

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

الجمهورية الجزائرية الديمقراطية الشعبية

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE

ECOLE SUPERIEURE EN SCIENCES APPLIQUEES
--T L E M C E N--



المدرسة العليا في العلوم التطبيقية
École Supérieure en
Sciences Appliquées

وزارة التعليم العالي والبحث العلمي

المدرسة العليا في العلوم التطبيقية
-تلمسان-

Mémoire de fin d'étude

Pour l'obtention du diplôme d'ingénieur

Filière : Génie industriel.

Spécialité : Management industriel et logistique.

Présenté par :

-NEHAD Mohamed Amine

-LOURMIL Djamel Eddine

Thème

La programmation et la supervision sous les outils STEP7
et WINCC-Flexible de la station MPS-testing

Soutenu publiquement, le 13/09/2020, devant le jury composé de :

❖ M. Mehdi SOUIER	MCA	Président
❖ Mme. Latéfa GHOMRI	MCA	Encadrant
❖ M. Mohammed El Amin MKEDDER	Ingénieur en R&D	Co-encadrant
❖ M. Fouad MALIKI	MCB	Examinateur
❖ M. Mohammed BENNEKROUF	MCB	Examinateur

Année universitaire : 2019/2020

Table des matières :

La liste des figures :.....	
Remerciements	
Introduction générale :	1
Chapitre I : Les systèmes automatisés.....	3
I. Introduction :.....	4
II.1.Historique :	4
II.2.Définition d'un système :	5
II.3.Structure d'un système automatisée :	7
II.4. La chaine d'information :.....	15
II.5. La chaine d'énergie :.....	16
II.6. Communication dans les systèmes automatisés :	18
II.7. L'avantage d'un système automatisé:	21
II.8.Les inconvénients d'un système automatisé :	22
III. Système MPS 500 : Présentation, description et Récolte des données :	22
III.1.MPS :	22
III.1.1. Historique :.....	22
III.1.2. Description du système MPS_FMS 500 :	23
III.1.3. Composante du système MPS500:	23
III.2. Notre Stations à étudier :	32
III.2.1.a. Description et fonctionnement :	32
III.2.1.b. Tableau des entrées / sorties / actionneurs :.....	33
III.2.1.c. Pupitre de la commande :	35
III.2.1.d. Pré-actionneurs :	35
III.2.1.e. Câblage :	36
IV. Conclusion :.....	36
Chapitre II : Les automates programmables	37
I. Introduction :.....	38
II. Les automates programmables :	38
II.1. Historique :	38
II.2. Définition :	39
II.3. Structure d'un API :	39
II.4. Structure interne d'un API :.....	42

II.5. Traitement du programme automate :	43
II.6. Principe de lecture et commande de l'API :	44
II.7. Critère de choix d'un automate :	45
III. L'automate programmable S7-300 :	45
III.1. Définition :	45
III.2. Structure interne de s7-300 :	46
III.3. Fonctionnalités :	47
III.4. Caractéristiques techniques:	47
III.5. Avantage :	47
IV. Langage de programmation :	48
IV.1. Graph7 :	48
IV.2. Langage CONT :	48
IV.3. Langage LOG :	49
IV.4. Langage LIST :	49
V. Logiciels utilisé pour la programmation et la simulation :	49
V.1. Logiciel Step7: (SIMATIC MANAGER - STEP7):	49
V.2. L'installation du logiciel SIEMENS SIMATIC STEP7 et PLCSIM :	51
VI. Conclusion :	58
Chapitre III : La supervision	59
I. Introduction :	60
II. La supervision :	60
III. L'interface homme machine :	67
IV. WINCC FLEXIBLE:	77
V. Conclusion :	83
Chapitre IV : La programmation et la création d'IHM.....	84
I. Introduction :	85
II. Création du projet dans l'éditeur SIMATIC Manager :	85
III. La création de notre IHM :	94
III.1. Introduction :	94
III.2. Les étapes à suivre :	94
III.3. Les modifications dans notre programme pour la réalisation d'IHM :	101
IV. Le résultat obtenu :	106
IV.1. Le cas de tri pas hauteur:	107
IV.2. Le tri par couleur :	114

V. Conclusion : 121
Conclusion générale : 122
Référence :
Résumé :

La liste des figures :

Figure I-1 Un système.

Figure I-2 Structure d'un système automatisé.

Figure I-3 Exemple d'un vérin.

Figure I-4 Exemple d'une vanne.

Figure I-5 Exemple d'un moteur.

Figure I-6 Exemple d'un afficheur.

Figure I-7 Fonctionnement d'un actionneur.

Figure I-8 Exemples d'actionneurs.

Figure I-9 Capteur de proximité.

Figure I-10 Capteur de pression.

Figure I-11 Capteur de position.

Figure I-12 Capteur de luminosité.

Figure I-13 Capteur d'humidité.

Figure I-14 Capteur de vitesse.

Figure I-15 Capteur de fumée.

Figure I-16 Les types d'information donné par le capteur.

Figure I-17 Un signal analogique.

Figure I-18 Un signal numérique.

Figure I-19 La chaîne d'information.

Figure I-20 Exemple d'une porte de garage.

Figure I-21 La chaîne d'information d'une porte de garage.

Figure I-22 La chaîne d'énergie.

Figure I-23 Fonctionnement de la chaîne d'énergie d'une porte de garage.

Figure I-24 Bus ASI.

Figure I-25 Bus CAN.

Figure I-26 Câble PROFIBUS.

Figure I-27 Câble Ethernet.

Figure I-28 La station MPS500.

Figure I-29 La station de distribution et contrôle.

Figure I-30 La sous station de distribution.

- Figure I-31** La station de contrôle.
- Figure I-32** La station de production.
- Figure I-33** La sous station de manipulation.
- Figure I-34** La sous station d'usinage.
- Figure I-35** La station d'assurance qualité.
- Figure I-36** La station d'assemblage.
- Figure I-37** La sous station rebot.
- Figure I-38** La station d'assemblage.
- Figure I-39** Le magasin.
- Figure I-40** La station de tri.
- Figure I-41** Le convoyeur.
- Figure I-42** Le pupitre de commande de la station de contrôle.
- Figure I-43** Pré-actionneur pneumatique.
- Figure I-44** Pré-actionneur électrique.
- Figure II-1** La structure d'un automate programmable.
- Figure II-2** La structure interne d'un automate programmable.
- Figure II-3** Les étapes de traitement d'un automate.
- Figure II-4** Le principe de lecture et commande dans un automate.
- Figure II-5** Les outils d'un API.
- Figure II-6** L'automate programmable s7-300.
- Figure II-7** La structure interne de s7-300.
- Figure II-8** Représentation de langage Graphe 7.
- Figure II-9** Représentation de langage CONT.
- Figure II-10** Représentation de langage LOG.
- Figure II-11** Représentation de langage LIST.
- Figure II-12** SIMATIC MANAGER.
- Figure II-13** Les diverses étapes pour réaliser un projet dans Step7.
- Figure II-14** Le choix de la langue.
- Figure II-15** L'acceptation des conditions de licence.
- Figure II-16** Le coche des barres qu'on a besoin.
- Figure II-17** L'acceptation des modifications des systèmes
- Figure II-18** Le choix de l'installation.
- Figure II-19** Le progrès de l'installation.
- Figure II-20** L'ouvrir de l'icône Simatickey pour l'installation des keys.

Figure II-21 L'installation des licences keys.

Figure II-22 Automation Licence Manager.

Figure II-23 Fin de l'installation

Figure III-1 Architecture de la supervision

Figure III-2 Représentation d'un synoptique.

Figure III-3 Représentation d'une courbe.

Figure III-4 Représentation des alarmes.

Figure III-5 Représentation d'une historisation.

Figure III-6 Représentation d'une gamme de fabrication.

Figure III-7 Des voyants.

Figure III-8 Exemples d'afficheurs.

Figure III-9 Exemples des organes d'émission des consignes.

Figure III-10 Les boutons poussoirs et les sélecteurs.

Figure III-11 Les terminaux d'exploitation.

Figure III-12 Lecteurs de carte.

Figure III-13 Tableau de bord.

Figure III-14 Imprimant.

Figure III-15 Invite de commande.

Figure III-16 Un opérateur écrit sur un écran graphique.

Figure III-17 Première souris.

Figure III-18 Micro-ordinateurs.

Figure III-19 Interface graphique

Figure III-20 Ecran.

Figure III-21 Représentation 3D

Figure III-22 Vocal.

Figure III-23 Différente représentations de fichier.

Figure III-24 Représentation en 2D.

Figure III-25 Représentation en 3D.

Figure III-26 Clavier.

Figure III-27 Dispositifs de pointage.

Figure III-28 Synthèse vocale.

Figure III-29 Code barre 2D.

Figure III-30 Code tactile.

Figure III-31 Lecteur d’empreinte.

Figure III-32 Reconnaissance visage.

Figure III-33 Les éditions de WinCC flexible.

Figure III-34 Acceptation les conditions de licence.

Figure III-35 Le choix de la langue.

Figure III-36 Le choix de l’installation complète.

Figure III-37 Automation Licence Manager.

Figure IV-1 Click sur l’icône de STEP7 MANAGER.

Figure IV-2 La création et l’insertion de projet.

Figure IV-3 La configuration matérielle.

Figure IV-4 La table mnémonique.

Figure IV-5 La liaison Ethernet.

Figure IV-6 La programmation de notre station Testing selon le cahier de charge.

Figure IV-7 Le lancement de PLCSIM.

Figure IV-8 Le chargement du bloc dans PLCSIM.

Figure IV-9 La visualisation.

Figure IV-10 La simulation de notre programme.

Figure IV-11 La pièce est au niveau de poste de contrôle de hauteur

Figure IV-12 La pièce est éjecté vers le poste 2

Figure IV-13 La pièce est éjecté vers le poste 3

Figure IV-14 L’icône de WinCC Flexible.

Figure IV-15 La création d’un projet.

Figure IV-16 Sélectionner petite machine.

Figure IV-17 Sélection du pupitre.

Figure IV-18 Informations sur le projet.

Figure IV-19 L’intégration dans le projet STEP7.

Figure IV-20 Le sélection du projet STEP 7.

Figure IV-21 L’icône de configuration.

Figure IV-22 La configuration sur HW Config.

Figure IV-23 La configuration sur Netpro.

Figure IV-24 Les modifications dans notre programme.

Figure IV-25 L’interface de travail sur WinCC.

Figure IV-26 La fenêtre des propriétés.

Figure IV-27 L'interface de liaison.

Figure IV-28 La vue de notre IHM.

Figure IV-29 La table de variable.

Figure IV-30 Le tri par hauteur en état initial.

Figure IV-31 La présence de la pièce rouge.

Figure IV-32 La présence de la pièce bleue.

Figure IV-33 La présence de la pièce bleue en bas.

Figure IV-34 L'éjection de la pièce bleue vers le poste 2.

Figure IV-35 La présence de la pièce rouge.

Figure IV-36 L'éjection de la pièce rouge vers le poste 3.

Figure IV-37 Le vérin en haut + Pas de pièce.

Figure IV-38 Le tri par couleur en état initial.

Figure IV-39 La présence de la pièce noire.

Figure IV-40 L'éjection de la pièce noire vers le poste 2.

Figure IV-41 La présence de la pièce non noir (rouge).

Figure IV-42 La présence de la pièce rouge en haut.

Figure IV-43 Ejection de la pièce rouge vers le poste 3.

Figure IV-44 Pas de pièces au niveau de haut.

Les tableaux :

Tableau 1- Les entrées/ sorties de la station de contrôle.

Remerciements

Au terme de ce travail, nous remercions **Dieu** le tout puissant de nous avoir donné la force et la patience pour réaliser ce travail;

Nous remercions nos parents pour tous leurs sacrifices, leurs amours, leurs tendresses et leurs soutiens tout au long de notre formation;

Nous tenons à exprimer notre profonde gratitude et nos vifs remerciements à nos encadrants **Madame Latifa GHOMRI** et **Monsieur Mohammed El Amin MKEDDER** pour tout le temps précieux qu'ils nous ont consacré dans le but de la réalisation de ce travail;

Nous tenons également à adresser nos sincères remerciements à tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail;

Nous tenons à remercier tous les enseignants de la filière « **Génie industriel** » de l'**Ecole Supérieure des Sciences Appliquées de Tlemcen** pour leur contribution à notre formation, et également nos camarades, amis pour leurs aides précieuses, leurs encouragements permanents et leurs soutiens moraux ;

Nos remerciements vont aussi aux membres du jury de nous avoir fait l'honneur de bien vouloir participer au jury de ce mémoire : **Monsieur Mehdi SOUIER** de nous avoir accepté de présider notre jury de mémoire, **Monsieur Fouad MALIKI** et **Monsieur Mohammed BENNEKROUF** pour toute l'attention qu'ils vont prêter à l'évaluation de notre travail.

