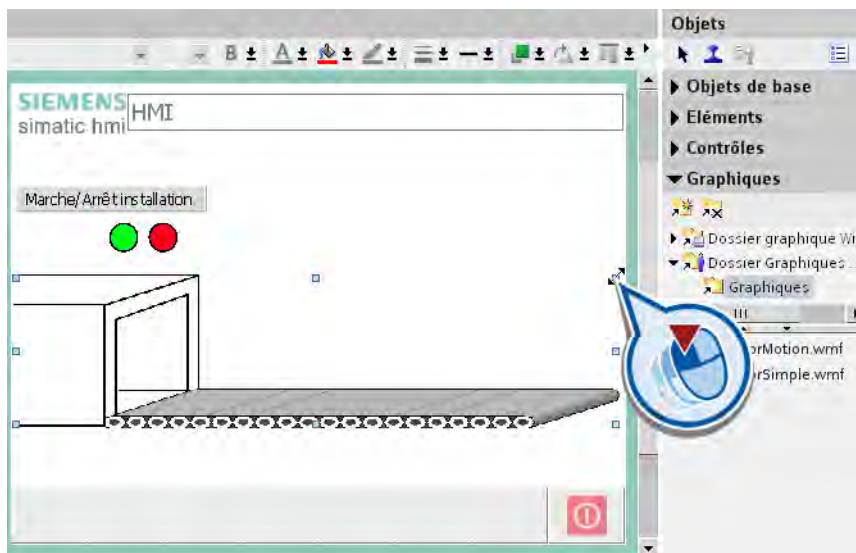
4. Désactivez l'option "Grandes icônes".



5. Placez l'objet graphique "ConveyorSimple.wmf" dans la vue IHM.



6. Mettez l'objet graphique à l'échelle.



Résultat

Vous avez copié l'objet graphique statique "Bande transporteuse" dans le projet. Si vous déplacez ou effacez le dossier "WinCC Graphics", seul le raccourci sera perdu. L'objet graphique sera conservé dans le projet.

Dans le paragraphe suivant, vous allez créer l'objet graphique "Bouteille" avec une animation de déplacement.

2.6.4.4 Objet graphique "Bouteille" avec simulation de déplacement

Introduction

Les étapes suivantes décrivent comment créer l'objet graphique "Bouteille" avec une animation de déplacement. Dans l'animation, la bouteille de lait se déplace de gauche à droite par le biais de la bande transporteuse. Utilisez une variable IHM interne pour animer les objets.

Variables IHM internes

Les variables IHM internes ne possèdent aucun lien avec l'automate. Elles sont enregistrées dans la mémoire du pupitre opérateur. Seul le pupitre opérateur peut accéder en lecture et en écriture à ces variables. Vous créez des variables IHM internes pour effectuer des calculs locaux indépendamment du programme de commande, par exemple.

Condition requise

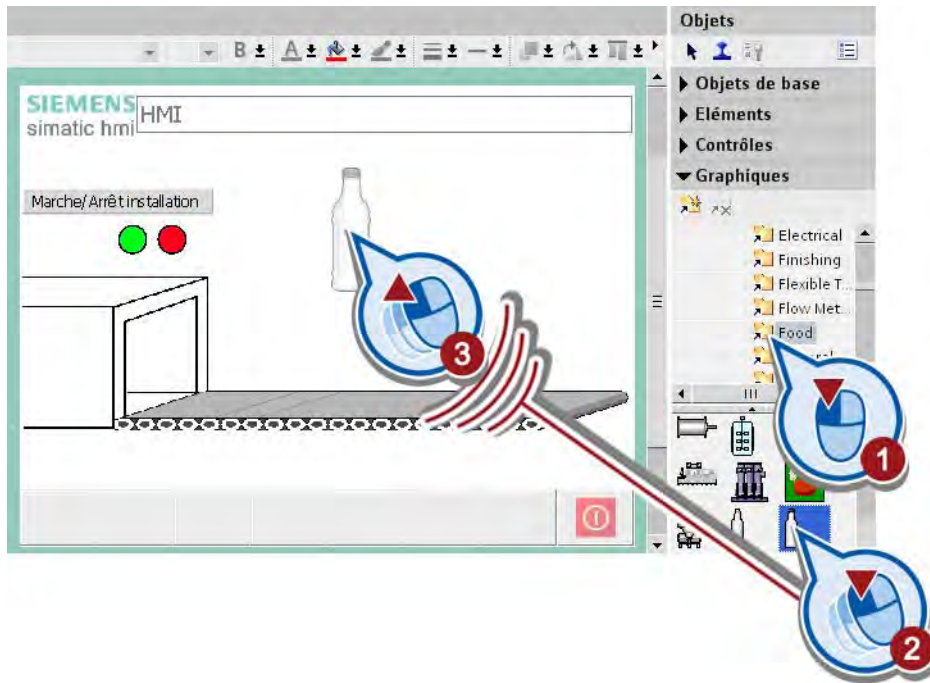
La vue IHM est ouverte.

Marche à suivre

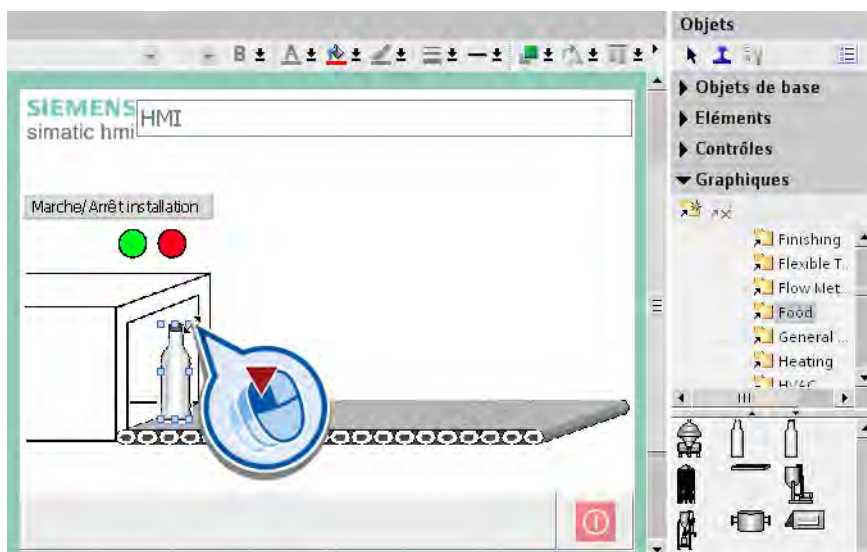
Procédez de la manière suivante pour créer l'objet graphique "Bouteille" et configurer un déplacement horizontal :

1. Copiez l'objet graphique "Bouteille" par glisser-déplacer du dossier graphique WinCC "Symbol Factory Graphics > SymbolFactory 256 Colors > Food" dans la zone libre de la vue au-dessus de l'objet "Bande transporteuse".

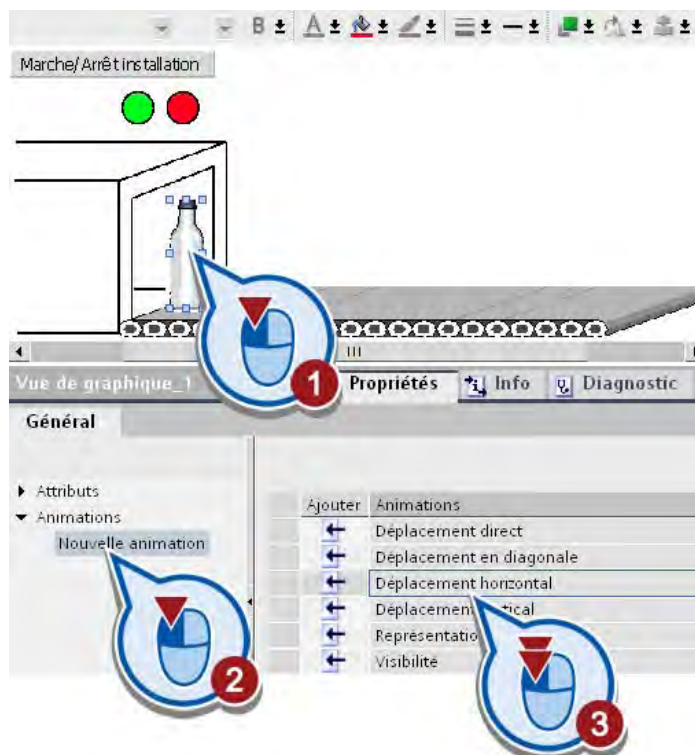
Prenez garde lors du placement à bien déposer la bouteille dans une zone libre de la vue IHM. Si vous faites glisser la bouteille directement sur la bande transporteuse, cette dernière sera remplacée par la représentation de la bouteille.



2. Mettez la bouteille à l'échelle pour lui donner une hauteur inférieure à celle de la goulotte.

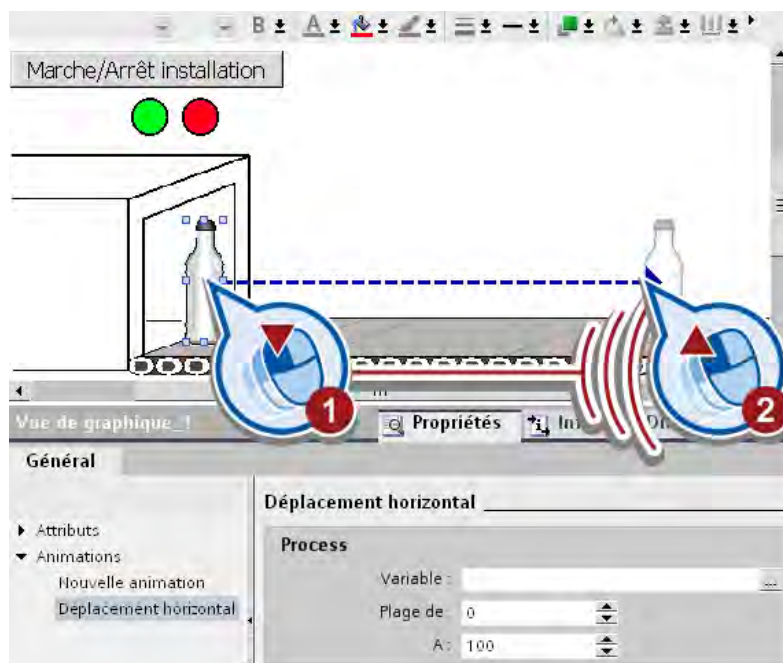


3. Créez une animation de déplacement horizontal pour l'objet graphique "Bouteille".



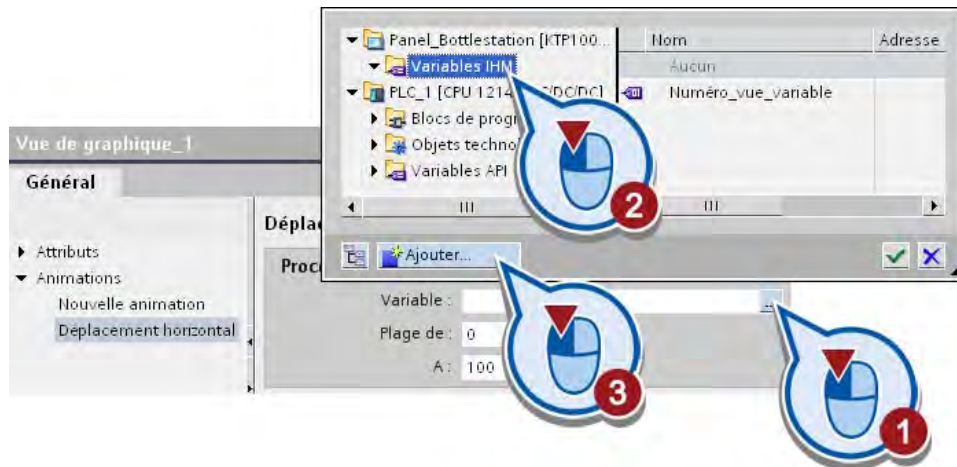
Une copie transparente de la bouteille, reliée à l'objet original par une flèche, s'affiche dans la zone de travail.

4. Déplacez la bouteille transparente à l'extrémité de la bande transporteuse.

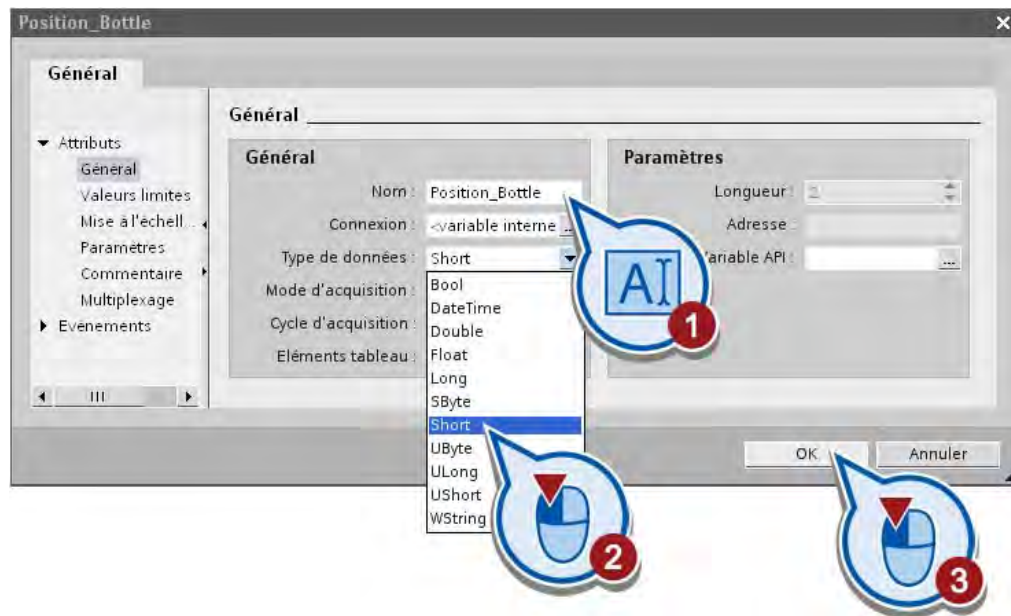


Les coordonnées en pixels de la position finale s'inscrivent automatiquement dans la fenêtre d'inspection.

- 5. Dans la fenêtre d'inspection, créez une nouvelle variable IHM pour l'animation de déplacement.



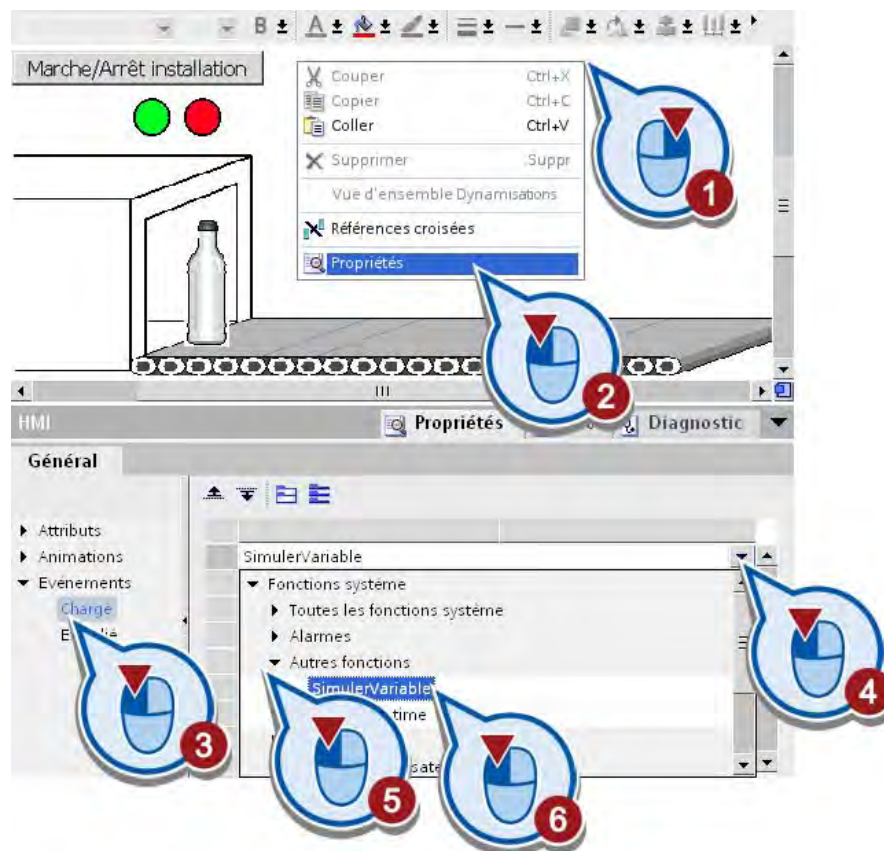
- 6. Utilisez "Position_Bottle" comme nom et "Short" comme type de données pour la variable.



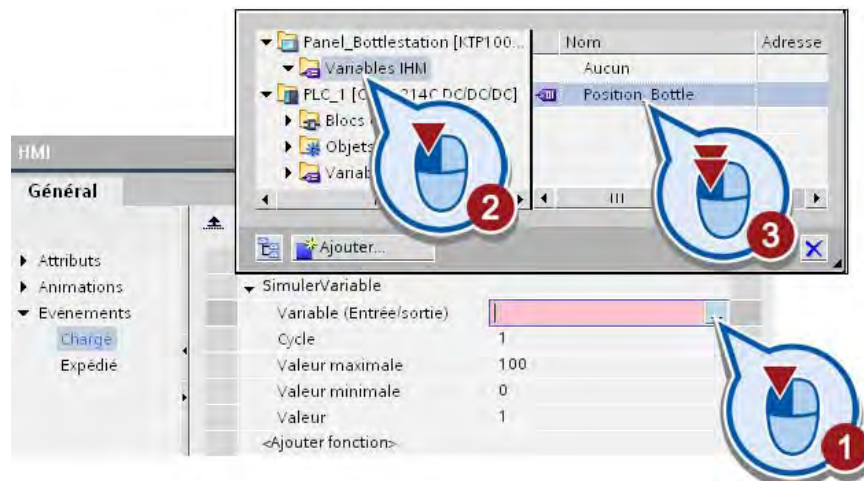
La position de la bouteille est associée à la variable. Lorsque la valeur de la variable est modifiée dans le programme en cours, la position de la bouteille change également.

Pour qu'un déplacement soit simulé, la valeur de la variable Position_Bottle doit changer automatiquement. La valeur de la variable doit s'incrémenter automatiquement après le chargement de la vue IHM. Dès que la valeur 100 est atteinte, il faut recommencer à 0. Les propriétés de la vue IHM permettent de simuler le changement de valeur pour la variable.

7. Ajoutez d'abord la fonction "SimulerVariable" à l'événement "Chargé" de la vue IHM.



8. Affectez la variable "Position_Bottle" à la fonction "SimulerVariable".



9. Enregistrez le projet.

Résultat

Vous avez créé l'objet graphique "Bouteille" avec une animation de déplacement. Lors du chargement de la vue IHM dans le pupitre opérateur, la valeur de la variable "Position_Bottle" s'incrémente de 1 après chaque cycle de base (200 ms). Dès que la valeur 100 est atteinte, la valeur de la variable est mise à "0". La position de la bouteille dépend de la valeur de la variable. Par exemple, si la variable a la valeur 50, la bouteille se trouve au milieu de la bande transporteuse.

Dans le paragraphe suivant, vous allez commander la visibilité de la bouteille au moyen de la variable "ON_OFF_Switch" du bouton "Marche/Arrêt installation".

2.6.4.5 Commande de la visibilité de l'animation de déplacement

Introduction

L'animation de déplacement de la bouteille est lancée automatiquement avec le chargement de la vue IHM. Les étapes suivantes décrivent comment configurer la visibilité de la bouteille animée dans la vue en fonction de la valeur de la variable API "ON_OFF_Switch".

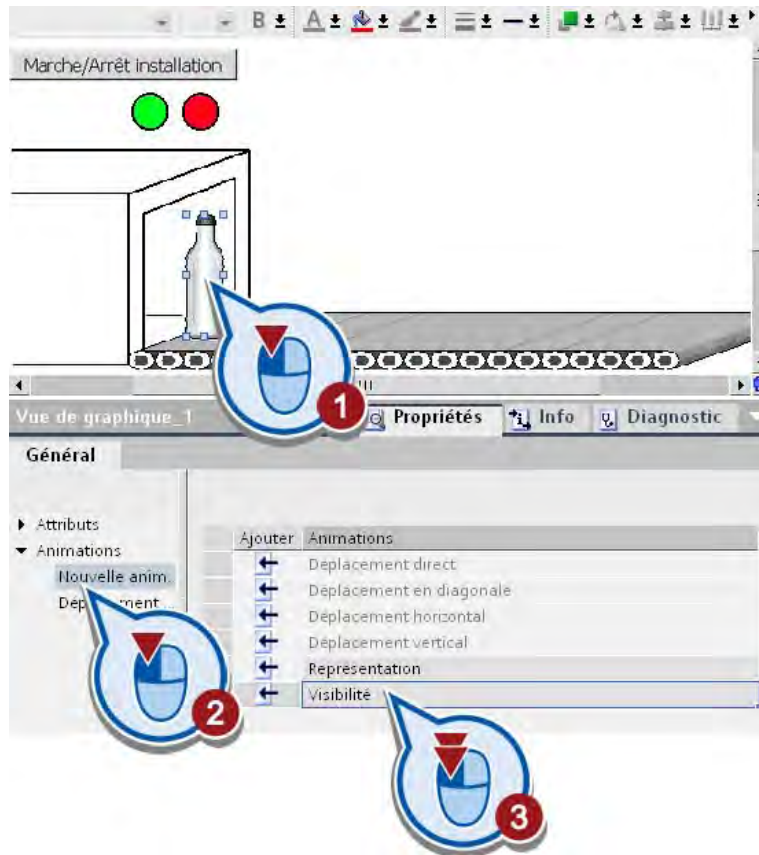
Condition requise

La vue IHM est ouverte.

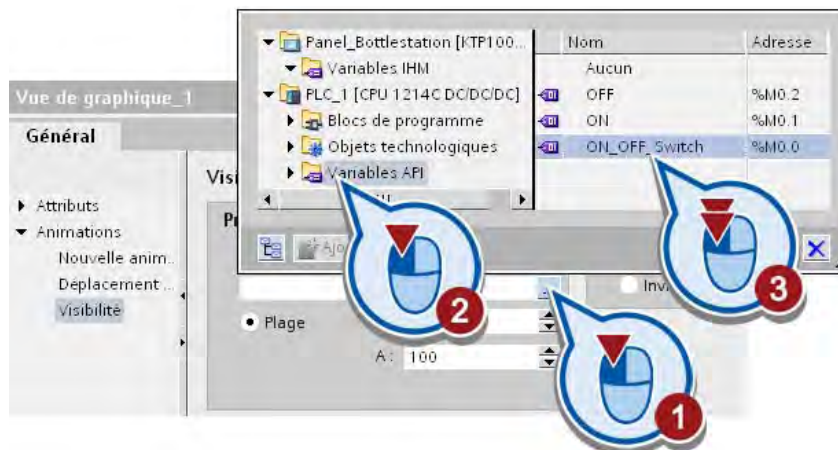
Marche à suivre

Procédez de la manière suivante pour configurer la visibilité de la bouteille dans la vue IHM :

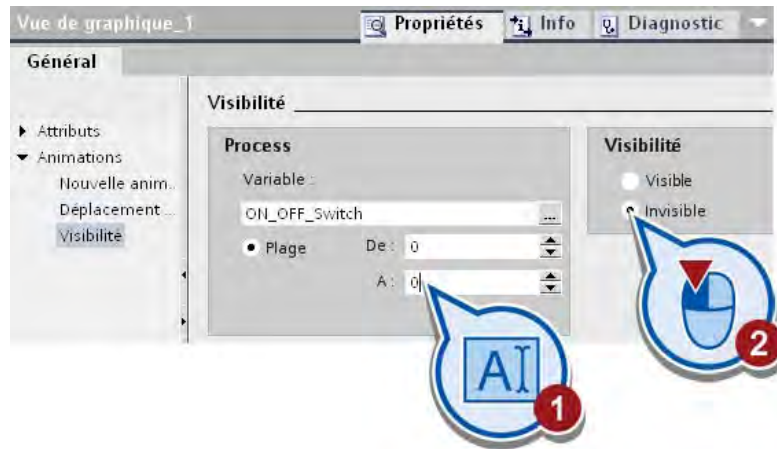
1. Créez une nouvelle animation "Visibilité" pour l'objet graphique "Bouteille".



2. Affectez la variable API "ON_OFF_Switch" à l'animation.



- Définissez la visibilité à "Invisible" pour la plage de variable "0" à "0".



- Enregistrez le projet à l'aide du bouton "Enregistrer le projet" de la barre d'outils.

Résultat

Vous avez configuré la visibilité de la bouteille dans la vue. La bouteille est visible lorsque l'installation est mise en marche et que la valeur de process de la variable "ON_OFF_Switch" prend la valeur "1".

2.7 Test de la vue IHM

2.7.1 Chargement de la vue IHM dans le pupitre opérateur

Introduction

Vous pouvez charger le projet Mise en route dans un pupitre opérateur et l'exécuter dans Runtime. Pour ce faire, une liaison doit être établie entre l'appareil de configuration et le pupitre opérateur. Si vous n'utilisez pas de pupitre opérateur, vous pouvez simuler le Runtime dans le portail TIA (voir Simulation Runtime (Page 91)).

Runtime dans le pupitre opérateur

On utilise des pupitres opérateur pour le contrôle-commande des tâches dans l'automatisation de la fabrication et des processus industriels.

Assurez-vous, si vous utilisez un pupitre opérateur pour l'exécution du projet Mise en route, qu'une liaison est établie entre le pupitre opérateur et l'automate.

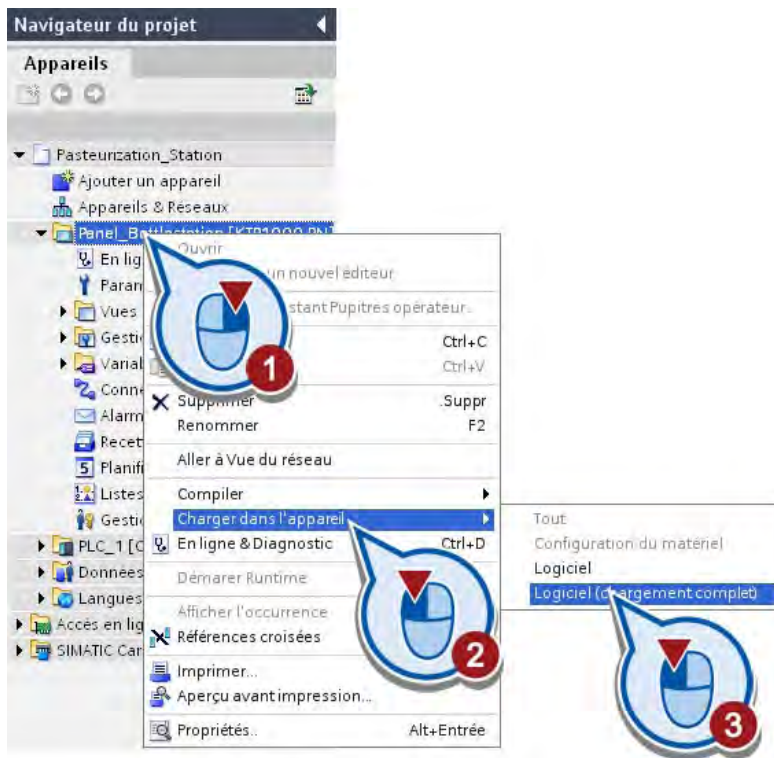
Comme ce sont avant tout des variables API qui sont utilisées dans la vue IHM du projet Mise en route, les animations des objets ne seront exécutées que si une liaison existe entre le pupitre opérateur et l'automate.

Condition requise

- Il existe une liaison avec le pupitre opérateur.
- Le pupitre opérateur est configuré correctement.
- Le pupitre opérateur est en mode Transfert.

Marche à suivre

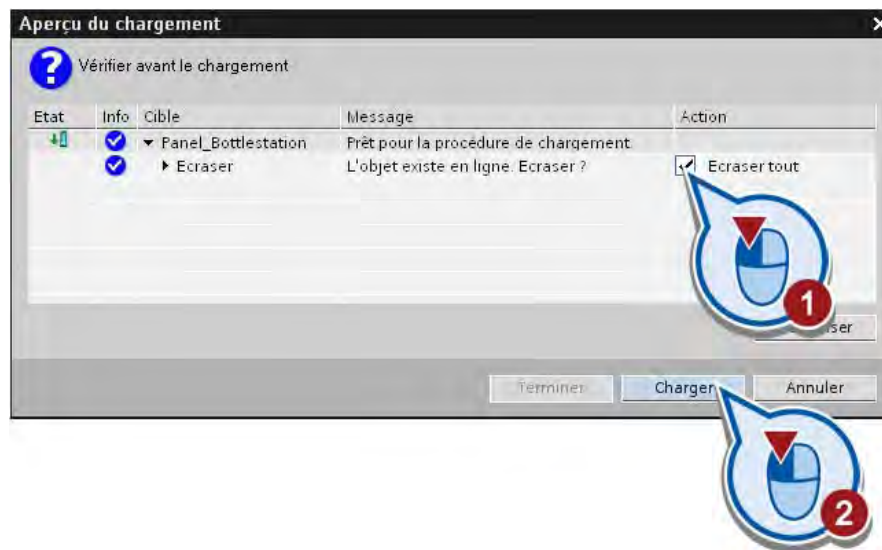
1. Démarrez la procédure de chargement du logiciel dans le pupitre opérateur. Le projet est automatiquement compilé avant le chargement.



PRUDENCE

Assurez-vous qu'aucune donnée significative n'est présente sur le pupitre opérateur. En effet, un logiciel déjà chargé sera écrasé.

2. Ecrasez le cas échéant le logiciel précédemment chargé sur le pupitre opérateur.



Remarque

Informations sur le thème IHM

Vous trouverez plus d'informations sur IHM et sur la configuration des pupitres opérateur dans le système d'information du portail TIA et dans le manuel de votre pupitre opérateur.

2.7.2 Simulation Runtime

Introduction

Si vous n'utilisez pas de pupitre opérateur, vous pouvez simuler toutes les variables API utilisées par le biais du simulateur Runtime.

Remarque

Lancez la simulation Runtime avec le simulateur de variables.

Si vous lancez la simulation sans le simulateur de variables, les boutons et éléments de commande ne seront pas actifs.

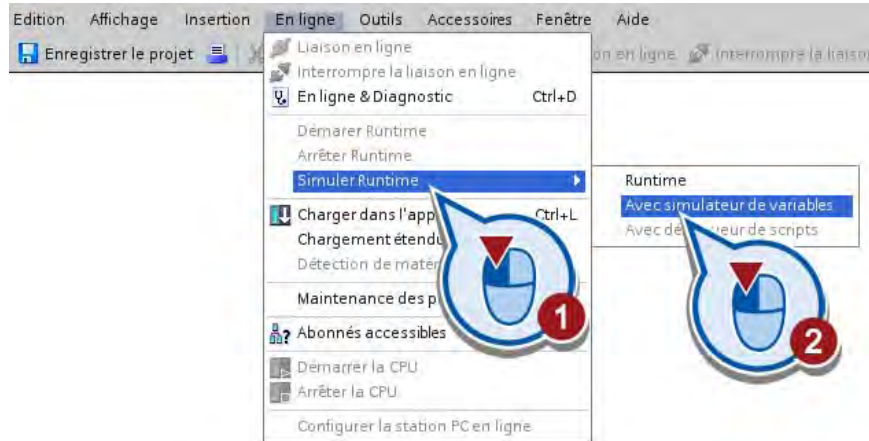
Simulateur Runtime

Le simulateur Runtime vous permet de simuler des valeurs de process des variables API associées indépendamment du programme. Dans la table du simulateur Runtime, vous pouvez sélectionner les variables API et forcer leur valeur. Les objets dans la vue IHM réagissent comme si les variables étaient définies par le programme s'exécutant dans l'automate.

Marche à suivre

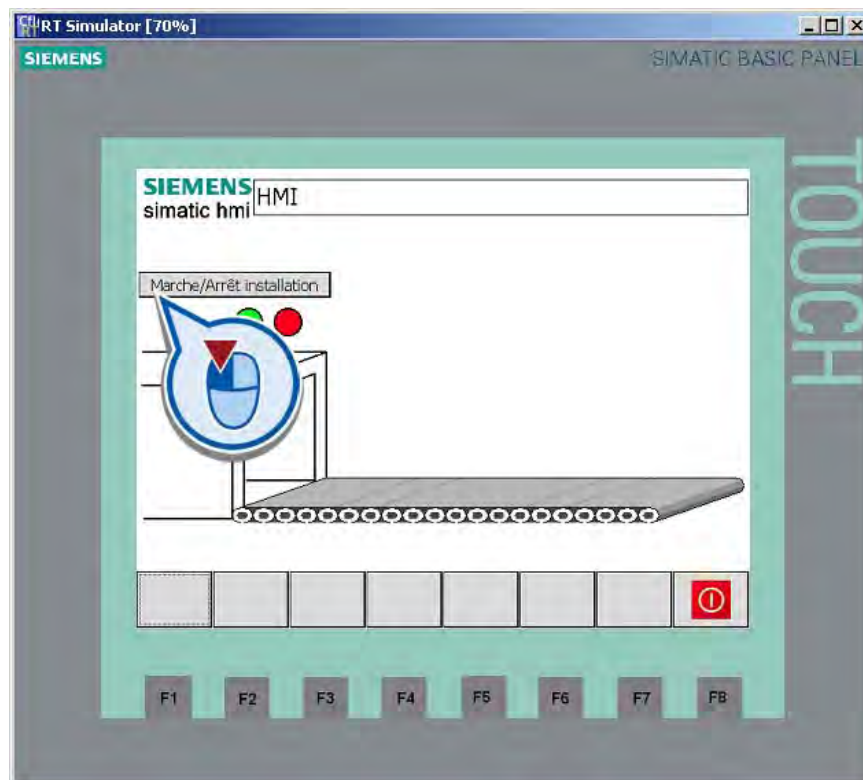
Procédez de la manière suivante pour démarrer la simulation de la vue IHM :

1. Démarrez la simulation Runtime par le biais de la barre des menus. Pour ce faire, la fenêtre IHM doit être active. Si le menu est inactif, cliquez d'abord sur une zone libre dans la vue IHM.



La simulation Runtime est lancée. Une fois la simulation lancée, la vue IHM est affichée dans la fenêtre "Simulateur RT" et la DEL rouge clignote (l'installation est arrêtée).

2. Démarrez l'installation.



Remarque

Messages de compilation

Si la simulation Runtime ne démarre pas en raison d'erreurs dans le projet, les messages correspondants sont affichés dans la fenêtre d'inspection sous "Info > Compiler". Double-cliquez sur un message pour naviguer automatiquement jusqu'à l'objet IHM mal configuré.

Résultat

La simulation de déplacement est démarrée. Simultanément, la DEL verte se met à clignoter à la place de la DEL rouge. Si vous cliquez une nouvelle fois sur le bouton "Marche/Arrêt installation", la bouteille n'est plus visible et la DEL rouge se met à clignoter à la place de la DEL verte. Vous pouvez mettre fin à la simulation Runtime en fermant la fenêtre ou en cliquant sur le bouton "Quitter Runtime".

Exemple avancé

3.1 Introduction

Chargement du projet

Si vous avez sauté le chapitre précédent, vous pouvez charger l'état du projet à la fin du chapitre (voir "Chargement de projets (Page 18)"). L'état à la fin du chapitre est sauvegardé dans le fichier "Simple_Example.ZIP".

Présentation

Dans la deuxième partie du projet-exemple, vous créez un autre programme servant à commander une station simple de pasteurisation de lait. La station se compose d'une bande transporteuse et d'une chambre de chauffe.

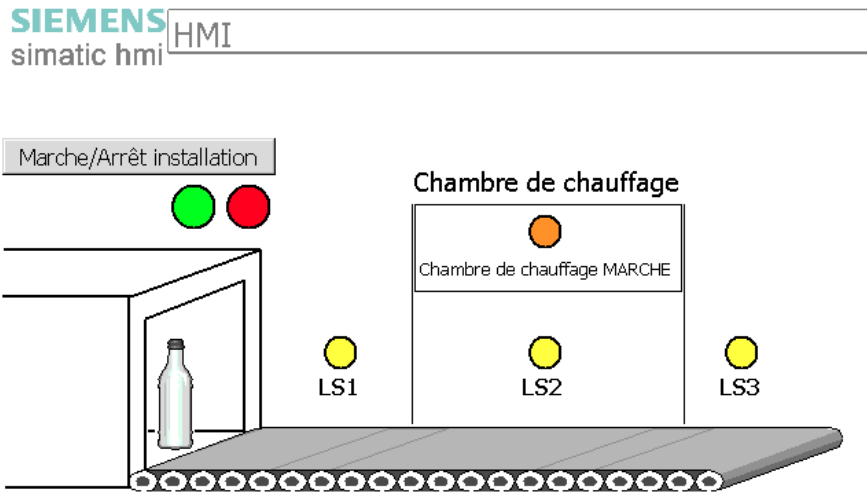
L'entraînement de la bande transporteuse est piloté par trois barrières photoélectriques :

- La première barrière photoélectrique (LS1) est activée lorsqu'une bouteille de lait se trouve au début de la bande transporteuse. L'entraînement de la bande transporteuse est activé et la bouteille est transportée dans la chambre de chauffe.
- La deuxième barrière photoélectrique (LS2) détecte la position d'une bouteille de lait dans la chambre de chauffe. Elle signale que la bande transporteuse doit être arrêtée et que le chauffage doit être mis en marche dans la chambre.

Dans la chambre de chauffe, les bouteilles sont chauffées à 75°C pendant 40 secondes. Un voyant d'état s'allume tant que le chauffage fonctionne. Une fois l'opération de chauffage achevée, l'entraînement de la bande est à nouveau activé et la bouteille est transportée jusqu'à l'extrémité de la bande transporteuse.

- Lorsqu'une bouteille atteint l'extrémité de la bande transporteuse, la troisième barrière photoélectrique (LS3) est activée et la bande transporteuse s'arrête. La station suivante peut alors venir chercher la bouteille.

La mise en marche et l'arrêt de l'ensemble de la station sont commandés par le bouton qui a été programmé dans la première partie du projet.



Etapes de travail

La deuxième partie du projet comprend les étapes de travail suivantes :

- Compléter le programme
Vous complétez le programme avec quatre réseaux supplémentaires.
- Charger le projet dans l'automate
Vous chargerez le projet complété dans l'automate et vous visualiserez l'exécution du programme dans la vue en ligne.
- Tester le programme avec la visualisation d'état du programme
Vous testerez l'exécution du programme en visualisant l'état du programme.
- Compléter la vue IHM
Pour visualiser le programme complété, vous insérerez d'autres objets graphiques dans la vue IHM existante.
- Simuler la vue IHM
Vous exécuterez la vue IHM en mode Runtime.

3.2 Compléter le programme

3.2.1 Définition de variables dans la table des variables API

Introduction

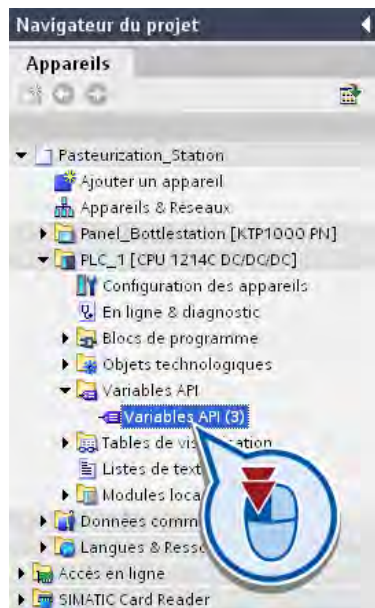
Les étapes suivantes décrivent comment définir les variables nécessaires pour compléter le programme. Vous créez les variables dans la table des variables API.

Vous trouverez des informations générales sur les variables API et la table des variables API au paragraphe "Qu'est-ce qu'une variable ? (Page 44)".

Marche à suivre

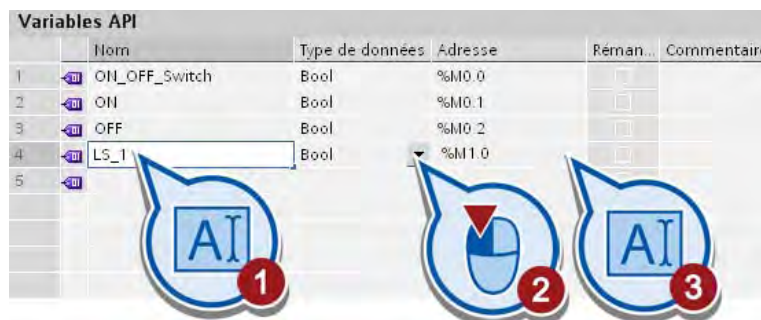
Procédez de la manière suivante pour définir les variables nécessaires :

1. Ouvrez la table des variables API.

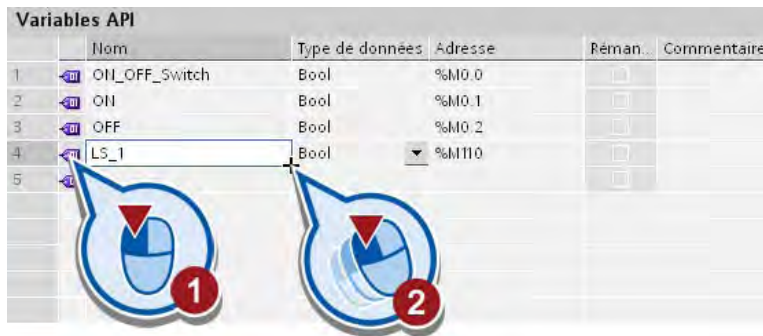


La table des variables s'ouvre dans la zone de travail. Les variables définies dans la première partie du projet apparaissent.

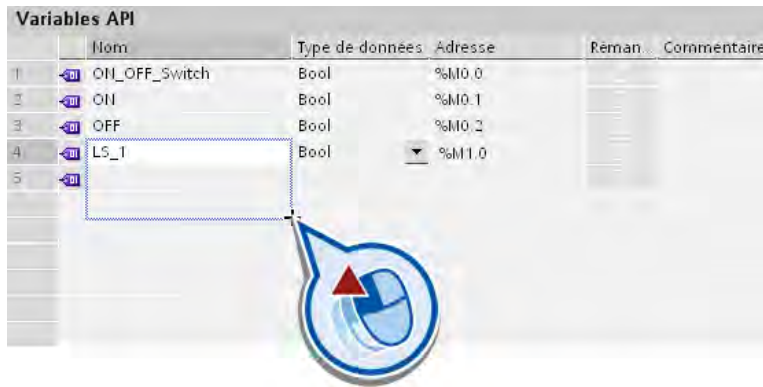
2. Définissez la variable "LS_1" d'adresse "M1.0".



3. Sélectionnez la cellule "Nom" de la variable "LS_1" et cliquez sur la poignée de recopie dans le coin inférieur droit de la cellule.

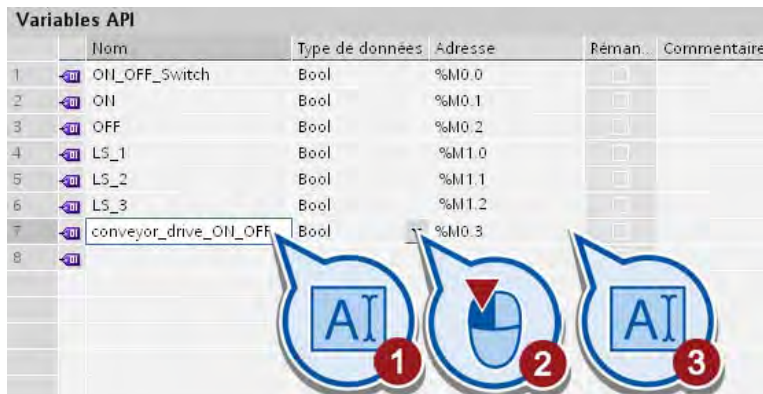


4. Transférez le contenu de la cellule dans les deux cellules inférieures suivantes.




Le contenu est copié avec des numéros consécutifs.

5. Définissez la variable conveyor_drive_ON_OFF.




6. Définissez la variable chamber_ON_OFF.

Variables API					
	Nom	Type de données	Adresse	Réman...	Commentaire
1	ON_OFF_Switch	Bool	%M0.0	<input type="checkbox"/>	
2	ON	Bool	%M0.1	<input type="checkbox"/>	
3	OFF	Bool	%M0.2	<input type="checkbox"/>	
4	LS_1	Bool	%M1.0	<input type="checkbox"/>	
5	LS_2	Bool	%M1.1	<input type="checkbox"/>	
6	LS_3	Bool	%M1.2	<input type="checkbox"/>	
7	conveyor_drive_ON_OFF	Bool	%M0.3	<input type="checkbox"/>	
8	chamber_ON_OFF	Bool	%M0.4	<input type="checkbox"/>	
9					



7. Définissez la variable milk_pasteurized.

Variables API				
	Nom	Type de donn...	Adresse	Rémanence
1	ON_OFF_Switch	Bool	%M0.0	<input type="checkbox"/>
2	ON	Bool	%M0.1	<input type="checkbox"/>
3	OFF	Bool	%M0.2	<input type="checkbox"/>
4	LS_1	Bool	%M1.0	<input type="checkbox"/>
5	LS_2	Bool	%M1.1	<input type="checkbox"/>
6	LS_3	Bool	%M1.2	<input type="checkbox"/>
7	conveyor_drive_ON_OFF	Bool	%M0.3	<input type="checkbox"/>
8	chamber_ON_OFF	Bool	%M0.4	<input type="checkbox"/>
9	milk_pasteurized	Bool	%M0.5	<input type="checkbox"/>
10				



Résultat

Vous avez défini les variables dont vous avez besoin pour compléter le programme. Les variables définies apparaissent dans la table des variables API.

Le tableau suivant présente toutes les variables définies jusqu'à maintenant et donne la signification de leurs valeurs respectives :

Nom	Type de données	Adresse	Fonction	Signification des valeurs de la variable
ON_OFF_Switch	BOOL	M0.0	Interrupteur poussoir	<ul style="list-style-type: none"> 1: L'installation se met en marche. 0: L'installation s'arrête.
ON	BOOL	M0.1	Fonctionnement de l'installation	<ul style="list-style-type: none"> 1: L'installation fonctionne. 0: Sans effet
OFF	BOOL	M0.2	Arrêt de l'installation	<ul style="list-style-type: none"> 1: L'installation est arrêtée. 0: Sans effet
LS_1	BOOL	M1.0	Barrière photoélectrique pour détecter la position de la bouteille au début de la bande transporteuse	<ul style="list-style-type: none"> 1: La barrière photoélectrique 1 est activée. 0: La barrière photoélectrique 1 est désactivée.
LS_2	BOOL	M1.1	Barrière pour détecter la position de la bouteille dans la chambre de chauffe	<ul style="list-style-type: none"> 1: La barrière photoélectrique 2 est activée. 0: La barrière photoélectrique 2 est désactivée.
LS_3	BOOL	M1.2	Barrière photoélectrique pour détecter la position de la bouteille à l'extrémité de la bande transporteuse	<ul style="list-style-type: none"> 1: La barrière photoélectrique 3 est activée. 0: La barrière photoélectrique 3 est désactivée.
conveyor_drive_ON_OFF	BOOL	M0.3	Etat de fonctionnement de la bande transporteuse	<ul style="list-style-type: none"> 1: La bande transporteuse se met en mouvement. 0: La bande transporteuse s'arrête.
chamber_ON_OFF	BOOL	M0.4	Etat de fonctionnement de la chambre de chauffe	<ul style="list-style-type: none"> 1: La chambre de chauffe se met en marche. 0: La chambre de chauffe est désactivée.
milk_pasteurized	BOOL	M0.5	Etat de l'opération de pasteurisation	<ul style="list-style-type: none"> 1: Le lait a été pasteurisé. 0: Le lait n'est pas encore pasteurisé.

3.2.2 Programmation des conditions pour démarrer la bande transporteuse

3.2.2.1 Interrogation de l'état de l'installation

Introduction

Les étapes suivantes décrivent comment programmer l'état de l'installation comme condition pour le démarrage de la bande transporteuse. La première condition est que la bande transporteuse ne peut être mise en mouvement que lorsque l'installation fonctionne. Pour mettre en oeuvre cette condition, utilisez un contact à fermeture (Page 39) que vous connectez à la variable "ON".

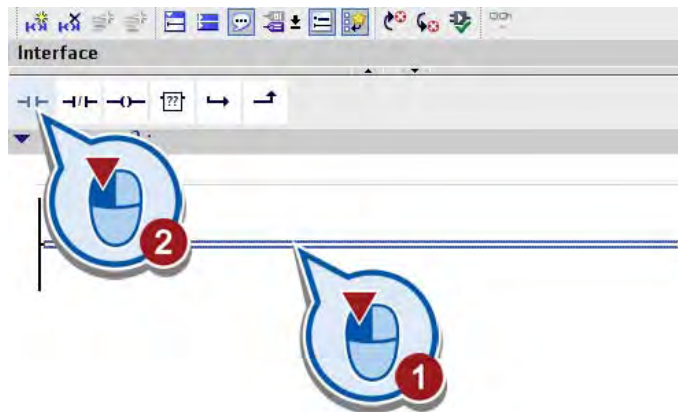
Condition requise

L'interrupteur poussoir de la première partie du projet a déjà été programmé dans le premier réseau du bloc d'organisation "Main [OB1]".

Marche à suivre

Procédez de la manière suivante pour définir l'état de l'installation comme condition pour le démarrage de la bande transporteuse :

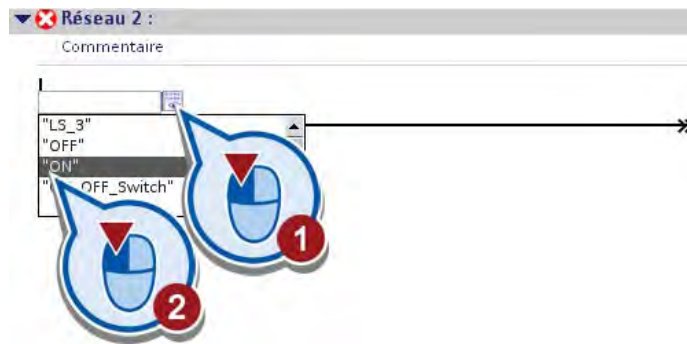
1. Ouvrez le bloc d'organisation "Main [OB1]" dans l'éditeur de programmes.
2. Insérez un contact à fermeture dans le réseau 2.



- 3. Double-cliquez dans l'emplacement réservé à l'opérande au-dessus du contact à fermeture.

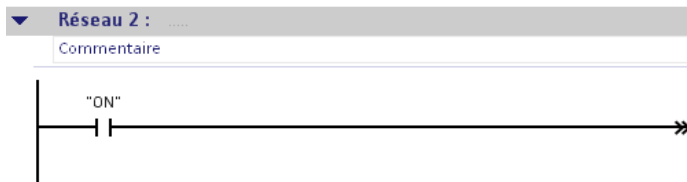


- 4. Connectez le contact à fermeture à la variable "ON".



Résultat

Vous avez inséré un contact à fermeture et l'avez connecté à la variable "ON".



Vous avez ainsi programmé la première condition pour le démarrage de la bande transporteuse. Lorsque l'installation se met en marche, la variable "ON" est mise à l'état logique "1", le contact à fermeture est fermé et le courant est conduit dans le circuit. Dans le paragraphe suivant, vous allez programmer d'autres conditions qui seront mises en oeuvre en fonction de l'état logique de la variable "ON".

3.2.2.2 Interrogation de la position de la bouteille et de l'état de la chambre de chauffe

Introduction

Les étapes suivantes décrivent comment programmer la position de la bouteille et l'état de la chambre de chauffe comme conditions pour le démarrage de la bande transporteuse.

Vous interrogez l'état de la chambre de chauffe à l'aide de l'instruction "Contact à ouverture".

Contact à ouverture

La figure suivante montre le symbole du contact à ouverture dans le programme :

<opérande>

---| / |---

L'activation d'un contact à ouverture dépend de l'état logique de l'opérande correspondant. Lorsque l'opérande fournit l'état logique "1", le contact s'ouvre et le flux de courant vers la barre conductrice droite est interrompu. Dans ce cas, la sortie de l'instruction fournit l'état logique "0".

Lorsque l'opérande fournit l'état logique "0", le contact à ouverture se ferme. Le courant circule à travers le contact à ouverture vers la barre conductrice droite et la sortie de l'instruction est mise à l'état logique "1".

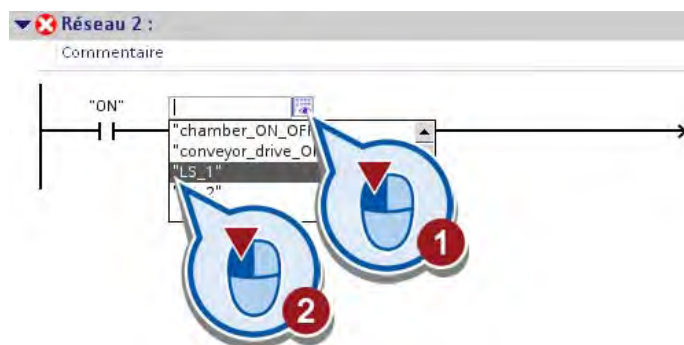
Condition requise

- Le bloc d'organisation "Main [OB1]" est ouvert.
- L'état de l'installation a été programmé comme première condition pour le démarrage de la bande transporteuse (voir Interrogation de l'état de l'installation (Page 101)).

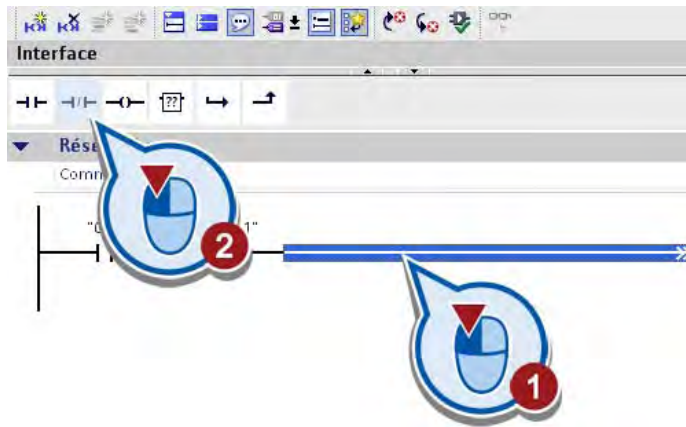
Marche à suivre

Procédez de la manière suivante pour définir la position de la bouteille et l'état de la chambre de chauffe comme conditions pour le démarrage de la bande transporteuse :

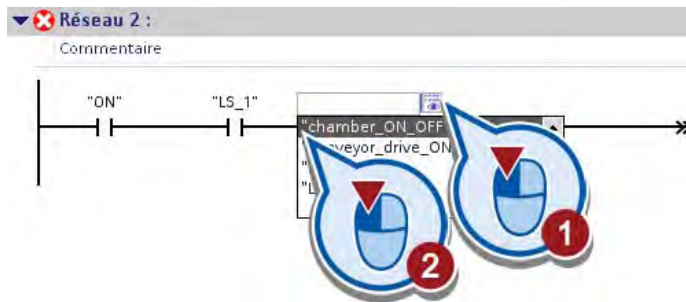
1. Insérez un deuxième contact à fermeture dans le réseau 2.
2. Connectez le contact à fermeture à la variable "LS_1".



3. Insérez un contact à ouverture dans le réseau 2.

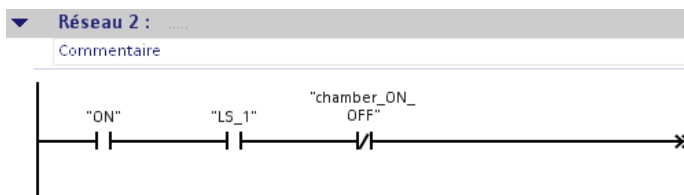


4. Connectez le contact à ouverture à la variable "chamber_ON_OFF".



Résultat

Vous avez programmé la position de la bouteille et l'état de la chambre de chauffe comme conditions pour le démarrage de la bande transporteuse.



3.2.2.3 Interrogation de l'avancement de la pasteurisation

Introduction

Les étapes suivantes décrivent comment commander la bande transporteuse en fonction de l'avancement de la pasteurisation. Lorsque le lait d'une bouteille est pasteurisé, la bande transporteuse se met en mouvement et la bouteille est transportée jusqu'à l'extrémité de la bande transporteuse. L'information sur l'avancement de la pasteurisation est sauvegardée dans la variable "milk_pasteurized" et est interrogée à l'aide d'un contact à fermeture.

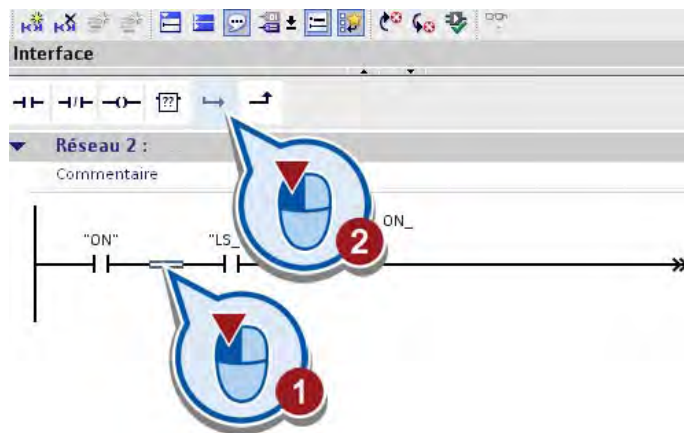
Condition requise

- Le bloc d'organisation "Main [OB1]" est ouvert.
- Les conditions décrites aux paragraphes "Interrogation de l'état de l'installation (Page 101)" et "Interrogation de la position de la bouteille et de l'état de la chambre de chauffe (Page 103)" ont déjà été programmées.

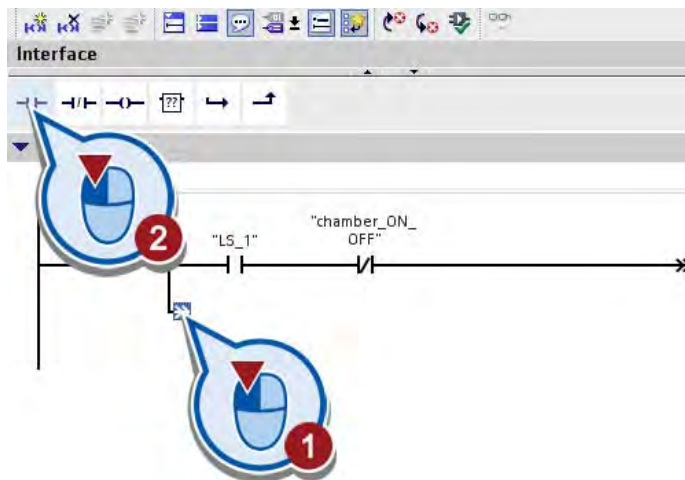
Marche à suivre

Procédez de la manière suivante pour commander la bande transporteuse en fonction de l'avancement de la pasteurisation :

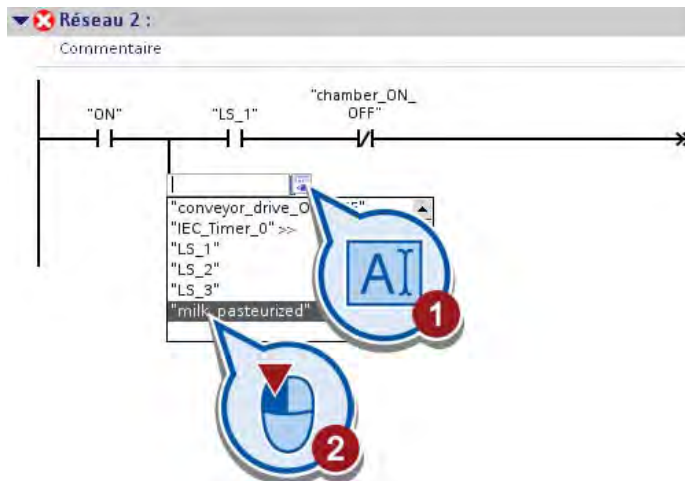
1. Insérez un embranchement dans le circuit entre le premier et le deuxième contact à fermeture.



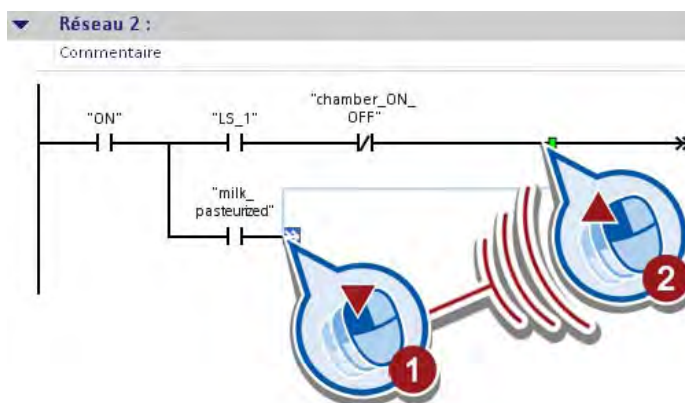
2. Insérez un contact à fermeture dans la branche ouverte.



3. Connectez le contact à fermeture à la variable "milk_pasteurized".

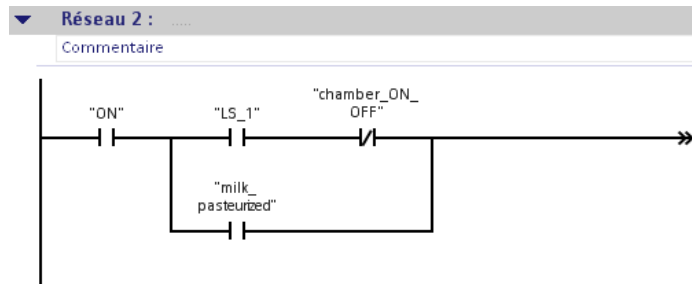


4. Fermez la branche ouverte.



Résultat

Vous avez programmé l'avancement de la pasteurisation comme nouvelle condition pour le démarrage de la bande transporteuse.



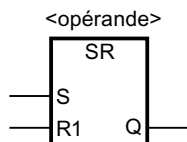
3.2.2.4 Commande de la bande transporteuse

Introduction

Les étapes suivantes décrivent comment mettre la bande transporteuse en mouvement en fonction des conditions programmées. Vous utiliserez l'instruction "Bascule mise à 1/mise à 0" pour commander la bande transporteuse.

Bascule "mise à 1/mise à 0"

La figure suivante montre la boîte de l'instruction "Bascule mise à 1/mise à 0" :



Cette instruction permet de mettre le bit d'un opérande indiqué à "1" ou à "0" en fonction de l'état logique aux entrées S et R1 de l'instruction.

- Si l'état logique est égal à "1" à l'entrée S et est égal à "0" à l'entrée R1, l'opérande indiqué est mis à "1".
- Si l'état logique est égal à "0" à l'entrée S et est égal à "1" à l'entrée R1, l'opérande indiqué est mis à "0", car l'entrée R1 a une priorité supérieure à celle de l'entrée S.
- Si l'état logique est égal à "1" aux deux entrées S et R1, l'état logique de l'opérande indiqué est mis à "0".

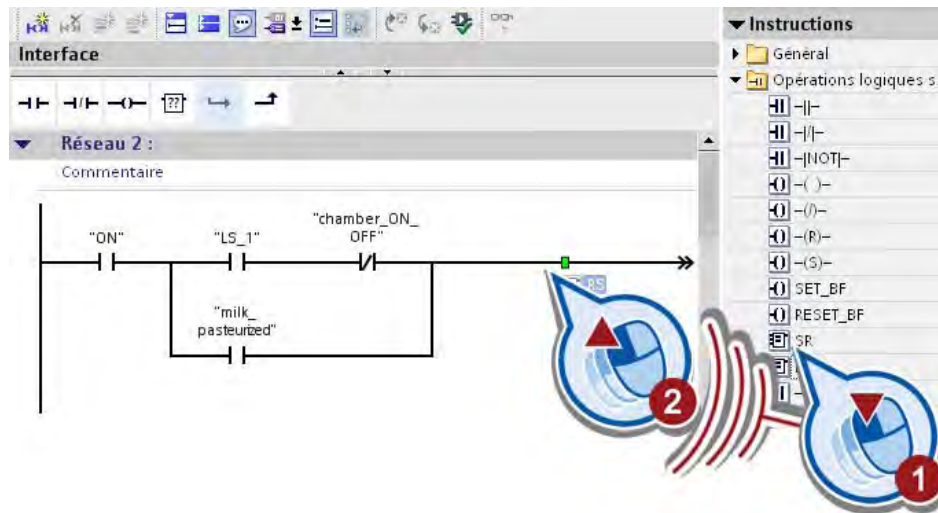
Condition requise

- Le bloc d'organisation "Main [OB1]" est ouvert.
- Les conditions décrites aux paragraphes "Interrogation de l'état de l'installation (Page 101)", "Interrogation de la position de la bouteille et de l'état de la chambre de chauffe (Page 103)" et "Interrogation de l'avancement de la pasteurisation (Page 105)" ont été programmées.

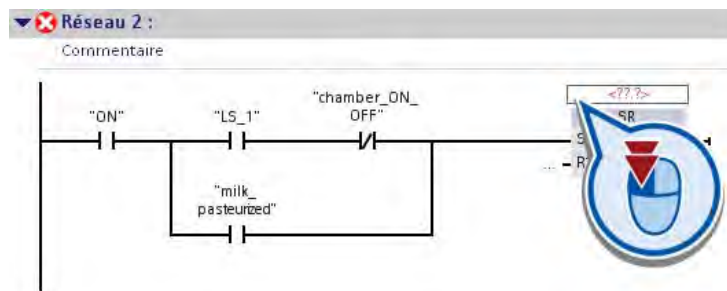
Marche à suivre

Procédez de la manière suivante pour programmer la commande de la bande transporteuse :

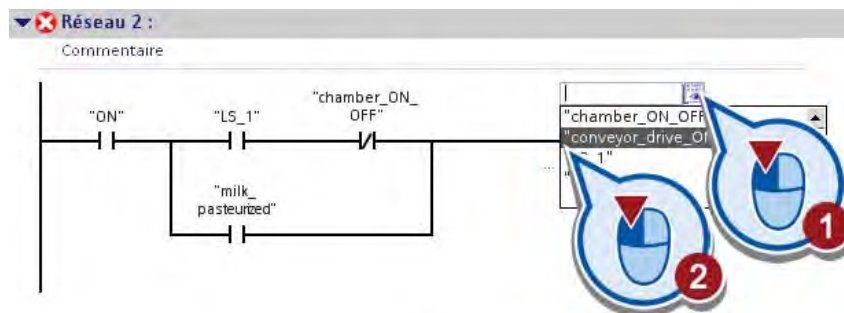
- 1. Insérez l'instruction "Bascule mise à 1/mise à 0" (SR) à la fin du deuxième réseau.



- 2. Double-cliquez dans l'emplacement réservé à l'opérande au-dessus de l'instruction.



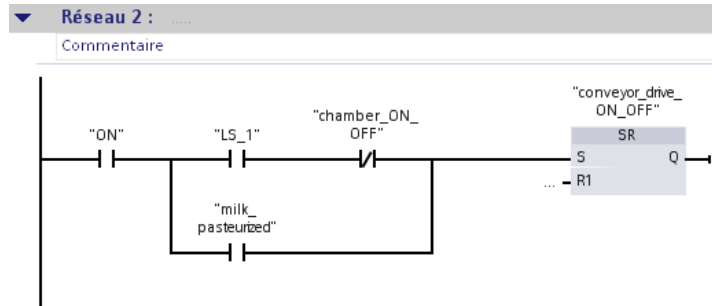
- 3. Connectez l'instruction à la variable "conveyor_drive_ON_OFF".



- 4. Enregistrez le projet à l'aide du bouton "Enregistrer le projet" de la barre d'outils.

Résultat

Vous avez programmé les conditions pour le démarrage de la bande transporteuse.



La bande transporteuse est mise en mouvement lorsque l'état logique de la variable "ON" est "1" et qu'au moins l'une des conditions suivantes est satisfaite :

- Une bouteille est détectée par la barrière photoélectrique (LS1) au début de la bande transporteuse et la chambre de chauffe ne fonctionne pas. Dans ce cas, la variable "LS_1" fournit l'état logique "1" et la variable "chamber_ON_OFF" l'état logique "0".
- La bouteille de lait a été pasteurisée. Dans ce cas, la variable "milk_pasteurized" fournit l'état logique "1". La bouteille est transportée jusqu'à l'extrémité de la bande transporteuse.

Lorsque la bande transporteuse est mise en mouvement, la variable "conveyor_drive_ON_OFF" fournit l'état logique "1".

Dans le paragraphe suivant, vous allez programmer les conditions pour arrêter la bande transporteuse.

3.2.3 Programmation des conditions pour arrêter la bande transporteuse

Introduction

Les étapes suivantes décrivent comment programmer les conditions pour l'arrêt de la bande transporteuse. Vous définissez ces conditions à l'entrée R1 de l'instruction "Bascule mise à 1/mise à 0" que vous avez insérée dans le bloc d'organisation "Main [OB1]" dans le paragraphe précédent.

Vous programmerez les conditions pour arrêter la bande transporteuse à l'aide des instructions suivantes :

- Contact à fermeture
- Contact à ouverture
- Bascule "mise à 1/mise à 0"

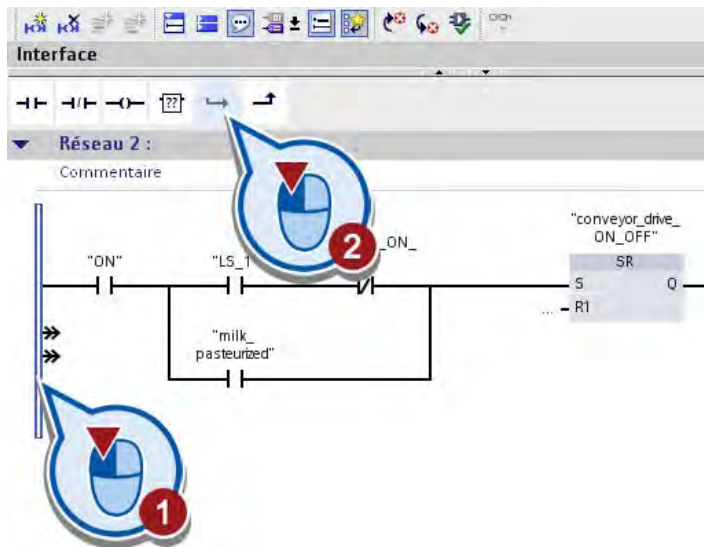
Condition requise

- L'interrupteur poussoir de la première partie du projet a déjà été programmé dans le premier réseau du bloc d'organisation "Main [OB1]".
- Les conditions pour le démarrage de la bande transporteuse ont déjà été programmées dans le bloc d'organisation "Main [OB1]".
Voir aussi : "Programmation des conditions pour démarrer la bande transporteuse (Page 101)"

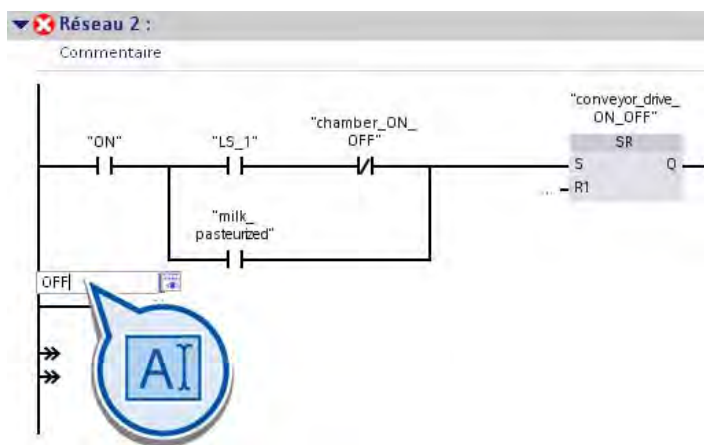
Marche à suivre

Procédez de la manière suivante pour programmer les conditions d'arrêt de la bande transporteuse :

1. Ouvrez le réseau 2 dans le bloc d'organisation "Main [OB1]".
2. Insérez trois nouveaux circuits.

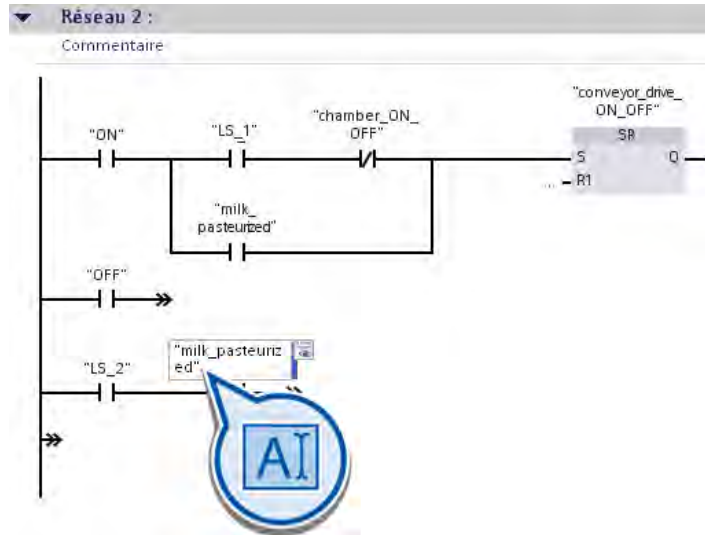


3. Insérez un contact à fermeture dans le premier nouveau circuit et connectez-le à la variable "OFF".



Avec la connexion de la variable "OFF", vous définissez la condition qui demande l'arrêt de la bande transporteuse si l'installation est arrêtée.

4. Insérez un contact à fermeture dans le deuxième nouveau circuit et connectez-le à la variable "LS_2".
5. Insérez un contact à ouverture à la fin du deuxième circuit et connectez-le à la variable "milk_pasteurized".

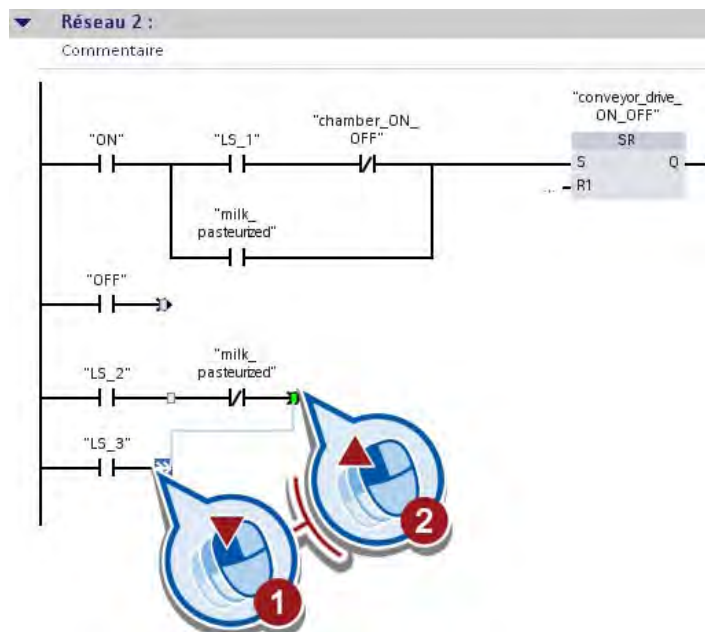


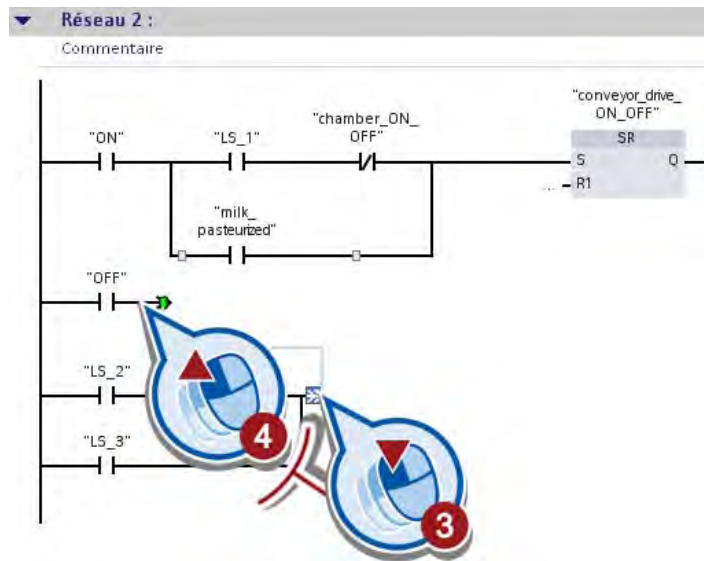
Avec la connexion des variables "LS_2" et "milk_pasteurized", vous définissez la condition qui demande l'arrêt de la bande transporteuse si une bouteille a atteint la chambre de chauffe et que le lait n'a pas encore été pasteurisé.

6. Insérez un contact à fermeture dans le troisième nouveau circuit et connectez-le à la variable "LS_3".

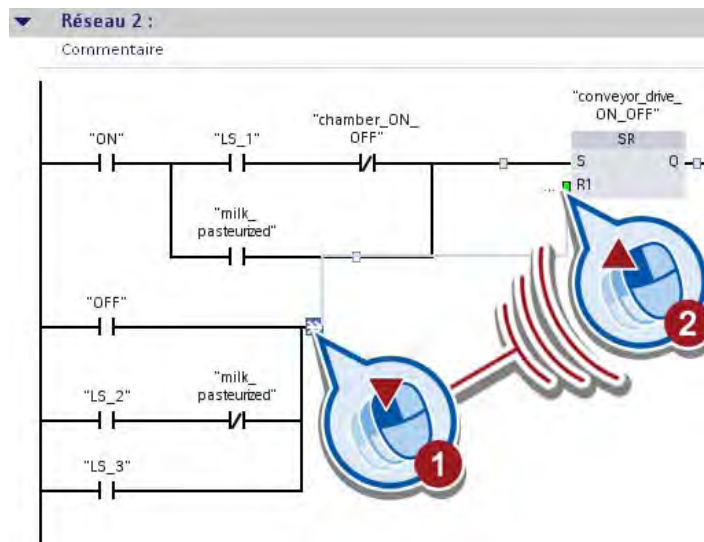
Avec la connexion de la variable "LS_3", vous définissez la condition qui demande l'arrêt de la bande transporteuse si une bouteille a atteint l'extrémité de la bande transporteuse.

7. Reliez les contacts à fermeture insérés dans un montage en parallèle.



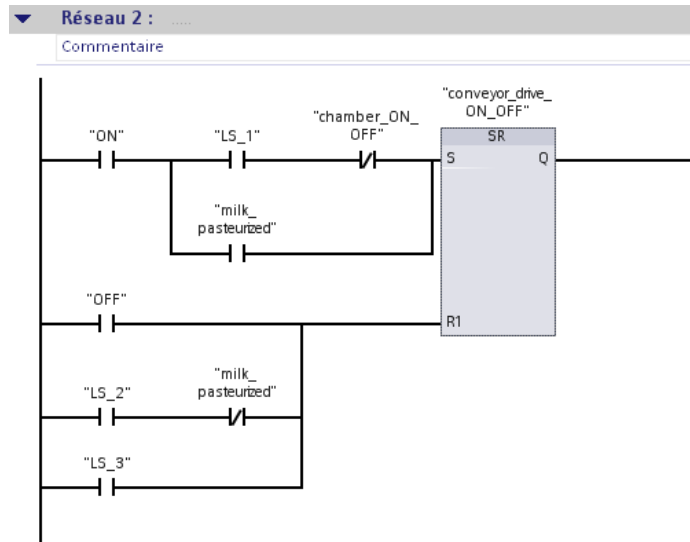


8. Reliez le montage parallèle à l'entrée R1 de l'instruction "Bascule mise à 1/mise à 0".



Résultat

Vous avez programmé la commande de l'entraînement de la bande transporteuse. La bande transporteuse est entraînée ou arrêtée en fonction de la position d'une bouteille de lait sur la bande transporteuse. La position des bouteilles de lait est détectée à l'aide des barrières photoélectriques "LS_1", "LS_2" et "LS_3".



La bande transporteuse s'arrête lorsque l'état logique de la variable "OFF" est "1" ou que l'une des conditions suivantes est satisfaite :

- La bouteille a été détectée par la deuxième barrière photoélectrique (LS2) dans la chambre de chauffe et le lait n'est pas encore pasteurisé. Dans ce cas, la variable "LS_2" fournit l'état logique "1" et la variable "milk_pasteurized" l'état logique "0".
- La bouteille a été détectée par la barrière photoélectrique (LS3) à l'extrémité de la bande transporteuse. Dans ce cas, la variable "LS_3" fournit l'état logique "1".

Lorsque la bande transporteuse s'arrête, la variable "conveyor_drive_ON_OFF" fournit l'état logique "0".

Dans le paragraphe suivant, vous allez programmer l'activation et la désactivation de la chambre de chauffe.

3.2.4 Programmation de la commande du chauffage

Introduction

Les étapes suivantes décrivent comment programmer la commande de l'opération de chauffage. L'opération de chauffage est commandée par l'activation et la désactivation de la chambre de chauffe. Dans le projet-exemple, on suppose pour simplifier que la température de 75°C dans la chambre de chauffe est atteinte dès l'activation.