- NEHAD Mohamed Amine
- LOURMIL Djamel Eddine

Introduction générale :

Les systèmes automatisés sont de plus en plus présents dans notre environnement. En effet, ils accomplissent les tâches pénibles et répétitives à notre place. Dans l'industrie, ils remplacent les ouvriers et effectuent des tâches de production, de manutention, de contrôle, de montage, etc. Ce qui a pour effet de diminuer les coûts de production, de la croissance de la productivité, l'amélioration de la sécurité de travail, l'augmentation de la qualité du produit et de la flexibilité de production. La commande des processus par l'automate programmable est la solution recherchée de plus en plus dans l'industrie vue la justesse des traitements qu'il effectue pour générer une commande exacte à tout moment et dans toutes les conditions. Le SIMATIC constitue une vaste plateforme d'automatisation offrant des solutions à des problèmes complexes pour tous les secteurs d'activité. Le logiciel STEP7 a été conçu dans un souci d'homogénéité et de complémentarité avec un système de contrôle et de commande, offrant des fonctions conviviales de conduite et de simulation du processus, ce qui simplifie d'une manière considérable la mise en œuvre de nombreuses caractéristiques du système de commande, notamment la gestion de base de données communes.

Les entreprises aujourd'hui recherchent, de plus en plus, des solutions globales qui regroupent les systèmes automatisés au sein d'un même processus et assurent, ainsi, un flux continu d'informations pour pouvoir suivre chaque phase du procédé et intervenir dans le cas échéant. En effet, avec le développement de l'informatique, il est devenu possible de traiter des données dans le domaine industriel, grâce à des vues préalablement créées et configurées, et à l'aide d'un logiciel adéquat.

Le logiciel de supervision WinCC Flexible est une entité capable de présenter à l'opérateur des informations utiles, afin qu'il prenne à temps les bonnes décisions pour la conduite du procédé. Il a essentiellement, pour mission de collecter les données et les mettre en forme (traitement), afin de les présenter à l'opérateur (supervision).

Le contexte dans lequel s'inscrit notre travail consiste à étudier le fonctionnement détaillé de la station MPS-testing qui se trouve dans le laboratoire MELT à l'université de Tlemcen, à programmer la station via Step7 avec le langage graphique et à développer une solution de commande et de supervision par WINCC Flexible à base d'un automate API S7-300.

Pour atteindre notre objectif, nous avons organisé notre travail comme suit :

-Le premier chapitre est consacré à une présentation sur les systèmes automatisés et les systèmes MPS et aussi à la présentation de notre station à étudier.

- Le deuxième chapitre contient une présentation sur les automates programmables industriels (API) et sur le logiciel Step7 ainsi ces langages de programmation et comment l'installer.
- Le troisième chapitre contient une présentation sur la supervision et les IHM ainsi une présentation sur le WinCC flexible 2008 et comment l'installer.
- Le quatrième chapitre sera consacré à la programmation de notre station MPS-testing avec le langage graphique et leur simulation via PLCSIM et aussi à la création d'une interface IHM de supervision par le WINCC Flexible qui nous permet de visualiser le fonctionnement de notre station MPS-testing et de nous donner la possibilité de la commander par l'interface crée. Nous terminerons par une conclusion générale.

Chapitre I : Les systèmes automatisés

I. Introduction :

Dans ce chapitre on présentera les notions théoriques relatives à notre travail. Nous allons voir les différents outils que nous avons utilisés pour la réalisation de ce travail. Ces outils sont la station MPS et ces sous-stations, la station de contrôle, et aussi les langages de programmation que nous avons utilisée pour la réalisation de ce travail (STEP7).

Nous avons eu besoin de comprendre toutes les notions présentées dans ce chapitre surtout le fonctionnement de la station MPS TESTING pour que nous puissions la programmer dans le STEP7.

II. Les systèmes automatisés:

II.1 Historique :

"Depuis toujours l'homme est en quête de bien être". Cette réflexion peut paraître bien éloignée d'un cours de Sciences Industrielles, pourtant c'est la base de l'évolution des sciences en général, et de l'automatisation en particulier. L'homme a commencé par penser, concevoir et réaliser. Lorsqu'il a fallu multiplier le nombre d'objets fabriqués, produire en plus grand nombre, l'automatisation des tâches est alors apparue : remplacer l'homme dans des actions pénibles, délicates ou répétitives.

Le développement des connaissances, et des outils mathématiques, ont conduit à un formidable essor des systèmes automatisés, dans la deuxième moitié du 20ème siècle.

Les systèmes automatisés industriels sont apparus à la fin des années soixante, à la demande de l'industrie automobile américaine (GENERAL MOTORS), qui réclamait plus d'adaptabilité de leurs systèmes de commande. Les coûts de l'électronique permettant alors de remplacer avantageusement les technologies actuelles.

- **Avant :**

Utilisation des relais électromagnétiques et des systèmes pneumatiques pour la réalisation de la partie commande —> La logique câblée. [1]

- **Inconvénients :**

Cher, pas de flexibilité, pas de communication possible.[1]

- **Solution :**

Utilisation des systèmes à la base de microprocesseur permettant une modification aisée des systèmes automatisée → La logique programmée.[1]

- **Contraintes du monde industriel :**

- **Influences externes :**

- Poussières.
- Températures.
- Humidité.
- Vibration.
- Parasites électromagnétiques.

- **Personnel :**

- Pas de langage de programmation complexe.
- Possibilité de modifier le système au cours de fonctionnement.

- **Matériel :**

- Evolutif.
- Modulaire. [1]

II.2.Définition d'un système :

II.2.1.Système :

La fonction globale de tout système (**Figure 0-1**) est de conférer une valeur ajoutée, à un ensemble de matières d'œuvre dans un ou un contexte donné. De plus, un système de production est dit « industriel » si l'obtention de cette valeur ajoutée, pour un ensemble de matières d'œuvre donné, a un caractère reproductible et peut être exprimée et quantifiée en termes économiques. [2]

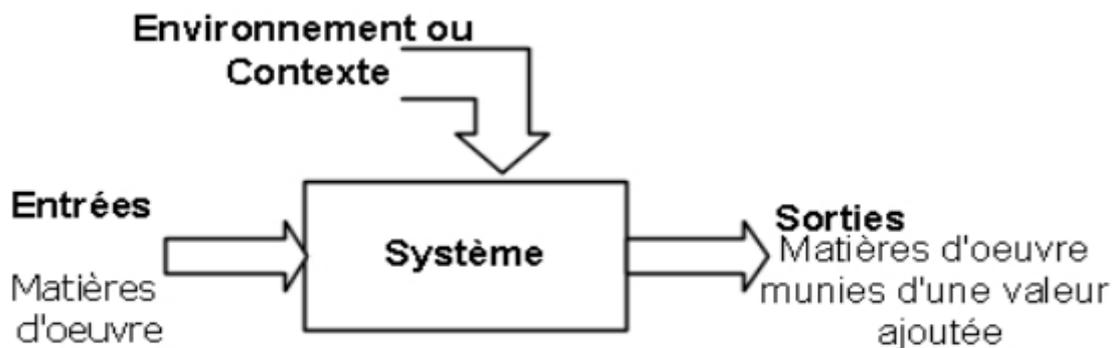


Figure I-1 Un système.

La matière d'œuvre peut se présenter sous plusieurs formes à savoir :

- Produit : liquide, solide, gazeux.
- Énergie : électrique, thermique, mécaniques.
- Information : physique, audiovisuel. [2]

La valeur ajoutée est caractérisée par sa nature, sa quantité et sa qualité. Elle peut être soit:

- Une modification physique: conversion d'énergie, mécanique, etc.
- Un arrangement particulier: montage, assemblage, etc.
- Un prélèvement d'information : mesure, contrôle, etc. [2]

Environnement ou contexte : c'est le contexte physique, social, économique, politique, etc. qui joue un rôle essentiel dans le fonctionnement du système et influe sur la qualité et la quantité de la valeur ajoutée. [2]

II.2.2. Système automatisée :

Un système automatisé est un ensemble d'éléments en interaction, et organisés dans un but précis : agir sur une matière d'œuvre afin de lui donner une valeur ajoutée. [1]

Un système est dit automatisé s'il exécute toujours le même cycle de travail après avoir reçu les consignes d'un opérateur, il réalise des opérations et pour lequel l'homme n'intervient que dans la programmation du système et dans son réglage. Les buts d'un système automatisé sont de réaliser des tâches complexes ou dangereuses pour l'homme, effectuer des tâches pénibles ou répétitives ou encore gagner en efficacité et en précision. [2]

II.3. Structure d'un système automatisée :

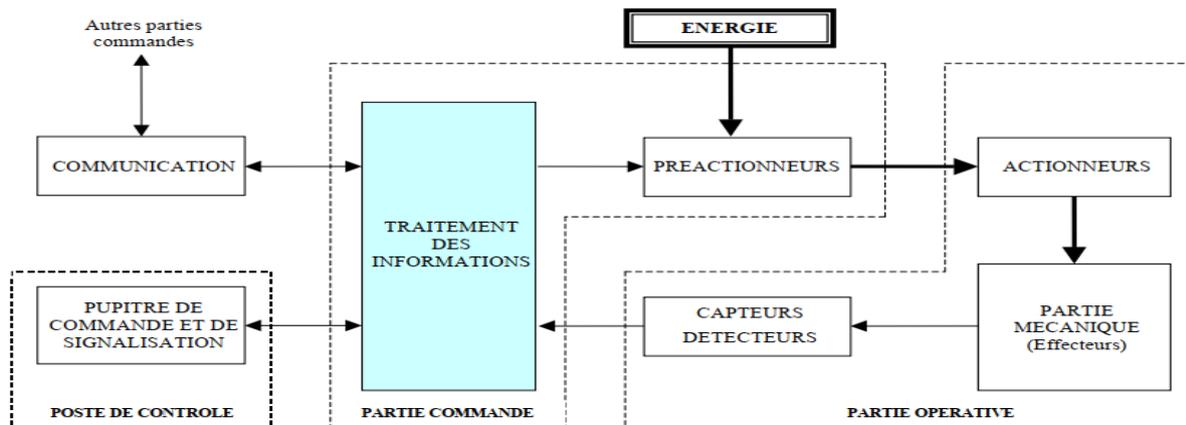


Figure I-2 Structure d'un système automatisé

- **Partie opérative :**

Elle agit sur la matière d'œuvre afin de lui donner sa valeur ajoutée.

Les actionneurs (moteurs, vérins) agissent sur la partie mécanique du système qui agit à son tour sur la matière d'œuvre.

Les capteurs / détecteurs permettent d'acquérir les divers états du système.

- **Poste de contrôle :**

Composé des pupitres de commande et de signalisation, il permet à l'opérateur de commander le système (marche, arrêt, départ cycle ...). Il permet également de visualiser les différents états du système à l'aide de voyants, de terminal de dialogue ou d'interface homme-machine (IHM).

- **Partie commande :**

Elle donne les ordres de fonctionnement à la partie opérative.

Les pré-actionneurs permettent de commander les actionneurs ; ils assurent le transfert d'énergie entre la source de puissance (réseau électrique, pneumatique ...) et les actionneurs. Exemple : contacteur, distributeur ...

Ces pré-actionneurs sont commandés à leur tour par le bloc traitement des informations.

Celui-ci reçoit les consignes du pupitre de commande (opérateur) et les informations de la partie opérative transmises par les capteurs / détecteurs.

En fonction de ces consignes et de son programme de gestion des tâches (implanté dans un automate programmable ou réalisé par des relais (on parle de logique câblée)),

elle va commander les pré-actionneurs et renvoyer des informations au pupitre de signalisation ou à d'autres systèmes de commande et/ou de supervision. [2]

II.3.a. Les actionneurs :

- **Définition :**

Un actionneur est un convertisseur électromécanique conçu pour mettre en mouvement un système mécanique à partir d'une commande électrique ou pour convertir une énergie en un autre.

- **Quelques exemples d'actionneurs :**

1-Les vérins :

Le vérin peut être pneumatique ou hydraulique, sert à créer un mouvement mécanique.

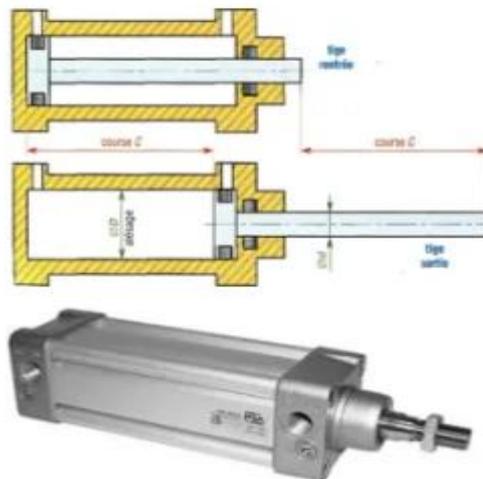


Figure I -3 Exemple d'un vérin

2-Les vannes :

La vanne est un dispositif destiné à contrôler le débit d'un fluide, liquide et gazeux en milieu libre ou un milieu fermé.



Figure I -4 Exemple d'une vanne

3-Les moteurs :

Un moteur est un appareil transformant une énergie quelconque en énergie mécanique.



Figure I -5 Exemple d'un moteur

4-Les afficheurs/voyants :

Permet de créer un échange visuel, soit par un voyant, soit par un message textuel ou numérique.



Figure I -6 Exemple d'un afficheur

Pour exécuter les ordres de la partie commande, la partie opérative est équipée de d'actionneurs.

Les actionneurs sont le plus souvent des composants électroniques capables de produire un phénomène physique (déplacement, dégagement de chaleur, émission de lumière...) à partir de l'énergie qu'il reçoit. [3]

II.3.b. Les Pré Actionnaires :

Le rôle du pré-actionneur est de distribuer, en le modulant si besoin, et sur ordre de la partie commande, l'énergie utile et importante aux actionneurs.