Vous programmez la commande de la chambre de chauffe dans le troisième réseau du bloc d'organisation "Main [OB1]" à l'aide des instructions suivantes :

- Contact à fermeture
- Bascule "mise à 1/mise à 0"

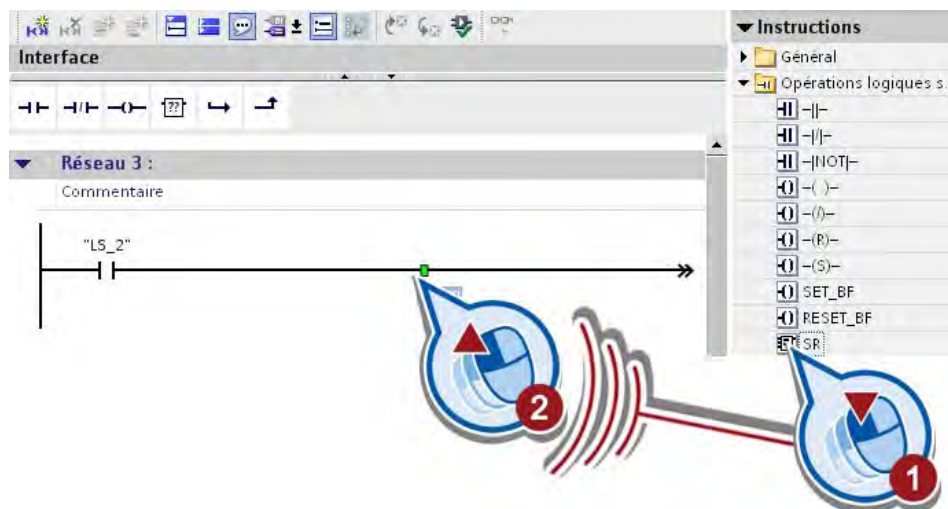
Condition requise

- Le bloc d'organisation "Main [OB1]" est ouvert.
- Les réseaux 1 et 2 du bloc d'organisation "Main [OB1]" ont été programmés.

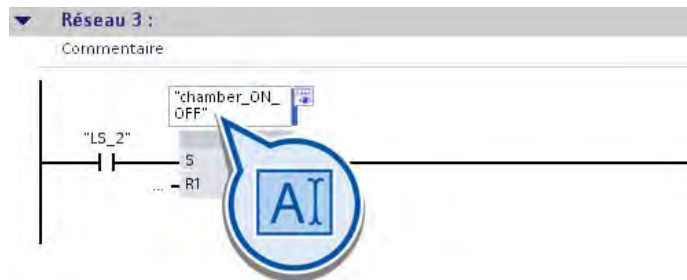
Marche à suivre

Procédez de la manière suivante pour programmer la commande de l'opération de chauffage :

1. Ouvrez le troisième réseau du bloc d'organisation "Main [OB1]".
2. Insérez un contact à fermeture.
3. Connectez le contact à fermeture à la variable "LS_2".
4. Insérez l'instruction "Bascule mise à 1/mise à 0".

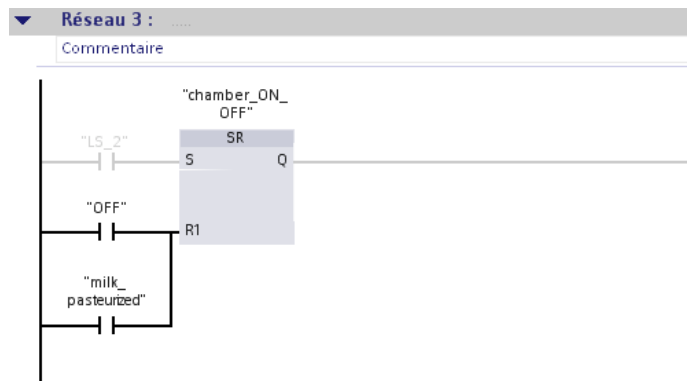


5. Cliquez dans l'emplacement réservé à l'opérande au-dessus de l'instruction "Bascule mise à 1/mise à 0" et connectez l'instruction à la variable "chamber_ON_OFF".



Lorsque la barrière photoélectrique (LS2) détecte une bouteille, la variable "LS_2" fournit l'état logique "1" au niveau de l'entrée S de l'instruction "Bascule mise à 1/mise à 0". La variable "chamber_ON_OFF" est ainsi mise à "1", ce qui active la chambre de chauffe.

6. Programmez les conditions suivantes pour la désactivation de la chambre de chauffe au niveau de l'entrée R1 de l'instruction :

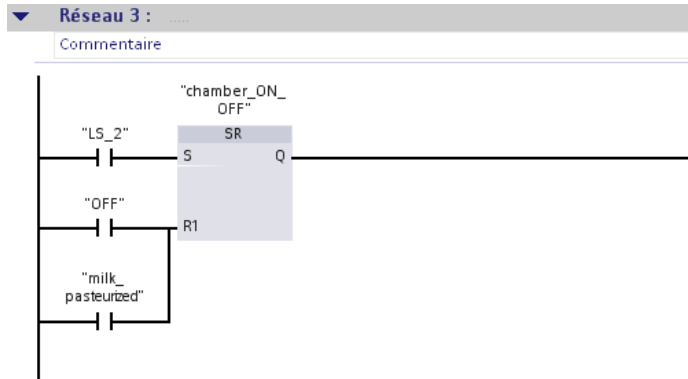


La variable "chamber_ON_OFF" est mise à "0" et le chauffage s'arrête lorsque l'une des conditions suivantes est satisfaite :

- L'installation a été arrêtée. Dans ce cas, la variable "OFF" fournit l'état logique "1".
- Le lait a été pasteurisé. Dans ce cas, la variable "milk_pasteurized" fournit l'état logique "1".

Résultat

Vous avez programmé la commande de l'opération de chauffage.



La chambre de chauffe est activée lorsqu'une bouteille de lait est détectée par la barrière photoélectrique "LS2". Lorsque la durée de chauffage prédéfinie s'est écoulée et que le lait a été pasteurisé, la chambre de chauffe est désactivée et la bouteille est transportée jusqu'à l'extrémité de la bande transporteuse.

Dans le paragraphe suivant, vous allez programmer les paramètres de la durée de chauffage.

3.2.5 Programmation de la durée de chauffage

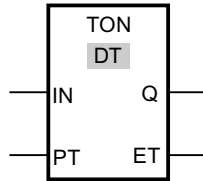
Introduction

Les étapes suivantes décrivent comment programmer la durée de l'opération de chauffage. Pour pasteuriser le lait, les bouteilles sont chauffées pendant 40 secondes dans la chambre de chauffe. Vous programmez ce paramétrage pour la durée de chauffage dans le quatrième réseau du bloc d'organisation "Main [OB1]" à l'aide des instructions suivantes :

- Contact à fermeture
- Retard à la montée
- Bobine de relais, sortie

Retard à la montée

La figure suivante montre la boîte de l'instruction "Retard à la montée" :



Cette instruction permet de mettre un opérande indiqué à l'état logique "1" en ménageant un retard. L'instruction est exécutée lorsque l'état logique à l'entrée IN de l'instruction passe de "0" à "1" (front montant). Le temps de retard (PT) commence à s'écouler au démarrage de l'instruction. Lorsque le temps est écoulé, la sortie Q fournit l'état logique "1".

Bobine de relais, sortie

La figure suivante montre le symbole de l'instruction "Bobine de relais, sortie" dans le programme :

<opérande>

---()---

L'instruction "Bobine de relais, sortie" permet de mettre à "1" le bit d'un opérande indiqué. Lorsque le résultat logique (RLG) à l'entrée de la bobine est égal à "1", l'opérande indiqué est mis à l'état logique "1". Lorsque l'entrée de la bobine est à l'état logique "0", le bit de l'opérande indiqué est mis à "0".

Condition requise

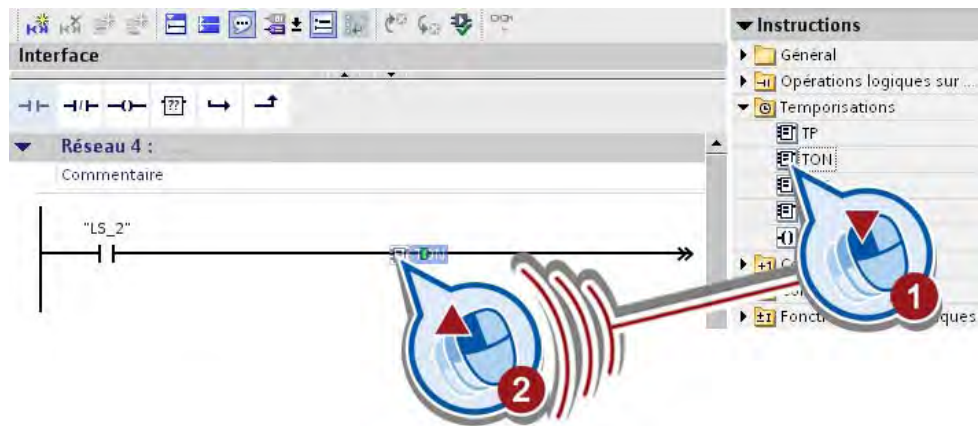
- Le bloc d'organisation "Main [OB1]" est ouvert.
- Les réseaux 1 à 3 du bloc d'organisation "Main [OB1]" ont été programmés.

Marche à suivre

Procédez de la manière suivante pour programmer la durée de l'opération de chauffage :

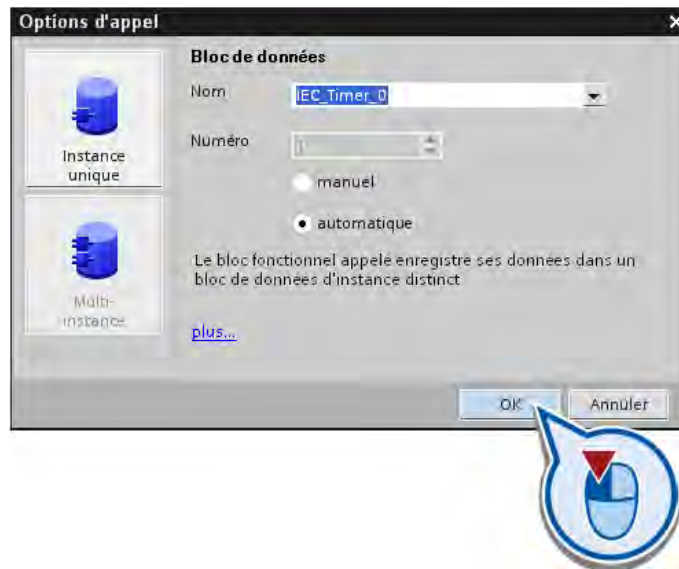
1. Ouvrez le quatrième réseau du bloc d'organisation "Main [OB1]".
2. Insérez un contact à fermeture.
3. Connectez le contact à fermeture à la variable "LS_2".

4. Insérez l'instruction "Retard à la montée" (TON).



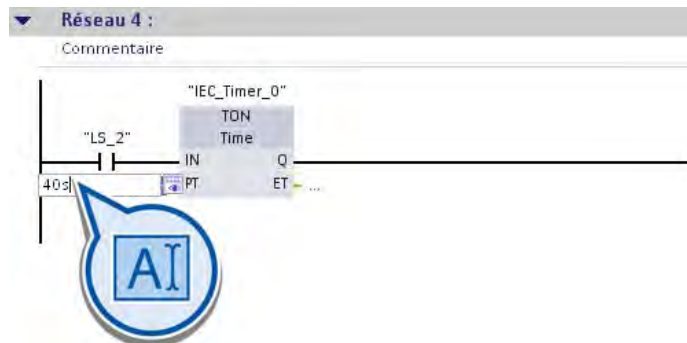
La boîte de dialogue "Options d'appel" pour la création d'un bloc de données s'ouvre.

5. Créez le bloc de données "IEC_Timer_0".



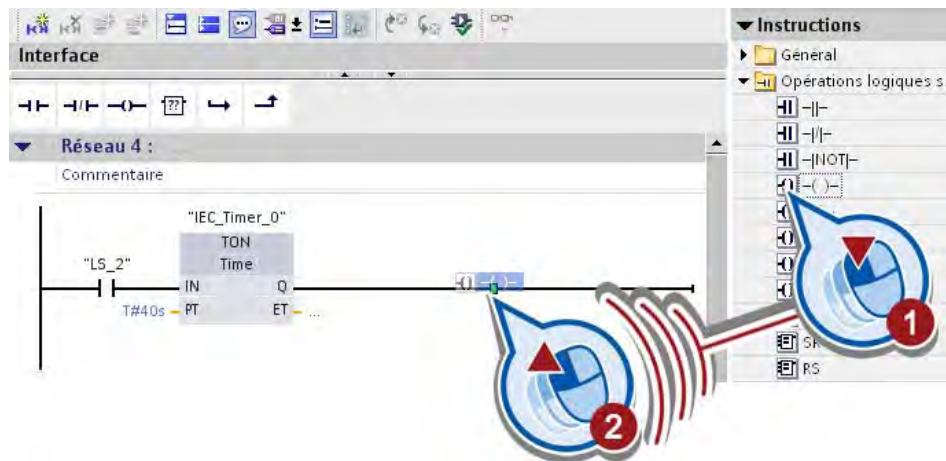
Ce bloc de données "IEC_Timer_0" est créé dans le dossier "Blocs de programme" et est affecté à l'instruction "Retard à la montée". Les données de la temporisation insérée seront sauvegardées dans ce bloc de données.

6. Saisissez la durée de chauffage de 40 secondes au niveau de l'entrée PT de la temporisation.

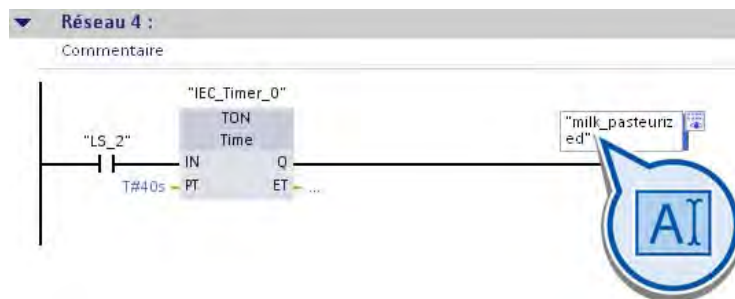


Lorsque la barrière photoélectrique dans la chambre de chauffe détecte une bouteille, l'état logique de la variable "LS_2" passe de "0" à "1". Le temps indiqué à l'entrée PT de la temporisation commence à s'écouler.

7. Insérez l'instruction "Bobine de relais, sortie" à la fin du circuit.



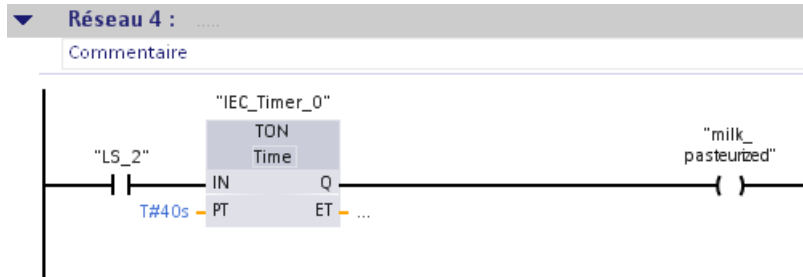
8. Connectez l'instruction "Bobine de relais, sortie" à la variable "milk_pasteurized".



La variable "milk_pasteurized" est mise à l'état logique "1" à l'expiration du temps indiqué au niveau de l'entrée PT.

Résultat

Vous avez programmé la durée de l'opération de pasteurisation.



Lorsqu'une bouteille de lait active la barrière photoélectrique "LS_2", la chambre de chauffe est activée et le temps de pasteurisation est démarré. La variable milk_pasteurized est mise à l'état logique "1" à l'expiration de ce temps. Cela met la bande transporteuse en mouvement et la bouteille de lait est transportée jusqu'à l'extrémité de la bande transporteuse.

Dans le paragraphe suivant, vous allez programmer un voyant d'état signalant l'état de fonctionnement de la chambre de chauffe.

3.2.6 Programmation du voyant d'état

Introduction

Les étapes suivantes décrivent comment programmer un voyant d'état. Le voyant d'état montre les états de fonctionnement de la chambre de chauffe. Lorsque la chambre de chauffe fonctionne, la variable "LED" est mise à l'état logique "1" et la vue sur le pupitre opérateur est activée. Lorsque la chambre de chauffe ne fonctionne pas, la variable "LED" fournit l'état logique "0" et la vue sur le pupitre opérateur est désactivée. Vous programmerez l'affichage sur le pupitre opérateur dans le paragraphe suivant.

Vous programmez le voyant d'état dans le cinquième réseau du bloc d'organisation "Main [OB1]" à l'aide des instructions suivantes :

- Contact à fermeture
- Bobine de relais, sortie

Condition requise

- Le bloc d'organisation "Main [OB1]" est ouvert.
- Les réseaux 1 à 4 du bloc d'organisation "Main [OB1]" ont été programmés.

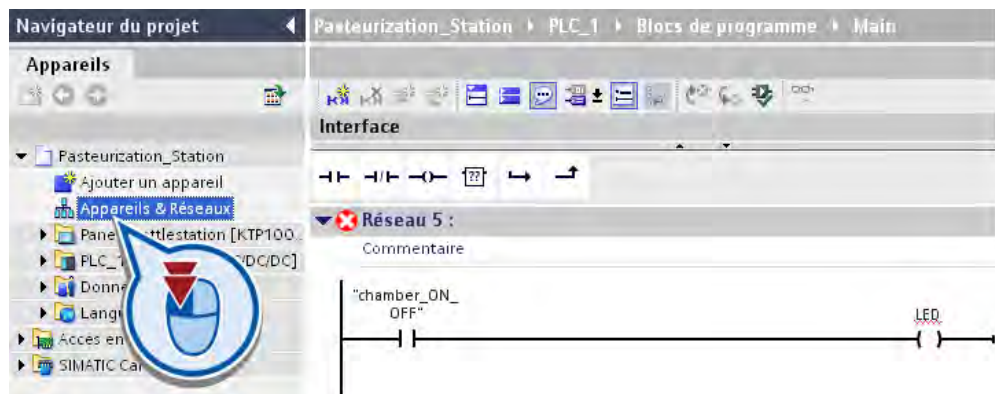
Marche à suivre

Procédez de la manière suivante pour programmer le voyant d'état signalant les états de fonctionnement de la chambre de chauffe :

1. Insérez un contact à fermeture dans le cinquième réseau du bloc d'organisation "Main [OB1]".
2. Connectez le contact à fermeture à la variable "chamber_ON_OFF".



3. Insérez l'instruction "Bobine de relais, sortie" à la fin du circuit.
4. Dans l'emplacement réservé à l'opérande de l'instruction "Bobine de relais, sortie", saisissez le nom "LED" et confirmez votre saisie avec la touche Entrée.
5. Ouvrez l'éditeur de matériels et de réseaux.

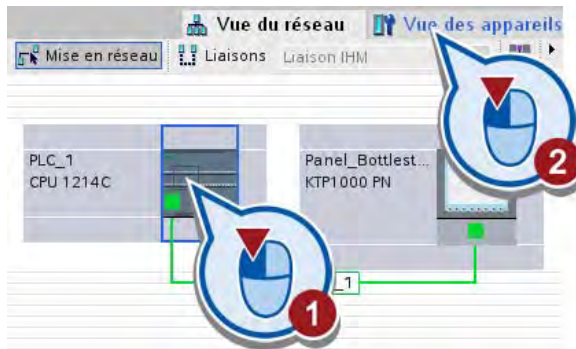


Remarque

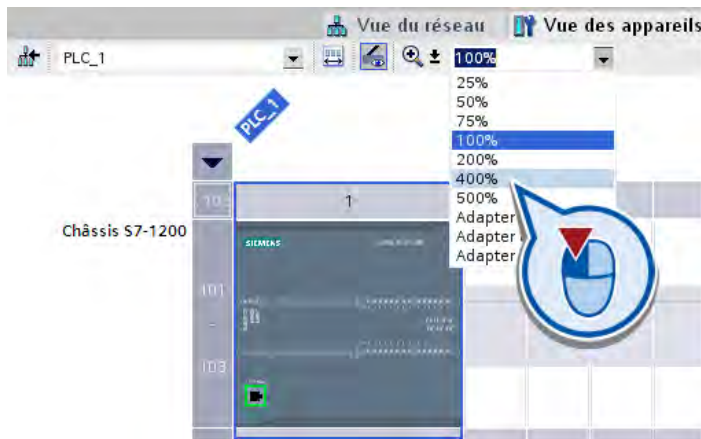
Réseau incomplet

Dans le réseau 5 s'affiche une croix blanche dans un cercle rouge. Ce symbole indique que la connexion ou la définition des variables n'est pas encore achevée dans ce réseau.

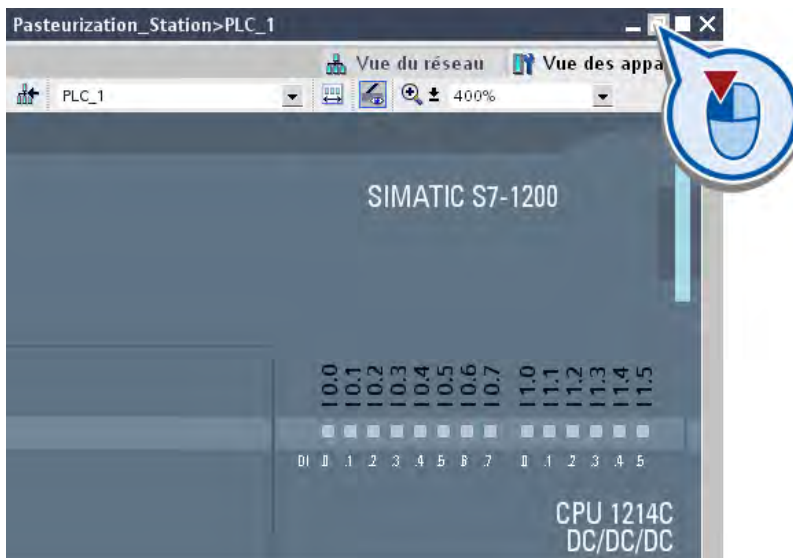
- 6. Affichez la vue des appareils.



- 7. Agrandissez la vue des appareils de manière à rendre bien visibles les entrées et sorties de l'automate.

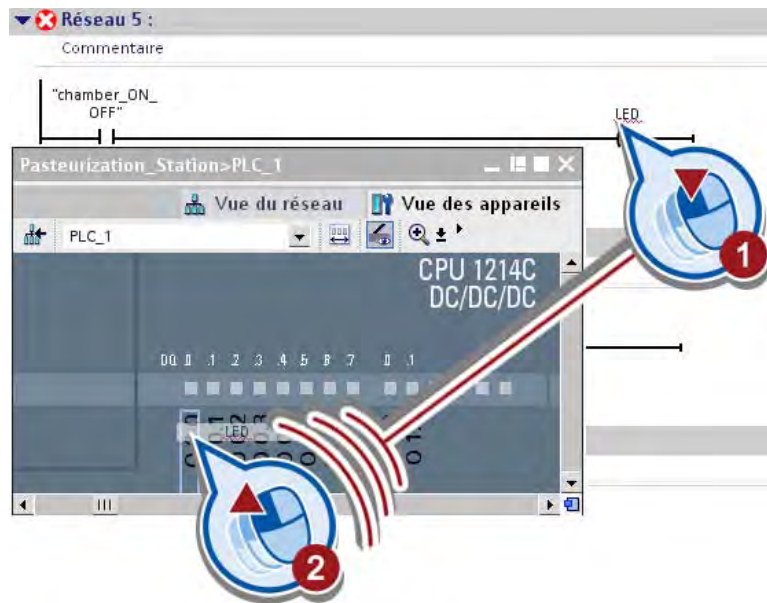


- 8. Décrochez la fenêtre de l'éditeur de matériels et de réseaux et positionnez-la à côté de l'éditeur de programmes.



- 9. Basculez dans le cinquième réseau du bloc d'organisation "Main [OB1]".

10. Associez la variable "LED" à la sortie "Q 0.0" de l'automate.



La variable "LED" est définie et affichée sur la sortie "Q 0.0" de l'automate. Les variables précédemment définies dans la table des variables API sont affichées au niveau des sorties de l'automate.

Remarque

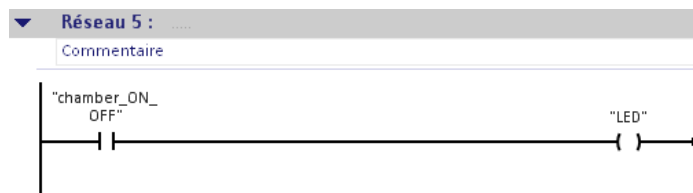
Définition des variables

Il est indifférent que vous définissiez les variables au niveau des entrées et sorties dans la vue des appareils, la table des variables API ou les réseaux. Les variables déjà définies sont également affichées dans la vue des appareils.

11. Enregistrez le projet à l'aide du bouton "Enregistrer le projet" de la barre d'outils.

Résultat

Vous avez programmé un voyant d'état pour l'affichage des états de travail de la chambre de chauffe.



Lorsque la chambre de chauffe est activée, la variable "chamber_ON_OFF" fournit l'état logique "1". La variable "LED" est mise à l'état logique "1" par l'instruction "Bobine de relais, sortie". La variable "LED" fournit l'état logique "1" tant que la chambre de chauffe fonctionne. Si la chambre de chauffe est désactivée, la valeur des variables "chamber_ON_OFF" et "LED" devient "0".

Dans les paragraphes suivants, vous allez tester le programme créé en visualisant l'état du programme.

3.3 Test du programme complété avec la visualisation de l'état du programme

Introduction

Les étapes suivantes décrivent comment tester le programme créé en visualisant l'état du programme.

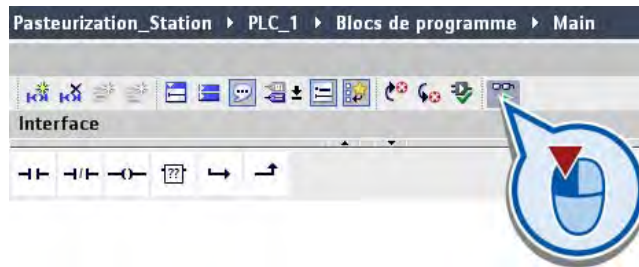
Condition requise

- L'automate est configuré.
- Aucune tension n'est appliquée aux entrées et sorties de l'automate, car les valeurs forcées sont écrasées par le module en mode en ligne.
- Le bloc d'organisation "Main [OB1]" est ouvert dans l'éditeur de programmes.

Marche à suivre

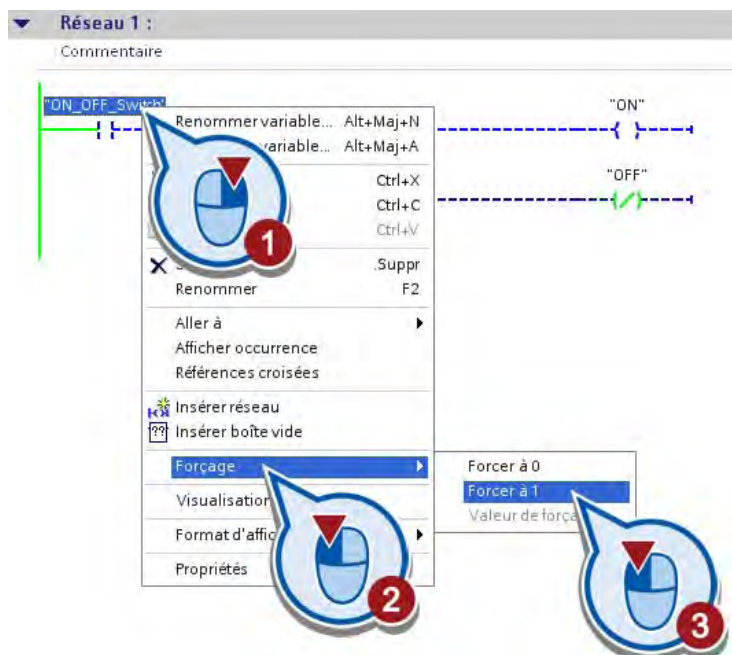
Procédez de la manière suivante pour tester le programme créé avec la visualisation d'état du programme :

1. Chargez le programme dans l'automate et activez la liaison en ligne. Pour plus d'informations, reportez-vous au paragraphe "Chargement du programme dans le système cible (Page 51)".
2. Cliquez sur le bouton "Activer/désactiver visualisation du programme" dans la barre d'outils de l'éditeur de programmes.

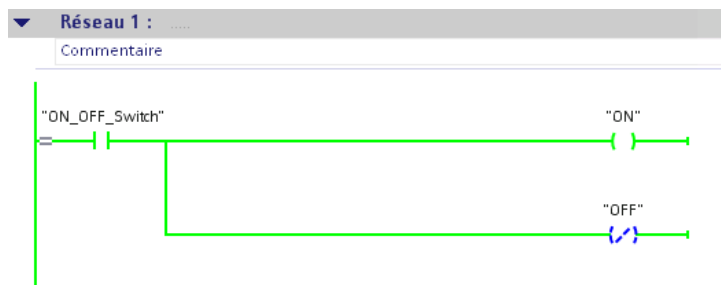


3.3 Test du programme complété avec la visualisation de l'état du programme

3. Dans le réseau 1, forcez la variable "ON_OFF_Switch" à "1".



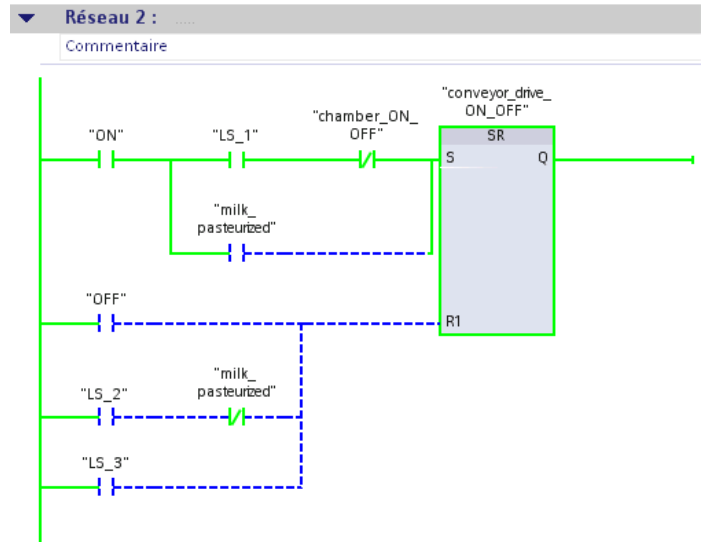
La variable "ON_OFF_Switch" est mise à l'état logique "1". Le courant traverse le contact à fermeture et atteint les bobines à la fin du réseau. La variable "ON" est mise à "1", ce qui met l'installation en marche. La variable OFF reste à l'état logique "0" et n'a donc aucun effet.



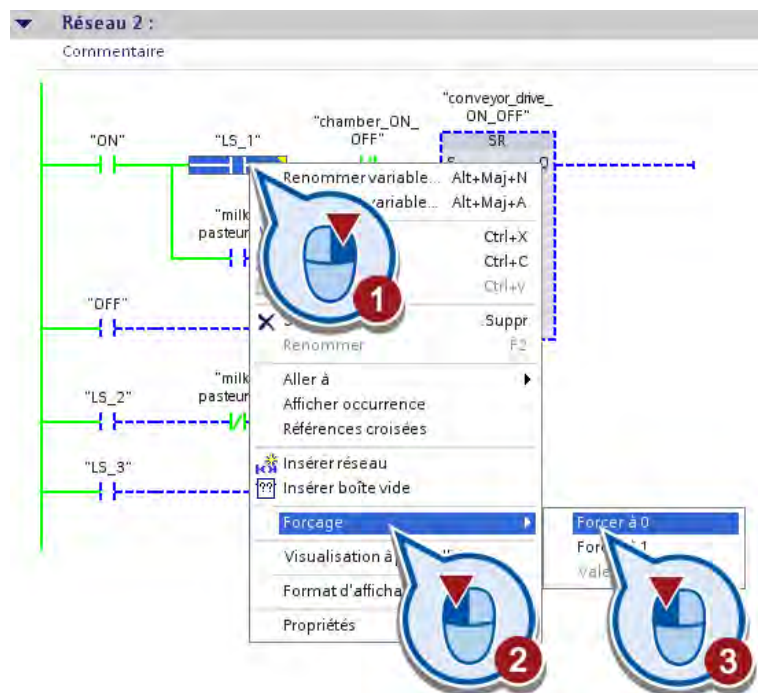
4. Dans le réseau 2, forcez la variable "LS_1" à "1".

La variable "LS_1" est mise à l'état logique "1". Cela simule la détection d'une bouteille par la barrière photoélectrique "LS1" au début de la bande transporteuse. La variable "chamber_ON_OFF" fournit l'état logique "0", car la chambre de chauffe est toujours désactivée lors de la simulation.

Le courant traverse les contacts du circuit principal et donne l'état logique "1" au niveau de l'entrée S de l'instruction "Bascule mise à 1/mise à 0". Cela provoque la mise à "1" de la variable conveyor_drive_ON_OFF et la bande transporteuse se met en mouvement.



5. Dans le réseau 2, forcez la variable "LS_1" à "0".



La variable "LS_1" est mise à l'état logique "0". Cela simule la désactivation de la barrière photoélectrique "LS1".

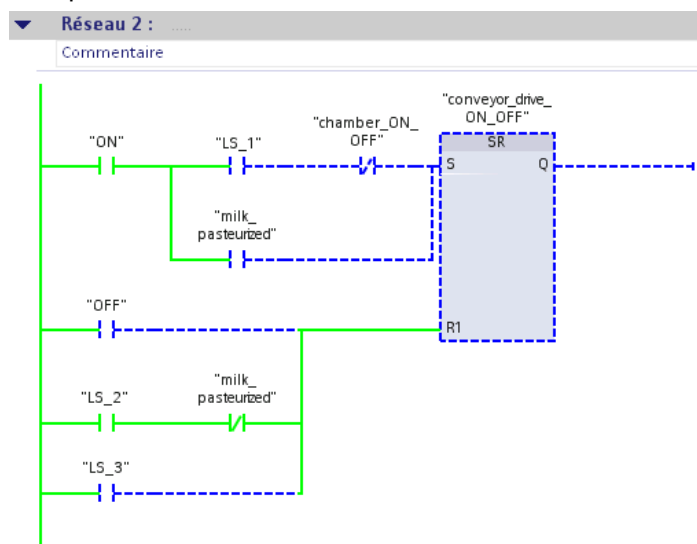
3.3 Test du programme complété avec la visualisation de l'état du programme

6. Dans le réseau 2, forcez la variable "LS_2" à "1".

Dans le réseau 2 :

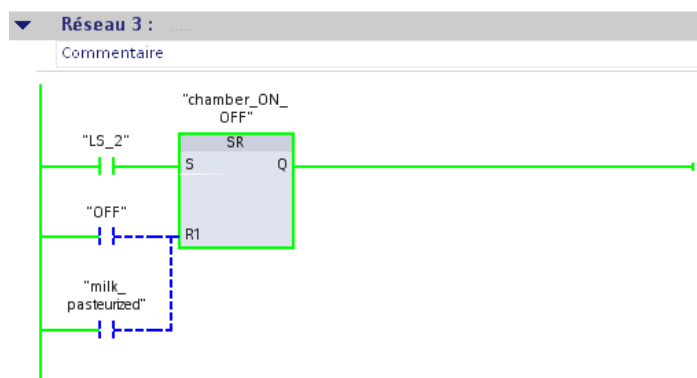
La variable "LS_2" est mise à l'état logique "1". Cela simule le transport de la bouteille dans la chambre de chauffe et l'activation de la barrière photoélectrique "LS2".

Le flux de courant est dévié vers l'entrée R1 de l'instruction "Bascule mise à 1/mise à 0". Cela met à "0" la variable "conveyor_drive_ON_OFF", ce qui arrête la bande transporteuse.



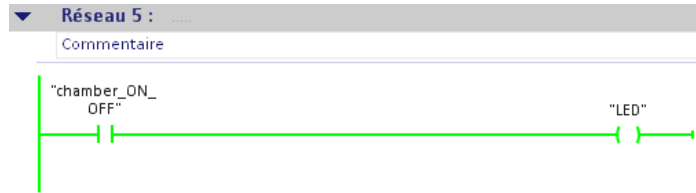
Dans le réseau 3 :

Le courant traverse le contact du circuit principal vers l'entrée S de l'instruction "Bascule mise à 1/mise à 0". Cela provoque la mise à l'état logique "1" de la variable chamber_ON_OFF, ce qui active la chambre de chauffe.



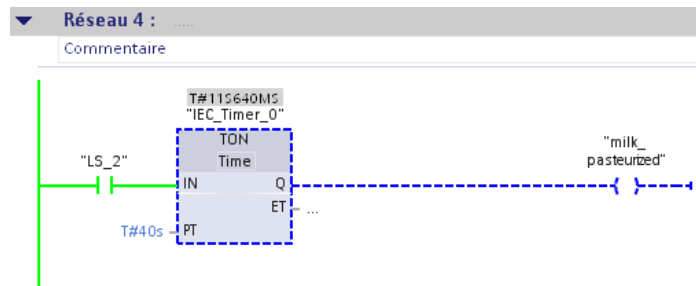
Dans le réseau 5 :

Dès que la chambre de chauffe fonctionne (réseau 3), la variable "LED" est mise à l'état logique "1" et le voyant d'état est activé.

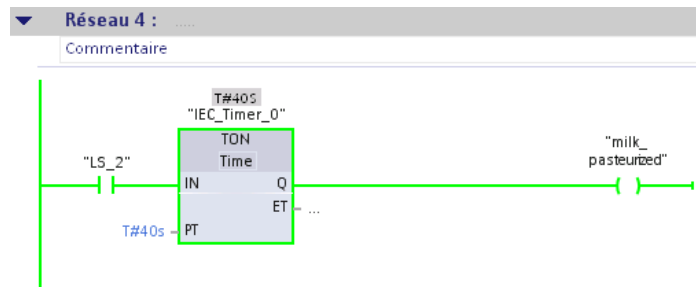


Dans le réseau 4 :

Un front montant est détecté au niveau de l'entrée IN de la temporisation "Retard à la montée" et la temporisation pour la pasteurisation est démarrée.

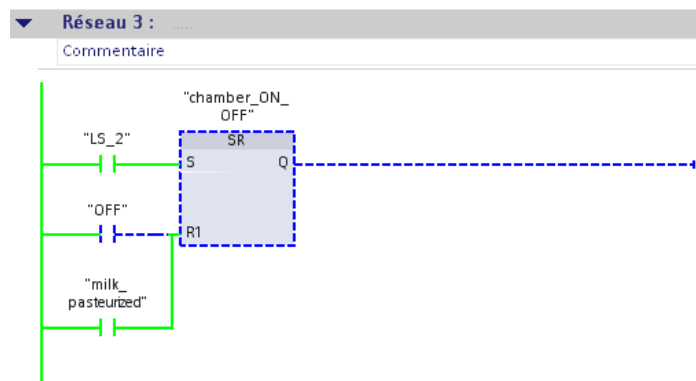


Une fois la temporisation écoulée, le lait est pasteurisé et la variable "milk_pasteurized" est mise à l'état logique "1".



Dans le réseau 3 :

La mise à "1" de la variable "milk_pasteurized" provoque la désactivation de la chambre de chauffe.



3.3 Test du programme complété avec la visualisation de l'état du programme

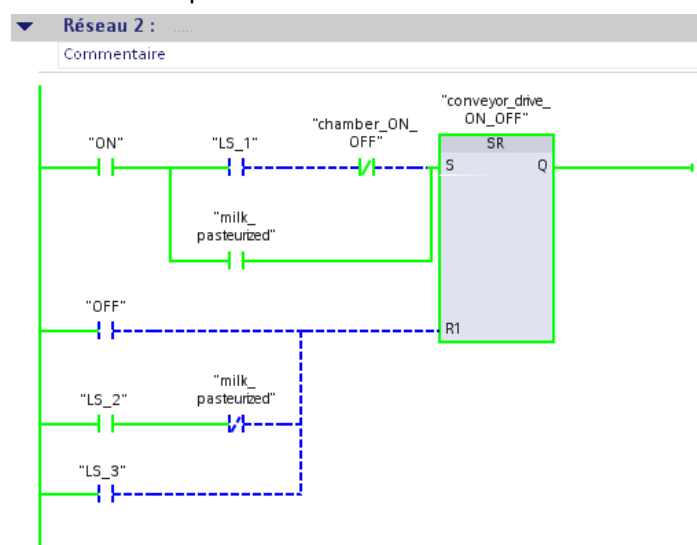
Dans le réseau 5 :

Lorsque la chambre de chauffe s'arrête, le voyant d'état est désactivé.



Dans le réseau 2 :

La bande transporteuse est remise en mouvement.