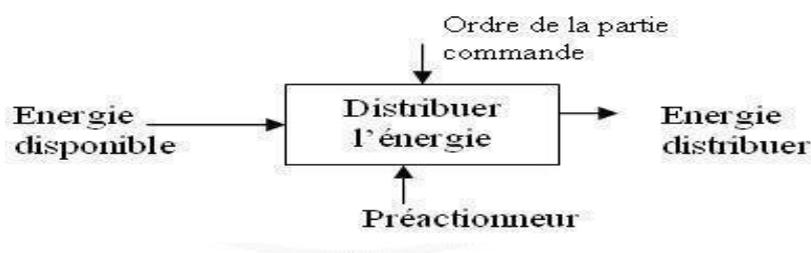


Figure I -7 Fonctionnement d'un actionneur

Si l'actionneur qui suit dans la chaîne fonctionnelle est électrique, le pré-actionneur sera aussi électrique (relais, contacteur, variateur, hacheur, carte de puissance).

Si l'actionneur est pneumatique, le pré-actionneur sera pneumatique (distributeur).

Certains pré-actionneurs (relais, contacteur, distributeur) sont dits « tout ou rien », c'est-à-dire qu'ils jouent le rôle d'interrupteur de la chaîne d'énergie.

Les autres pré-actionneurs (variateur, hacheur, carte de puissance) laissent passer seulement une partie de l'énergie source, c'est-à-dire qu'ils régulent le débit d'énergie, on parle alors de « pré-actionneur proportionnel ». [4]



Figure I -8 Exemples d'actionneurs

II.3.c. Les capteurs :

- **Définition :**

Le capteur est un dispositif transformant l'état d'une grandeur physique observée en une grandeur utilisable.

Les capteurs se trouvent beaucoup dans notre vie, par exemple : lorsque nous passons les portes de notre supermarché, les portes s'ouvrent automatiquement dès notre détection, une balance donne la masse d'un objet, un radar flash en cas ou en dépasse la vitesse régulière, un thermomètre donne la température ambiante.

- **Quelque exemple des capteurs :**

1-Capteur de proximité :



Figure I -9 Capteur de proximité

Il s'appelle aussi un détecteur de présence.

Exemple d'application : détecteur de présence dans un jardin pour allumer une ampoule.

2-Capteur de pression :



Figure I -10 Capteur de pression

Une sonde de pression est un dispositif destiné à convertir les variations de pression en variations de tension électrique.

Exemples d'application : un pèse-personne, pression de l'air dans un pneu

3-Capteur de position, de fin de course :

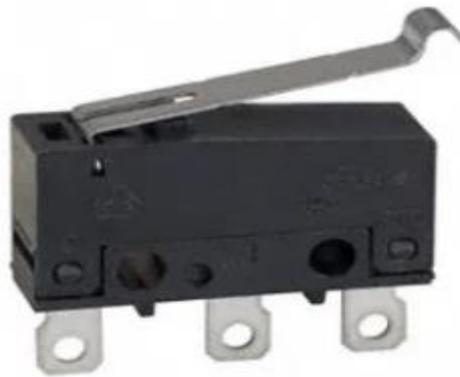


Figure I -11 Capteur de position

Un capteur de position est un dispositif qui permet de recueillir des informations sur la position d'un objet dans un espace de référence.

Exemples d'application : butée de fin de course dans un ascenseur, un portail.

4-Capteur de luminosité :

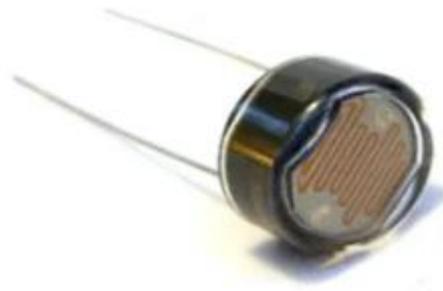


Figure I -12 Capteur de luminosité

Un capteur de luminosité est un capteur qui détecte la luminosité ambiante dans un environnement.

Exemples d'application : activation automatique des phares dans une voiture en cas d'obscurité.

5-Capteur d'humidité :

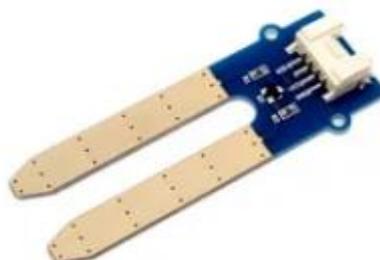


Figure I -13 Capteur d'humidité

Un capteur d'humidité permet de mesurer l'humidité ambiante dans un environnement.

Exemples d'application : arrosage automatique d'une plante lorsqu'elle manque d'eau

6-Capteur de vitesse :



Figure I -14 Capteur de vitesse

Le capteur de vitesse, un composant indispensable pour assurer le fonctionnement de plusieurs systèmes embarqués, par exemple permet de mesurer la vitesse de rotation magnétique afin de fournir une tension qui correspond à la vitesse de rotation dans le secteur automobile. [5]

Exemples d'application : Mesure de la vitesse d'un moteur, traduit en km/h pour l'utilisateur.

7-Capteur de fumée :



Figure I -15 Capteur de fumée

Un capteur de fumée est un élément de sécurité qui réagit à la présence de fumée ou de particules de vapeur dans l'air.

Exemples d'application : Détecteur de fumée avec une alarme pour prévenir un incendie. [3]

La partie opérative est également équipée de capteurs.

Un capteur est un élément capable de détecter (avec ou sans contact) un phénomène physique

dans son environnement (présence ou déplacement d'un objet, chaleur, lumière) et de rendre compte de ce phénomène à la partie commande.

Il convertit les informations physiques de la PO en grandeurs électriques exploitables par la PC. L'information peut être :

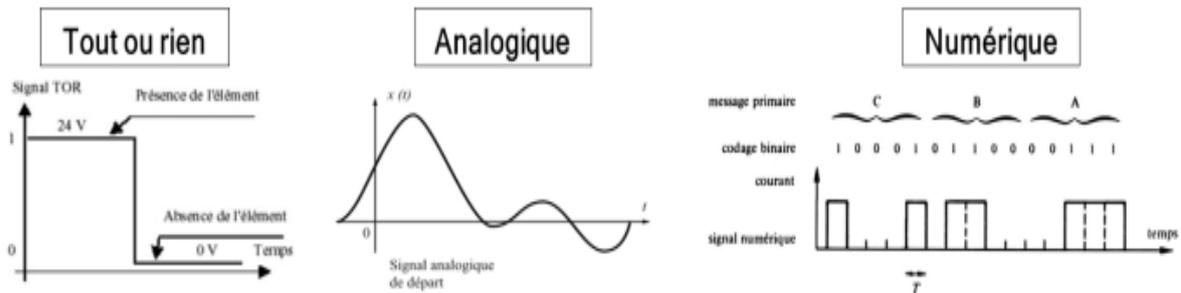


Figure I -16 Les types d'information donnés par le capteur

- Tout ou rien (TOR) :

En automatisme, le concept TOR (tout ou rien), se ramène au système binaire c'est-à-dire 0 ou 1, cela signifie que l'information traitée peut prendre que 2 états : marche ou arrêt.

Il y a des capteurs de type TOR, ces capteurs ne renverront que deux niveaux logiques :

0 = absence d'objet.

1 = présence d'objet.

- La différence entre un signal analogique et numérique :

o Définition d'un signal analogique :

Un signal analogique est un signal variant continuellement dans le temps.

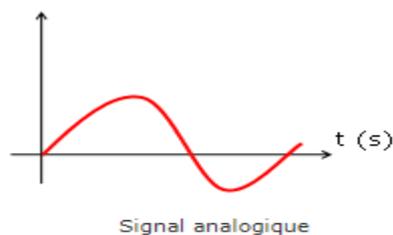


Figure I -17 Un signal analogique

o Définition d'un signal numérique :

Un signal numérique est un signal variant de façon discontinue dans le temps.

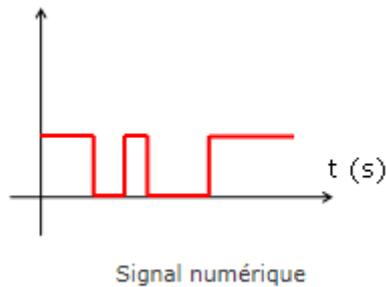


Figure I -18 Un signal numérique

Le signal analogique est également continu en amplitude ce qui signifie qu'il peut prendre toutes les valeurs possibles (même des valeurs à virgules contrairement à un signal numérique qui n'est composé que de nombres entiers). [6]

II.3.d. Effectuer :

- **Définition :**

Un effecteur est un ensemble qui utilise de l'énergie, sous la forme qui lui est adaptée, pour produire un effet utile sur la matière d'œuvre en lui conférant une certaine valeur ajoutée. Dans une chaîne d'action, l'effecteur est le dispositif terminal qui agit directement sur la matière d'œuvre traitée par le système. [7]

II.4. La chaîne d'information :

- **Définition :**

La chaîne d'information est la partie du système automatisé qui capte l'information et qui la traite. On peut découper cette chaîne en plusieurs blocs fonctionnels. [3]

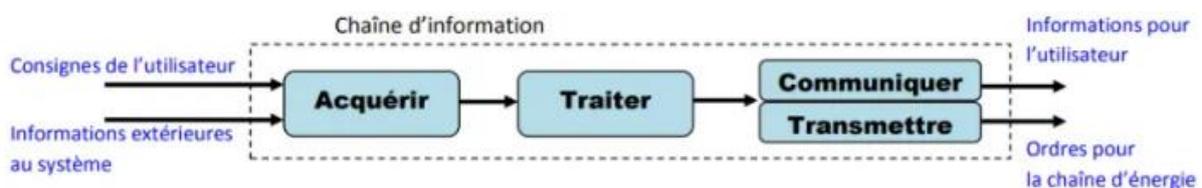


Figure I -19 La chaîne d'information

Acquérir : Fonction qui permet de prélever des informations à l'aide de capteurs.

Traiter : C'est la partie commande composée d'un automate ou d'un microcontrôleur.

Communiquer : Cette fonction assure l'interface l'utilisateur et/ou d'autres systèmes.

Transmettre : Cette fonction assure l'interface avec l'environnement de la partie commande.

Exemple de chaîne d'information : La porte de garage.

L'opérateur appuie sur le bouton de la télécommande pour fermer la porte du garage (consigne de l'utilisateur). La chaîne d'informations, composée d'un boîtier électronique et de

capteurs, détecte le signal et ordonne, lorsqu'elle en reçoit l'ordre, la mise en route du moteur afin d'ouvrir la porte (ordre). [3]

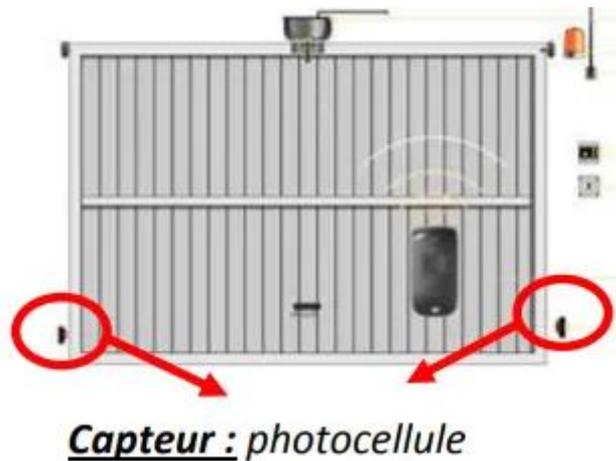


Figure I -20 Exemple d'une porte de garage

La photocellule empêche la fermeture de la porte si elle détecte La présence d'un objet (personne, voiture, animal...) Il y a aussi des capteurs qui permettent de connaître l'état de la porte (ouverte ou fermée). [3]

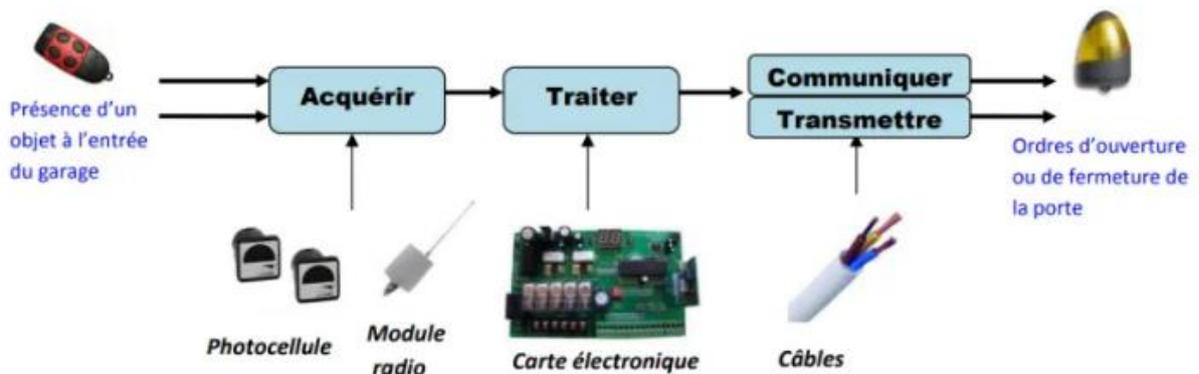


Figure I -21 La chaîne d'information d'une porte de garage

II.5. La chaîne d'énergie :

- **Définition :**

Dans un système automatisé, on appelle une chaîne d'énergie l'ensemble des procédés qui vont réaliser une action. On peut découper cette chaîne en plusieurs blocs fonctionnels. [3]

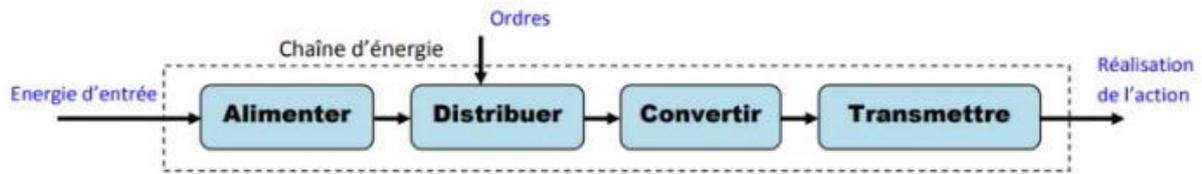


Figure I -22 La chaîne d'énergie

Alimenter : Mise en forme de l'énergie externe en énergie compatible pour créer une action.

Distribuer : Distribution de l'énergie à l'actionneur réalisée par un distributeur ou un contacteur.

Convertir : L'organe de conversion d'énergie appelé actionneur peut être un vérin, un moteur...

Transmettre : Cette fonction est remplie par l'ensemble des organes mécaniques de transmission de mouvement et d'effort : engrenages, courroies, accouplement, embrayage.....

Exemple de chaîne d'énergie : la porte de garage. [3]

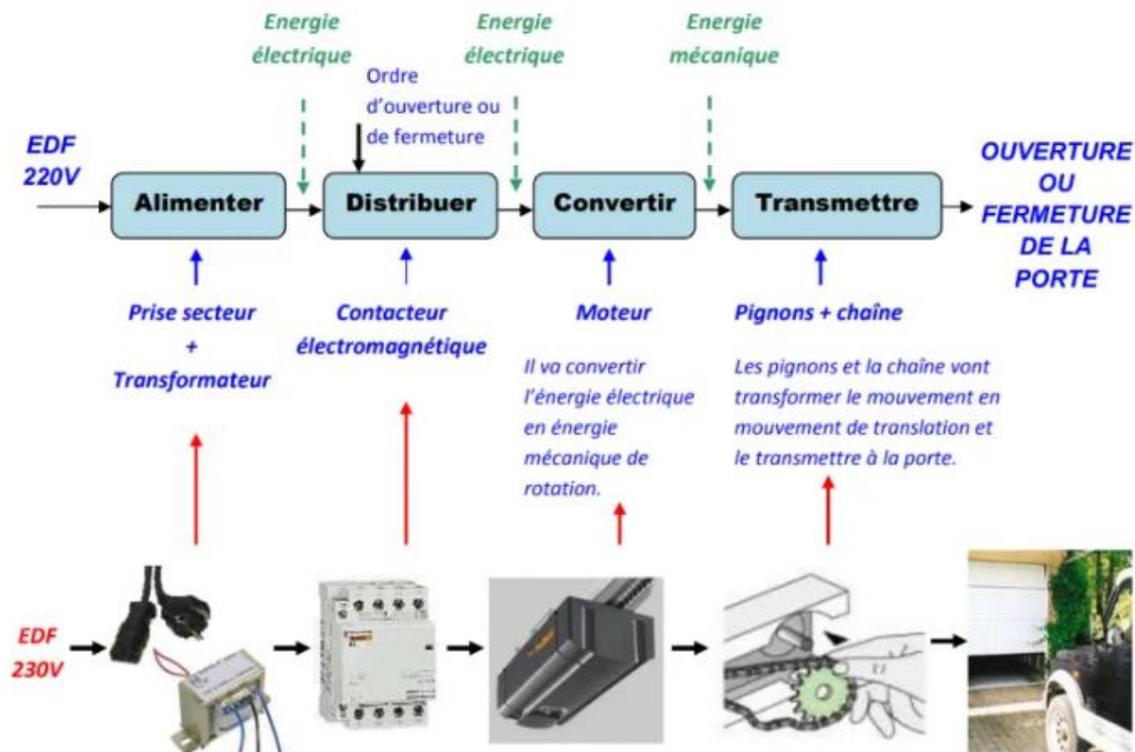


Figure I -23 Fonctionnement de la chaîne d'énergie d'une porte de garage

II.6. Communication dans les systèmes automatisés :

- **Bus ASI :**

Le bus AS-I étant un bus série, il permet de diminuer le nombre de câbles nécessaire à l'utilisation d'un grand nombre de capteurs ou de pré-actionneurs.

L'AS-Interface (AS-i) – ou plus précisément l'Actuator-Sensor Interface – est un système de mise en réseau simple et efficace pour le niveau de terrain. Système de bus ouvert et non propriétaire, il assure la transmission des signaux tout ou rien et analogiques au niveau des machines et joue également le rôle d'interface universelle entre le niveau de commande et les capteurs et actionneurs binaires simples. Sa simplicité et son efficacité sont telles qu'il représente de loin la solution de mise en réseau la plus économique face à tous les autres systèmes de bus de terrain. Rien d'étonnant donc à ce que l'AS-i se soit imposé comme un véritable standard dans le monde de l'automatisation industrielle. Car le système ne se distingue pas seulement par sa simplicité de mise en œuvre et son installation quasi-instantanée. Il offre aussi une souplesse d'extension incomparable et une robustesse extrême, même dans les conditions les plus sévères. [8]

Le réseau AS-i est un bus de terrain niveau 0 permettant de réduire les coûts d'ingénierie et de câblage des équipements simples. [9]



Figure I -24 Bus ASI

- **Bus CAN:**

Le bus CAN (Control Area Network) est un moyen de communication série qui supporte des systèmes embarqués en temps réel avec un haut niveau de fiabilité. Ses domaines d'application s'étendent des réseaux moyens débits aux réseaux de multiplexages faibles coûts. Il est avant tout à classer dans la catégorie des réseaux de terrain utilisés dans l'industrie. [10]

Il a été normalisé avec la norme ISO 11898. Développé, au milieu des années 80 d'après la collaboration entre l'université de Karlsruhe et BOSCH.

Il existe pour le moment 2 norme couvrant les couches 1 et 2 du modèle OSI :

- Le CAN standard ou CAN 2.0 A
- Le CAN étendu ou CAN 2.0 B

Ces 2 norme sont compatible, il peut circuler sur un même réseau des Message suivant la norme 2.0A et des messages suivant la norme 2.0B. [8]



Figure I -25 Bus CAN

- **PROFIBUS :**

PROFIBUS est un réseau de terrain ouvert, non propriétaire, répondant aux besoins d'un large éventail d'applications dans les domaines du manufacturier et du processus. Son universalité (indépendance vis-à-vis du constructeur) et son ouverture sont garanties par les normes européennes EN 50170, EN 50254. et international IEC61158 PROFIBUS autorise le dialogue de matériels multi-constructeurs, sans passer par des interfaces spécialisées. Il se prête aussi bien à la transmission de données exigeant des actions réflexes, en des temps de réaction très courts, qu'aux échanges de grandes quantités de d'informations complexes. En perpétuelle évolution, PROFIBUS reste le réseau de communication industriel du futur. [8]

Il existe trois types de PROFIBUS :

- PROFIBUS DP (périphérie décentralisée).
- PROFIBUS PA (automatisation de process).
- PROFIBUSFMS (Fieldbus message spécification).



Figure I -26 Câble PROFIBUS

- **PROFINET (Ethernet industriel) :**

Ethernet industriel est une technologie de réseau qui se développe très vite. Au future l'Ethernet avec le protocole de réseau TCP/IP (Transmission Control Protocol/Internet Protocol) va être le raccord au niveau de la zone de mémoire (par exemple capteurs, commandes) bien établie. En général les débits de transmission sont divisés comme suit:

- SHARED ETHERNET = 10 Mbit/s
- FAST ETHERNET = 100 Mbit/s
- GIGABIT ETHERNET = 1000 Mbit/s (1 Gbit/s). [8]

L'Ethernet est la technologie filaire habituelle utilisée pour les réseaux LAN (Local Area Network), permettant aux appareils connectés de communiquer entre eux grâce à un protocole – il s'agit d'une norme mise en place pour faciliter les échanges réseau. En d'autres termes c'est la technologie qui permet à deux appareils connectés grâce à un câble RJ45 de communiquer. [11]



Figure I -27 Câble Ethernet

II.7. L'avantage d'un système automatisé:

- Éliminer la manipulation de documents papier : Tout se fait de façon numérique, alors il est possible de facilement stocker l'information sur le système, donc de réduire les coûts additionnels d'entreposage de documents physiques.
- Le temps de recherche de l'information est aussi optimisé.
- Maximiser la productivité des tâches que seulement les employés peuvent faire : L'un des principaux attraits de l'automatisation est qu'elle libère votre talent humain pour faire ce que les humains font de mieux : la création de plans d'affaires et de plans ingénieux pour éliminer des coûts, prendre des décisions importantes, etc. Toutes des méthodes créatives et innovantes qu'un système automatique ne peut faire.
- Se concentrer sur les tâches plus importantes. [12]
- Définir vos processus d'affaires : Pour automatiser, vous devez marquer tout ce qui est impliqué dans le processus de votre entreprise avant de pouvoir configurer l'automatisation. Ce marquage rend tout visible, vous obtenez une image beaucoup plus claire de ce qui est dans votre entreprise! [12]
- Faire de meilleures projections : Avec un BPA (Business Process Automation), vous avez les outils pour surveiller tout, tout le temps. Ce qui, à son tour, aboutit à des ensembles de données beaucoup plus précis et à des projections plus précises des besoins et des revenus futurs des entreprises. [12]
- Identifier les ressources inutilisées : Au-delà de meilleures prévisions sur ce que vous allez vendre et ce dont vous aurez besoin pour fabriquer vos produits, un bon BPA vous permettra de détecter les ressources sous-utilisées ou inutilisées (par exemple, les téléphones cellulaires rendus orphelins par un employé partant). L'automatisation des processus vous permettra d'identifier et de réaffecter vos ressources inutilisées. [12]
- Permettre et faciliter de nouvelles opportunités d'affaires : L'automatisation vous permet de configurer et de gérer des infrastructures de vente et de support dans des domaines que vous n'auriez jamais pu gérer en utilisant uniquement des ressources humaines. [12]
- Économisez de l'argent, même pendant que vous dormez : Vous êtes humain, et vous devez dormir, vous allez littéralement endommager votre santé si vous ne vous reposez pas assez. Tôt ou tard, vous devrez cesser de gérer physiquement vos opérations, ne serait-ce que pour donner à votre pauvre corps épuisé un repos bien mérité. Toutefois, cette limitation ne s'applique pas à vos systèmes automatisés. Les

plateformes automatisées peuvent être configurées pour fonctionner toute la journée et toute la nuit, sans relâche et sans fin.

- Gérez facilement les équipes décentralisées dans différents fuseaux horaires. [12]

II.8. Les inconvénients d'un système automatisé :

- Un test automatisé n'est pas forcément identique à un test manuel. [13]
- Un test automatisé peut-être moins précis qu'un test manuel.
- Les tests automatisés ne testent que les cas connus.
- Un test automatisé est soit trop simple soit trop compliqué. [13]
- Les tests automatisés sont difficiles à maintenir.
- L'environnement d'exécution de tests est coûteux à maintenir.
- Les tests automatisés ne sont pas développés par les experts métier. [13]
- En cas d'échec, l'analyse des résultats peut-être fastidieux.
- Les tests automatisés ne peuvent pas être créés en même temps que le développement. [13]

III. Système MPS 500 : Présentation, description et Récolte des données :

III.1. MPS :

III.1.1. Historique :

Depuis 1991, les stations du système de production modulaire MPS® sont les « équipements sportifs » des championnats du monde des mécatroniciens. Le MPS® a prouvé dans des compétitions nationales et internationales que sa conception, ses stations et commandes ainsi que les fonctions qui y sont réalisées offrent exactement ce qui caractérise la fabrication automatisée dans le monde entier : l'intégration de mécanique, électrotechnique et la technique d'information de la mécatronique.

Cela signifie que le formateur qui enseigne avec le MPS® peut considérer que beaucoup d'entreprises, écoles et universités du monde entier en font de même. Les stations du système de production modulaire sont à l'origine et le modèle de presque tous les systèmes de formation à la mécatronique. [14]

III.1.2. Description du système MPS_FMS 500 :

Le système MPS_FMS 500 (Modular Production System-flexible manufacturing system et le chiffre 500 signifie la disposition actuelle au alentour de Convoyeur) est installé actuellement au sein de laboratoire de recherche MELT (Productique) de l'université ABOU BEKR BELKAID de Tlemcen.