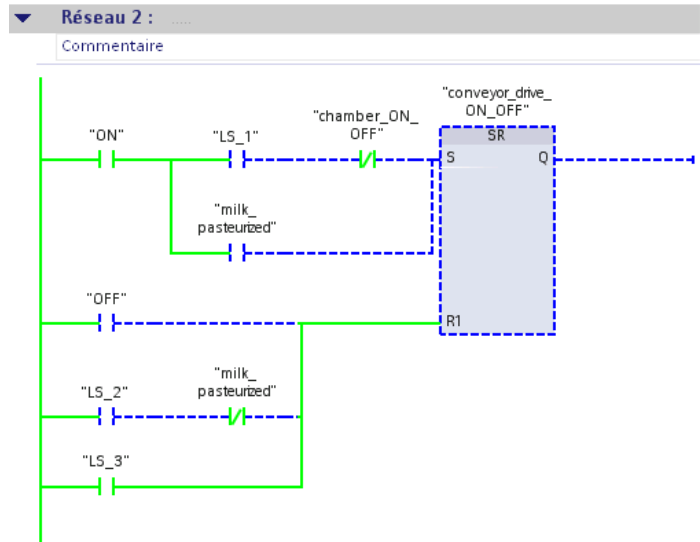
7. Dans le réseau 2, forcez la variable "LS_2" à "0".

La variable "LS_2" est mise à l'état logique "0". Cela simule la sortie de la bouteille de la chambre de chauffe.

8. Dans le réseau 2, forcez la variable "LS_3" à "1".

La variable "LS_3" est mise à l'état logique "1". Cela simule le transport de la bouteille jusqu'à l'extrémité de la bande transporteuse et sa détection par la barrière photoélectrique "LS3".

La variable "conveyor_drive_ON_OFF" est mise à l'état logique "0" et la bande transporteuse s'arrête.



9. Coupez la liaison en ligne.

Résultat

Vous avez testé le programme et vérifié le déroulement de ses opérations.

Dans le paragraphe suivant, vous allez compléter la vue IHM avec d'autres objets graphiques. Cela permettra de visualiser le déroulement des opérations du programme complété.

3.4 Compléter la vue IHM

3.4.1 Objet graphique "Chambre de chauffe"

Introduction

Les étapes suivantes décrivent comment compléter la vue IHM avec la représentation graphique d'une chambre de chauffe. Vous utilisez des objets graphiques statiques simples pour la représentation de la chambre de chauffe.

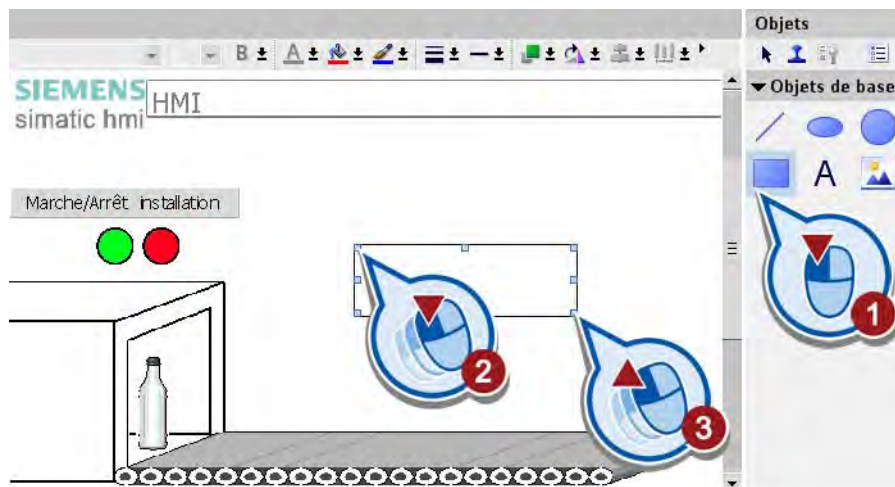
Condition requise

- La vue IHM du chapitre "Exemple simple" existe déjà ou le projet "Simple Example (Page 18)" a été chargé.
- La vue IHM créée sous "Panel_Bottlestation" est ouverte.

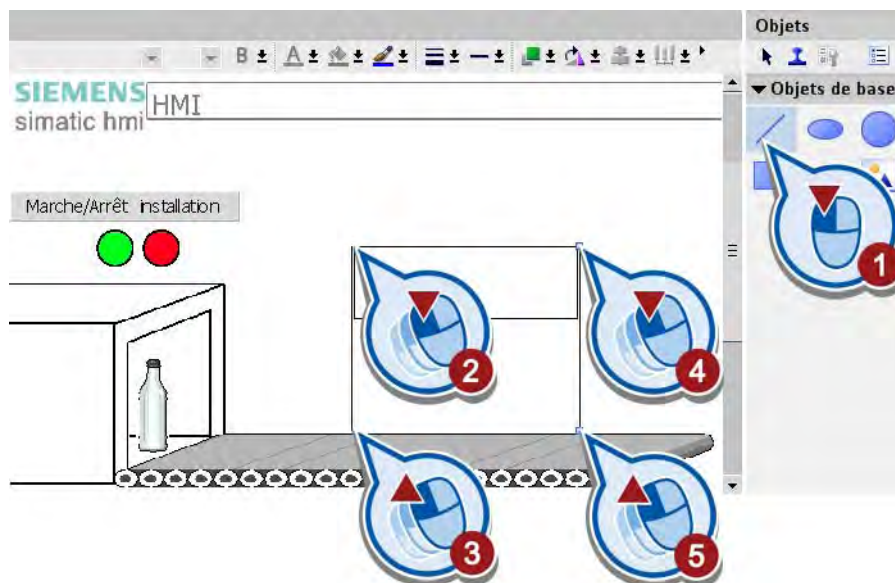
Marche à suivre

Procédez de la manière suivante pour compléter la vue IHM avec la représentation d'une chambre de chauffe :

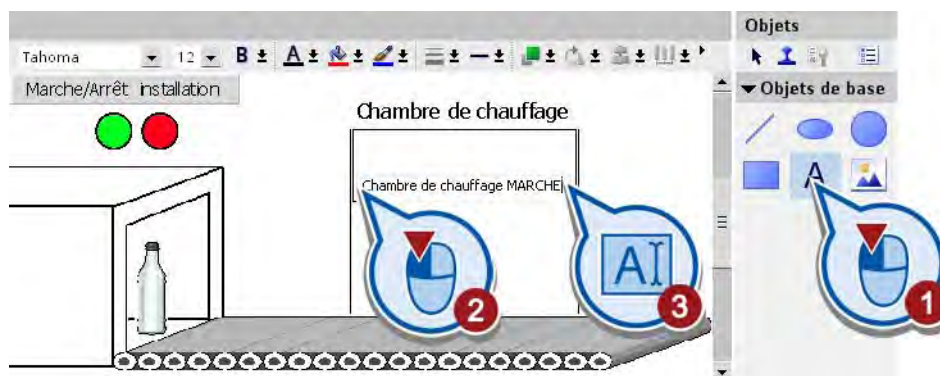
1. Dessinez un rectangle au-dessus de la bande transporteuse au milieu de la vue.



- Tracez une ligne verticale à gauche et à droite du rectangle.



- Créez deux zones de texte "Chambre de chauffe" et "Chambre de chauffe activée".



- Adaptez la mise en forme du texte sous "Général" et "Représentation" dans la fenêtre d'inspection.

Résultat

Vous avez complété la vue IHM avec les éléments graphiques statiques de la chambre de chauffe.

3.4.2 Objet graphique "DEL de la chambre de chauffe"

Introduction

Les étapes suivantes décrivent comment compléter la vue IHM avec la représentation graphique d'une DEL et comment animer cette DEL avec la variable API "LED".

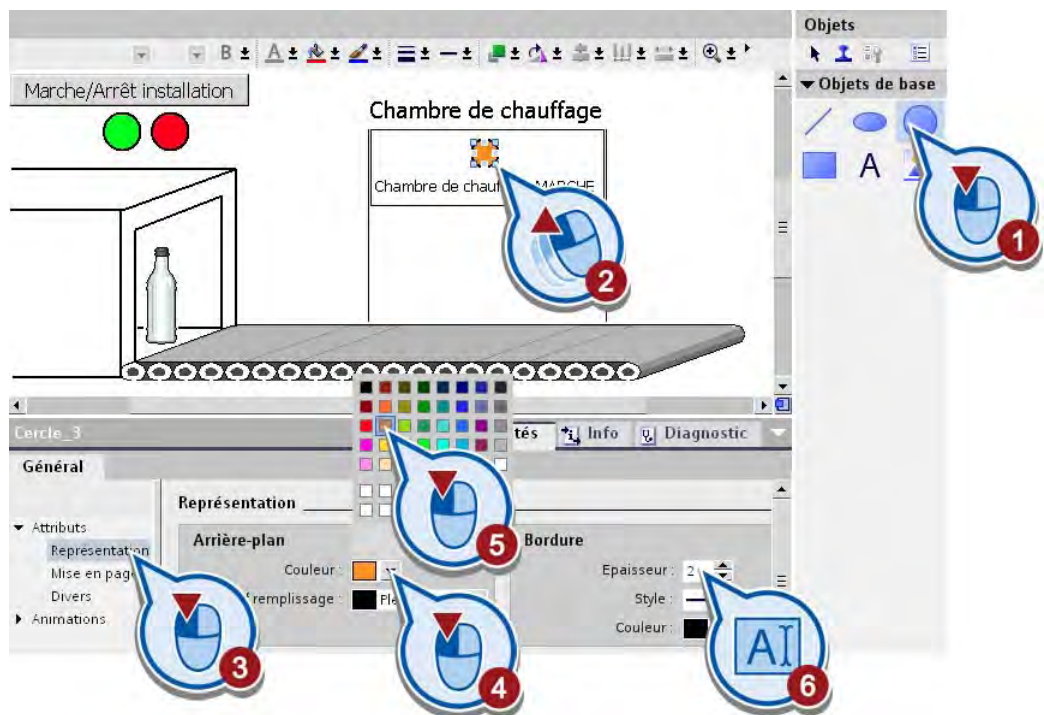
Condition requise

- Le programme a été créé.
- La vue IHM est ouverte.

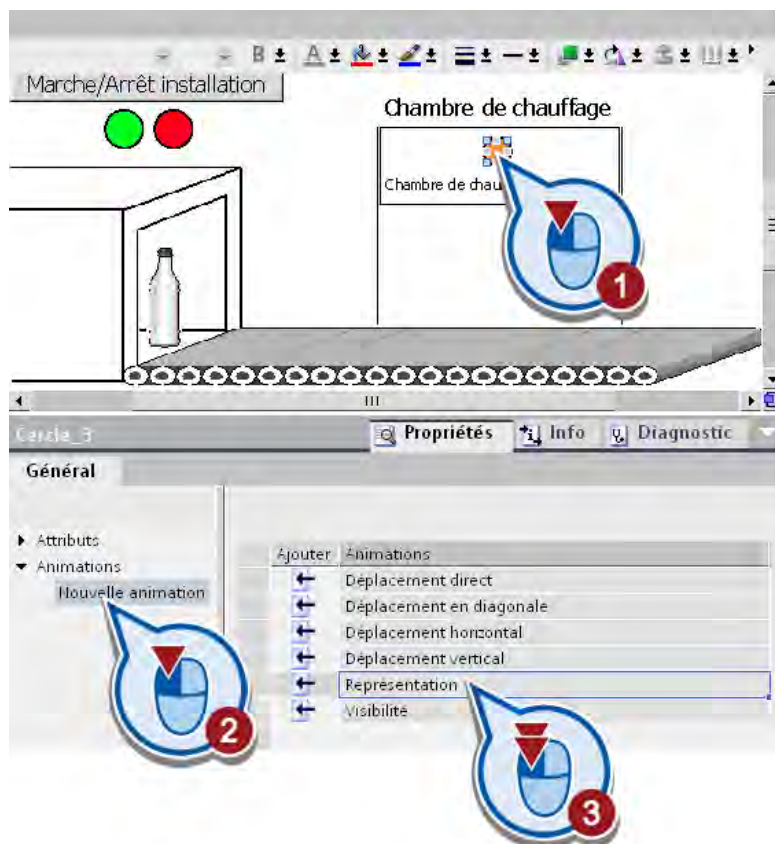
Marche à suivre

Procédez de la manière suivante pour créer et animer la DEL pour la chambre de chauffe :

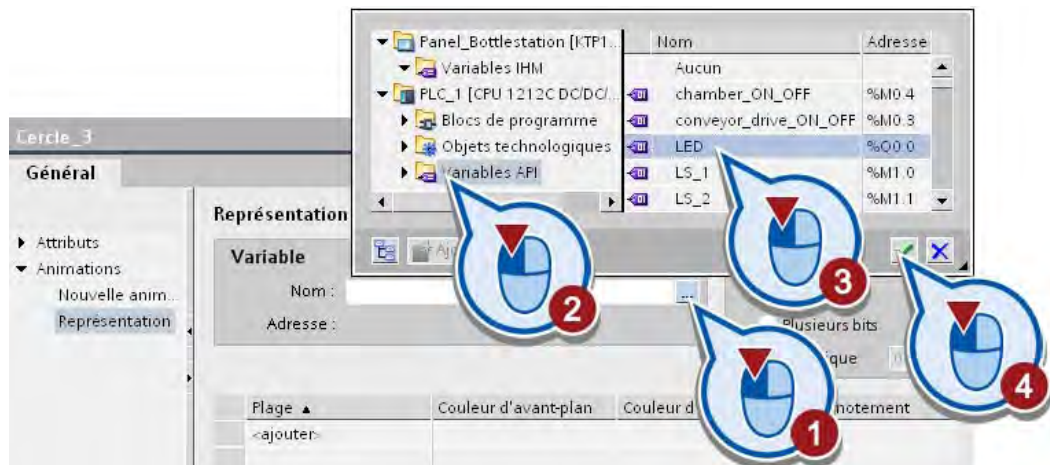
1. Dans le rectangle, insérez un cercle orange représentant la DEL de la chambre de chauffe en le centrant au-dessus de la zone de texte "Chambre de chauffe activée".



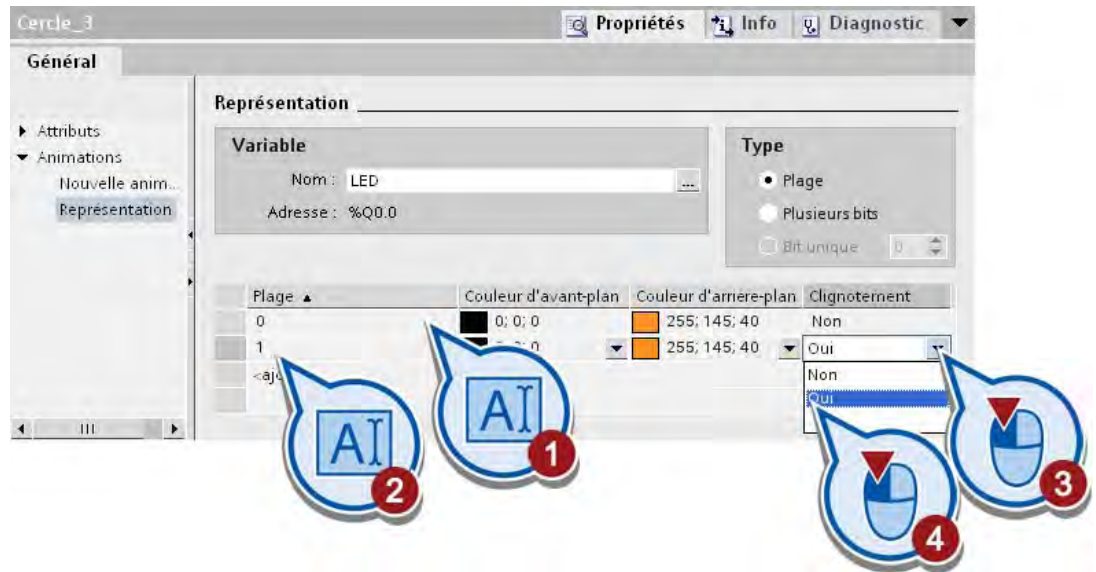
2. Créez une nouvelle animation pour la DEL de la chambre de chauffe.



3. Associez l'animation à la variable "LED".



4. Activez "Clignotement" pour la plage de valeurs "1" de la variable.



Résultat

Vous avez complété la vue IHM avec une DEL. Lorsque la variable "LED" prend la valeur "1", l'activation de la chambre de chauffe est signalée par le clignotement dans la vue IHM de la DEL créée.

Dans le paragraphe suivant, vous allez compléter la vue IHM avec la représentation graphique des barrières photoélectriques.

3.4.3 Objets graphiques "Barrières photoélectriques"

Introduction

Les étapes suivantes décrivent comment compléter la vue IHM avec les barrières photoélectriques "LS1", "LS2" et "LS3". Vous utiliserez l'objet élémentaire "Cercle" pour représenter les barrières photoélectriques. Vous associerez les barrières photoélectriques que vous avez créées aux variables du programme et vous animerez la représentation.

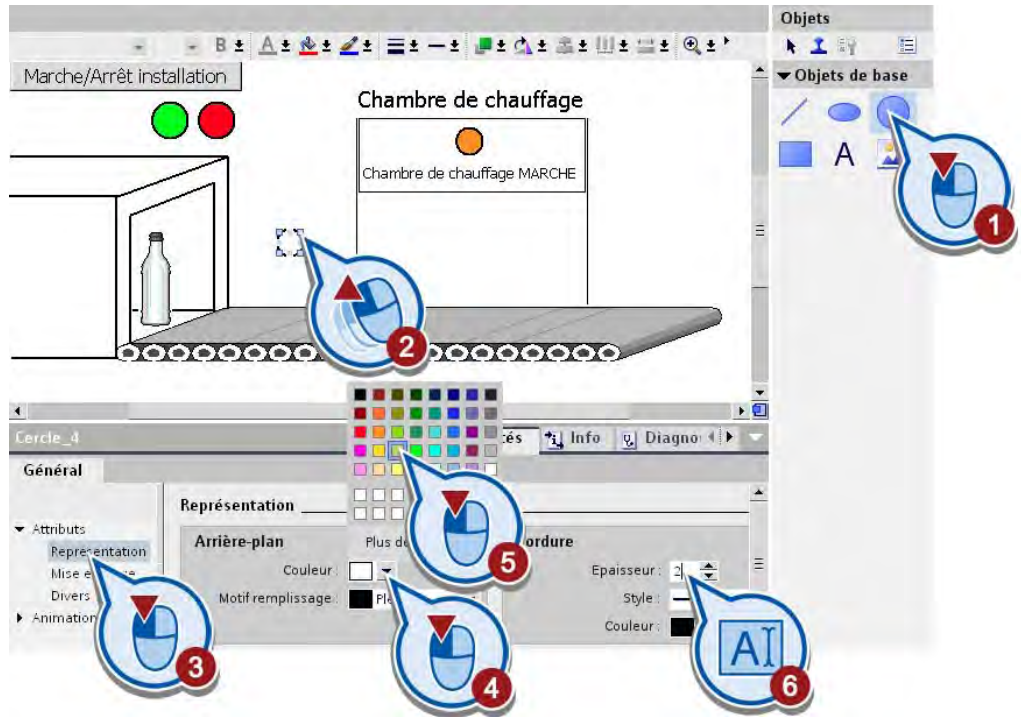
Condition requise

- Le programme de commande existe.
- La vue IHM est ouverte.

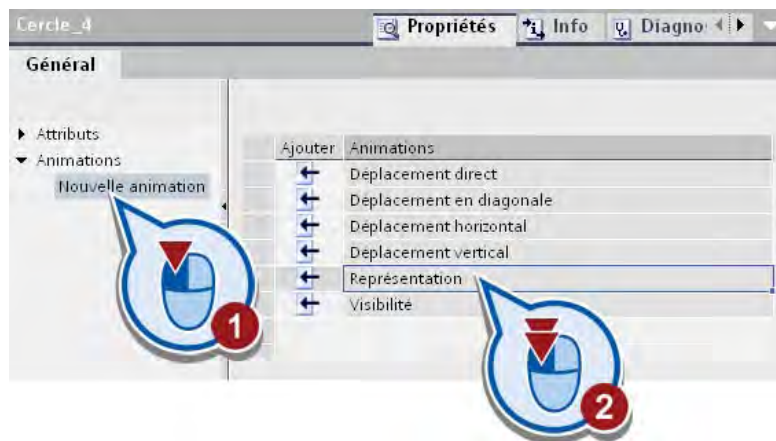
Marche à suivre

Procédez de la manière suivante pour insérer la représentation des barrières photoélectriques dans la vue IHM :

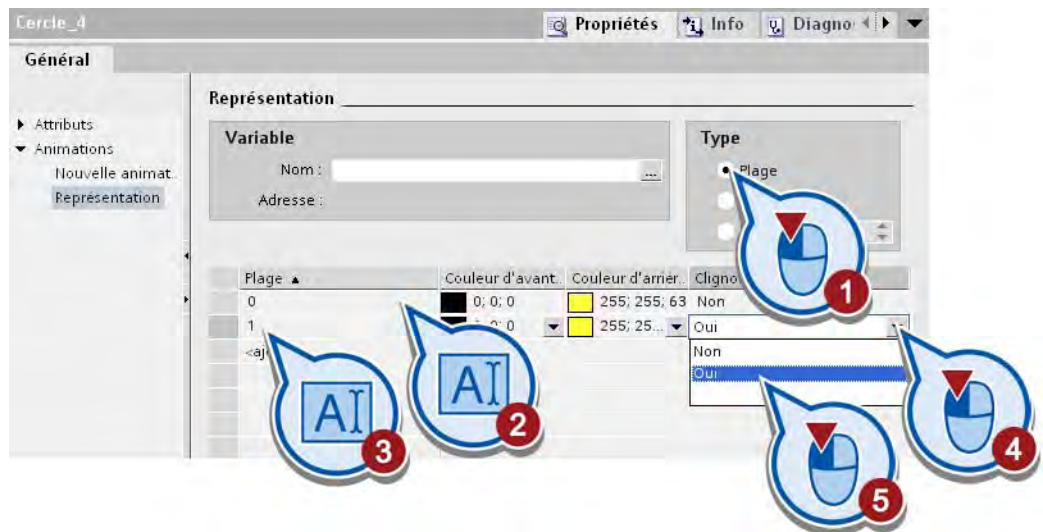
1. Insérez un objet cercle avec un arrière-plan en jaune et une largeur de bord de "2" pour représenter une barrière photoélectrique.



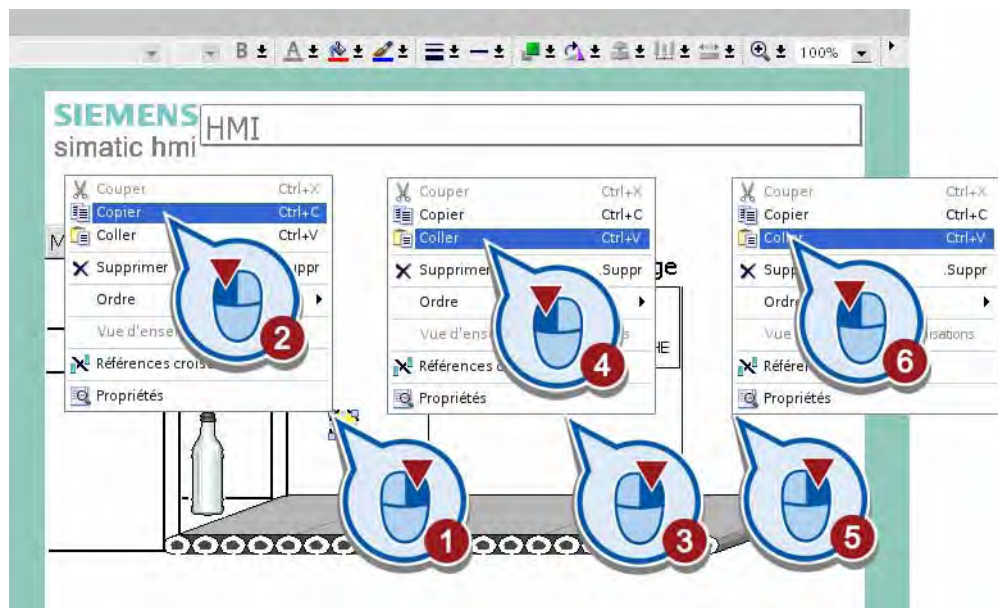
2. Créez une animation "Représentation" pour la barrière photoélectrique.



3. Activez la fonction "Clignotement" pour la plage de variable "1" sans définir de variable préalablement.

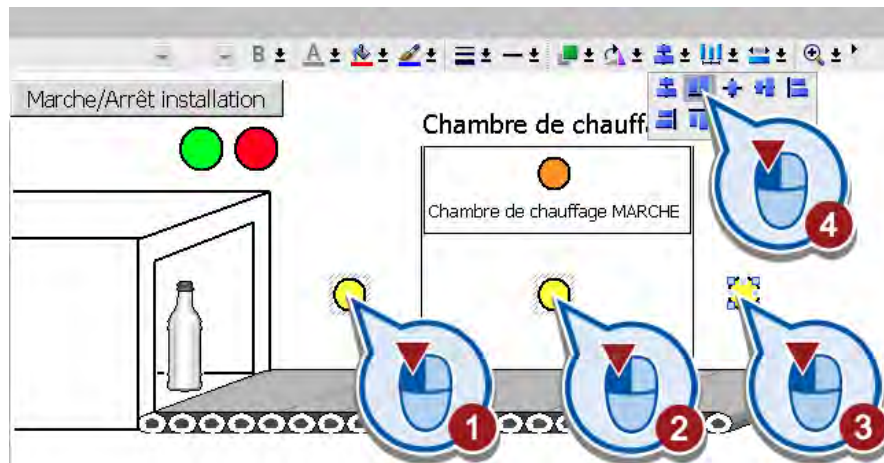


4. Créez deux autres barrières photoélectriques par copier-coller.

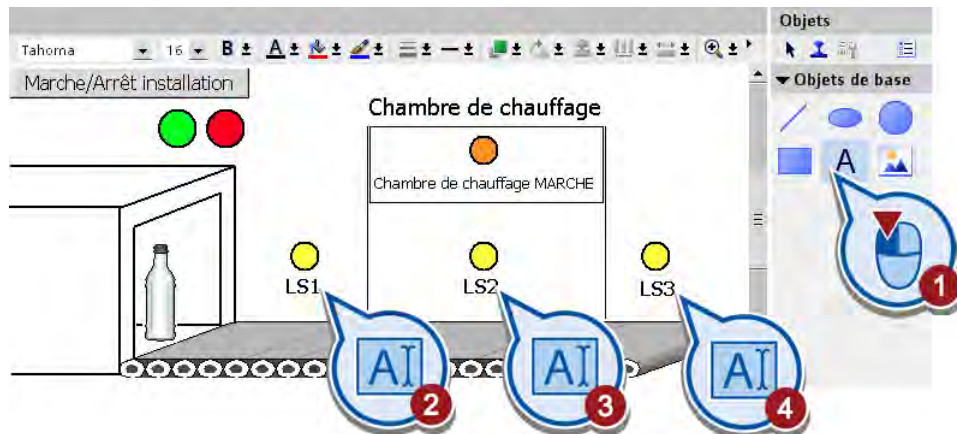


5. Positionnez la barrière photoélectrique au milieu de la chambre de chauffe et à la fin de la bande transporteuse.

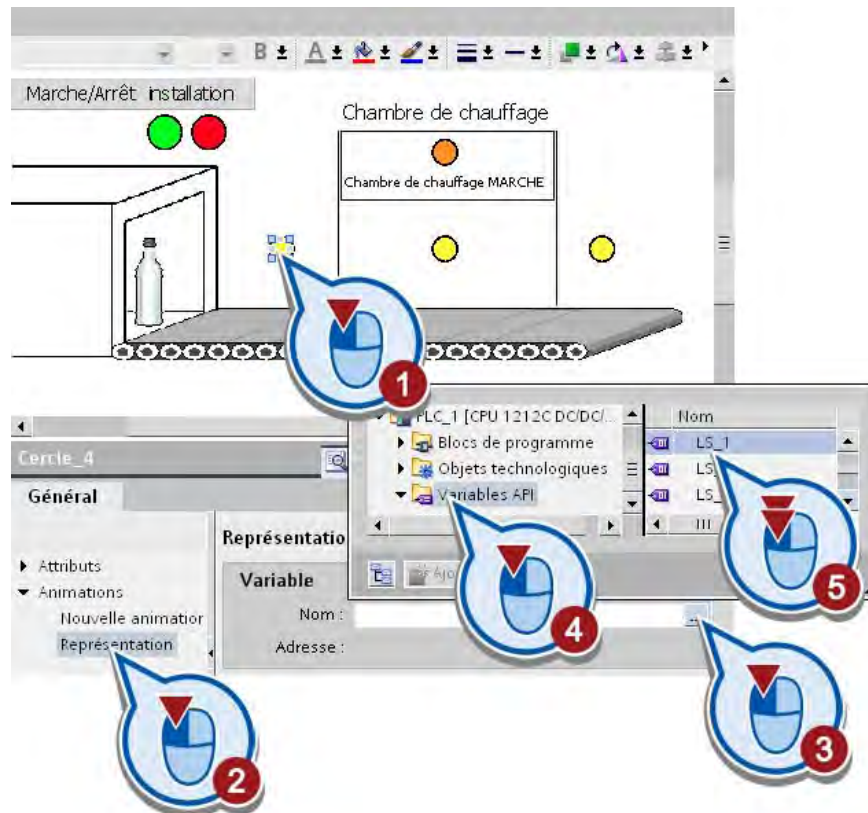
- Sélectionnez les trois barrières photoélectriques en maintenant la touche Maj enfoncée et alignez-les.



- Nommez les barrières photoélectriques "LS1", "LS2" et "LS3" respectivement.



- Associez l'animation de la représentation de la première barrière photoélectrique à la variable API "LS_1".



- Associez l'animation de la représentation des deuxième et troisième barrières photoélectriques aux variables API "LS_2" et "LS_3" respectivement.
- Enregistrez le projet à l'aide du bouton "Enregistrer le projet" de la barre d'outils.

Résultat

Vous avez complété la vue avec les barrières photoélectriques "LS1", "LS2" et "LS3" que vous avez associées à des variables et animées.

Dans le paragraphe suivant, vous allez simuler la vue pour tester le comportement des objets dynamiques.

3.5 Simulation de la vue IHM

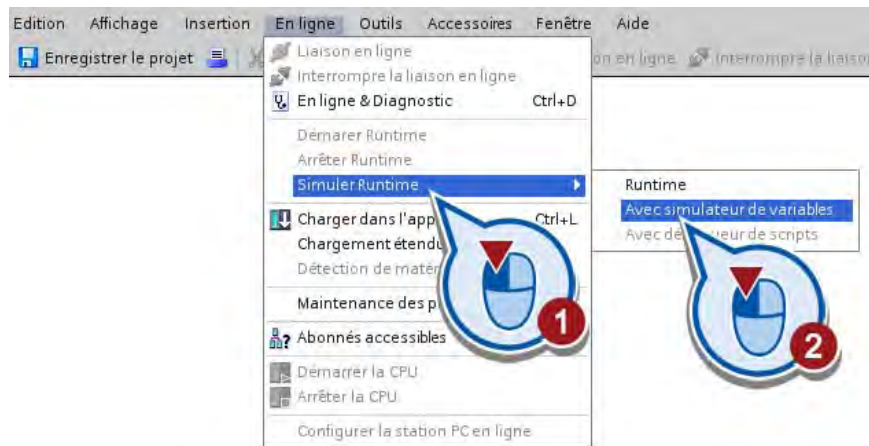
Introduction

Les étapes suivantes décrivent comment tester la vue IHM créée avec le simulateur Runtime. Avec le simulateur Runtime, vous simulez l'activation de l'entrée de l'automate pour la barrière photoélectrique "LS1". Reportez-vous au paragraphe "Simulation Runtime (Page 91)" pour plus d'informations sur le simulateur Runtime.

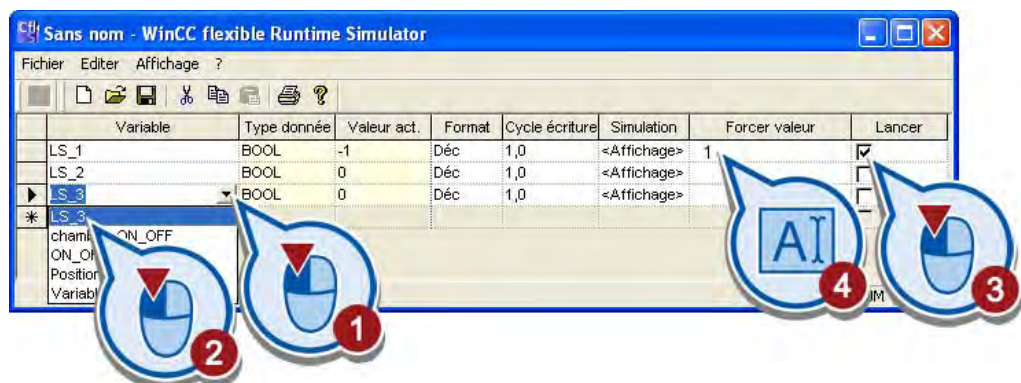
Marche à suivre

Procédez de la manière suivante pour démarrer la simulation de la vue IHM créée :

1. Démarrez la simulation Runtime par le biais de la barre des menus. Pour ce faire, la fenêtre du pupitre opérateur doit être active. Si le menu est inactif, cliquez d'abord sur une zone libre dans la vue IHM.



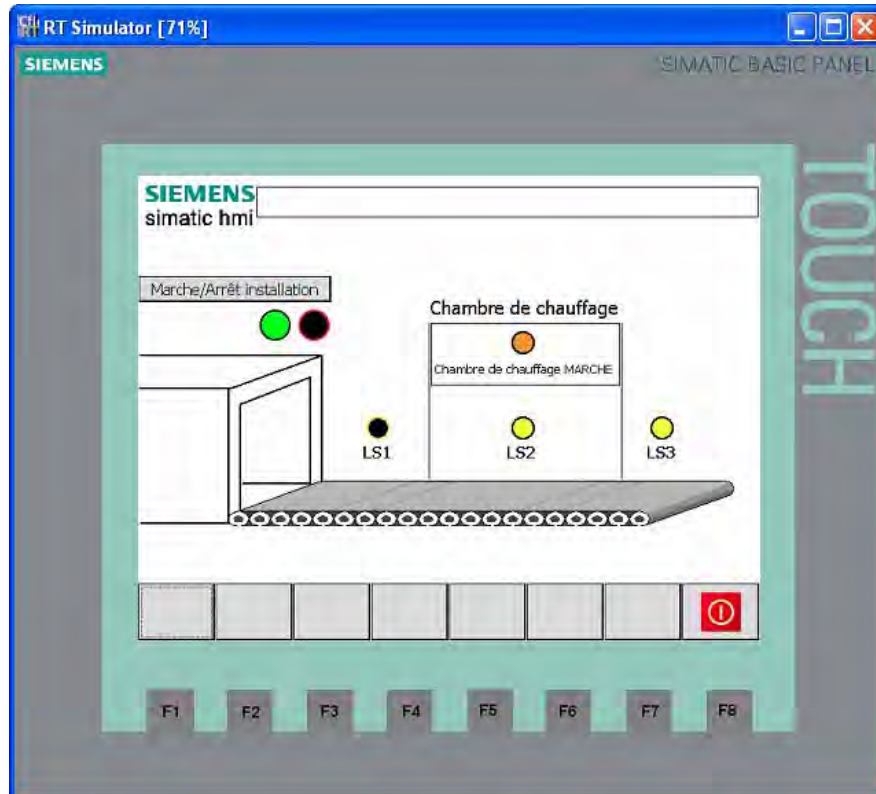
2. Mettez la valeur de la variable "LS_1" à "1". Appliquez la valeur donnée via la touche Entrée.



3. Basculez dans la fenêtre "Simulateur RT".

Résultat

La valeur pour la variable API "LS_1" est simulée. La barrière photoélectrique "LS1" clignote.



Vous pouvez simuler les valeurs pour toutes les variables API utilisées en affectant des valeurs dans la colonne "Définir valeur". Toutes les variables API sont des valeurs booléennes ; elles ne peuvent donc prendre que la valeur "0" ou la valeur "1".

Remarque

Variables dans le simulateur Runtime

Ce sont les variables IHM qui sont affichées dans la sélection de variables du simulateur Runtime. Les variables API qui ne sont associées ni à des événements ni à une animation de la vue IHM ne figurent pas dans la colonne "Variable".

Test dans le pupitre opérateur

Vous pouvez également tester la vue IHM sur un pupitre opérateur. Pour plus d'informations, reportez-vous au paragraphe "Chargement de la vue IHM dans le pupitre opérateur (Page 89)".

Exemple "Régulation PID"

4.1 Introduction

Chargement du projet

Si vous avez sauté les chapitres précédents, vous pouvez charger l'état du projet à la fin du dernier chapitre (voir "Chargement de projets (Page 18)"). L'état à la fin du dernier chapitre est sauvegardé dans le fichier "Extended_Example.ZIP".

Introduction

Une régulation est nécessaire lorsqu'une grandeur physique définie, par exemple une température, une pression ou une vitesse, doit avoir une valeur précise dans le processus et que cette valeur peut varier suite à des conditions externes non prévisibles.

Définition de "régulateur PID"

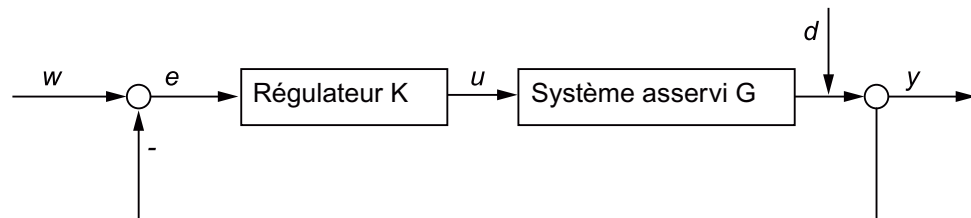
Un régulateur PID est constitué d'une action proportionnelle, d'une action intégrale et d'une action dérivée. Il détecte en continu la valeur mesurée de la grandeur réglée à l'intérieur de la boucle de régulation et la compare à la consigne désirée. Le régulateur PID calcule à partir de l'écart de régulation résultant une grandeur réglante qui amène la grandeur réglée le plus rapidement et de la manière la plus stable possible à la valeur de consigne.

Définition de "boucle de régulation"

Une boucle de régulation se compose d'un système asservi, du régulateur, d'un organe de mesure (capteur) et d'un actionneur :

- Dans cet exemple, la régulation PID SIMATIC S7-1200 est utilisée comme régulateur.
- Dans cet exemple, l'organe de mesure est un capteur qui mesure la température à l'intérieur de la chambre de chauffe.
- L'actionneur est l'élément chauffant auquel l'automate accède directement.

Le schéma de circuit suivant montre une boucle de régulation typique :



La consigne (w) est imposée. Dans l'exemple suivant, la consigne est la température désirée de 75°C à l'intérieur de la chambre de chauffe. L'écart de régulation (e) est calculé à partir de la consigne (w) et de la mesure (y). L'écart de régulation est converti par le régulateur (K) en une valeur de réglage (u). La valeur de réglage modifie la mesure (y) par le biais du système asservi (G). Dans cet exemple, le système asservi (G) correspond à l'adaptation de température dans la chambre de chauffe, par exemple par augmentation ou diminution de l'apport d'énergie.

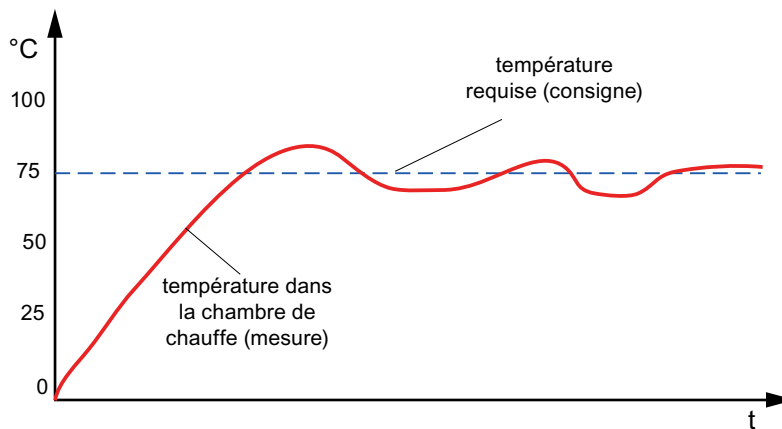
La mesure (y) peut être modifiée non seulement par le système asservi (G) mais également par des grandeurs perturbatrices (d). Dans cet exemple, une grandeur perturbatrice pourrait être une variation de température indésirable dans la chambre de chauffe, provoquée par exemple par une variation de la température extérieure.