Ce système « Festo Didactic » c'est une chaîne de production modulaire qui répond aux exigences le plus diverses: fonctions différenciés, unités individuelles et combinées, différents techniques d'actionnement, flux matière et flux d'informations et concepts de commande moderne et variable, Il nous permet d'étudier, d'analyser, de comprendre et de maîtriser les interactions entre la mécanique, l'électricité, le pneumatique, contrôle et les interfaces de communications pour mieux gérer les systèmes industries complexes

Il est un système de plusieurs stations reliées par un système de transport qui se charge de transporter les pièces entre les différentes stations. [14]

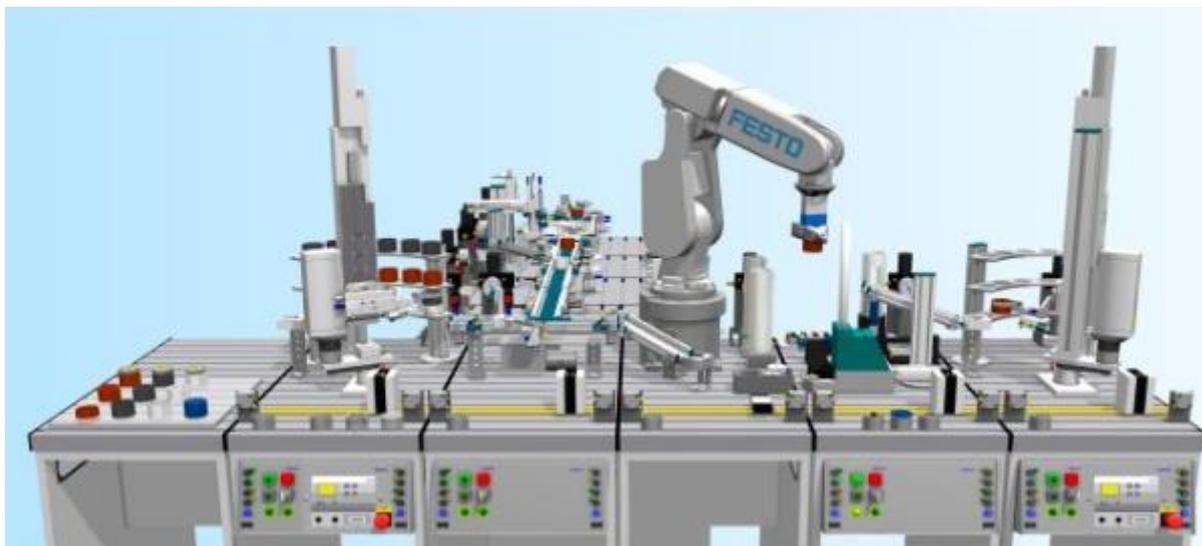


Figure I -28 La station MPS500

III.1.3. Composante du système MPS500:

III.1.3.1. Station distribution et control :

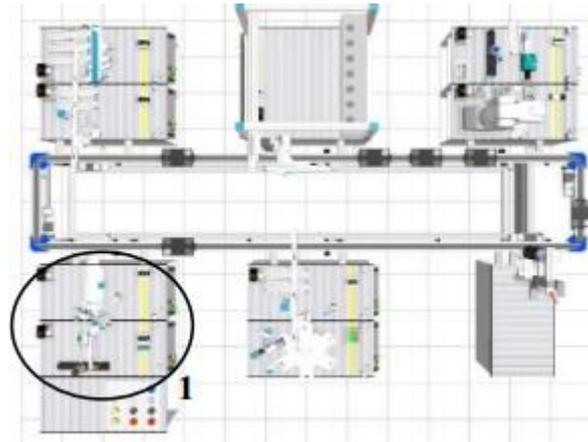


Figure I -29 La station de distribution et contrôle

1-La sous station du Distribution :

Le flux de travail commence à ce stage, Le magasin éjecte une à une des pièces à usiner dans un magasin. Jusqu'à 8 pièces à usiner peuvent être Stocké dans un ordre quelconque dans le tube du magasin. Les pièces à usiner doivent être insérées côté ouvert vers le haut.

Un vérin à double effet éjecte la pièce à usiner du bas du magasin jusqu'en butée mécanique. La présence d'une pièce à usiner dans le tube du magasin est détectée au moyen d'une barrière photoélectrique. La position du vérin d'éjection est détectée par un capteur électromagnétique.

Les pièces à usiner sont saisies par une ventouse. Les pièces à usiner sont déplacées par un vérin oscillant. L'angle de rotation peut être réglé de manière variable entre 0° et 180° à l'aide de butées de fin de course mécaniques. La détection de fin de course s'effectue par des capteurs électriques de fin de course (électromécaniques). Ce module de transfert peut être exploité avec une ventouse qui nous permet d'aspirer et d'éjecter les Pièces à traiter. [14]

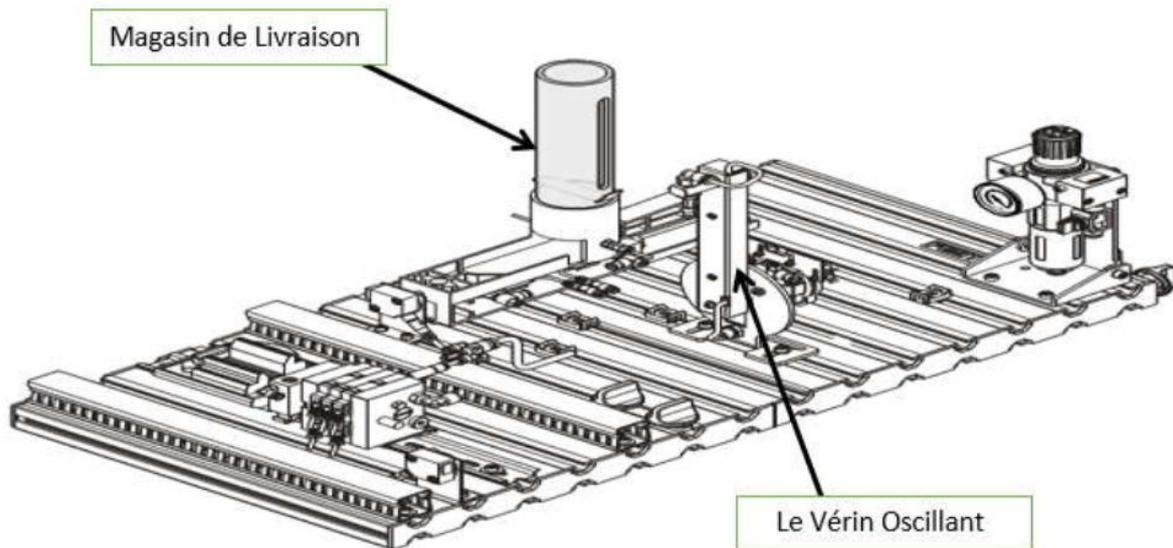


Figure I -30 La sous station de distribution

2-La sous station du Contrôle :

La station de contrôle détermine les caractéristiques des pièces à usiner posées. Le module de détection effectue l'identification de la couleur de la pièce à usiner. Un capteur capacitif identifie chaque pièce à usiner indépendamment de sa couleur. Un détecteur à réflexion détecte les pièces à usiner métalliques et les pièces à usiner rouges. Les pièces à usiner noires ne sont pas reconnues.

Une barrière à réflexion surveille si la zone de travail située au-dessus du réceptacle (Lieu qui reçoit des choses venues de divers endroits) est libre avant que la pièce à usiner ne soit soulevée par le module de levage. Le capteur analogique du module de mesure détermine la hauteur de la pièce à usiner.

Le signal de sortie est soit numérisé par une carte de comparaison avec des valeurs seuils réglables soit transmis par un système de traitement des signaux analogiques B5 au moyen du bloc de connexion d'un API.

Un vérin linéaire achemine les pièces conformes vers la station en aval en empruntant la glissière à coussin d'air supérieure. Les autres pièces à usiner sont rebutées sur la glissière inférieure. [14]

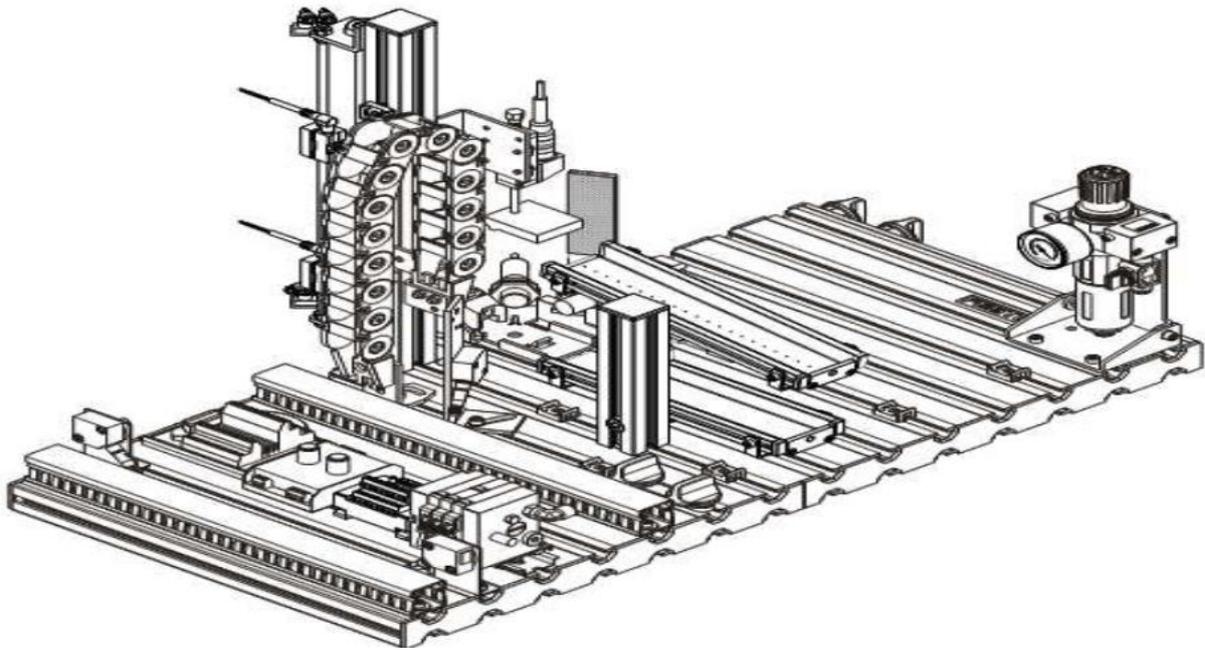


Figure I -31 La station de contrôle

III.1.3.2. Station de production :

C'est la deuxième entité de système, elle se compose aussi de deux sous stations (La station de manipulation et la station d'usinage)

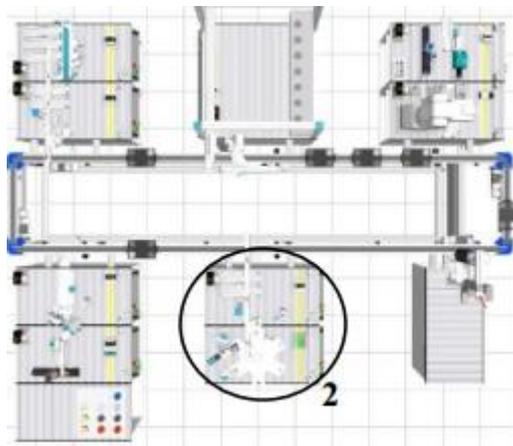


Figure I -32 La station de production

1-La Sous- Station de Manipulation :

La station de manipulation vient juste après le convoyeur, elle est équipée d'un manipulateur flexible à deux axes. Les pièces à usiner insérées sont détectées dans le réceptacle par une barrière photo-électrique. [14]

Le manipulateur y prélève les pièces à usiner à l'aide d'une pince pneumatique. Un capteur optique équipe la pince. Le capteur distingue les pièces « noires » et les pièces « autres que noires ». Les pièces à usiner sont alors acheminées, en fonction de ces critères,

vers des glissières distinctes. D'autres critères de tri peuvent être définis en cas de combinaison de la station avec d'autres stations.[14]

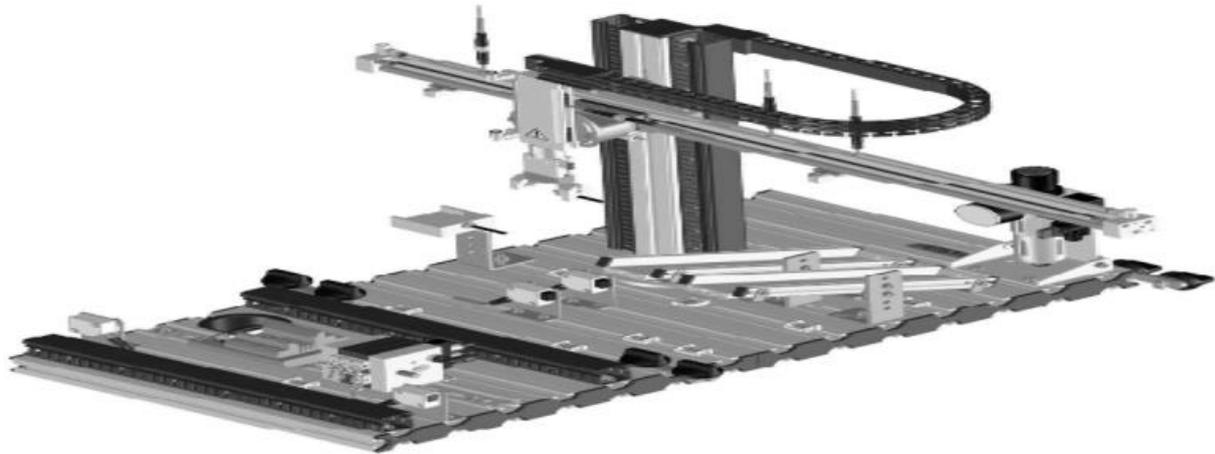


Figure I -33 La sous station de manipulation

2-La Sous-station d'usinage :

Dans la station d'usinage, il est procédé au contrôle des caractéristiques des pièces à usiner (position correcte, perçage) et à l'usinage de pièces sur un plateau à indexation. Le plateau à indexation dispose de 6 réceptacles à semi-ouverts et est entraîné par un moteur à courant continu. Le positionnement du plateau à indexation s'effectue au moyen d'un circuit à relais, la position du plateau à indexation est détectée par un capteur inductif.

Les pièces à usiner sont contrôlées et usinées en 2 procédures parallèles sur le plateau à indexation. Une bobine dotée d'un induit contrôle si les pièces ont été insérées dans la bonne position. La pièce à usiner est fixée par un dispositif de serrage électrique lors de l'usinage par une perceuse électrique. [14]



Figure I -34 La sous station d'usinage

III.1.3.3. La station d'assurance qualité :

La station d'assurance qualité sert à vérifier la qualité de la pièce usinée juste avant qu'elle soit assemblée avec le reste des composants du produit final ; ceci se fait à l'aide d'une caméra équipée d'un programme de traitement d'image qui contient les informations et les données principales de la conformité du produit final.[14]

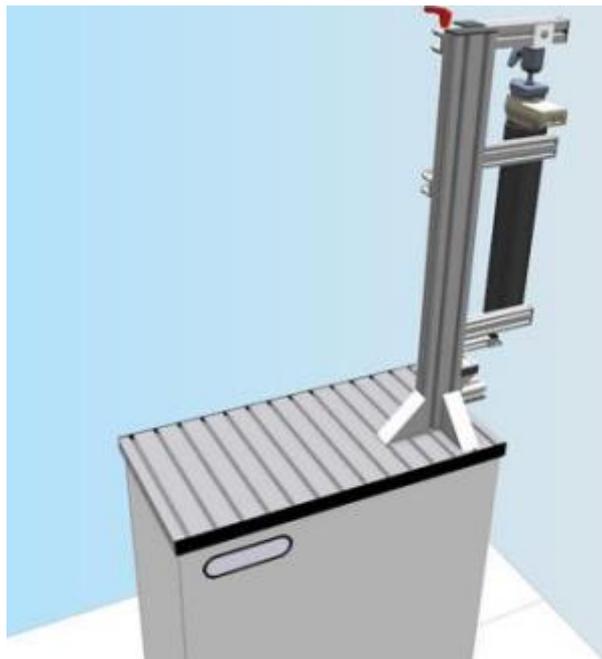


Figure I-35 La station d'assurance qualité

III.1.3.4. La station d'assemblage :



Figure I-36 La station d'assemblage

1- La Sous-station Robot:

La station de robot transporter les pièces qui sont introduites dans la rampe et les placer dans le dispositif de retenue de montage. Le capteur dans la pince permet au robot de distinguer les pièces par leur couleur (noir / non noir). Le capteur dans le dispositif de retenue de montage surveille l'orientation de la pièce. De la retenue d'assemblage, le robot classe les pièces dans divers entrepôts ou les passe à la station suivante, La combinaison avec la station d'assemblage, permet l'assemblage de pièces. [14]



Figure I -37 La sous station robot

2- Station d'assemblage :

La station d'assemblage c'est un complément de la station de robot. Il fournit les composants cylindriques pour le processus d'assemblage: Un vérin double effet extrait le stock du cylindre de l'entrepôt. Les pistons sont sur une palette. Un vérin à double effet pousse le ressort hors de l'entrepôt. [14]



Figure I -38 La station d'assemblage

III.1.3.5. Magasin :

La station de magasin central automatisé est à disposition à la position de travail 5 pour le stockage des matériaux. Les pièces à usiner peuvent y être stockées et déstockées. La station du magasin central automatisé est un magasin de production et peut être utilisé dans un processus complet comme magasin d'entrée, intermédiaire ou de sortie. Le magasin central automatisé sert de plaque tournante logistique pour toutes les pièces finies ou les produits semi-finis. Utilisé comme magasin à rayonnage pour pièces, les points de réception (intégrés pour pièces de 40 mm de diamètre) peuvent stocker et déstocker jusqu'à 35 pièces à usiner dans les 5 allées du rayon. Le stockage et le déstockage a lieu du même côté frontal du magasin. Le magasin central automatisé permet l'application de différents principes pour la gestion des stocks : FIFO (First In First Out) ou LIFO (Last In First Out). [14]

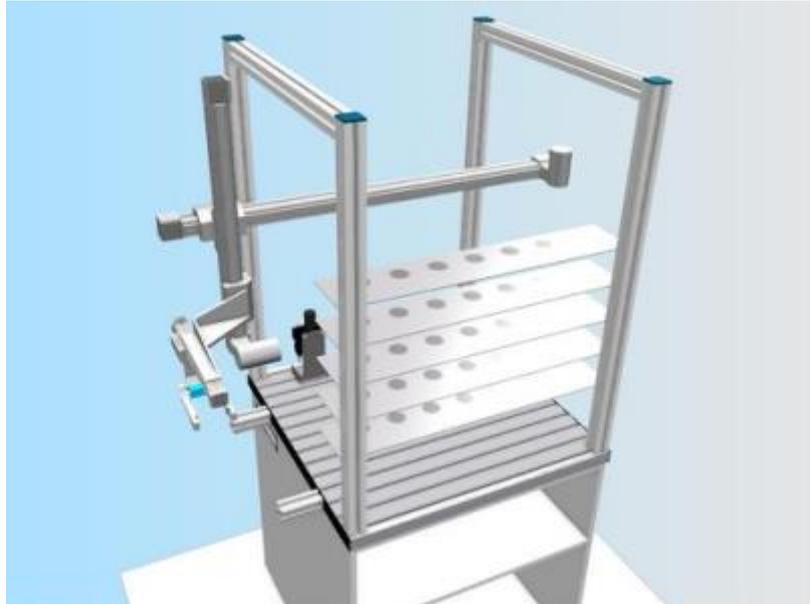


Figure I -39 Le magasin

III.1.3.6. La station de tri:

Cet atelier est le dernier de la ligne, il se trouve à la dernière position de travail et se compose lui aussi de deux stations (Station de manipulation et la station de tri). La station de manipulation, situé directement à côté du système de transport permet grâce à son manipulateur flexible à deux axes d'acheminer la pièce du convoyeur vers la station de tri. Celui-ci s'occupe de trier les pièces selon leurs couleurs, Pour cela, il existe trois emplacements (Buffers) pour chaque type de pièce (Rouge, Noire ou métallique). [14]

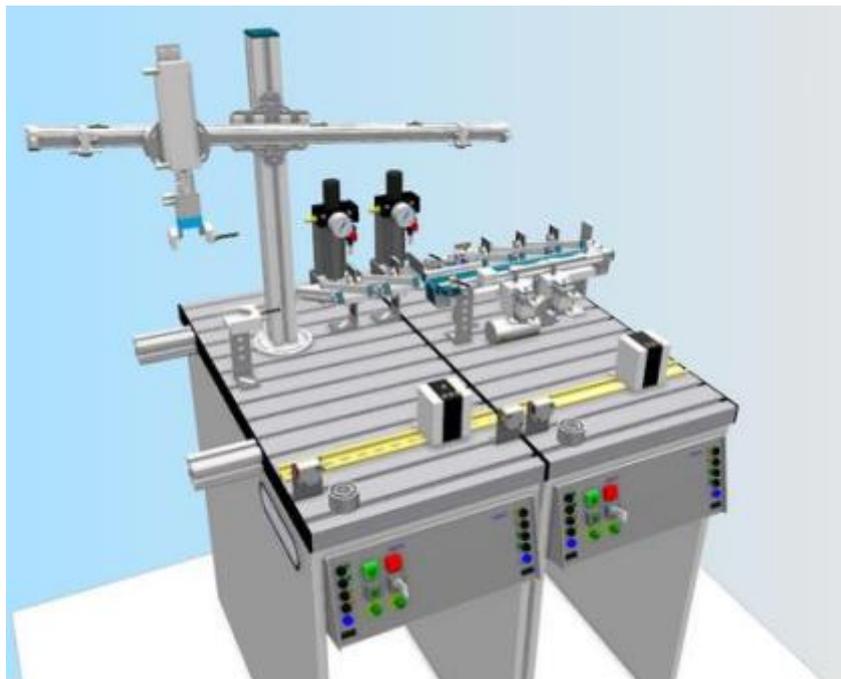


Figure I -40 La station de tri

III.1.3.7. Le convoyeur :

Le convoyeur peut être considéré comme l'élément le plus important dans le système, car il assure la manutention (transport) des pièces sur des palettes à travers les stations. [15]



Figure I -41 Le convoyeur

III.2. Notre Stations à étudier :

III.2.1. Sous Station de contrôle :

III.2.1.a. Description et fonctionnement :

Le système permet le tri des pièces selon la hauteur ou la couleur. Il compose :

- 01 vérin **2M2** double effet permettant la réception et l'évacuation des pièces dans le poste1.
- 01 vérin sans tige **1M2** double effet permettant le transfert des pièces vers le poste de contrôle de la hauteur.
- 01 glissière à coussin d'air **3M1** permettant l'évacuation des pièces de hauteur déterminée vers le poste2.

À l'état initial :

Le chariot du vérin **1M1** doit être en position basse **1B2**.

Le vérin **2M2** doit être en position entrée **2B1**.

1- Tri selon la hauteur :

Le système doit faire le tri entre des pièces de deux hauteurs différentes. Les pièces dont la hauteur est égale à une valeur prédéterminée et des pièces dont la hauteur est inférieure à la valeur prédéterminée

Lorsqu'une pièce est détectée par le capteur (**Part av**) démarrent le cycle dans l'ordre suivant :

Transfert de la pièce vers le poste de contrôle de la hauteur.

Si la hauteur est égale à la valeur prédéterminée, la pièce est évacuée vers le poste2.

Si la hauteur est inférieure à la valeur prédéterminée, la pièce doit être transférée vers le

poste1, puis évacuée vers le poste3.

Le contrôle de la hauteur est effectué par un comparateur qui compare la hauteur de la pièce avec la valeur prédéterminée.

2- Tri selon la couleur :

le système tri entre des pièces de couleur noir et des pièces de couleur rouge ou argentée.

Une pièce de couleur noir est évacuée vers le poste3, alors qu'une pièce de couleur rouge ou argentée est évacuée vers le poste2.

La détection de la couleur est effectuée par deux capteurs au poste1.

Le capteur inductif (**Part av**) détecte les trois couleurs (noir, rouge et argentée), alors que le détecteur photoélectrique (**B2**) détecte les couleurs (rouge et argentée).

Remarque :

Si le capteur de sécurité (**B4**) détecte un objet dans la zone de déplacement du vérin **1M2**, le transfert des pièces du poste 1 vers le poste2 ne peut pas être effectué.

III.2.1.b. Tableau des entrées / sorties / actionneurs :

Dans ce tableau on va présenter les différentes variables E/S entrées/Sorties (Capteurs et Actionneur) qui existent dans cette sous station (contrôle, pupitre) avec leur adresse physique et simulation.

Tableau 1- Les entrées/ sorties de la station de controle

Variable	Désignation	Adressage Simulation	Adressage Physique	Utilité
Part_AV	Capteur Capacitif	E0.0	E124.0	Pièce présente
B2	Capteur Optique de proximité	E0.1	E124.1	Pièce à usiner autre que noire
B4	Capteur optique à réflexe	E0.2	E124.2	Barrière lumineuse de sécurité
B5	Capteur potentiométrique	E0.3	E124.3	Hauteur de pièce à usiner correcte
1B1	Capteur électromagnétique	E0.4	E124.4	Vérin de levage en haut
1B2	Capteur	E0.5	E124.5	Vérin de levage en bas

	électromagnétique			
2B1	Capteur électromagnétique	E0.6	E124.6	Vérin d'éjection rentré
IP_FI	Capteur Optique de proximité	E0.7	E124.7	Station en aval libre
S1	Capteur électromécanique	E1.0	E125.0	Touche START
S2	Capteur électromécanique	E1.1	E125.1	Touche STOP (contact à ouverture)
S3	Capteur électromécanique	E1.3	E125.3	Touche mise en référence/RESET
S_Auto	Capteur électromécanique	E1.2	E125.2	Automatique/manuel
Em_Stop	Capteur électromécanique	E1.5	E125.5	Coupure d'urgence déverrouillée
1M1	Actionneur	A0.0	A124.0	Vérin de levage vers le bas
1M2	Actionneur	A0.1	A124.1	Vérin de levage vers le haut
2M1	Actionneur	A0.2	A124.2	Vérin d'éjection vers l'avant
3M1	Actionneur	A0.3	A124.3	Glissière à coussin d'air activée
IP_N_FO	Actionneur	A0.7		Station occupée
P3	Actionneur	A1.3	A125.3	Voyant matériau allumé
P1	Actionneur	A1.0	A125.0	Voyant START allumé
P2	Actionneur	A1.1	A125.1	Voyant position de repos (Reset)

III.2.1.c. Pupitre de la commande :

Le pupitre de commande MPS permet de commander la station MPS, chaque station dans MPS 500 contient un pupitre de commande qui réalise les différentes opérations:

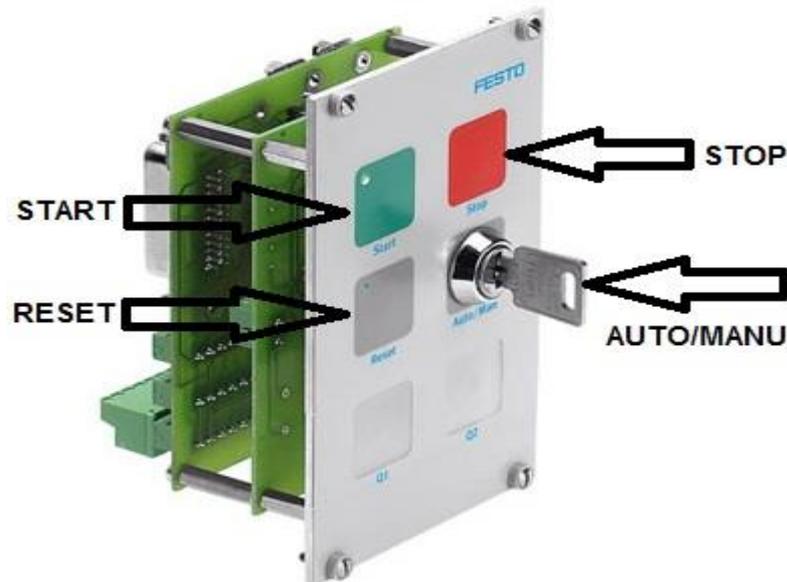


Figure I-42 Le pupitre de commande de la station de contrôle

- Le bouton START pour le lancement du système,
- Le bouton RESET pour le redémarrage du système
- Le bouton STOP pour l'arrêt total du système.
- Il y'a aussi un boutons pour l'arrêt d'urgence.
- Sélecteur à clé Manu/Auto : sert à sélectionner le Mode manuel et Mode automatique.