Utilisation du régulateur PID

Dans le projet-exemple, le régulateur PID est utilisé pour atteindre la température désirée de 75°C aussi rapidement que possible et pour maintenir la consigne de la manière la plus constante possible.

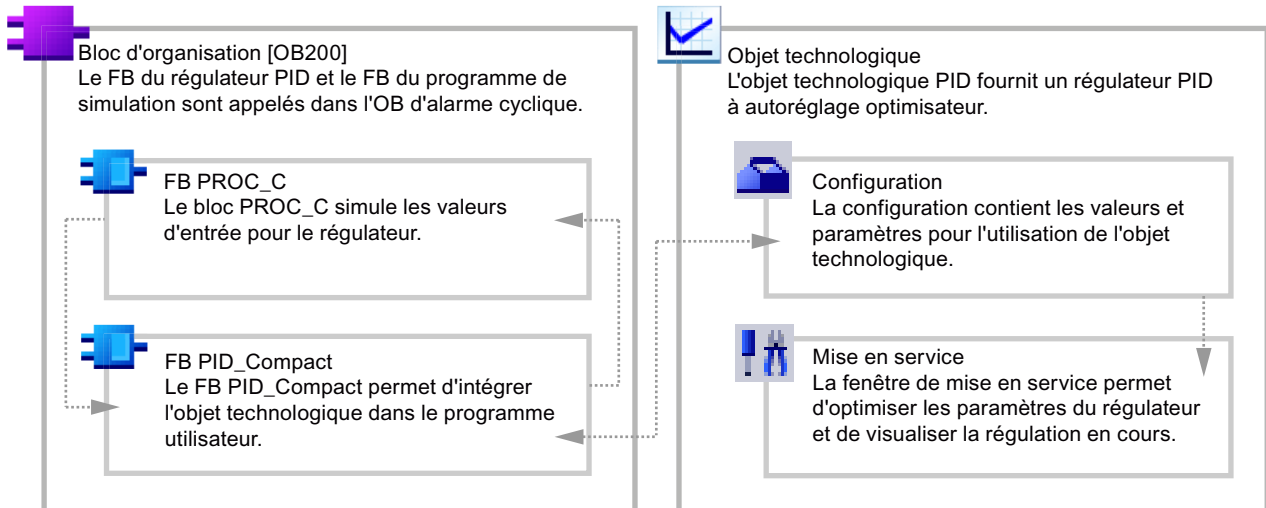
On pourrait avoir un dépassement de la consigne dans cet exemple si l'élément chauffant dégageait toujours de la chaleur même une fois arrêté. Cet effet est appelé dépassement et apparaît s'il se produit un retard entre la régulation et l'opération de mesure de la valeur de mesure.

La figure suivante montre un tracé possible de la courbe de température à la première mise en marche de l'installation :



Etapes de travail

La figure suivante contient une vue d'ensemble des objets que vous allez créer :



Vous procéderez de la manière suivante pour créer ces objets :

- Vous créez un deuxième bloc d'organisation [OB200] dans lequel les blocs pour le régulateur PID seront appelés.
- Vous créez l'objet technologique "PID_Compact".
- Vous chargerez le bloc de simulation "PROC_C" dans le bloc d'organisation [OB200]. Grâce à l'utilisation du bloc de simulation, vous n'aurez pas besoin de matériel en plus de l'automate.
- Vous configurerez l'objet technologique "PID_Compact".
 - Vous sélectionnez le type de régulation.
 - Vous définirez une consigne pour la régulation.
 - Vous connecterez la mesure et la valeur de réglage de l'objet technologique "PID_Compact" au bloc de simulation "PROC_C".
- Vous chargerez le programme utilisateur et vous exécuterez une optimisation du régulateur dans la fenêtre de mise en service de l'objet technologique.

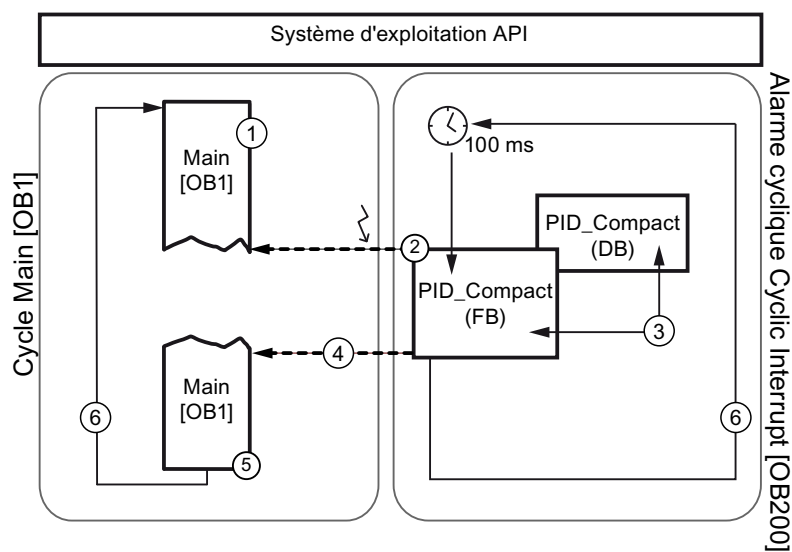
4.2 Création du bloc d'organisation pour le régulateur PID

Introduction

Les blocs pour le régulateur PID sont créés dans un nouveau bloc d'organisation. Nous utiliserons comme nouveau bloc d'organisation un OB d'alarme cyclique que vous allez créer dans la suite.

Les blocs d'organisation d'alarme cyclique servent à démarrer des programmes à intervalles réguliers indépendamment du traitement cyclique du programme. Le traitement cyclique du programme est alors interrompu par l'OB d'alarme cyclique, puis reprend une fois l'OB traité.

La figure suivante montre l'exécution du programme avec un OB d'alarme cyclique :



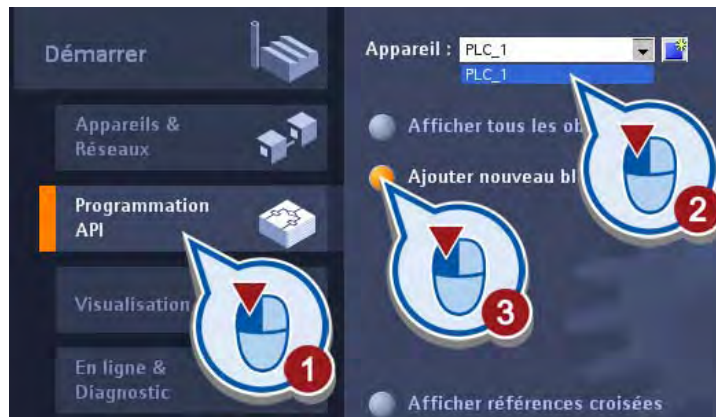
- ① Le programme commence avec Main [OB1].
- ② Toutes les 100 ms se déclenche une alarme cyclique qui interrompt le programme en cours à n'importe quel endroit (par exemple, pendant Main [OB1]) et qui exécute le programme figurant dans l'OB d'alarme cyclique.
Ici, le programme se compose du bloc fonctionnel PID_Compact.
- ③ PID_Compact est exécuté et les valeurs sont écrites dans le bloc de données PID_Compact (DB).
- ④ Une fois l'OB d'alarme cyclique traité, l'exécution de Main [OB1] reprend où elle avait été interrompue. Les valeurs sont conservées.
- ⑤ Main [OB1] est achevé.
- ⑥ Le cycle du programme reprend du début.

Dans le projet-exemple, vous utilisez l'OB d'alarme cyclique pour appeler l'objet technologique "PID_Compact". L'objet technologique "PID_Compact" est l'image du régulateur PID dans le logiciel. Cet objet technologique vous permet de configurer un régulateur PID, de l'activer et de contrôler l'état d'exécution.

Marche à suivre

Procédez de la manière suivante pour créer un OB d'alarme cyclique pour le régulateur PID :

1. Ouvrez la vue du portail.
2. Insérez un nouveau bloc dans l'automate déjà présent.



3. Créez un OB d'alarme cyclique de nom "PID". Veillez à ce que la case d'option "Ajouter nouveau et ouvrir" soit cochée.



Résultat

L'OB d'alarme cyclique créé s'ouvre dans l'éditeur de programmes dans la vue du projet. Si le bloc ne s'ouvre pas automatiquement, c'est que la case d'option "Ajouter nouveau et ouvrir" n'était pas cochée dans la boîte de dialogue. Dans ce cas, basculez dans la vue du projet et ouvrez le bloc de programme dans le navigateur du projet.

Dans le paragraphe suivant, vous allez appeler l'objet technologique "PID_Compact" dans le bloc créé.

4.3 Création de l'objet technologique pour le régulateur PID

Introduction

Les étapes suivantes décrivent comment appeler l'objet technologique "PID_Compact" dans l'OB d'alarme cyclique "PID [OB200]".

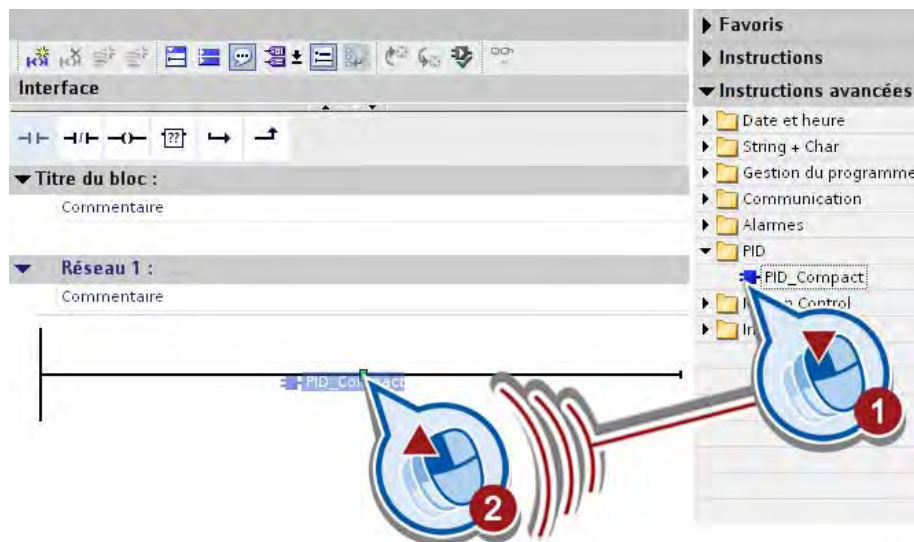
Condition requise

- Un projet avec un automate S7-1200 a été créé.
- Un OB d'alarme cyclique existe déjà et est ouvert dans la vue du projet.

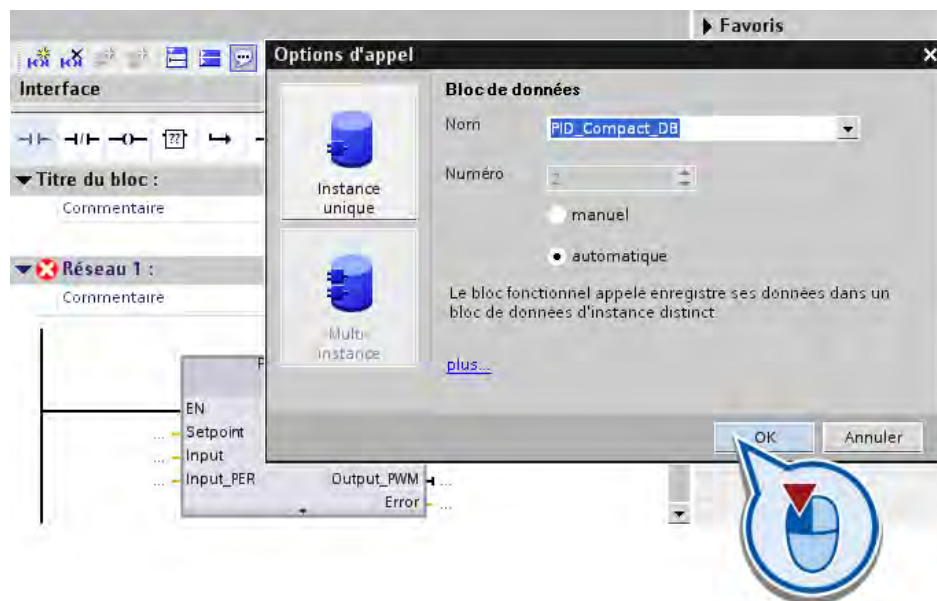
Marche à suivre

Procédez de la manière suivante pour appeler l'objet technologique "PID_Compact" dans l'OB d'alarme cyclique "PID [OB200]" :

1. Créez l'objet technologique "PID_Compact" dans le premier réseau du bloc d'organisation "PID [OB200]".



2. Confirmez la création du bloc de données pour l'objet technologique "PID_Compact".



Résultat

Vous avez programmé l'appel de l'objet technologique "PID_Compact" dans l'OB d'alarme cyclique "PID [OB200]" et créé le bloc de données "PID_Compact_DB".

Dans le paragraphe suivant, vous allez charger un bloc de simulation dans le programme pour simuler les valeurs d'entrée et de sortie du régulateur PID.

4.4 Chargement du bloc de simulation

Introduction

Les étapes suivantes décrivent comment charger le bloc "PROC_C" dans le projet-exemple. Le bloc simule des valeurs d'entrée et de sortie du régulateur PID. Pour utiliser ce bloc, vous devez charger une bibliothèque dans le projet-exemple et créer le bloc dans le deuxième réseau.

Condition requise

Le bloc d'organisation "PID [OB200]" est ouvert dans la vue du projet.

Marche à suivre

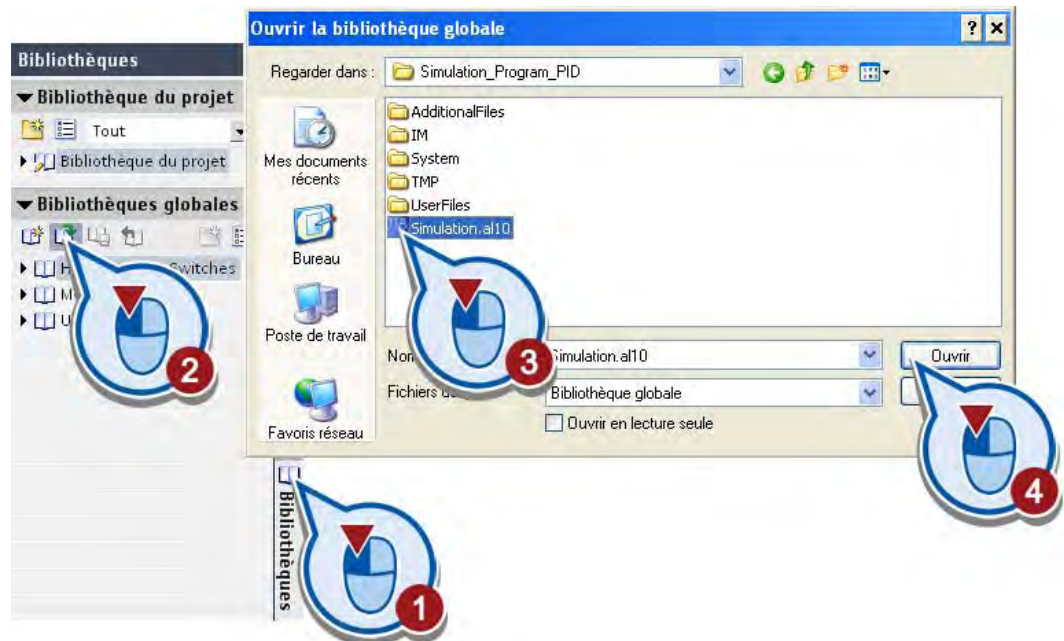
Procédez de la manière suivante pour ouvrir la bibliothèque et copier le bloc :

1. Copiez le fichier "Simulation Program PID.ZIP" de l'adresse Internet suivante sur votre disque dur local et extrayez le fichier :

<http://support.automation.siemens.com/WWW/view/fr/40263542>

Cliquez sur l'icône "Infos" pour afficher les fichiers ZIP.

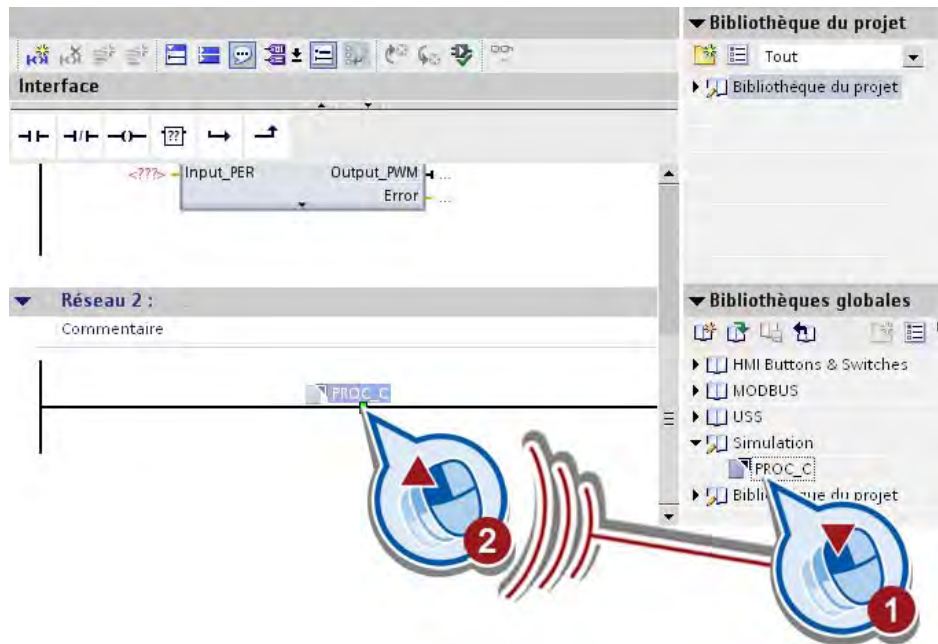
2. Extrayez le fichier "Simulation Program PID.ZIP".
3. Ouvrez via la Task Card "Bibliothèques" la bibliothèque globale "Simulation" dans le répertoire du fichier extrait.



La bibliothèque est chargée.

4.4 Chargement du bloc de simulation

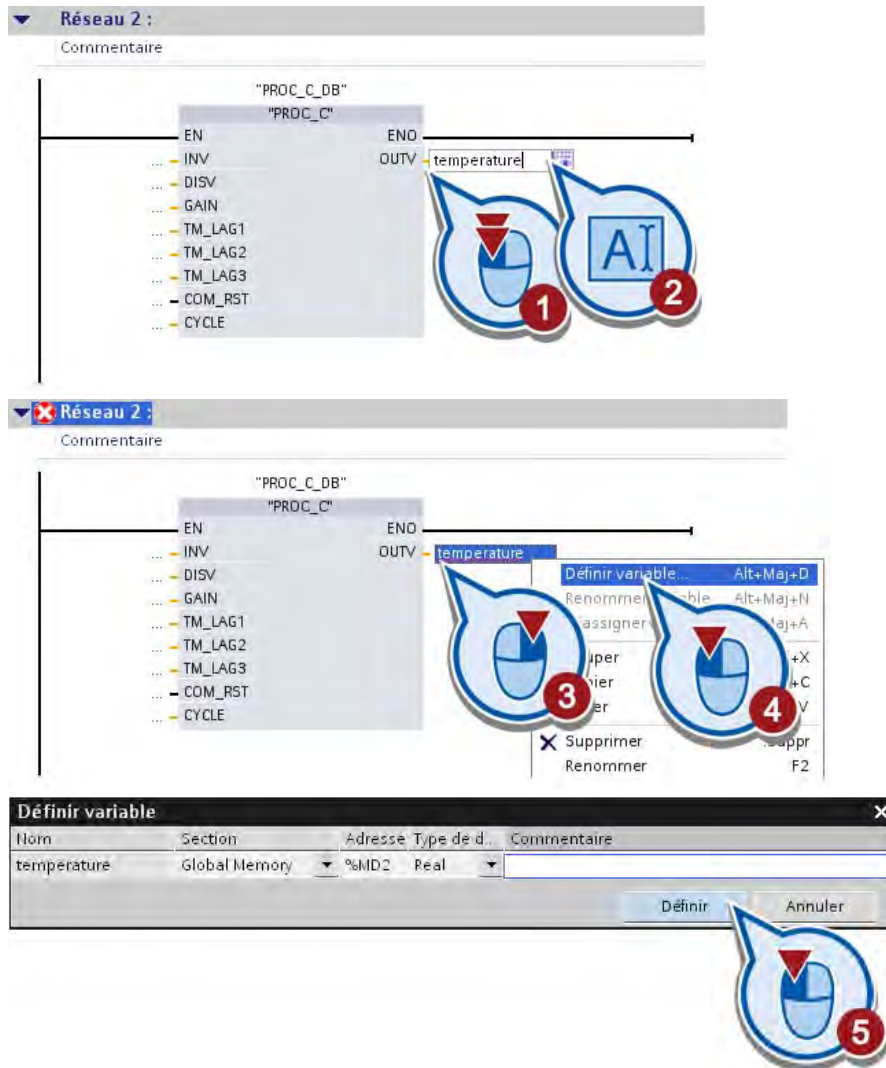
- 4. Copiez le bloc de simulation "PROC_C" dans le deuxième réseau du bloc d'organisation "PID [OB200]".



- 5. Confirmez la création du bloc de données pour le bloc de simulation "PROC_C".



6. Définissez une variable "temperature" pour le paramètre OUTV.

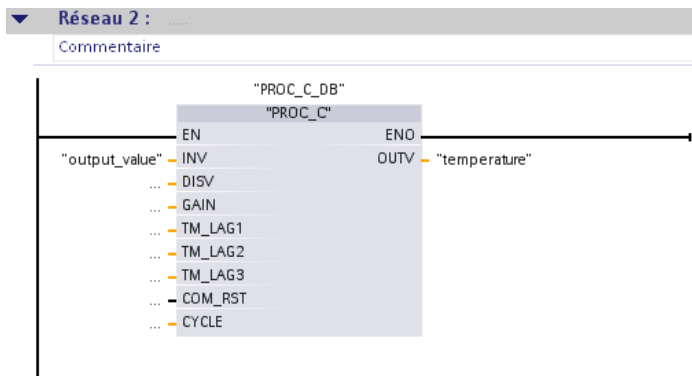


La valeur du paramètre OUTV est sauvegardée dans la variable "temperature". La valeur de ce paramètre est la valeur de température qui est simulée par le bloc "PROC_C".

7. Définissez de la même manière une variable "output_value" pour le paramètre INV.

Résultat

Vous avez chargé dans le projet-exemple le bloc pour la simulation des valeurs d'entrée et de sortie du régulateur PID.



Lors de l'exécution de l'OB d'alarme cyclique "PID (OB200)", le bloc "PROC_C" simule des valeurs d'entrée et de sortie et les sauvegarde dans le bloc de données d'instance "PROC_C_DB". Les valeurs des paramètres INV et OUTV sont copiées dans les variables lors de l'exécution du programme.

Dans le paragraphe suivant, vous allez configurer le régulateur PID grâce à l'objet technologique "PID_Compact" et associer ses entrées et sorties aux valeurs correspondantes du bloc de simulation.

4.5 Configuration du régulateur PID

Introduction

Les étapes suivantes décrivent comment configurer le régulateur PID grâce à l'objet technologique "PID_Compact".

Paramétrages pour la configuration du régulateur PID

- Type de régulation

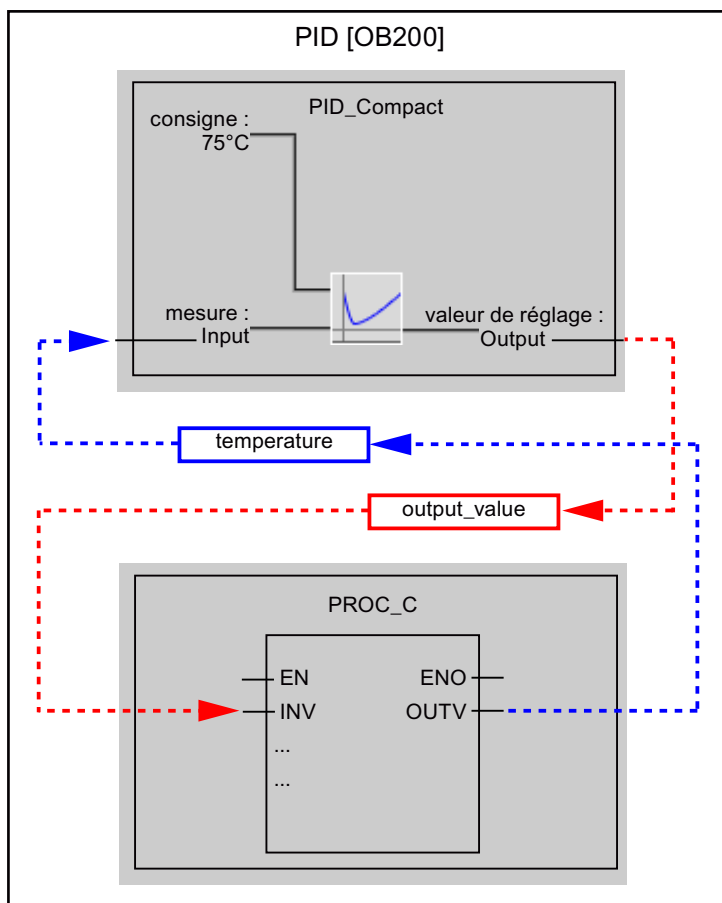
Le type de régulation permet d'effectuer une présélection pour l'unité de la valeur à réguler. Dans cet exemple, nous utilisons "Température" avec l'unité "°C" comme type de régulation.

- Paramètres d'entrée/sortie

Dans cette zone, vous donnez des valeurs aux paramètres d'entrée et de sortie pour la consigne, la mesure et la valeur de réglage de l'objet technologique "PID_Compact". Pour utiliser le régulateur PID sans matériel supplémentaire, vous associez les paramètres d'entrée et de sortie de "PID_Compact" aux variables "output_value" et "temperature" connectées au bloc de simulation "PROC_C" :

- La mesure est simulée par "PROC_C" et utilisée comme entrée de "PID_Compact".
Dans l'exemple, la mesure correspond à la température qui est mesurée dans la chambre de chauffe et qui est représentée dans la variable "temperature".
- La valeur de réglage est calculée par l'objet technologique "PID_Compact" et est un paramètre de sortie du bloc. La valeur de réglage est représentée dans la variable "output_value" et est utilisée comme valeur d'entrée de "PROC_C".

La figure suivante montre comment l'objet technologique "PID_Compact" et le bloc de simulation "PROC_C" sont interconnectés.



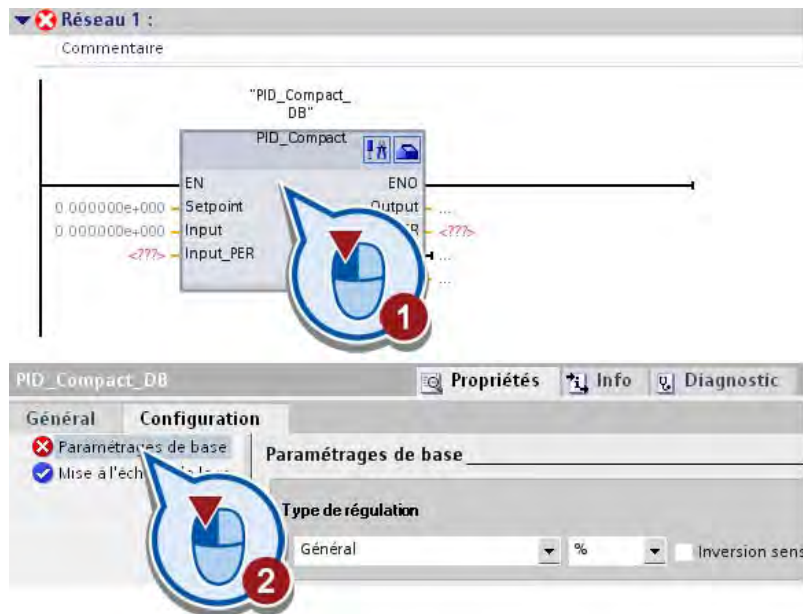
Condition requise

- L'OB d'alarme cyclique "PID [OB200]" est ouvert.
- Le bloc "PID_Compact" est appelé dans le bloc d'organisation "PID [OB200]".
- Le bloc de simulation "PROC_C" est appelé dans le bloc d'organisation "PID [OB200]".

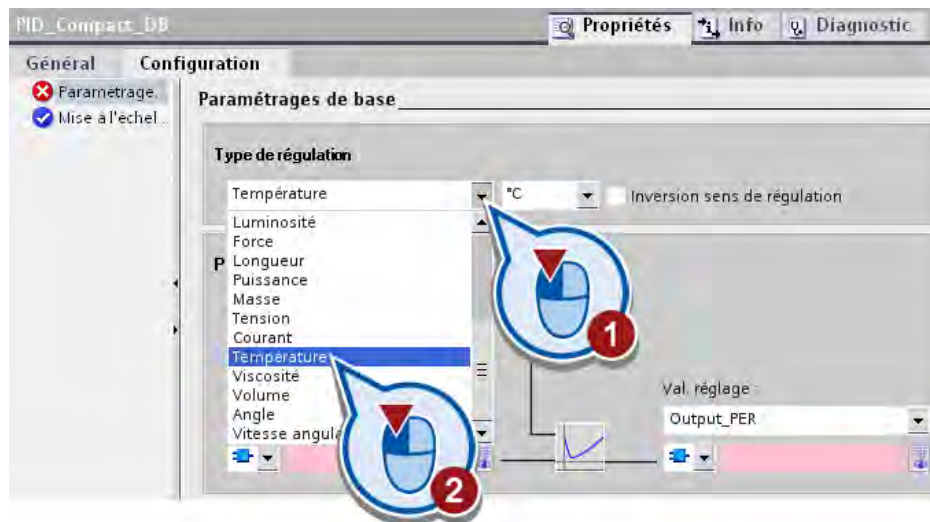
Marche à suivre

Procédez de la manière suivante pour configurer l'objet technologique ""PID_Compact" et pour le connecter au bloc de simulation "PROC_C" :

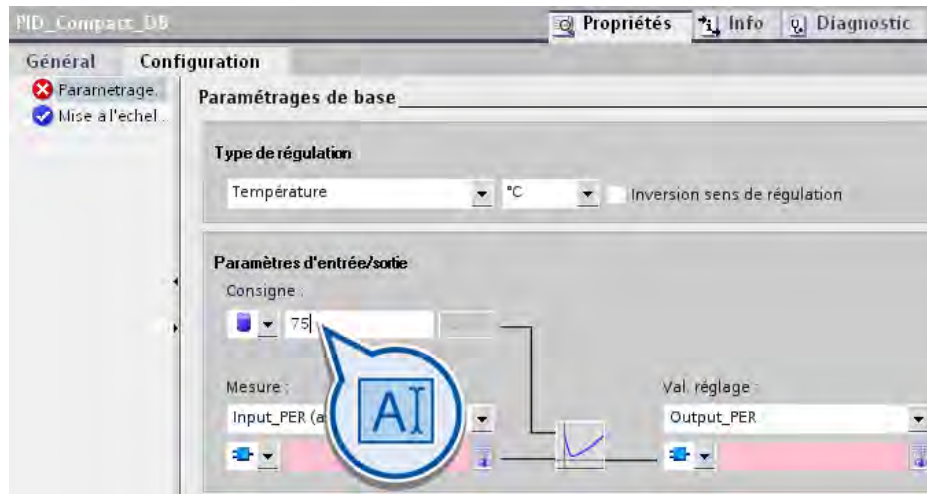
1. Ouvrez la configuration du régulateur PID dans la fenêtre d'inspection.



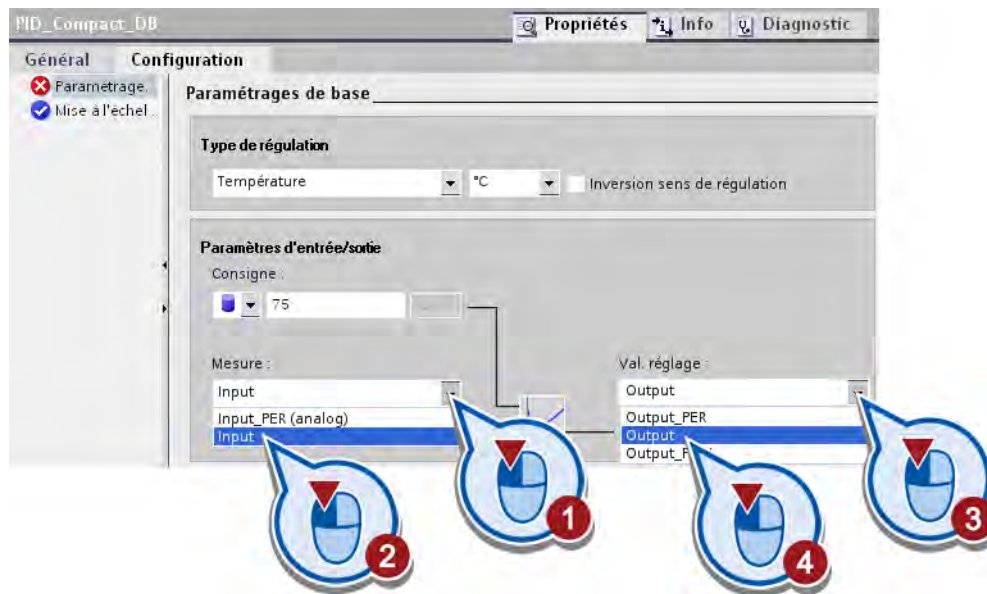
2. Sélectionnez le type de régulation.



- 3. Saisissez la consigne pour la régulation.



- 4. Sélectionnez respectivement "Input" et "Output" pour la mesure et la valeur de réglage. Vous indiquez ainsi qu'il faut utiliser les valeurs provenant d'une variable du programme utilisateur.

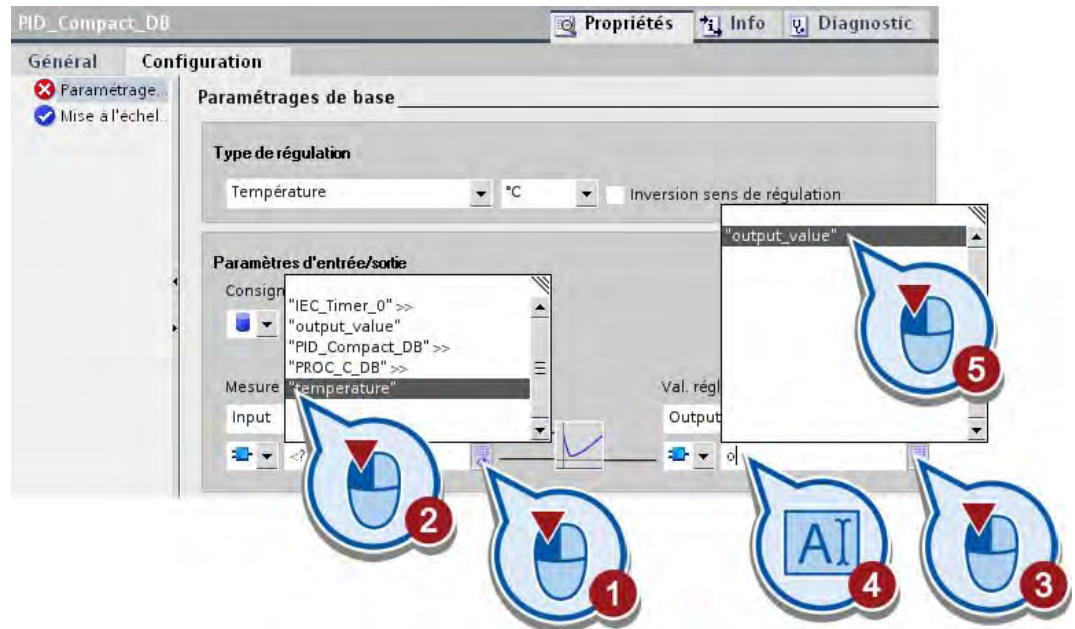


Remarque

Input(_PER) - Output(_PER)

Avec Input et Output, vous fournissez respectivement une mesure et une valeur de réglage du programme utilisateur aux paramètres d'entrée et de sortie. Avec Input_PER et Output_PER, vous pouvez utiliser une entrée analogique comme mesure et une sortie analogique comme sortie de valeur de réglage. Avec Output_PWM, vous pouvez utiliser une sortie de commutation TOR qui est commandée par le biais d'une modulation de largeur d'impulsion. Dans ce cas, la valeur de réglage est formée via des temps d'activation et de désactivation variables.

5. Connectez la variable "temperature" à la mesure et la variable "output_value" à la valeur de réglage. Lorsque vous saisissez la première lettre de la variable, la sélection est filtrée de manière correspondante par Intellisense®.



Remarque

Connexion directe des blocs à des paramètres

La variable "temperature" est encore utilisée dans la suite. Sinon, vous pourriez connecter le bloc "PID_Compact" directement aux paramètres du bloc de simulation. L'adressage des paramètres se fait comme suit : "*nom de bloc*".*paramètre*.

Dans la liste déroulante à gauche du champ d'entrée de la mesure, vous pouvez choisir s'il faut utiliser comme valeur le paramètre connecté ou la variable connectée au bloc fonctionnel ou bien la valeur en cours du DB d'instance de l'instruction PID_Compact.

Résultat

Vous avez connecté le régulateur PID (PID_Compact) et le bloc de simulation "PROC_C". En cas de démarrage de la simulation, le régulateur PID reçoit une nouvelle mesure à chaque appel du bloc d'organisation "PID [OB200]".

4.6 Modification de la commande de la chambre de chauffe

Introduction

Les étapes suivantes décrivent comment programmer la commande de l'activation et de la désactivation de la chambre de chauffe en fonction du fonctionnement de l'installation. La chambre de chauffe est activée lorsque l'installation est mise en marche. Dans cette partie du projet-exemple, la température de la chambre de chauffe n'est pas considérée comme constante et est pilotée par le régulateur PID créé.

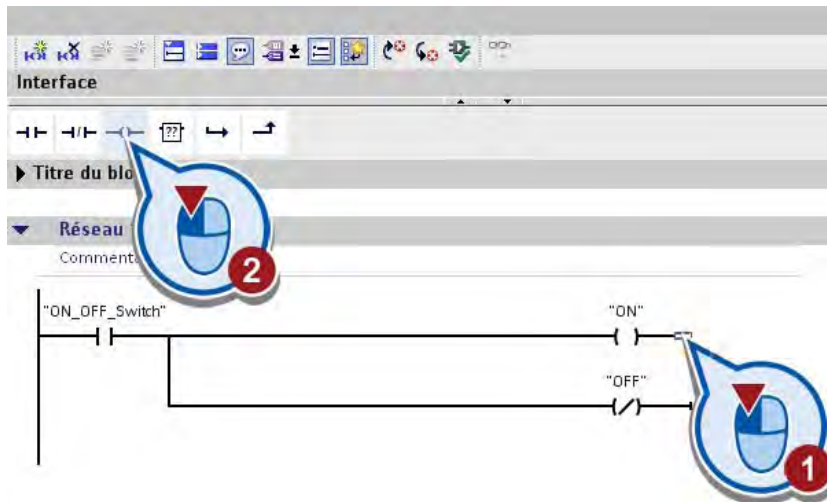
Condition requise

- Le bloc d'organisation "Main [OB1]" est ouvert.
- Les réseaux du bloc d'organisation "Main [OB1]" ont été programmés.

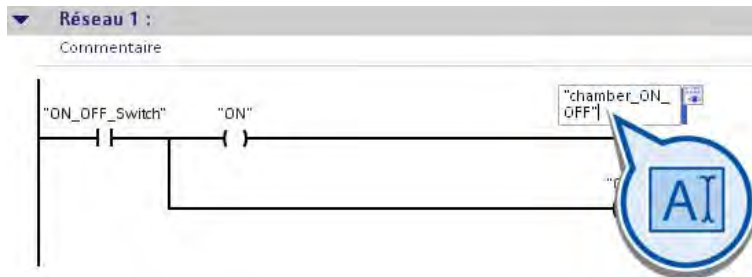
Marche à suivre

Procédez de la manière suivante pour modifier la commande de l'activation et de la désactivation de la chambre de chauffe :

1. Ouvrez le premier réseau du bloc d'organisation "Main [OB1]".
2. Insérez l'instruction "Bobine de relais, sortie" à la fin du circuit principal.