III.2.1.d. Pré-actionneurs :

C'est un organe capable de réaliser la commande d'un actionneur. Il distribue à l'actionneur un courant fort tout en étant commandé par un courant électrique faible provenant de la partie commande. Il est intégré à la partie opérative ou à l'interface et dimensionné en fonction de l'énergie demandée par l'actionneur.

1-Le Distributeur pneumatique PA1 :

Un pré-actionneur qui fournit de l'énergie pneumatique aux différents actionneurs pneumatiques de la station.

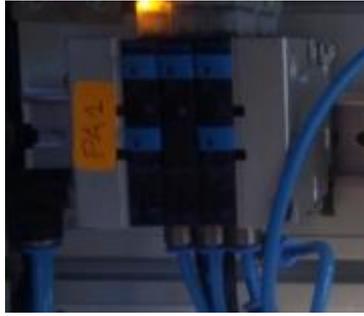


Figure I -43 Pré-actionneur pneumatique

2-Le Distributeur électrique PA2 :

Un Pré-actionneur qui fournit de l'énergie électrique aux différents actionneurs électrique de la station.



Figure I -44 Pré-actionneur électrique

III.2.1.e. Câblage :

Il existe en général deux types de câble de distribution d'énergie : câbles électriques et câbles/tubes pneumatiques, les premiers servent à transmettre l'électricité aux différents composants électriques, et les derniers permettent de transmettre l'air comprimé vers les actionneurs pneumatiques comme les vérins.

IV. Conclusion :

Dans ce chapitre nous avons vue plusieurs notions et généralités concernant les systèmes automatisés en général et la station MPS500 en particulier, Nous avons bien compris le fonctionnement de la station de travail au niveau du système MPS500.

Le chapitre contient les outils matériels que nous avons bien maitrisés pour continuer notre travail.

Chapitre II : Les automates programmables

I. Introduction :

Dans ce chapitre nous allons parler des automates programmables industriels, surtout l'automate que nous avons utilisé et qui est l'API S7-300. Sa constitution et son critère de choix. Nous allons aussi parler des outils logiciels que nous avons utilisés pour la programmation de la station de testing.

Les logiciels utilisés sont STEP7 et PLCSIM. Nous allons les présenter ainsi que leurs étapes d'installation sur un PC.

II. Les automates programmables :

II.1. Historique :

Les Automates Programmables Industriels (API) sont apparus aux Etats-Unis vers la fin des années soixante, à la demande de l'industrie automobile américaine (General Motors) qui réclamait plus d'adaptabilité de leurs systèmes de commande.

Les premières applications des automates programmables furent d'abord le remplacement des horloges de contrôle du temps des employés ; par la suite, leurs multiples utilisations industrielles, en particulier sur les lignes de production des usines, deviennent indispensables non seulement au point de vue contrôle, mais aussi du côté économique pour l'espace et l'entretien. C'est alors que de nombreux systèmes à relais durent céder leur place. Les premiers automates programmables n'effectuaient que la commutation ON/OFF avec la possibilité de temporisation, comme les relais. Leurs applications étaient limitées seulement aux procédés répétitifs ainsi qu'à certaines machines. Par contre, leurs avantages consistaient dans une installation plus facile, la visualisation des étapes; ils possédaient des indicateurs diagnostiques permettant la localisation des pannes. C'était déjà mieux que les relais, en plus de pouvoir être reprogrammé advenant un changement de fonction ou de procédé.

De 1970 à 1974, la technologie des microprocesseurs (du moins les premiers) ajoutèrent une plus grande flexibilité et une « intelligence » à l'automate programmable. Les capacités d'interface avec l'utilisateur s'améliorent. L'automate programmable peut maintenant exécuter les opérations arithmétiques en plus des opérations logiques; il manipule les données et les adresses ; il effectue la communication avec d'autres automates ou ordinateurs, donnant ainsi une nouvelle dimension aux applications de l'automate programmable.

Les automates programmables utilisent une mémoire non-volatile (RAM+Pile, EEPROM ou EAPROM par exemple) pour emmagasiner les instructions. Ces derniers accompliront des fonctions logiques, arithmétiques, de temporisation, de comptage et de manipulation des

données. En plus, les fonctions de contrôle PID et d'autres fonctions complexes comme le contrôle numérique de processus sont présentes. Puisque les automates programmables ont été conçus pour accomplir des opérations semblables à celles des relais, la programmation est basée généralement sur la nomenclature des diagrammes en échelle. Des langages de haut niveau ont été aussi implantés sur certains automates afin de produire une plus grande flexibilité de programmation. [16]

II.2. Définition :

L'automate programmable est un appareil électronique de traitement des informations, adapté à l'environnement industriel, qui réalise des fonctions d'automatisme pour assurer la commande de pré-actionneurs et d'actionneurs à partir d'information logique, analogique ou numériques, il effectue des fonctions d'automatisme programmées comme : La logique combinatoire, le séquencement, la temporisation, le comptage, les calculs numériques, asservissements, régulation. va nous permettre de réaliser les fonctions d'un système de production grâce à un programme qui va gérer les informations (consigne opérateur) et donner les ordres (commande des pré actionneurs), c'est de la logique programmée. Les domaines d'utilisation sont très divers : métallurgie et sidérurgie (sécurité), mécanique et automobile (montage, banc d'essais, ...), chimique, pétrolière, alimentaires, ... [17]

II.3. Structure d'un API :

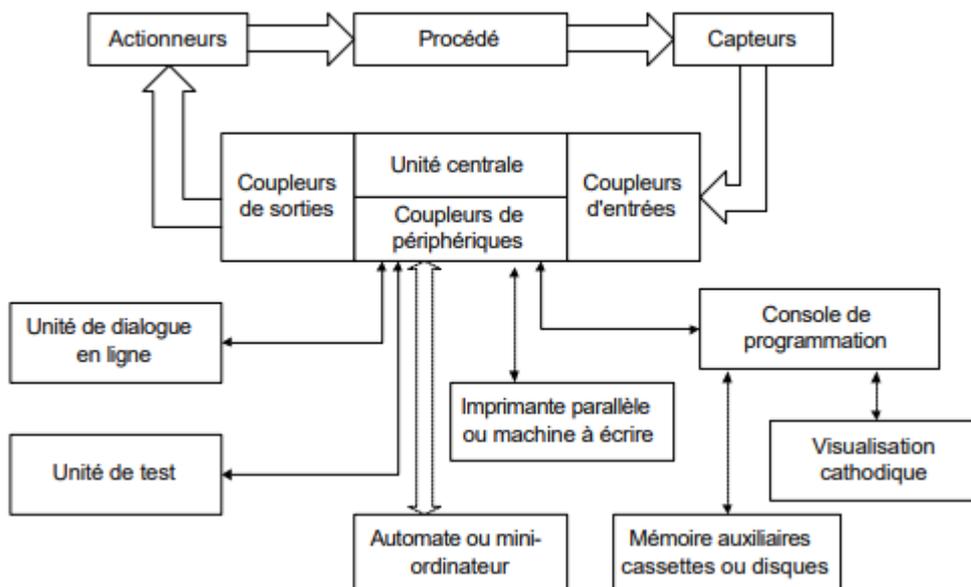


Figure II-1 La structure d'un automate programmable

II.3.1.L'unité centrale (UC) :

C'est le cœur de la machine, comporte le(s) processeur(s) et la mémoire(s) :

1-Processeur :

Appelé unité de traitement, il assure le contrôle de l'ensemble de la machine et effectue les traitements demandés par les instructions du programme. Il réalise les fonctions logiques, temporisation, comptage, calcul. Il comporte un certain nombre de registres (compteur ordinal, registre d'instructions, registre d'adresse, registres de données, accumulateurs, ... Il est connecté aux autres éléments (mémoires, interfaces d'E/S, ...) par l'intermédiaire des bus. Le processeur est organisé autour d'un certain nombre de registres, ce sont des mémoires rapides permettant la manipulation des informations qu'elles retiennent, ou leur combinaison avec des informations extérieures. [17]

Les principaux registres existants dans un processeur sont :

- **L'accumulateur :**

Un accumulateur électrique est un système destiné au stockage de l'énergie électrique sous une forme différente.

C'est le registre où s'effectuent les opérations du jeu d'instruction, les résultats sont contenus dans ce registre spécial. [17]

- **Le registre d'instruction :**

Il reçoit l'instruction à exécuter et décode le code opération. Cette instruction est désignée par le pointeur. [17]

- **Le registre d'adresse :**

Ce registre reçoit, parallèlement au registre d'instruction, la partie opérande de l'instruction. Il désigne le chemin par lequel circulera l'information lorsque le registre d'instruction validera le sens et ordonnera le transfert. [17]

- **Le registre d'état :**

C'est un ensemble de positions binaires décrivant, à chaque instant, la situation dans laquelle se trouve précisément la machine. [17]

- **Les piles :**

Une organisation spéciale de registres constitue une pile, ces mémoires sont utilisées pour contenir le résultat de chaque instruction après son exécution. Ce résultat sera utilisé ensuite

par d'autres instructions, et cela pour faire place à la nouvelle information dans l'accumulateur. [17]

2- Mémoire :

La mémoire centrale est découpée en plusieurs zones :

- Zone mémoire programme.
- Zone mémoire des données :(états des E/S, valeurs des compteurs, temporisations, ...)
- Zone où sont stockés des résultats : Ce sont les résultats de calcul utilisé ultérieurement dans le programme.
- Zone pour les variables internes. [18]

-Il existe différents types de mémoires :

○ Mémoires vives :

RAM ce sont des mémoires volatiles : elles perdent l'information en cas de coupure de l'alimentation. Certaines d'elles sont équipées de batteries de sauvegarde (autonomie réduite). Elles sont accessibles en lecture et en écriture. [18]

○ Mémoires mortes :

Les contenus sont figés. Ce sont des mémoires à lecture seule. Les informations sont conservées en permanence sans source externe. [18]

○ ROM :

Mémoire programmé par le fabricant et ineffaçables.

○ PROM :

Vendues vierges et programmables une seule fois par l'utilisateur.

○ REEPROM ou EPROM :

Utilisables plusieurs fois (écriture / effacement). Effacement à l'UV (Ultra-violet) pendant 10 à 30 minutes. Elles ne peuvent être reprogrammées qu'après un effacement total. [18]

○ EEPROM :

Effacement électrique et reprogrammation rapide sur place. [18]

II.3.2. Bloc d'alimentation :

Permet de fournir à l'automate l'énergie nécessaire à son fonctionnement. Ils délivrent, à partir du 220 V alternatif, des sources de tension nécessaires à l'automate tels que : +5V, 12V et 24V en continu. [18]

II.3.3. Coupleurs :

Ce sont des cartes électroniques qui assurent la communication entre les périphériques (modules d'E/S ou autres) et l'unité centrale. En général, les échanges entre l'UC et les modules d'E/S s'effectuent par l'intermédiaire d'un bus interne (liaison parallèle codée) alors que ceux avec les périphériques de l'automate (console, lecteur de cassette, ...) s'effectuent par un bus externe (liaison parallèle ou série). [18]

II.3.4. Les carte d'E/S :

Les E/S des automates programmables revêtent une importance évidente au plan technique. Leur coût dépasse fréquemment la moitié de l'investissement total d'une configuration. Ces facteurs justifient une étude détaillée de leur architecture générale, suivi de celle des E/S industrielles typiques. [18]

II.3.5. Les console :

- Console d'exploitation :

Permet le paramétrage et les relevés d'informations (Modification des valeurs et visualisation) sur site. [18]

- Console de programmation :

Réglage et exploitation. Cette dernière effectue dans la phase de programmation, l'écriture, la modification, l'effacement et le transfert d'un programme dans la mémoire de l'automate ou dans une mémoire REEPROM. Dans la phase de réglage et d'exploitation, elle permet d'exécuter le programme pas à pas, de le visualiser, de forcer ou de modifier des données telles que les entrées, les sorties, les bits internes, les registres de temporisation, les compteurs, ..., la sortie sur imprimante du programme en cas de présence d'un port de sortie. La console, généralement équipée d'un écran à cristaux liquides, peut afficher le résultat de l'auto-test comprenant l'état des modules d'E/S, l'état de la mémoire, de la batterie. [18]