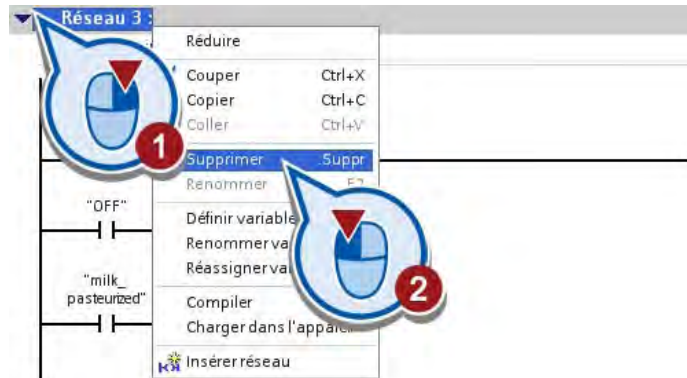


3. Connectez l'instruction insérée à la variable "chamber_ON_OFF".

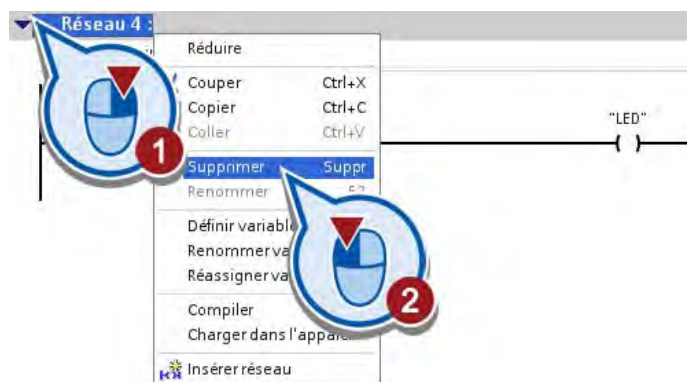


4. Effacez le troisième réseau du bloc d'organisation "Main [OB1]".

Cette partie du programme n'est plus nécessaire, car l'activation et la désactivation de la chambre de chauffe se font par le programme de pilotage de l'installation dans le premier réseau du bloc d'organisation.



5. Effacez le quatrième réseau dans lequel le voyant d'état de la chambre de chauffe est programmé. Le voyant d'état n'est plus nécessaire, car le fonctionnement de la chambre de chauffe se fait parallèlement au fonctionnement de l'installation.



Résultat

Vous avez programmé la commande de l'activation et de la désactivation de la chambre de chauffe de manière à ce que le fonctionnement de la chambre de chauffe soit parallèle à celui de l'installation.

Dans le paragraphe suivant, vous allez programmer une interrogation des valeurs de température et l'intégrer dans le programme comme condition pour le pilotage de la bande transporteuse.

4.7 Intégration dans le programme de commande d'une comparaison de température comme condition

Introduction

Les étapes suivantes décrivent comment évaluer les valeurs de température du régulateur PID et les intégrer comme condition pour la commande de l'installation. Le processus de pasteurisation est piloté par les blocs d'organisation suivants :

- Main [OB1]

Le programme pour la commande de l'installation en fonction des valeurs de température du régulateur PID est exécuté dans le bloc d'organisation "Main [OB1]".

- PID [OB200]

L'objet technologique "PID_Compact" du régulateur PID et le bloc de simulation "PROC_C" sont exécutés dans le bloc d'organisation "PID [OB200]".

Lien entre le régulateur PID et le programme de commande

La surveillance de température du régulateur PID s'intègre comme suit dans le programme du bloc d'organisation "Main [OB1]" :

La mesure de température de la variable "temperature" qui est utilisée comme entrée pour le régulateur PID doit être lue par le programme dans le bloc d'organisation "Main (OB1)" et comparée à une valeur minimale et une valeur maximale.

- Si la mesure de la température est comprise entre une valeur minimale de 73°C et une valeur maximale de 77°C (consigne +/- 2°C), la bande transporteuse est entraînée.
- Si la valeur minimale de 73°C n'est pas encore atteinte ou que la valeur maximale de 77°C est dépassée, la variable "conveyor_drive_ON_OFF" est mise à "0".

Condition requise

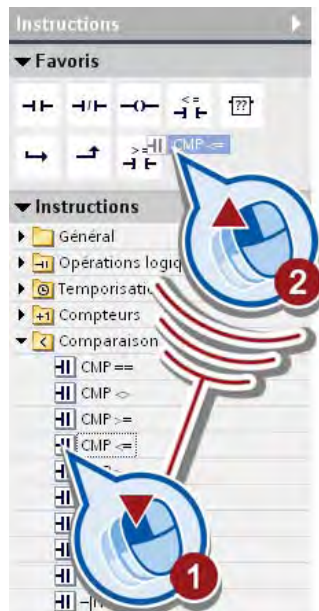
- Le bloc d'organisation "Main [OB1]" est ouvert.

4.7 Intégration dans le programme de commande d'une comparaison de température comme condition

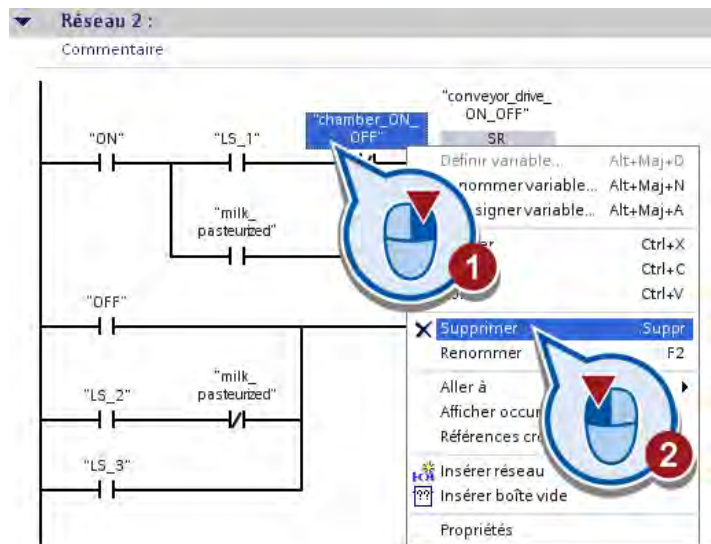
Marche à suivre

Procédez de la manière suivante pour insérer une comparaison de valeur de la variable "temperature" comme condition pour l'entraînement de la bande transporteuse :

1. Insérez les instructions "Supérieur ou égal à (CMP>=)" et "Inférieur ou égal à (CMP<=)" dans la liste des favoris.



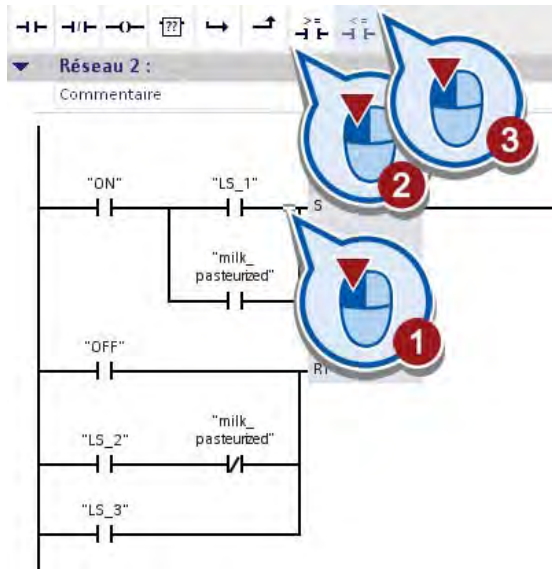
2. Ouvrez le deuxième réseau du bloc d'organisation "Main [OB1]".
3. Effacez le contact à ouverture avec la variable "chamber_ON_OFF".



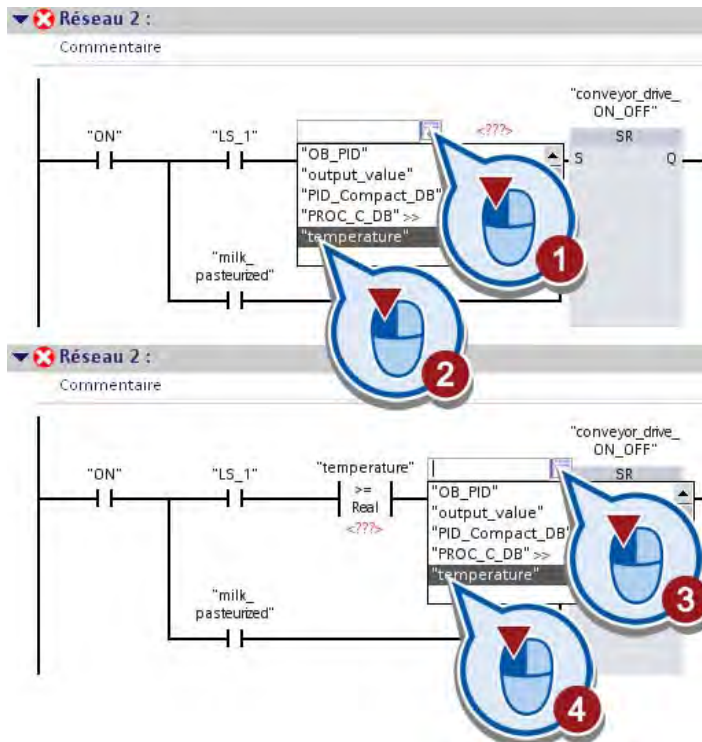
Exemple "Régulation PID"

4.7 Intégration dans le programme de commande d'une comparaison de température comme condition

- 4. Insérez les instructions "Supérieur ou égal à (CMP>=)" et "Inférieur ou égal à (CMP<=)" à partir de la liste des favoris.

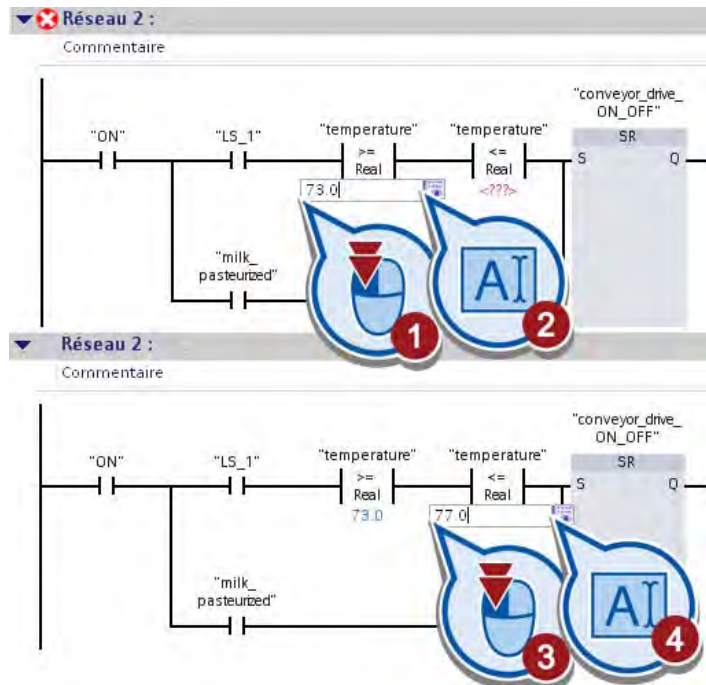


- 5. Connectez les instructions "Supérieur ou égal à (CMP>=)" et "Inférieur ou égal à (CMP<=)" à la variable "temperature".



4.7 Intégration dans le programme de commande d'une comparaison de température comme condition

6. Entrez la valeur "73.0" dans l'emplacement réservé à l'opérande sous l'instruction "Supérieur ou égal à" et la valeur "77.0" dans l'emplacement réservé à l'opérande sous l'instruction "Inférieur ou égal à".



7. Enregistrez le projet à l'aide du bouton "Enregistrer le projet" de la barre d'outils.

Résultat

Vous avez programmé la valeur de température comme condition pour l'entraînement de la bande transporteuse. La bande transporteuse est mise en mouvement lorsque l'installation fonctionne et qu'au moins l'une des conditions suivantes est satisfaite :

- La barrière photoélectrique "LS1" est activée et la température dans la chambre de chauffe est comprise entre 73°C et 77°C. Cela signifie :
 - que la variable "LS_1" a l'état logique "1" ;
 - que la valeur de la variable "temperature" est comprise entre "73" et "77".
- Le lait a été pasteurisé. Dans ce cas, la variable "milk_pasteurized" fournit l'état logique "1".

Dans le paragraphe suivant, vous allez adapter l'association du voyant d'état dans la vue IHM.

4.8 Adaptation de la vue IHM

Introduction

Les étapes suivantes décrivent comment remplacer le voyant d'état de la chambre de chauffe par un champ d'E/S.

Dans l'exemple complété, la chambre de chauffe était mise en marche lorsque la variable "chamber_ON_OFF" avait la valeur "1". Comme le pilotage de la température se fait maintenant par le biais du régulateur PID, cette variable API n'est plus utilisée. Au lieu du voyant d'état, c'est maintenant la valeur de température effective qui doit être affichée dans un champ d'E/S.

Champs d'E/S

Un champ d'E/S (champ d'entrée/sortie) permet d'entrer des valeurs du processus ou de les afficher dans la vue IHM.

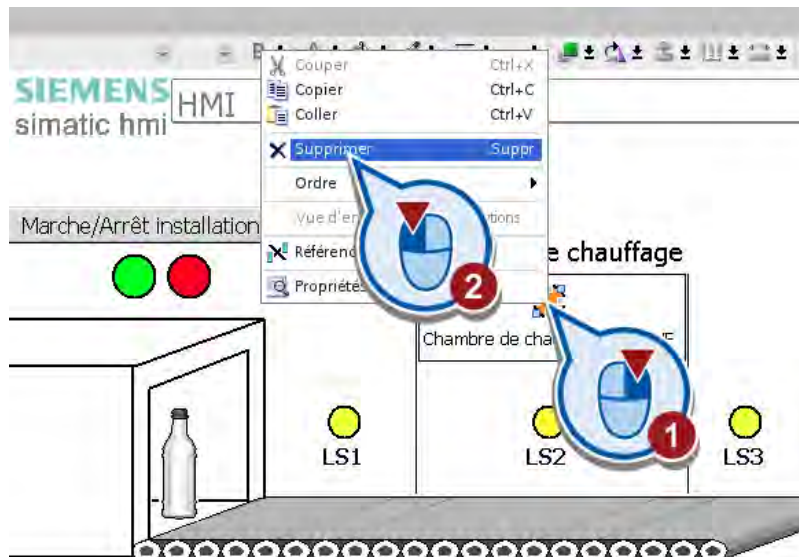
Condition requise

- Le programme a été créé.
- La vue IHM est ouverte dans l'éditeur.

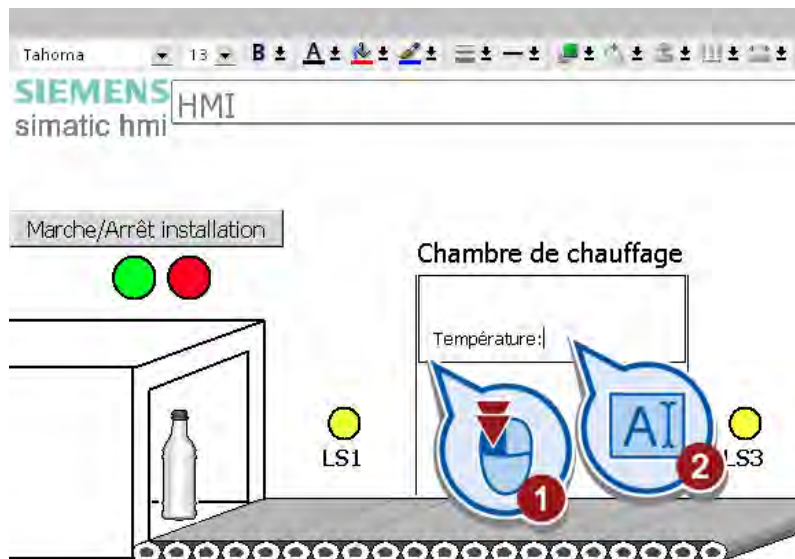
Marche à suivre

Procédez de la manière suivante pour afficher la valeur de température dans la vue IHM :

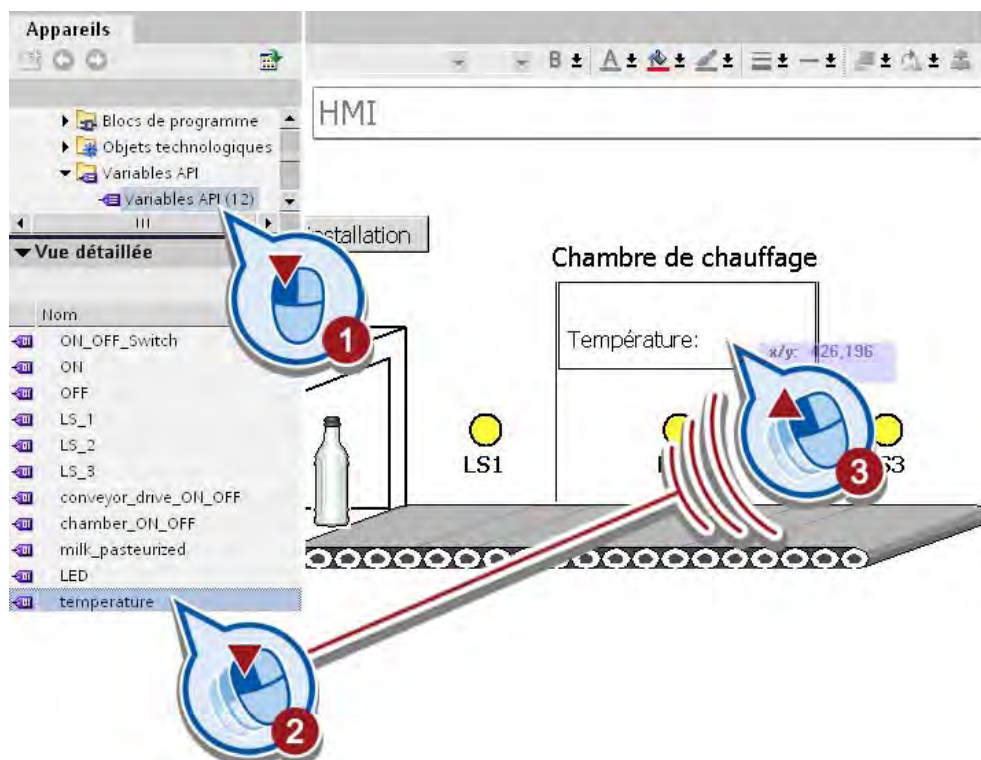
1. Effacez le cercle orange.



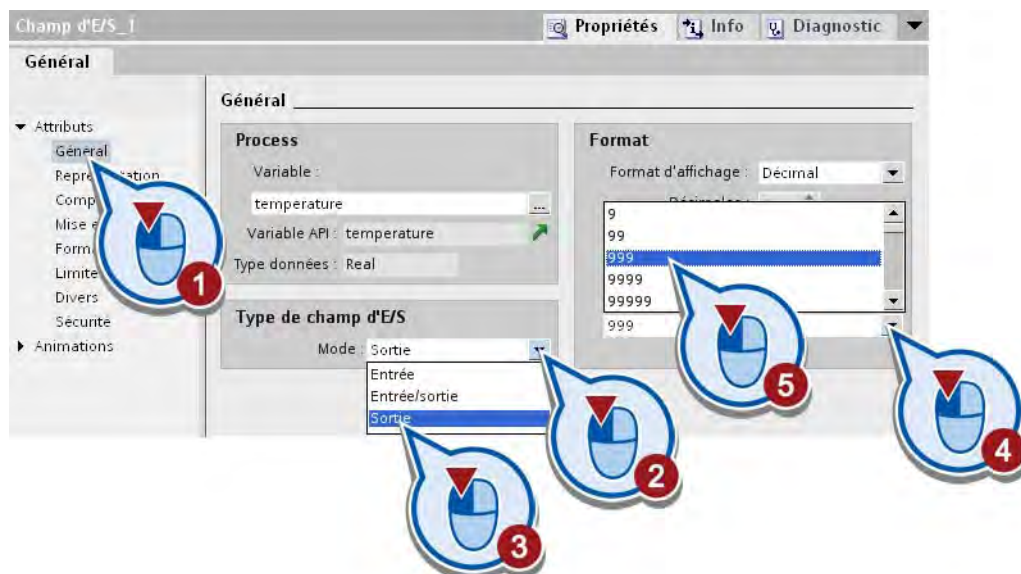
2. Changez le contenu de la zone de texte en "Température".



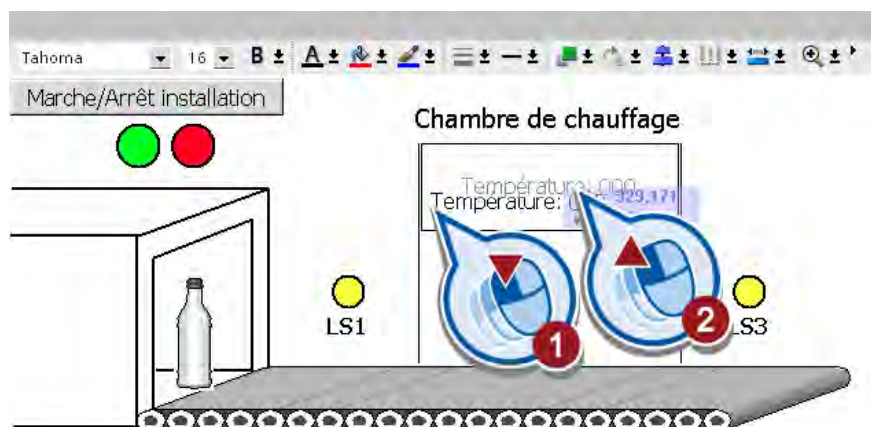
3. Insérez un champ d'E/S dans la vue IHM par le biais de la vue détaillée des variables API.



4. Modifiez le type et le format du champ d'E/S.



5. Adaptez la taille et la position de la zone de texte et du champ d'E/S.



6. Enregistrez le projet à l'aide du bouton "Enregistrer le projet" de la barre d'outils.

Résultat

Vous avez créé un champ de sortie pour la valeur de la variable "temperature". La valeur en cours de la variable sera affichée sur le pupitre opérateur lors de l'exécution du programme utilisateur.

4.9 Activation du régulateur PID en mode en ligne

Introduction

Les étapes suivantes décrivent comment simuler le fonctionnement du régulateur PID. Pour ce faire, vous chargerez d'abord le programme dans l'automate et vous activerez le mode en ligne pour le régulateur PID. Une fois le régulateur optimisé, vous visualiserez le déroulement de la régulation dans la fenêtre des courbes.

Optimisation du régulateur

Vous pouvez adapter le régulateur au système asservi grâce à l'optimisation. Vous disposez de deux choix pour l'optimisation du régulateur :

- Autoréglage à la première mise en route

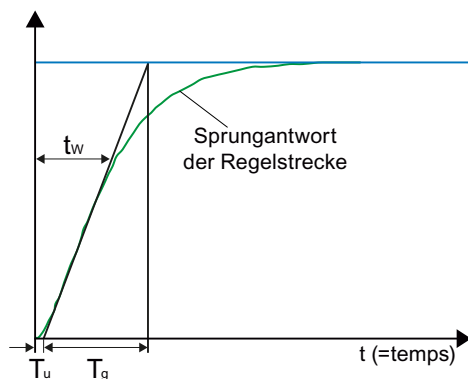
On utilise la méthode de tangente dans un point d'inflexion pour l'autoréglage à la première mise en route. Cette méthode permet de déterminer les constantes de temps de la réponse indicielle. Un point d'inflexion apparaît dans la réponse indicielle du système asservi. On trace au point d'inflexion une tangente grâce à laquelle les paramètres du système temps de retard (T_u) et constante de temps (T_g) sont déterminés. On détermine les paramètres optimisés du régulateur en se basant sur les paramètres du système obtenus. Pour la détermination des paramètres avec la méthode de tangente dans un point d'inflexion, l'écart entre la consigne et la mesure doit être d'au moins 30%. Si ce n'est pas le cas, l'essai en limite de pompage via la fonction "Autoréglage au point de fonctionnement" est automatiquement utilisé pour déterminer les paramètres du régulateur.

- Autoréglage au point de fonctionnement

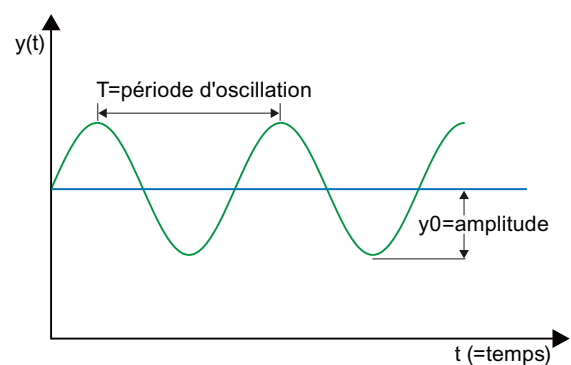
L'autoréglage au point de fonctionnement utilise l'essai en limite de pompage pour l'optimisation des paramètres du régulateur. On utilise cette méthode pour déterminer indirectement le comportement du système asservi. On augmente le gain jusqu'à ce que la limite de stabilité soit atteinte et que la grandeur réglée oscille de manière uniforme. On calcule les paramètres de régulation à partir de la période des oscillations.

La figure suivante montre la réponse indicielle du système asservi avec la méthode de tangente au point d'inflexion (autoréglage à la première mise en route) et avec l'essai en limite de pompage (autoréglage au point de fonctionnement).

Méthode de tangente au point d'inflexion



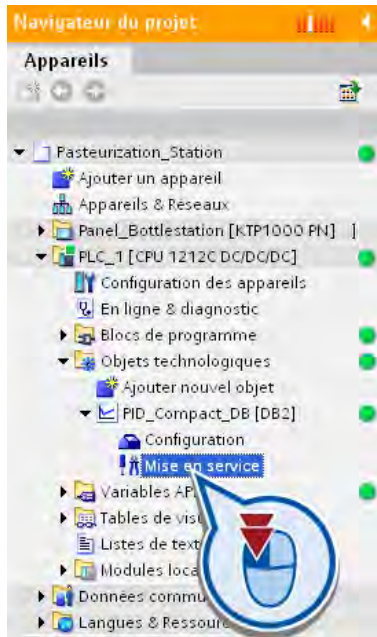
Essai en limite de pompage



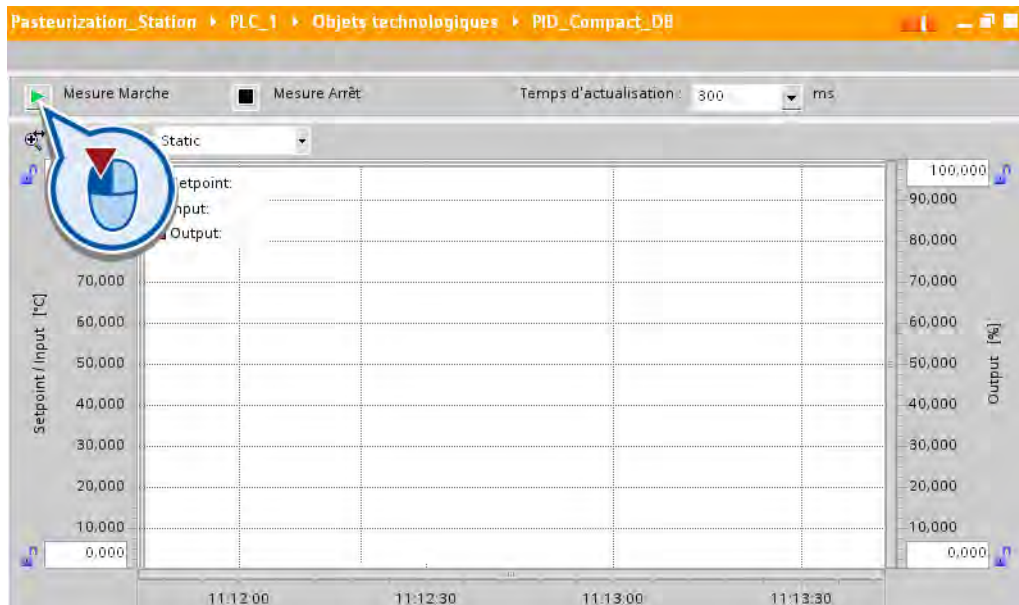
Marche à suivre

Procédez de la manière suivante pour activer le régulateur PID et pour démarrer la simulation :

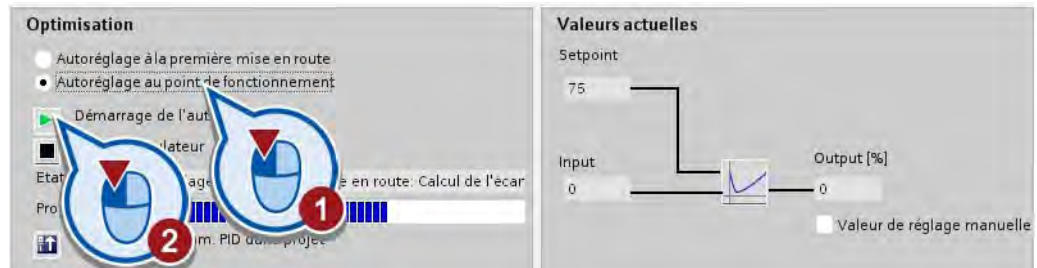
1. Chargez le programme dans l'automate et activez la liaison en ligne. Pour plus d'informations, reportez-vous au paragraphe "Chargement du programme dans le système cible (Page 51)".
2. Démarrez la mise en service dans le navigateur du projet.



3. Activez les fonctions de la fenêtre de mise en service.

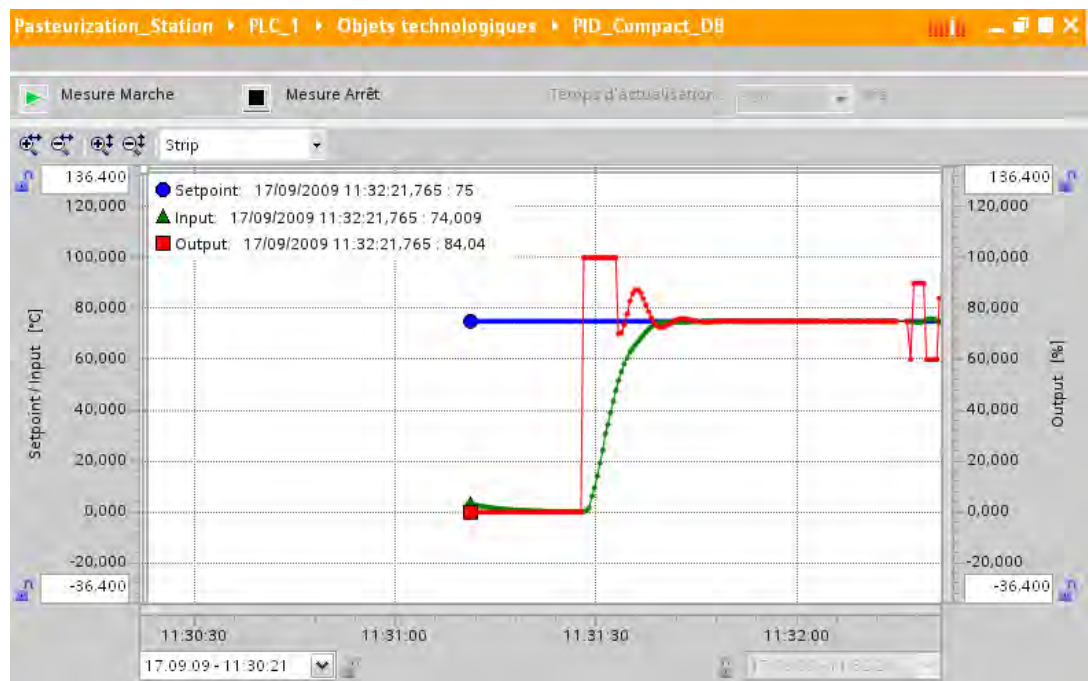


4. Effectuez un réglage fin de l'optimisation du régulateur en lançant l'autoréglage au point de fonctionnement.



L'autoréglage démarre. La zone "Etat" indique les étapes de travail en cours ainsi que les erreurs éventuelles. La barre de progression montre l'avancement de l'étape de travail en cours. Plusieurs étapes de travail sont exécutées pour l'optimisation du régulateur. L'affichage de la progression est remis à "0" pour chaque étape de travail. Si l'autoréglage s'achève sans message d'erreur, les paramètres PID ont bien été optimisés. Le régulateur PID passe en mode automatique et utilise les paramètres optimisés.

5. Observez l'allure de la courbe dans la partie supérieure de la fenêtre "Mise en service".



Remarque

Allure de la courbe

Avec le temps, l'allure de la courbe s'approche de plus en plus de la consigne. Pour visualiser l'ensemble du tracé de la courbe, utilisez la barre de défilement dans la partie inférieure de la représentation graphique.

Une fois que le régulateur est passé en mode automatique, il réagit aux variations de température simulées par le bloc "PROC_C" en adaptant automatiquement la valeur de réglage.

Résultat

Vous avez activé le régulateur PID en mode en ligne. Les paramètres PID optimisés avant le démarrage du mode automatique sont conservés en cas de mise sous tension et de redémarrage de l'automate. Si vous voulez réutiliser les paramètres PID optimisés dans la CPU lors d'un nouveau chargement des données du projet dans la CPU, vous pouvez sauvegarder les paramètres PID dans le projet.

Exemple "Motion"

5.1 Introduction

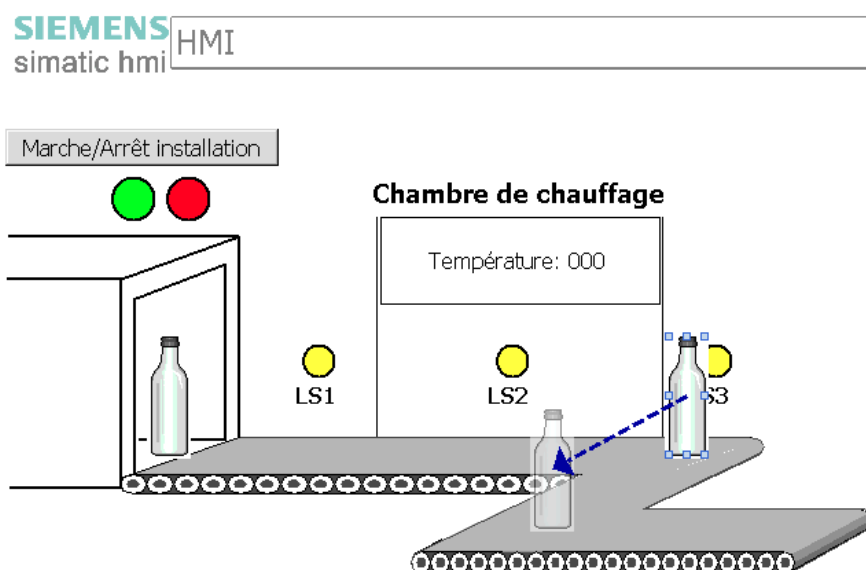
Chargement du projet

Si vous avez sauté les chapitres précédents, vous pouvez charger l'état du projet à la fin du dernier chapitre en ouvrant le projet "Extendet_Example_PID". Le résultat à la fin de ce chapitre est également sauvegardé sous forme d'un projet ayant le nom "Extendet_Example_Motion".

Reportez-vous au paragraphe "Chargement de projets (Page 18)" pour plus d'informations sur le chargement des projets.

Utilisation de Motion

Dans ce chapitre, vous allez compléter l'exemple avec une nouvelle bande transporteuse. La bouteille va être transportée de la position de la barrière photoélectrique "LS3" sur une autre bande transporteuse par le biais d'un segment de liaison.



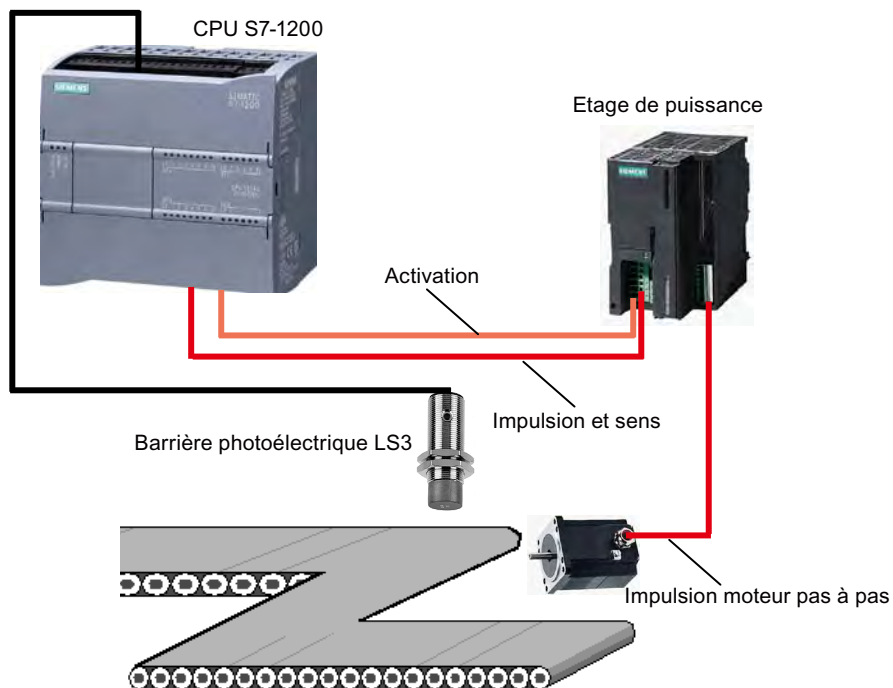
Vous utilisez l'objet technologique "Axe" pour commander la liaison entre les deux bandes transporteuses. La commande est intégrée dans le projet de la manière suivante :

- Si l'installation fonctionne, l'axe pour la commande d'un moteur pas à pas sur le segment de liaison est débloqué.
- L'axe du moteur est commandé si la barrière lumineuse "LS3" est activée.
- La bouteille est déplacée vers la deuxième bande transporteuse par rapport à la position de la barrière photoélectrique "LS3".
- La deuxième bande transporteuse est activée si cette position est atteinte.

Configuration matérielle

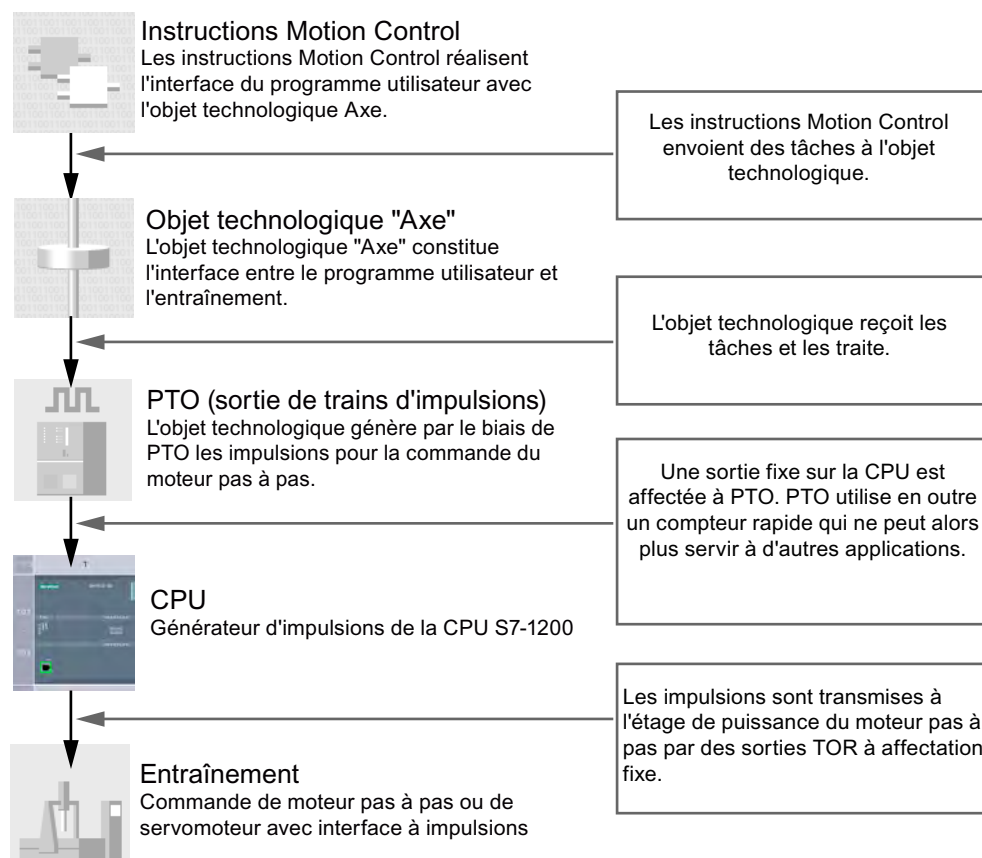
Vous n'avez pas besoin de matériel en plus de l'automate pour réaliser cet exemple. En cas d'utilisation d'un automate à sorties à relais, un Signal Board supplémentaire est nécessaire pour les sorties de PTO. La fonction PTO (sortie de trains d'impulsions) génère au niveau des sorties des suites d'impulsions rapides qui sont utilisées pour la réalisation d'opérations de commande de mouvement.

La figure suivante montre quelle pourrait être la configuration matérielle pour cet exemple :



Déroulement de la génération d'impulsions

La figure suivante décrit le déroulement de la génération d'impulsions par le biais d'instructions Motion Control.



Une fois que les impulsions pour la commande du moteur pas à pas ont été générées par le biais des sorties et transmises à l'étage de puissance du moteur pas à pas, elles sont converties en un déplacement d'axe par l'étage de puissance du moteur pas à pas. La liaison entre les bandes transporteuses est entraînée par le biais de l'axe.

Étapes de travail

Vous procéderez de la manière suivante dans ce chapitre :

- Vous créez l'objet technologique "Axe".
- Vous configurez l'objet technologique "Axe" :
 - en affectant un générateur d'impulsions à l'objet technologique,
 - en configurant le générateur d'impulsions en tant que PTO (sortie de trains d'impulsions),
 - en affectant un compteur rapide (sélectionné automatiquement) au générateur d'impulsions.
- Vous créez deux instructions Motion Control :
 - une pour le déblocage de l'axe (MC_Power),
 - une pour le positionnement relatif par rapport à la position de la barrière photoélectrique "LS3" (MC_MoveRelative).

5.2 Insertion de l'objet technologique "Axe"

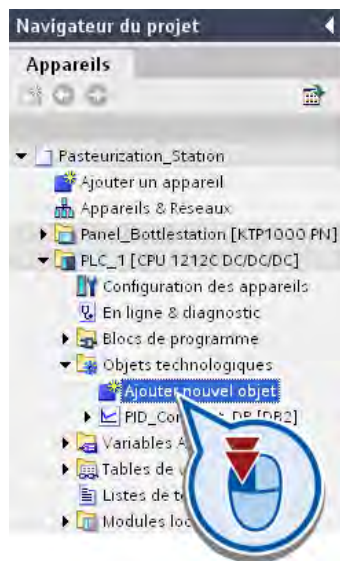
Introduction

Les étapes suivantes décrivent comment créer un nouvel objet technologique "Axe".

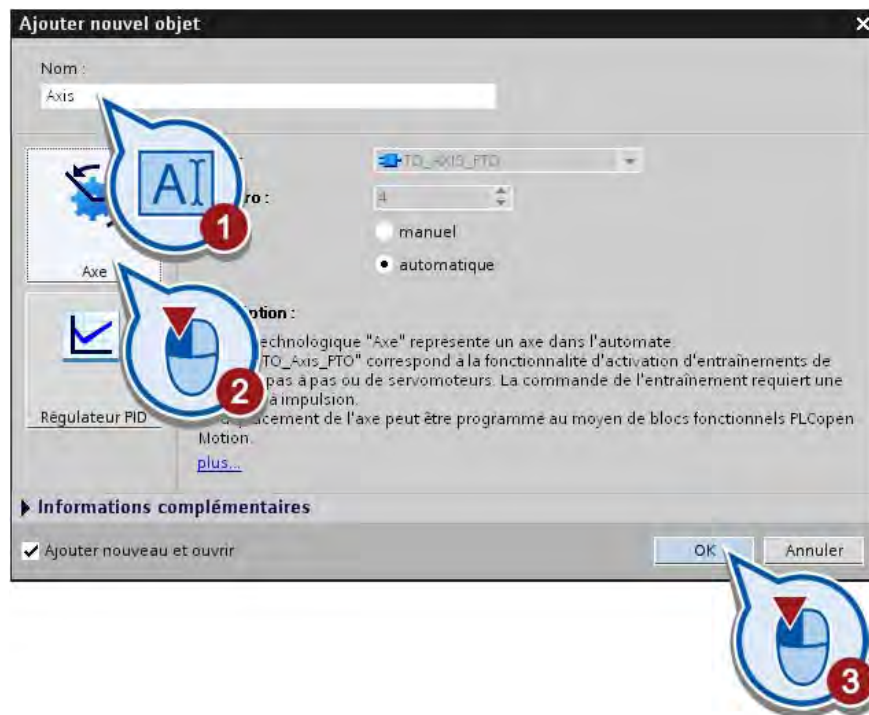
Marche à suivre

Procédez de la manière suivante pour créer l'objet technologique "Axe" :

1. Créez un nouvel objet technologique dans le navigateur du projet.



2. Créez un nouvel objet "Axis".



La configuration de l'objet technologique est démarrée dans la zone de travail.

Résultat

Vous avez créé l'objet technologique "Axe". Cet objet est rangé dans le dossier "Objets technologiques" dans le navigateur du projet.

Dans le paragraphe suivant, vous allez configurer l'objet technologique créé.

5.3 Configuration de l'objet technologique "Axe"

Introduction

Les étapes suivantes décrivent comment configurer l'objet technologique "Axe".

Condition requise

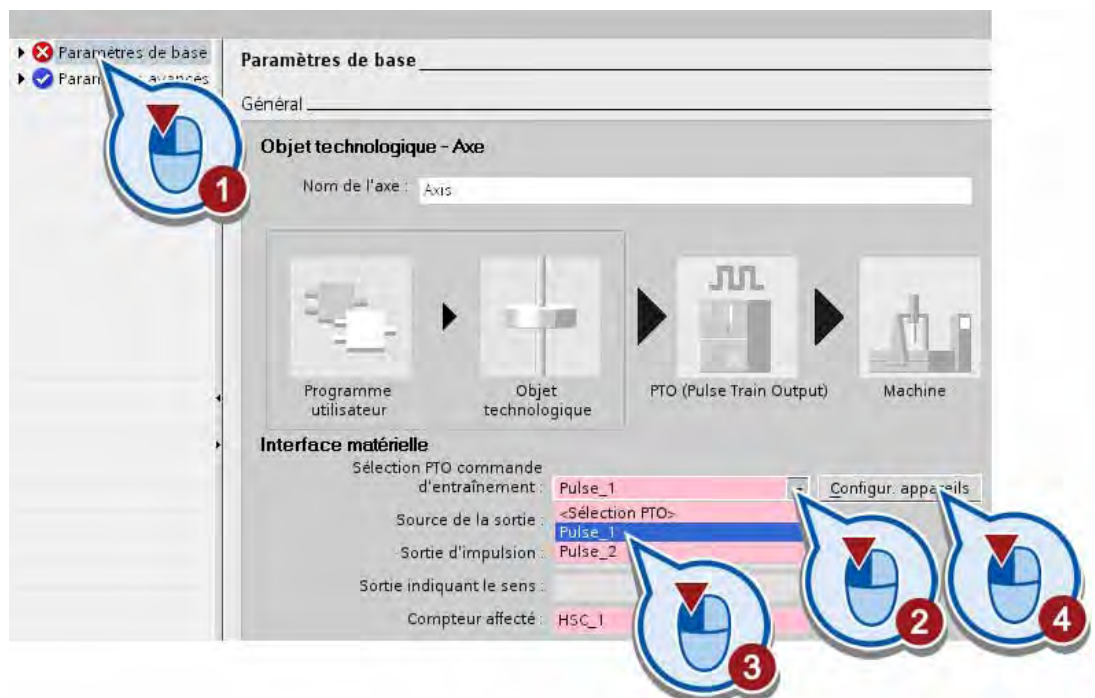
- L'objet technologique "Axis" a été créé.
- La configuration de l'objet technologique a été démarrée.

Si la configuration ne s'est pas ouverte automatiquement, démarrez-la dans le navigateur du projet.

Marche à suivre

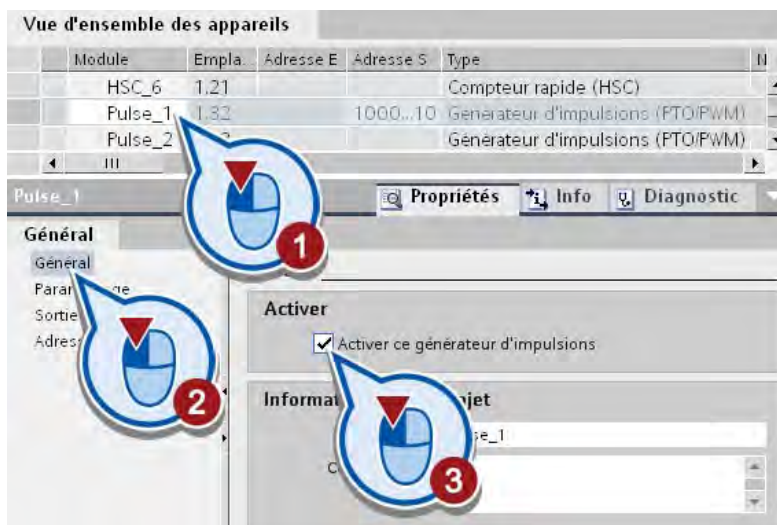
Procédez de la manière suivante pour configurer la commande de l'entraînement :

1. Sélectionnez le générateur d'impulsions "Pulse_1" pour la commande d'entraînement PTO et basculez dans la configuration des appareils.

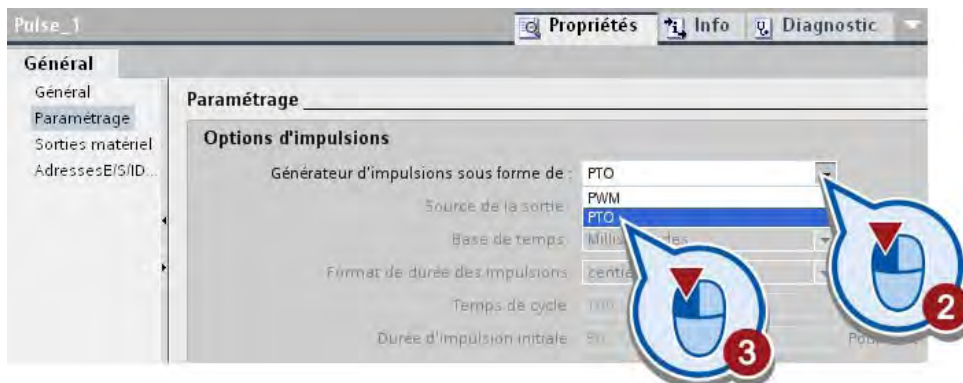


Le compteur rapide "HSC_1" est automatiquement affecté lors de la sélection de "Pulse_1". La vue des appareils s'ouvre.

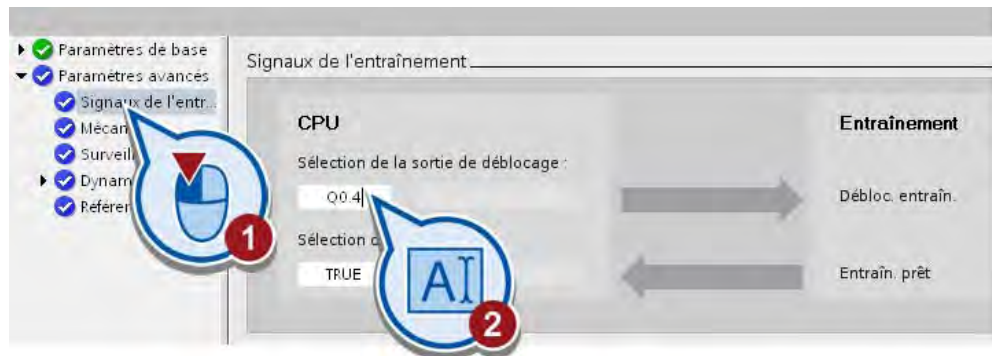
2. Activez le générateur d'impulsions.



3. Paramétrez le générateur d'impulsions en tant que PTO.



4. Basculez vers l'éditeur de configuration de l'objet technologique "Axis". Indiquez par quelle sortie de l'automate doit se faire le déblocage de l'entraînement.



5. Enregistrez le projet à l'aide du bouton "Enregistrer le projet" de la barre d'outils.

Résultat

Vous avez affecté le générateur d'impulsions "Pulse_1" à l'objet technologique "Axis" créé et l'avez paramétré en tant que PTO. Le compteur rapide "HSC 1" a été automatiquement activé. Le compteur rapide permet de compter les impulsions émises par le générateur d'impulsions.

Si vous avez configuré l'objet technologique correctement, l'état des zones "Paramètres de base" et "Paramètres avancés" s'affiche en vert dans la fenêtre de configuration.

5.4 Déblocage de l'axe

Introduction

Les étapes suivantes décrivent comment débloquent l'axe du moteur de la bande transporteuse.

Pour débloquent ou bloquer un axe, vous utilisez l'instruction Motion Control "MC_Power". Cette instruction doit être appelée une fois par axe dans le programme.

L'instruction Motion Control "MC_Power" permet de débloquent ou de bloquer l'axe de manière centralisée :

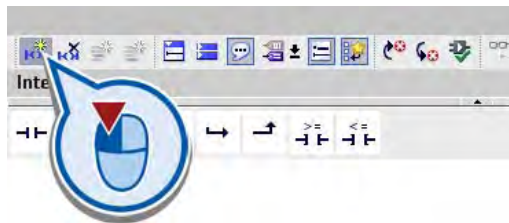
- Si l'axe est débloquent, le débloquent vaut pour toutes les instructions Motion Control auxquelles cet axe a été affecté.
- Si l'axe est bloqué, aucune autre instruction Motion Control n'a d'effet sur cet axe. Toutes les tâches en cours sont annulées.

Condition requise

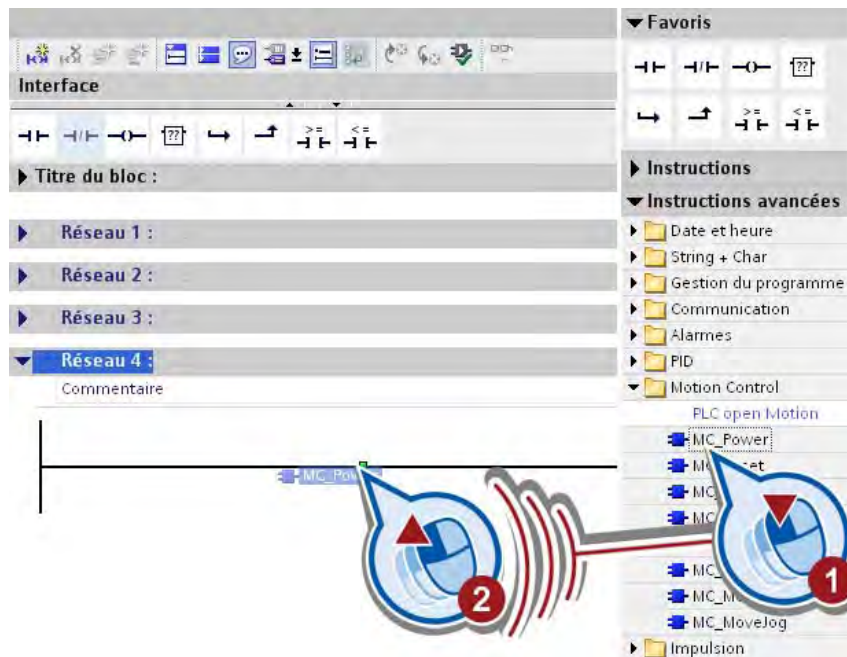
- L'objet technologique a été configuré correctement.
- Le bloc d'organisation "Main [OB1]" est ouvert.

Marche à suivre

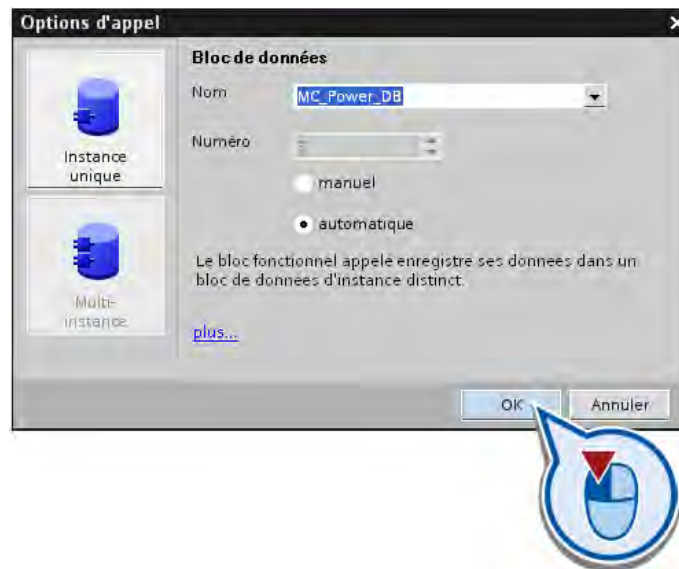
1. Créez un nouveau réseau dans le bloc d'organisation "Main [OB1]". Un nouveau réseau est automatiquement inséré dès que vous créez un élément dans un réseau vide. Si un réseau vide existe déjà, vous pouvez ignorer cette étape.



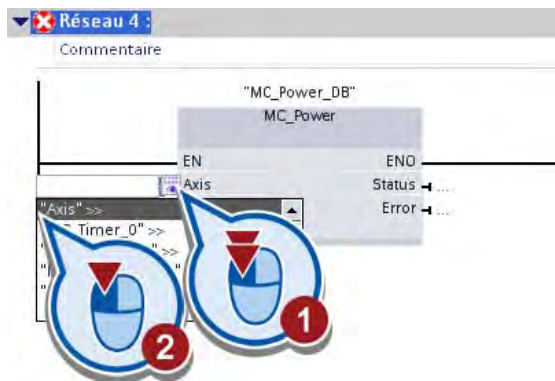
2. Créez le bloc Motion Control "MC_Power" dans le nouveau réseau.



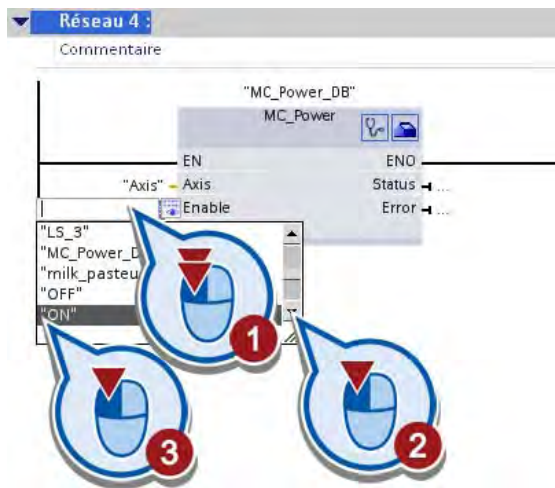
3. Confirmez la création du nouveau bloc de données.



4. Sélectionnez l'axe "Axis" précédemment configuré pour l'entrée "Axis".



5. Sélectionnez la variable "ON" pour l'entrée "Enable".



6. Enregistrez le projet.

Résultat

Vous avez inséré dans le programme l'instruction "MC_Power" pour le déblocage de l'axe et l'avez affectée à l'objet technologique "Axis".

Le déblocage de l'axe dépend de la valeur de la variable "ON" au niveau de l'entrée "Enable" :

- Si le bit de la variable "ON" a la valeur "0" (installation arrêtée), l'axe est bloqué.
- Si le bit de la variable "ON" a la valeur "1" (installation en marche), l'axe est débloquent.

Dans le paragraphe suivant, vous allez programmer le mouvement de la bande transporteuse relativement à un point de départ.

5.5 Positionnement relatif de l'axe

Introduction

Les étapes suivantes décrivent comment programmer le mouvement sur la deuxième bande transporteuse relativement à une position initiale à l'aide de l'instruction Motion Control "MC_MoveRelative"

Le mouvement est défini comme suit :

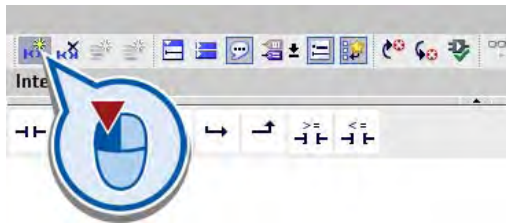
- La position initiale du mouvement est la position de la barrière photoélectrique "LS3".
- La position finale se situe à 0,5 m en sens positif sur l'axe entre la première et la deuxième bande transporteuse.
- La deuxième bande transporteuse est activée lorsque la position finale est atteinte.

Condition requise

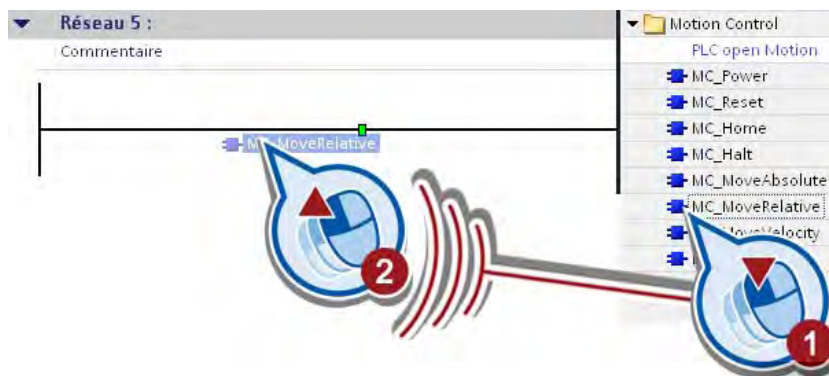
- L'objet technologique "Axe" a été créé et correctement configuré.
- L'instruction Motion Control "MC_Power" figure dans le bloc d'organisation "Main [OB1]".
- Le bloc d'organisation "Main [OB1]" est ouvert.

Marche à suivre

1. Créez un nouveau réseau dans le bloc d'organisation "Main [OB1]". Un nouveau réseau est automatiquement inséré dès que vous créez un élément dans un réseau vide. Si un réseau vide existe déjà, vous pouvez ignorer cette étape.



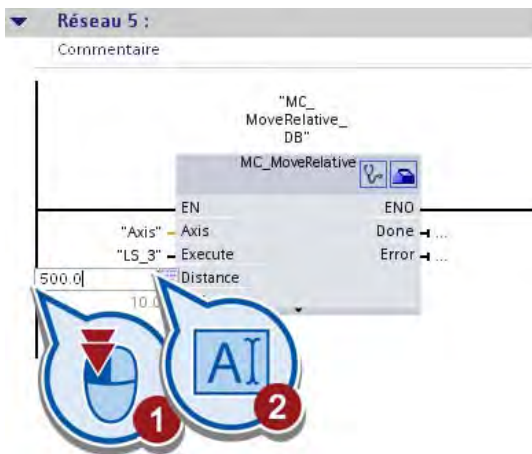
2. Créez le bloc "MC_MoveRelative" dans le nouveau réseau.



3. Dans la boîte de dialogue "Options d'appel", confirmez la création du nouveau bloc de données avec "OK".

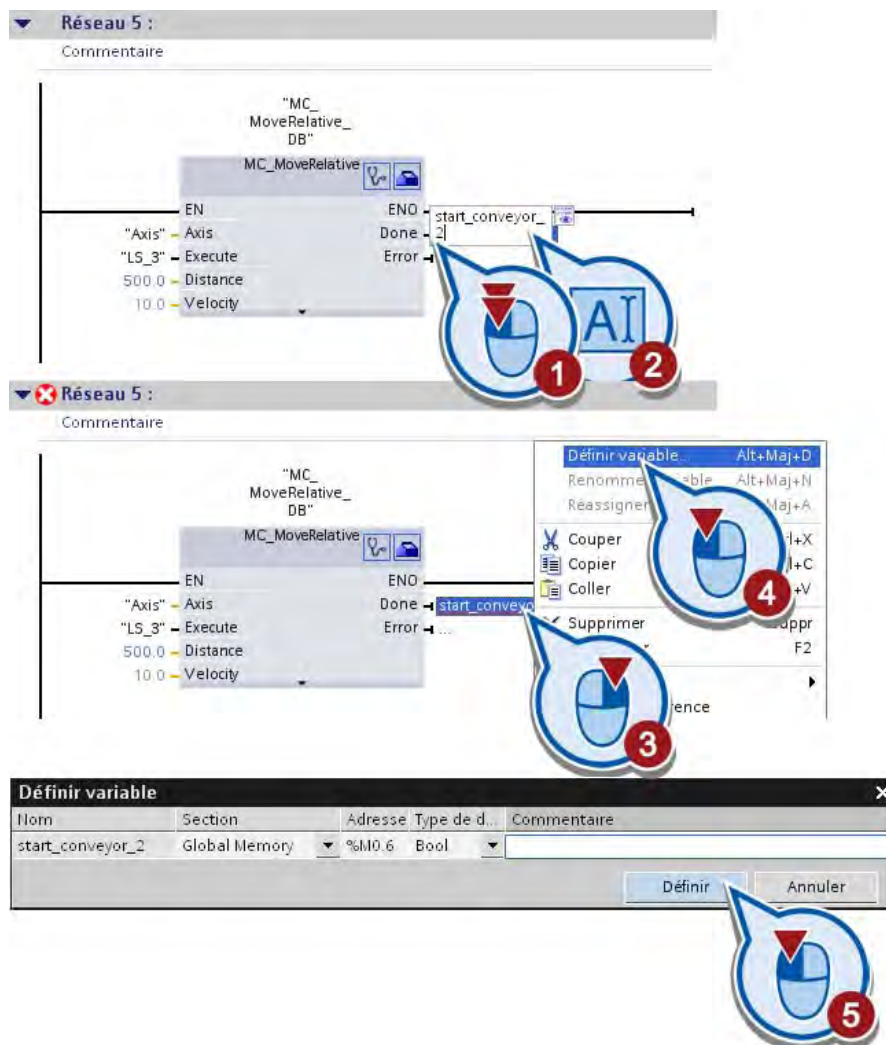


4. Sélectionnez l'axe "Axis" précédemment configuré pour l'entrée "Axis" de la boîte.
5. Sélectionnez la variable "LS_3" pour l'entrée "Execute" de la boîte.
6. Entrez la valeur "500.0" pour l'entrée "Distance" et confirmez avec la touche Entrée.



L'unité par défaut pour "Distance" est le millimètre et a été précédemment validée dans la configuration.

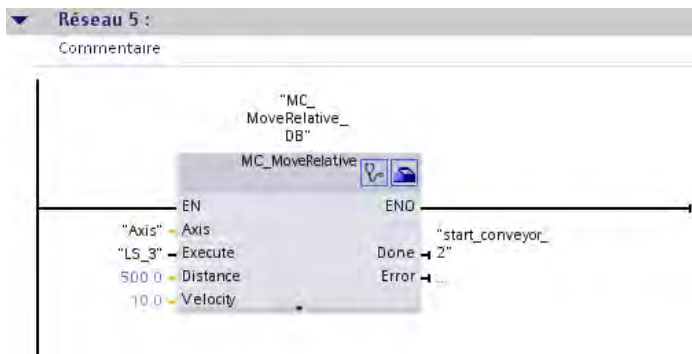
7. Créez une nouvelle variable "start_conveyor_2" pour la sortie "Done".



8. Enregistrez le projet à l'aide du bouton "Enregistrer le projet" de la barre d'outils.

Résultat

Vous avez programmé le mouvement de la bande transporteuse relativement à la position de la barrière photoélectrique "LS3".



Le mouvement démarre lorsque la barrière photoélectrique "LS3" est activée. Une fois la position finale atteinte, le bit de la variable "start_conveyor_2" est mis à "1" au niveau de la sortie "Done", ce qui met en mouvement la deuxième bande transporteuse.

Dans le paragraphe suivant, vous allez compléter la vue IHM avec d'autres éléments afin de visualiser le déroulement des processus programmés.

5.6 Compléter la vue IHM

5.6.1 Modification de l'objet graphique Bande transporteuse

Introduction

Les étapes suivantes décrivent comment remplacer dans la vue IHM l'objet graphique "Bande transporteuse" existant par le graphique "ConveyorMotion.wmf". Au lieu d'une seule bande transporteuse, ce graphique contient deux bandes transporteuses reliées entre elles.

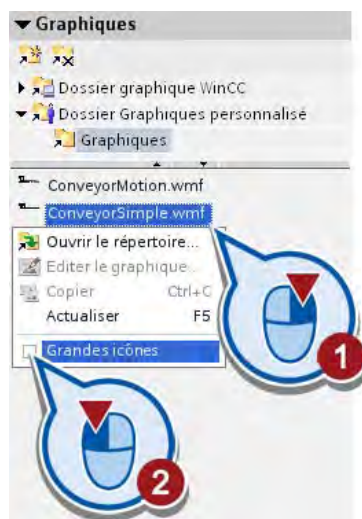
Condition requise

Le fichier ZIP "WinCC Graphics" a été extrait et stocké dans un répertoire local. Un raccourci a été créé pour le répertoire dans la palette "Graphiques" de la Task Card "Outils". Voir le paragraphe Objet graphique "Bande transporteuse" (Page 78).

Marche à suivre

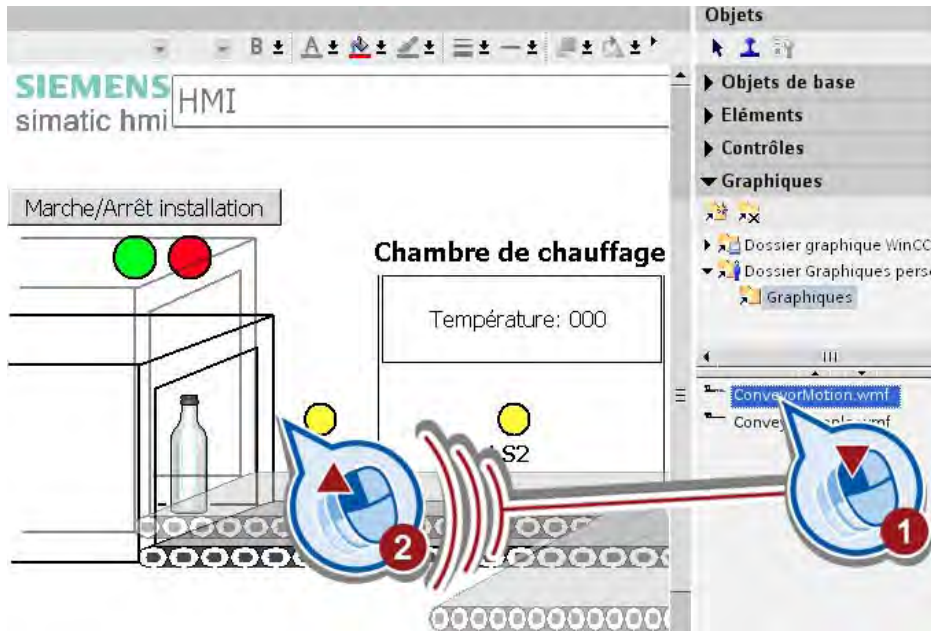
Procédez de la manière suivante pour compléter la vue IHM :

1. Ouvrez la vue IHM.
2. Ouvrez le raccourci vers le dossier local "WinCC Graphics" dans la palette "Graphiques" de la Task Card "Outils".
3. Désactivez l'option "Grandes icônes".

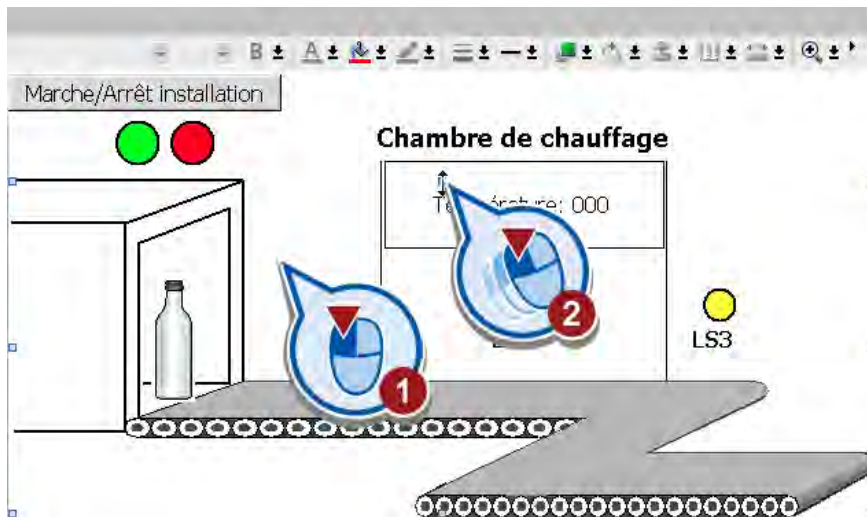


4. Faites glisser et déposez l'objet graphique "ConveyorMotion.wmf" dans la zone de l'éditeur pour remplacer la bande transporteuse existante.

Veillez à ce qu'une flèche apparaisse comme symbole à côté du pointeur de la souris lors de l'insertion. Si le pointeur de la souris est représenté par un signe plus, le graphique sera ajouté et non substitué.



5. Si nécessaire, mettez l'objet graphique à l'échelle.



Résultat

Vous avez remplacé l'objet graphique "Bande transporteuse".

Dans le paragraphe suivant, vous allez compléter la vue IHM avec un deuxième objet graphique animé.

5.6.2 Création d'un deuxième objet graphique Bouteille

Introduction

Dans la suite, vous allez créer une deuxième bouteille avec une animation de déplacement de la première vers la deuxième bande transporteuse.

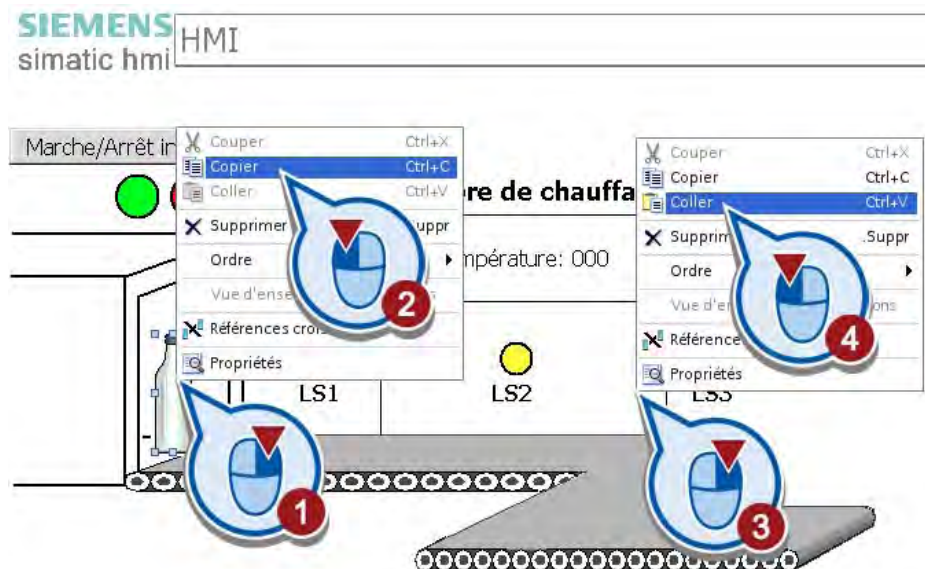
Condition requise

La vue IHM est ouverte.

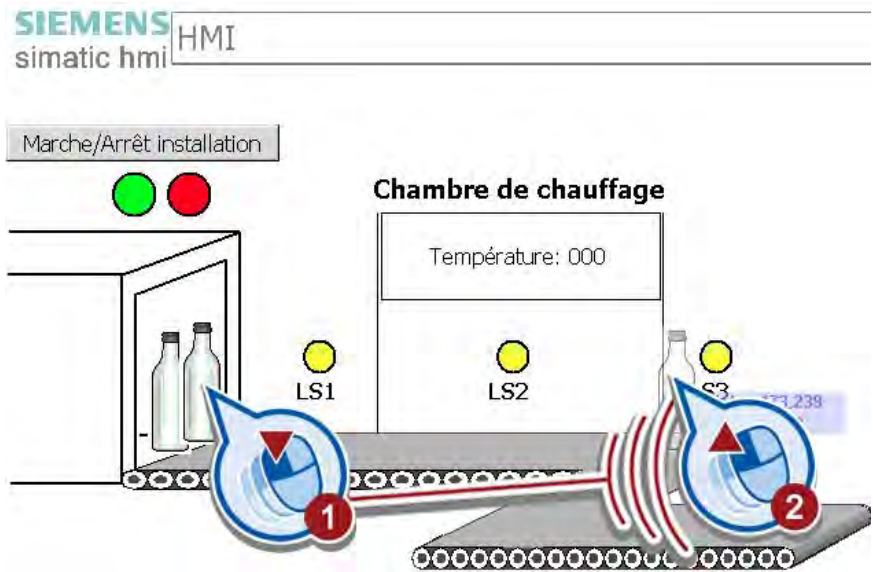
Marche à suivre

Procédez de la manière suivante pour animer la deuxième bouteille dans la vue IHM :

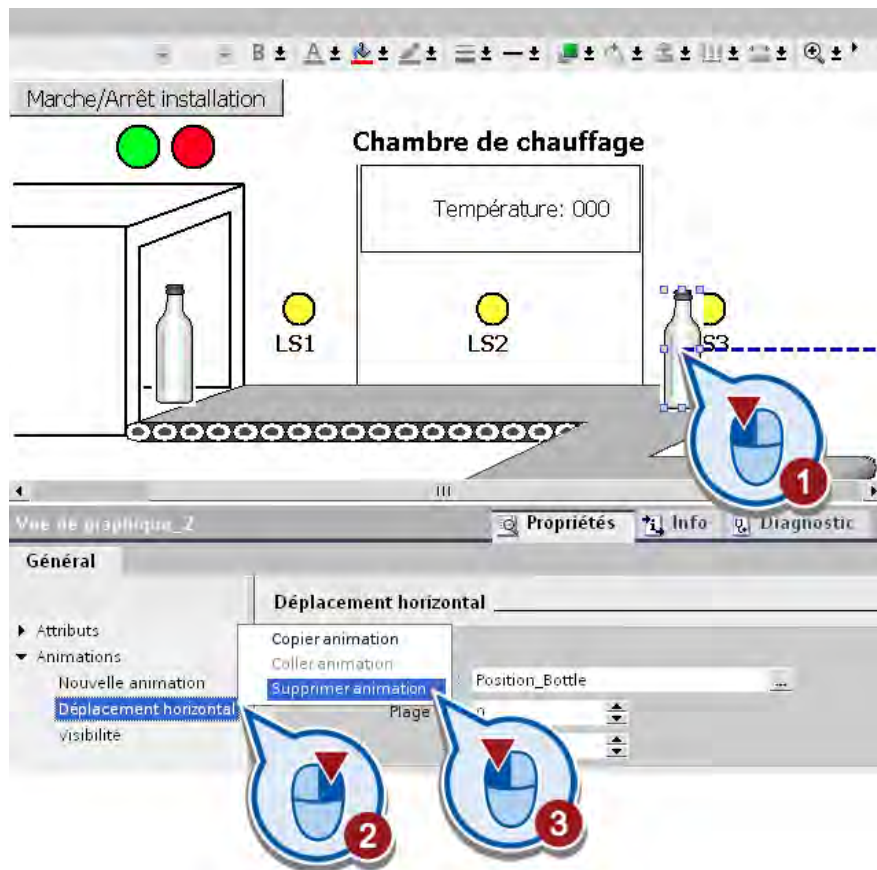
1. Copiez la bouteille existante.



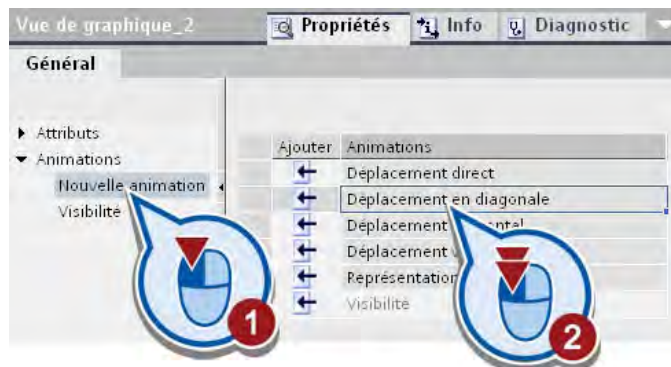
- 2. Positionnez la bouteille copiée à la fin de l'animation de déplacement de la première bouteille devant la barrière photoélectrique "LS3".



- 3. Effacez l'animation "Déplacement horizontal" de la bouteille copiée.

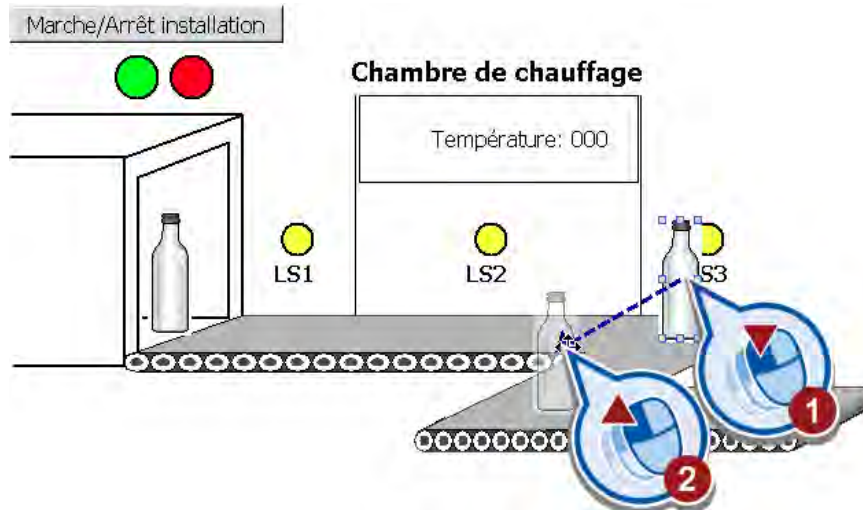


4. Créez une animation "Déplacement en diagonale".

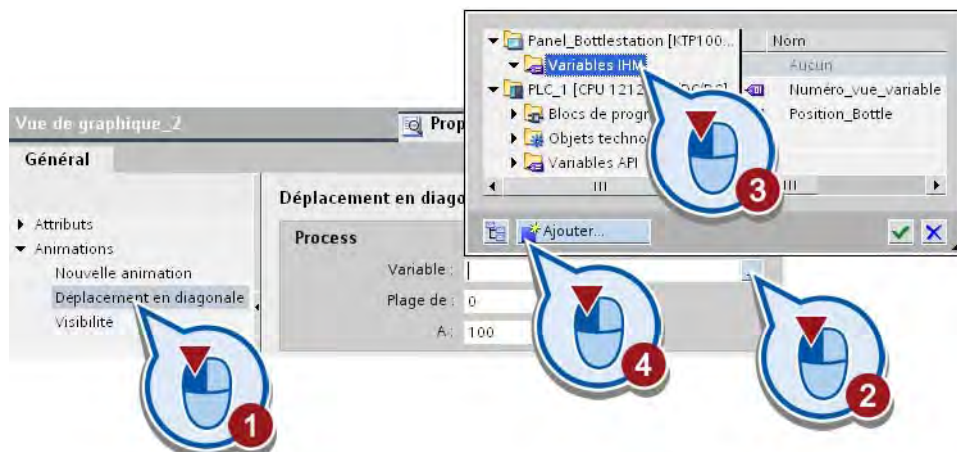


L'animation de déplacement est représentée avec une flèche bleue.

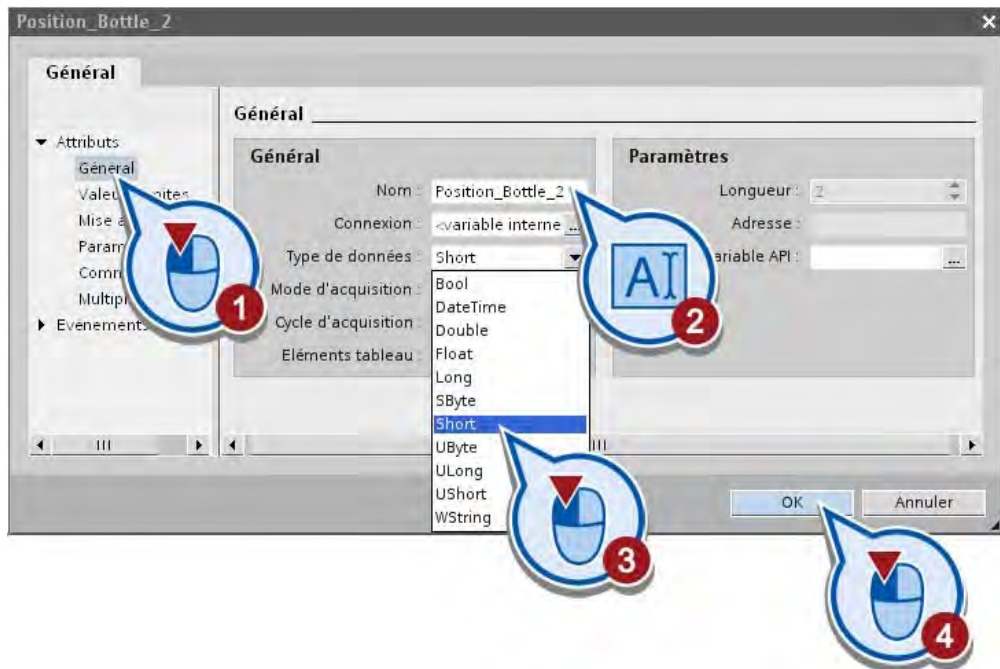
5. Faites glisser la représentation transparente de la bouteille à la pointe de la flèche sur la deuxième bande transporteuse.



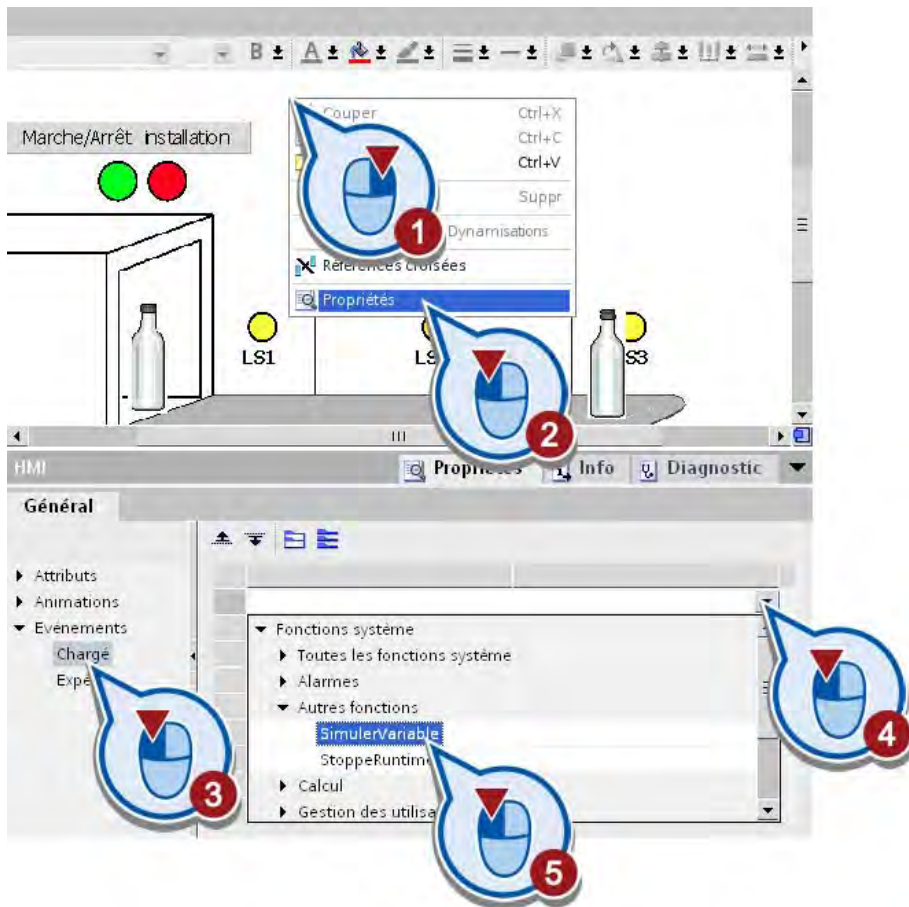
6. Créez une nouvelle variable IHM pour l'animation de déplacement en diagonale.



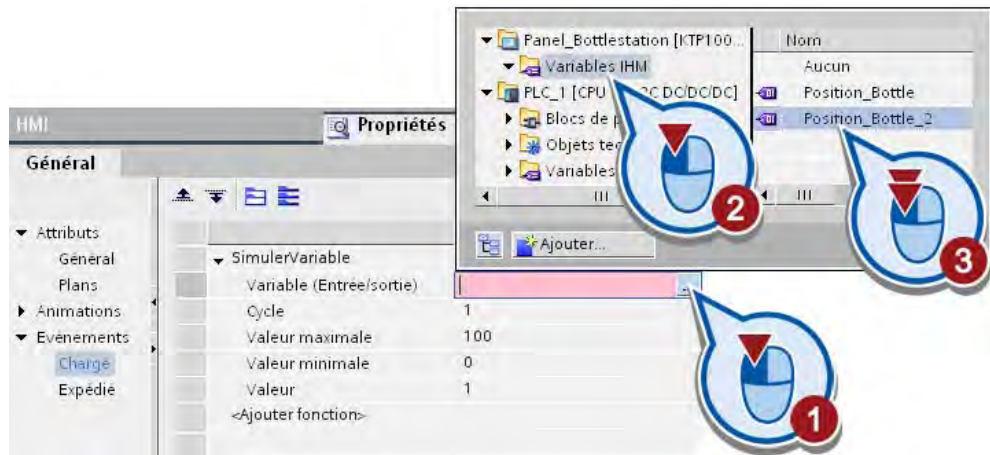
7. Utilisez "Position_Bottle_2" comme nom et "Short" comme type de données.



8. Ajoutez la fonction "SimulerVariable" à l'événement "Chargé" de la vue IHM.



9. Affectez la variable "Position_Bottle_2" à la fonction "SimulerVariable".



Résultat

Vous avez créé une animation qui visualise le déplacement de la deuxième bouteille de la première vers la deuxième bande transporteuse.

Dans le paragraphe suivant, vous allez modifier les paramètres de visibilité des bouteilles sur la bande transporteuse.

5.6.3 Association d'objets IHM avec une instruction Motion

Introduction

Les étapes suivantes décrivent comment animer la visibilité des bouteilles dans la vue IHM en fonction de la progression du programme.

Vous utilisez l'état logique du paramètre "Busy" de l'instruction "MC_MoveRelative" pour les paramètres de la visibilité. Ce paramètre indique si l'instruction Motion Control est actuellement en cours d'exécution ou non.

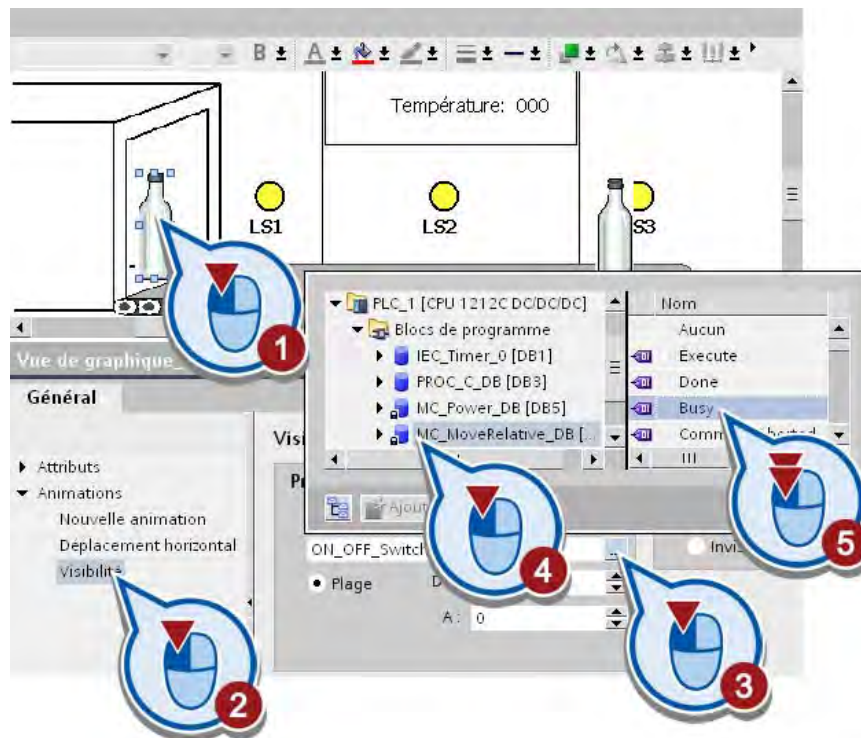
Condition requise

- L'instruction "MC_MoveRelative" a été insérée dans le bloc d'organisation "Main [OB1]".
- La vue IHM est ouverte.
- L'animation de déplacement pour la deuxième bouteille a été créée.

Marche à suivre

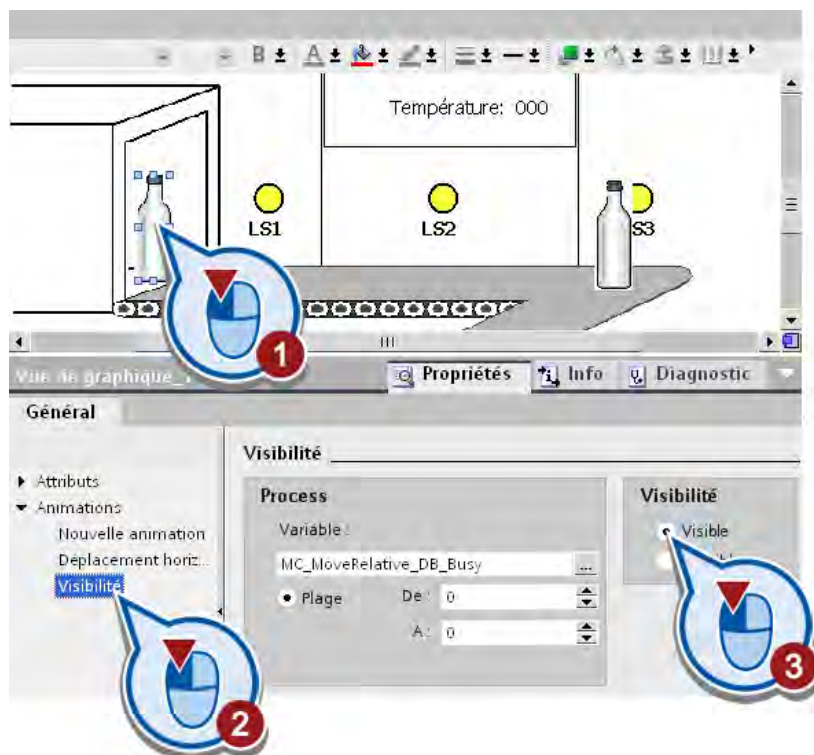
Procédez de la manière suivante pour animer la visibilité des bouteilles dans la vue IHM :

1. Associez l'animation "Visibilité" de la première bouteille au paramètre "Busy" du bloc de données "MC_Move Relative".



2. Associez également l'animation "Visibilité" de la deuxième bouteille au paramètre "Busy" du bloc de données "MC_Move Relative".

3. Modifiez la visibilité de la première bouteille pour la plage "0" à "0" en "Visible".



Laissez le paramétrage de la visibilité de la deuxième bouteille à "Invisible" pour la plage "0" à "0".

4. Enregistrez le projet à l'aide du bouton "Enregistrer le projet" de la barre d'outils.

Résultat

Vous avez paramétré la visibilité des bouteilles dans la vue IHM en fonction de l'état logique du paramètre "Busy" de l'instruction "MC_MoveRelative".

Lorsque l'instruction "MC_MoveRelative" est exécutée, la deuxième bande transporteuse est mise en mouvement. Le paramètre "Busy" de l'instruction fournit dans ce cas l'état logique "1", ce qui entraîne les conséquences suivantes pour la vue IHM :

- La première bouteille devient invisible.
- La deuxième bouteille devient visible et se déplace de la position de la troisième barrière photoélectrique vers la deuxième bande transporteuse.

Lorsque l'instruction "MC_MoveRelative" n'est pas exécutée, le paramètre "Busy" de l'instruction fournit l'état logique "0", ce qui entraîne les conséquences suivantes pour la vue IHM :

- La première bouteille devient visible et se déplace de la position de la première barrière photoélectrique vers la position de la troisième barrière photoélectrique.
- La deuxième bouteille n'est plus visible.

5.7 Simulation de la vue IHM

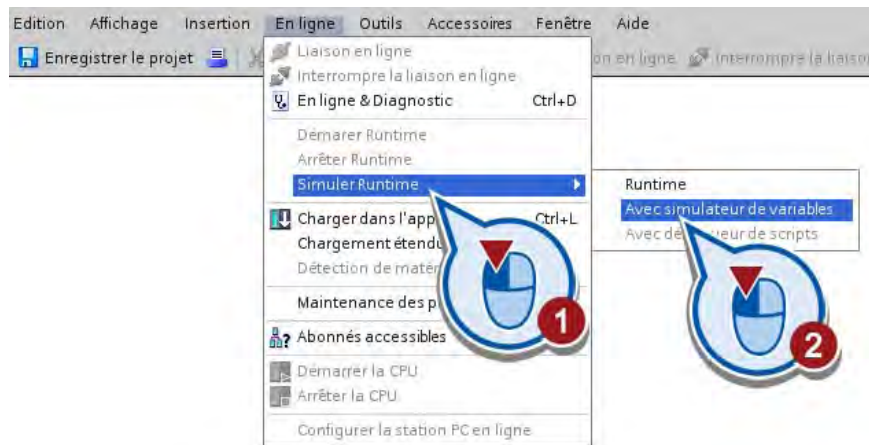
Introduction

Les étapes suivantes décrivent comment tester avec le simulateur Runtime la vue IHM créée. Avec le simulateur Runtime, vous simulez l'activation de l'entrée de l'automate pour la barrière photoélectrique "LS3".

Marche à suivre

Procédez comme suit pour démarrer la simulation de la vue IHM créée :

1. Ouvrez la vue IHM.
2. Démarrez la simulation Runtime par le biais de la barre des menus.



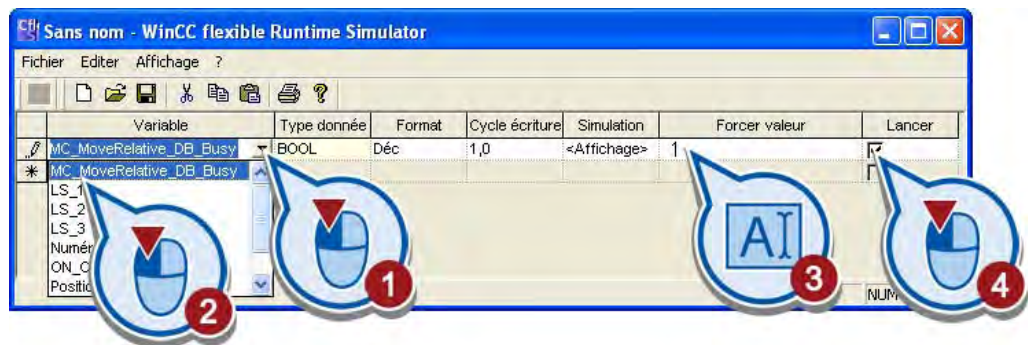
La simulation Runtime est lancée. Une fois la simulation lancée, la vue IHM est affichée dans la fenêtre "Simulateur RT" et la DEL rouge clignote (l'installation est arrêtée).

3. Démarrez l'installation.



La DEL verte clignote. Rien ne change dans la visibilité de la première bouteille, car cette animation dépend maintenant de la valeur de bit du paramètre "MC_MoveRelative_DB_Busy" qui a été automatiquement créé en tant que variable IHM.

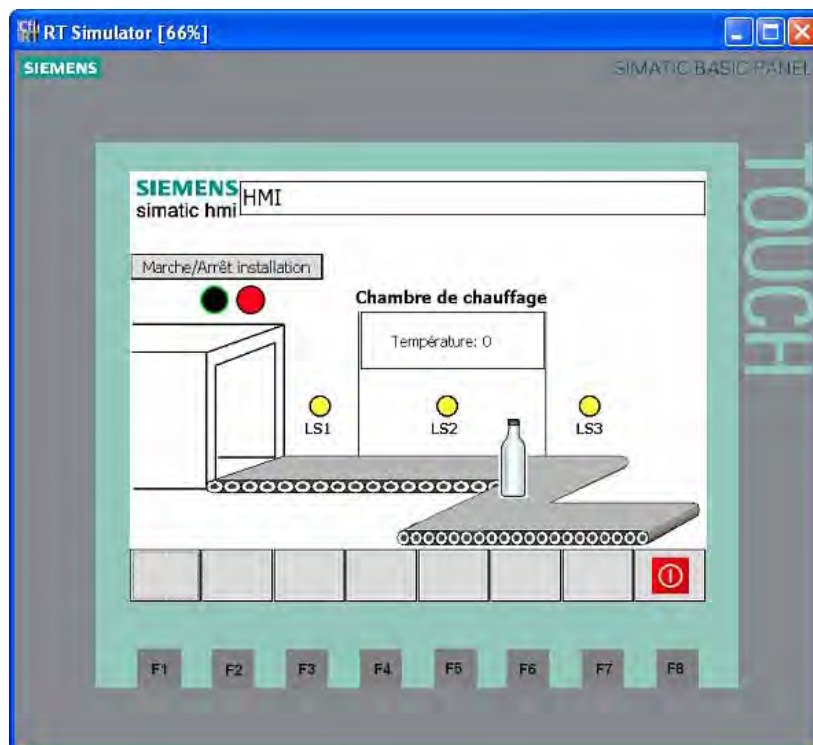
- Mettez la valeur de la variable IHM "MC_MoveRelative_DB_Busy" à "1".



- Basculez dans la fenêtre "Simulateur RT".

Résultat

La valeur de la variable "MC_MoveRelative_DB_Busy" est simulée. La bouteille se déplace sur la bande transporteuse relativement à la position de la barrière photoélectrique "LS3".



5.8 Démarrage de la vue de diagnostic

Introduction

Les étapes suivantes décrivent comment démarrer la vue de diagnostic de l'objet technologique "Axis". Vous pouvez surveiller les tâches de déplacement ainsi que les messages d'état et de défaut les plus importants de l'axe du moteur à l'aide de la fonction de diagnostic.

Condition requise

- L'objet technologique "Axe" a été créé et correctement configuré.
- L'instruction Motion Control "MC_Power" figure dans le bloc d'organisation "Main [OB1]".
- L'instruction Motion Control "MC_MoveRelativ" figure dans le bloc d'organisation "Main [OB1]".

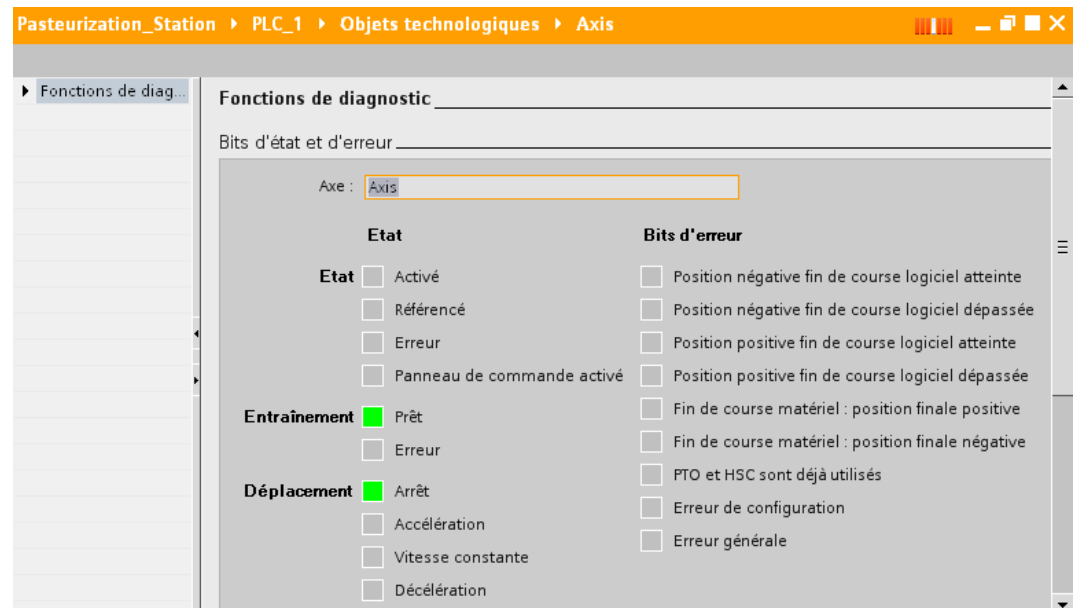
Marche à suivre

Procédez de la manière suivante pour démarrer la fonction de diagnostic :

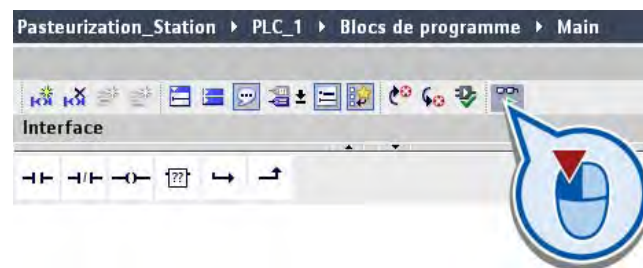
1. Chargez le programme dans l'automate et activez la liaison en ligne. Pour plus d'informations, reportez-vous au paragraphe "Chargement du programme dans le système cible (Page 51)".
2. Ouvrez la fenêtre de dialogue de l'objet technologique "Axis".



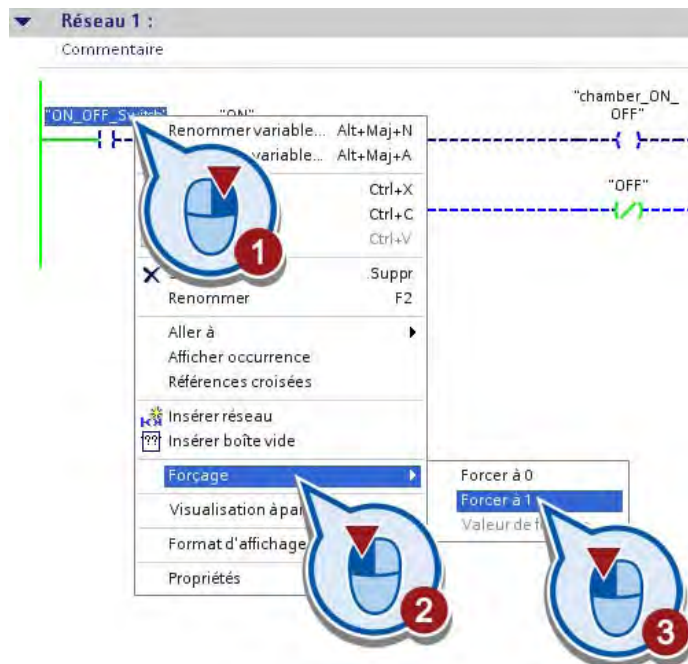
La fenêtre de diagnostic s'ouvre.



3. Ouvrez le bloc d'organisation "Main [OB1]".
4. Cliquez sur le bouton "Activer/désactiver visualisation du programme" dans la barre d'outils de l'éditeur de programmes.



- Dans le réseau 1, forcez la variable "ON_OFF_Switch" à "1".

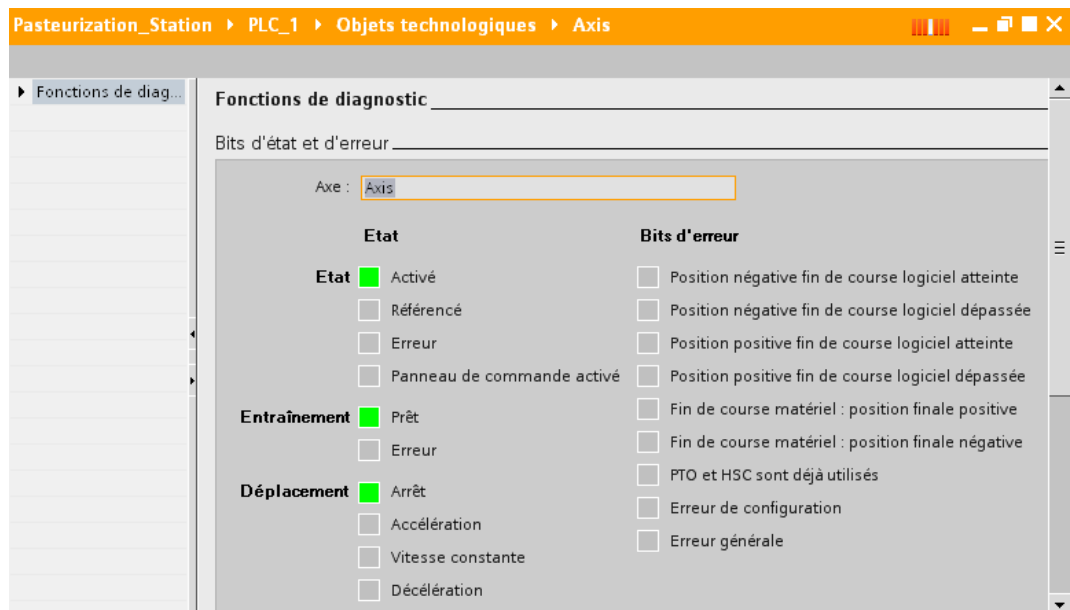


La variable ON_OFF_Switch est mise à l'état logique "1" dans le réseau 1. Le courant traverse le contact à fermeture et atteint les bobines à la fin du réseau. La variable "ON" est mise à "1", ce qui met l'installation en marche.

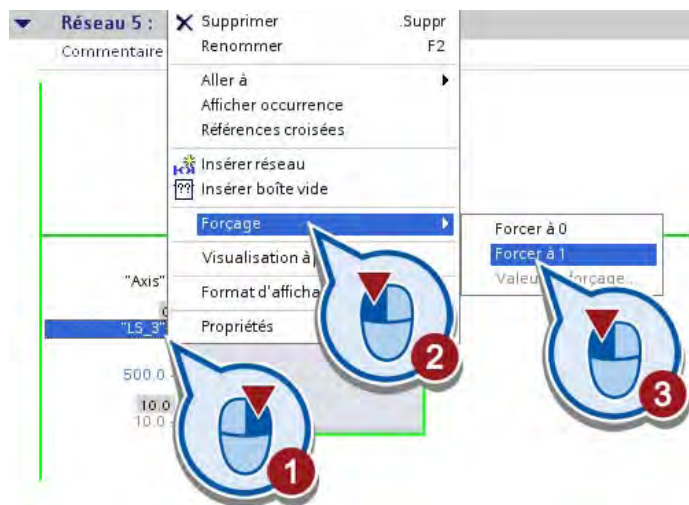
Dans le réseau 4, l'instruction "MC_Power" est exécutée et l'axe du moteur de la bande transporteuse est débloqué.

- Basculez dans le diagnostic de l'objet technologique "Axis".

Le déblocage de l'axe est visualisé dans la zone "Etat" de la fenêtre de diagnostic.



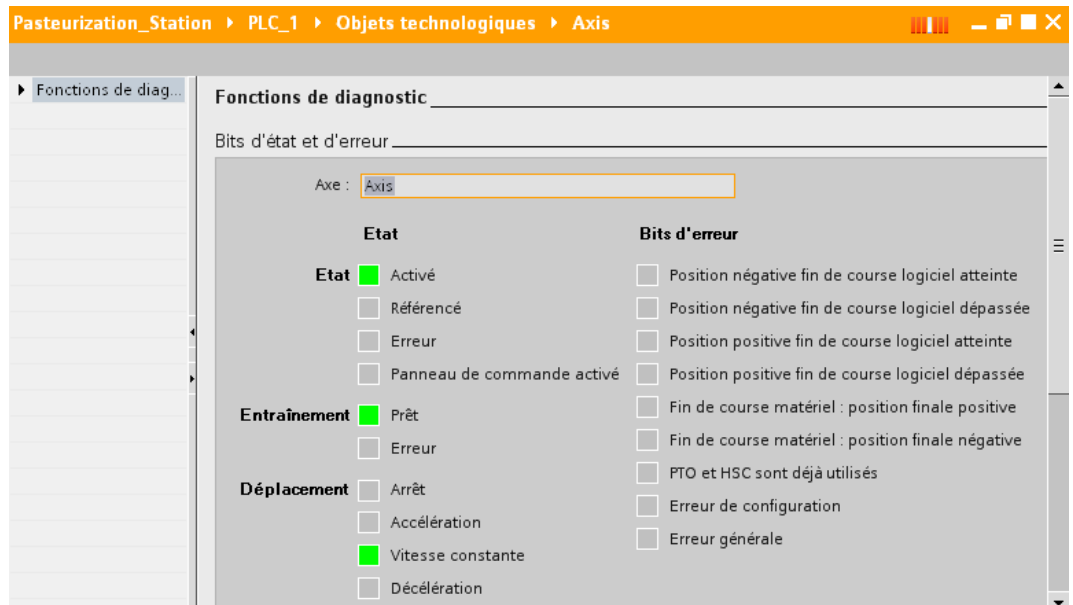
7. Dans le réseau 5, forcez la variable "LS_3" à "1".



L'instruction "MC_MoveRelative" est exécutée et l'axe du moteur de la bande transporteuse est mis en mouvement.

8. Basculez dans le diagnostic de l'objet technologique "Axis".

La zone "Déplacement" de la fenêtre de diagnostic montre que l'axe du moteur se déplace à vitesse constante.



9. Affichez l'état de déplacement.



La position actuelle de l'axe du moteur est visualisée dans le champ "Position actuelle". Lorsque la position cible de plus 500 mm relativement à la position initiale est atteinte, l'axe est arrêté.

Résultat

Vous avez testé le fonctionnement correct de l'axe du moteur à l'aide de la fonction de diagnostic.

Glossaire

Adressage

Affectation d'une adresse dans le programme utilisateur. Les adresses peuvent être affectées à des opérands ou plages d'opérands spécifiques. Exemples : entrée I12.1 ; mot de mémentos MW25.

Adresse

Caractérisation d'une adresse précise dans la zone des entrées, des sorties ou des mémentos de la CPU.

Alarme cyclique

Les OB d'alarme cyclique servent à démarrer des programmes à intervalles réguliers indépendamment du traitement cyclique du programme. Les moments de déclenchement d'un OB d'alarme cyclique sont indiqués via la périodicité et le décalage de phase.

API

Les automates programmables sont des commandes électroniques dont la fonction est sauvegardée sous forme de programme dans l'appareil de commande. Le montage et le câblage de l'appareil ne dépendent donc pas de la fonction de la commande. Un automate programmable est composé d'au moins un module d'alimentation, d'une CPU et de modules d'entrées et de sorties.

Bloc

Il structure le programme utilisateur en sections indépendantes. Il est possible de subdiviser des parties du programme utilisateur en blocs que l'on peut réutiliser à différents endroits ou qui permettent de clarifier la structure du programme utilisateur.

Bloc de données (DB)

Bloc dans le programme utilisateur pour la sauvegarde de valeurs ou de chaînes de caractères. Il existe des blocs de données globaux auxquels il est possible d'accéder à partir de n'importe quel bloc de code et des blocs de données d'instance affectés à un appel de FB spécifique.

Bloc de données d'instance

Un bloc de données d'instance contient les paramètres formels et les données statiques de blocs fonctionnels. Il peut être affecté à un appel de FB ou à une hiérarchie d'appel de blocs fonctionnels.

Bloc d'organisation

Les blocs d'organisation constituent l'interface entre le système d'exploitation de la CPU et le programme utilisateur. L'ordre de traitement du programme utilisateur y est défini.

Bloc fonctionnel (FB)

Conformément à CEI 1131-3, un bloc fonctionnel est un bloc de code avec des données statiques. Il permet de transmettre des paramètres dans le programme utilisateur. C'est la raison pour laquelle les blocs fonctionnels conviennent à la programmation de fonctions complexes récurrentes, telles que les régulations ou la sélection de modes de fonctionnement. Comme un FB dispose d'une mémoire (bloc de données d'instance), il est possible d'accéder à tout moment à ses paramètres à n'importe quel endroit du programme utilisateur.

Bobine

Les bobines permettent de commander des opérands binaires. Elles peuvent mettre à "1" ou à "0" un opérande binaire en fonction de l'état du résultat logique.

Boîte

Les boîtes sont des éléments du programme contenant des fonctions complexes. La boîte vide constitue une exception. Il s'agit d'une boîte générique à laquelle l'opération souhaitée sera substituée.

Champ d'E/S

Le champ d'E/S est un champ d'entrée et de sortie qui sert à l'affichage et à la modification de valeurs de variables.

Configuration

On entend par "configuration" la disposition, le paramétrage et la mise en réseau des appareils et des modules dans la vue des appareils ou dans la vue de réseau. Les châssis sont représentés de manière symbolique. Comme les châssis "réels", ils permettent d'enficher un nombre défini de modules.

Console de programmation

Ordinateur personnel de conception compacte spéciale pour environnements industriels. Une console de programmation (PG) est complètement équipée pour la programmation des systèmes d'automatisation SIMATIC.

Contact

Les contacts permettent d'établir ou d'interrompre une liaison de conduction de courant entre deux éléments. La conduction du courant s'effectue de gauche à droite. Les contacts vous permettent d'interroger l'état logique ou la valeur d'un opérande et de contrôler le flux de courant en fonction du résultat.

CPU

Le programme utilisateur est sauvegardé et traité dans l'unité centrale (CPU) d'un système d'automatisation. L'unité centrale contient le système d'exploitation, l'unité de traitement et les interfaces de communication.

Entrée

Zone de mémoire dans la mémoire système de la CPU (mémoire image des entrées) ou connexion à un module d'entrées.

Forcer des variables

La fonction "Forcer des variables" permet de forcer des variables d'un programme utilisateur et d'affecter des valeurs fixes à des variables spécifiques à un point prédéfini dans l'exécution du programme utilisateur.

Langage de programmation

Un langage de programmation permet de créer des programmes utilisateur et fournit à cet effet des éléments de langage spécifiques sous forme d'instructions graphiques ou textuelles. Ces instructions sont saisies par l'utilisateur dans un éditeur, puis compilées en un programme utilisateur exécutable.

Mémento

Zone de mémoire dans la mémoire système d'une CPU. Il est possible d'y accéder en écriture et en lecture (par bit, octet, mot et double mot). La zone des mémentos permet à l'utilisateur de sauvegarder des résultats intermédiaires.

Mémoire image

Les états logiques des modules d'entrées et de sorties TOR sont enregistrés dans la CPU, dans une mémoire image. On distingue la mémoire image des entrées (MIE) et la mémoire image des sorties (MIS).

La mémoire image des sorties (MIS) est transmise aux modules de sorties avant le traitement du programme utilisateur et la lecture de la mémoire image des entrées par le système d'exploitation.

La mémoire image des entrées (MIE) est lue par les modules d'entrées avant le traitement du programme utilisateur par le système d'exploitation.

Motion Control

Composant logiciel pour la commande d'entraînements. Les instructions Motion Control commandent l'objet technologique "Axe". L'objet technologique "Axe" est la représentation d'un axe dans l'automate et sert à la commande de moteurs pas à pas et de servomoteurs avec interface à impulsions.

Paramètres du bloc

Paramètres génériques dans des blocs réutilisables auxquels des valeurs effectives sont affectées à l'ouverture du bloc correspondant.

Programme

Un programme permet de résoudre une tâche d'automatisation constituant un tout logique. Il est affecté à un module programmable et peut être structuré en unités plus petites, p. ex. en blocs.

PTO

Abréviation de "Pulse Train Output" (sortie de trains d'impulsions). Certaines CPU, tel le S7-1200, peuvent générer via les sorties des suites d'impulsions rapides qui sont utilisées pour la réalisation d'opérations de commande de mouvement.

Pupitre opérateur

Appareil muni d'un écran pour l'affichage de l'état et de l'avancement du processus ainsi que pour la commande du programme utilisateur.

Régulateur PID

Le régulateur PID détecte en continu la valeur mesurée de la grandeur réglée à l'intérieur de la boucle de régulation et la compare à la consigne désirée. Il calcule à partir de l'écart de régulation résultant une grandeur réglante qui amène la grandeur réglée le plus rapidement et de la manière la plus stable possible à la valeur de consigne.

Régulation

Une régulation est un processus dans lequel le résultat (grandeur de sortie) influe sur une grandeur réglante par le biais d'une rétroaction.

Réseau

Le programme d'un bloc est subdivisé en réseaux. Les réseaux servent à structurer les programmes.

Runtime

Le logiciel Runtime exécute le projet en mode processus et permet le contrôle-commande de processus.

Sortie

Zone de mémoire dans la mémoire système de la CPU (mémoire image des sorties) ou connexion à un module de sorties.

Sous-réseau

Un sous-réseau englobe tous les abonnés du réseau reliés entre eux sans routeur. Il peut contenir des répéteurs.

Système cible

Système d'automatisation sur lequel s'exécute le programme utilisateur.

Système d'automatisation

Un système d'automatisation est un automate programmable (API) composé d'un châssis de base, d'une CPU et de différents modules d'entrées/sorties.

Système d'exploitation de la CPU

Le système d'exploitation organise l'ensemble des fonctions et opérations de la CPU qui ne sont pas liées à une tâche d'automatisation spécifique.

Table de visualisation

Sert à regrouper des variables du programme utilisateur afin de les visualiser, de les forcer et/ou de les forcer de manière permanente.

Table des variables

Table de définition des variables valables dans l'ensemble de la CPU.

Temps de cycle

Le temps de cycle est le temps requis par la CPU pour une exécution unique du programme utilisateur.

Type de données

Définit comment utiliser la valeur d'une variable ou d'une constante dans le programme utilisateur. Une variable de type BOOL, par exemple, ne peut prendre que les valeurs 1 ou 0.

Variable

Une variable est constituée d'une adresse et d'un nom symbolique qui est généralement utilisé plusieurs fois dans le projet. L'adresse (par exemple, d'une entrée ou d'un memento) est utilisée dans la communication avec le système d'automatisation. Les variables sont utilisées pour que, lors d'une modification d'adresse (d'une entrée, par exemple), la modification se fasse de manière centralisée au lieu de devoir être effectuée dans tout le programme utilisateur.