II.4. Structure interne d'un API :

Voici la structure interne d'un automate programmable :

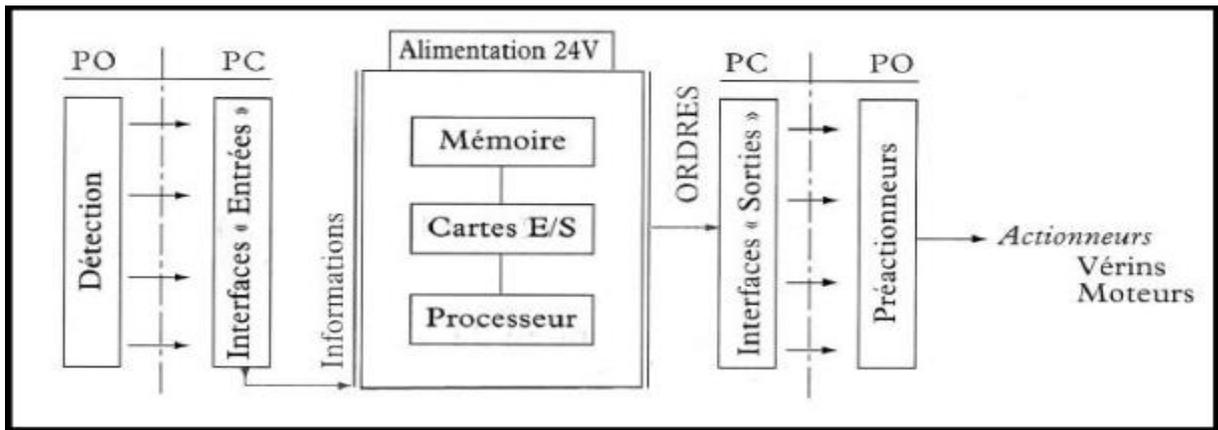


Figure II-2 La structure interne d'un automate programmable

II.5. Traitement du programme automate :

Toutes les automates fonctionnent selon cette structure :

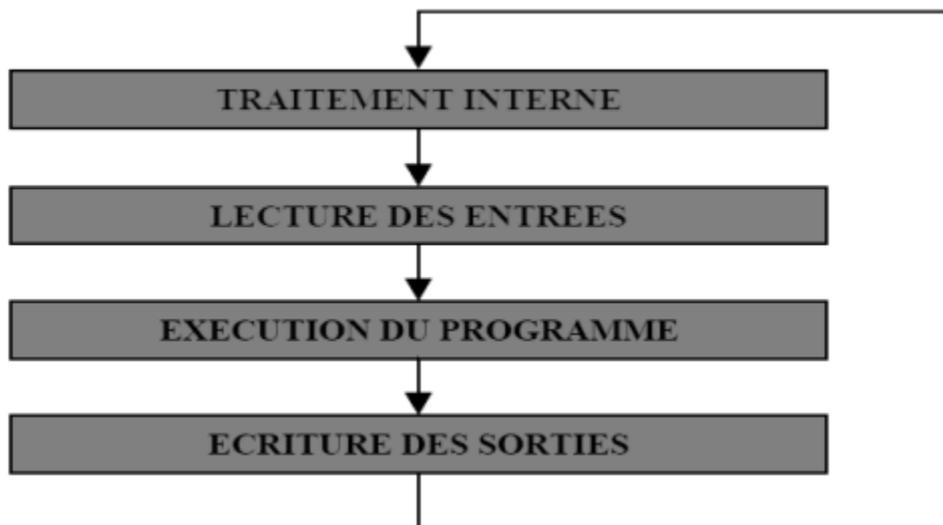


Figure II-3 Les étapes de traitement d'un automate

- **Traitement interne :**

C'est-à-dire que l'automate fait des opérations de contrôle et met à jour certains paramètres de système.

- **Lecture des entrées :**

C'est-à-dire que l'automate lit les entrées de façon synchrone et les recopies dans la mémoire image des entrées.

- **Exécution du programme :**

C'est-à-dire que l'automate exécute le programme de façon synchrone (c'est-à-dire instruction par instruction) et écrit les sorties dans la mémoire image des sorties.

- Ecriture des sorties :

L'automate ordonne les différentes sorties aux positions définies dans la mémoire image des sorties. [19]

II.6. Principe de lecture et commande de l'API :

Les entrées destinées à fournir des informations sur l'état du processus, et après l'automate qui traite les différentes informations d'entrée afin d'élaborer les ordres.

Les sorties transmettent les sorties élaborés par l'automate aux différents actionneurs ou pré-actionneurs. [20]

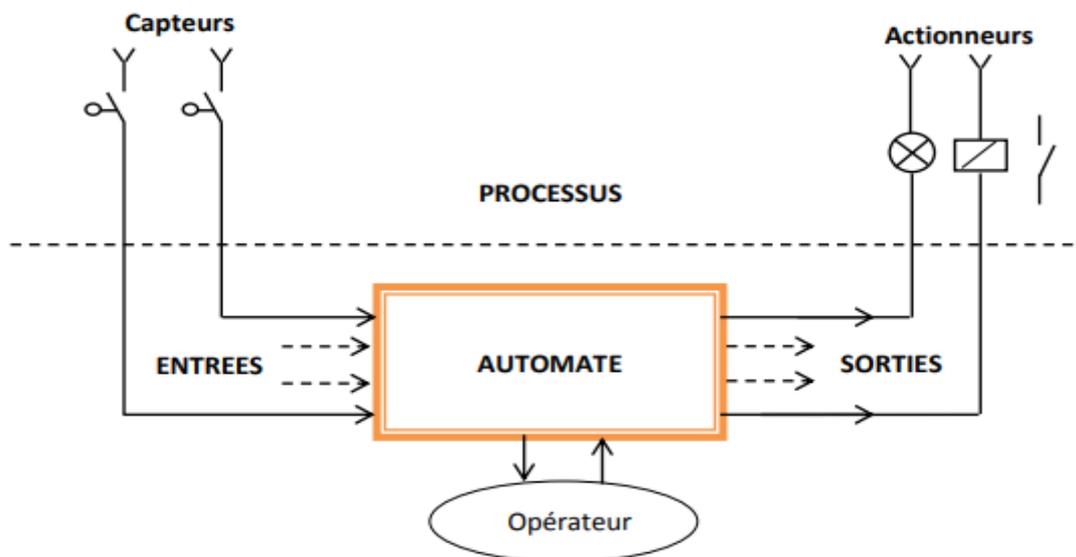


Figure II-4 Le principe de lecture et commande dans un automate

Pour créer un projet à partir d'un API, les outils nécessaires comme :

- Un API avec des caractéristiques compatibles au système à commander : nombre d'entrées/sorties, type d'entrées.
- Un logiciel de programmation adapté à l'API.
- Un câble liaison entre le PC et l'API. [20]

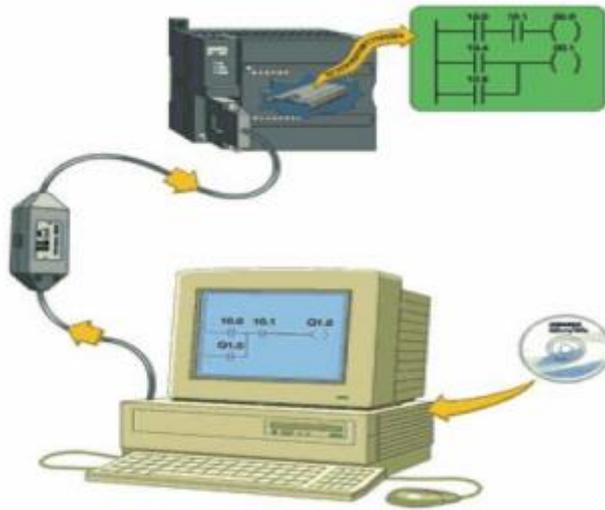


Figure II-5 Les outils d'un API

II.7. Critère de choix d'un automate :

- Le critère de familiarité :

On a une tendance à choisir un automate car on le maîtrise déjà.

- Le temps de cycle :

Certains automates ont un temps de cycle plus rapide face aux d'autres, ce qui peut être important pour des systèmes nécessitant une certaine réactivité.

- Le critère de standardisation :

Si on développe des standards de programme sur un type d'automate particulier, on a intérêt à se focaliser sur un type d'automate spécifique.

- La disponibilité en termes de SAV et de composants rechange :

Certaines marques d'automate ont plus d'autorité ce qui leur permet d'offrir des services après ventes et d'accompagnement client meilleurs.

- Le critère de renommé :

Certaines marques d'automates sont plus connues que d'autres ce qui témoigne de leur qualité de service.

- Le critère de cout :

Pour de petit projet d'automatisation, il est préférable de choisir des micros ou mini automates ce qui va nous empêcher « d'acheter un char pour tuer un moustique ». [20]

III. L'automate programmable S7-300 :

III.1. Définition :

Le S7-300 est un mini-automate modulaire pour les applications d'entrée et de milieu de gamme, avec possibilité d'extensions jusqu'à 32 modules, et une mise en réseau par l'interface

Multipoint (MPI), PROFIBUS et Industriel Ethernet.

Ils permettent de réaliser de nombreuses autres fonctions grâce à des modules intelligents que l'on dispose sur un ou plusieurs racks. Ces modules ont l'avantage de ne pas surcharger le travail de la CPU car ils disposent bien souvent de leur propre processeur.

Il se compose d' :

- Une alimentation
- Une unité centrale(CPU). [21]



Figure II-6 L'automate programmable s7-300

III.2. Structure interne de s7-300 :

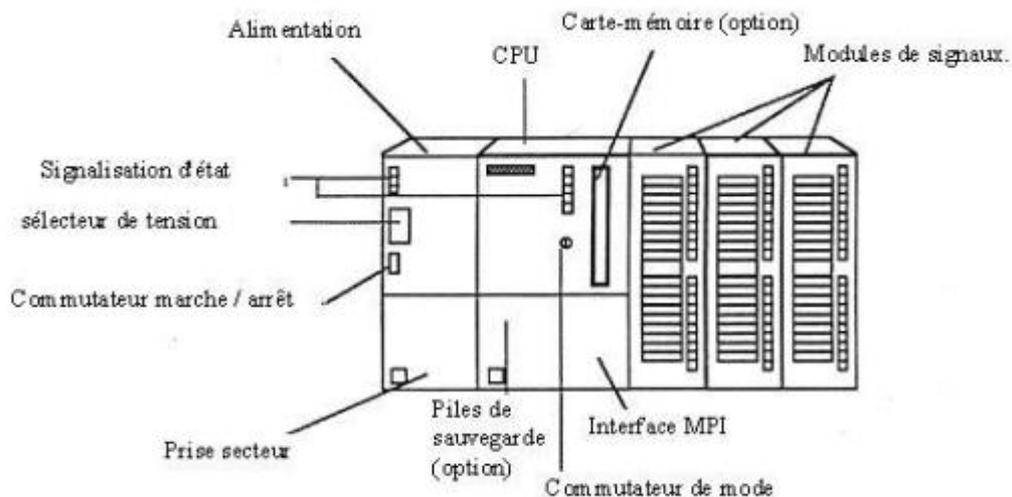


Figure II-7 La structure interne de s7-300

III.3. Fonctionnalités :

Le S7-300 est l'automate conçu pour des solutions dédiées au système manufacturier et constitue à ce titre une plate-forme d'automatisation universelle pour les applications avec des architectures centralisées et décentralisées.[22]

III.4. Caractéristiques techniques:

Le S7-300 offre une gamme échelonné de 24 CPU, des CPUs standard parmi lesquelles la première CPU avec interface Ethernet/PROFINET intégrée, des CPUs de sécurité, des CPUs compactes avec fonctions technologiques et périphériques intégrées et CPU technologiques pour la gestion des fonctions motion control.

Le S7-300 offre également une très large palette de modules d'E/S TOR et analogiques pour la quasi-totalité des signaux avec possibilité de traitement des interruptions et du diagnostic. A disposition également des modules pour emploi dans des zones à atmosphère explosive, des modules de fonction technologiques comme par exemple : régulation et came électronique et des modules de communication point a point ou par bus ASI, PROFIBUS ou industriel Ethernet. Sa simplicité de montage et sa grande densité d'implantation avec des modules au modulo 32 permettent un gain de place appréciable dans les armoires électriques. [22]

III.5. Avantage :

Une construction compacte et modulaire, libre de contraintes de configuration. Une riche gamme de modules adaptés à tous les besoins u marché est utilisable en architecture centralisée ou décentralisée, qui réduit grandement le stock de pièces de rechange. Une large gamme de CPU adaptée à toutes les demandes certaines étant dotée de fonctions technologiques intégrées comme par exemple : le comptage, la régulation ou le positionnement.

Une énorme d'ingénierie en utilisant les outils orientés application et normalisés CEI 1131-3 tels que les langages évolués SCL ou des logiciels exécutifs orientés technologie pour le contrôle des mouvements. [22]

IV. Langage de programmation :

IV.1. Graph7 :

Cette abréviation signifie graphe fonctionnelle commande étape et transition. Il fait la description graphique d'un système automatisé il est sous forme étape représenté par un seul carré et transition par un trait horizontal qui représente la condition, il commence toujours par une étape initiale qui est la seule l'étape représentée par double carré elle est toujours active au début.[23]

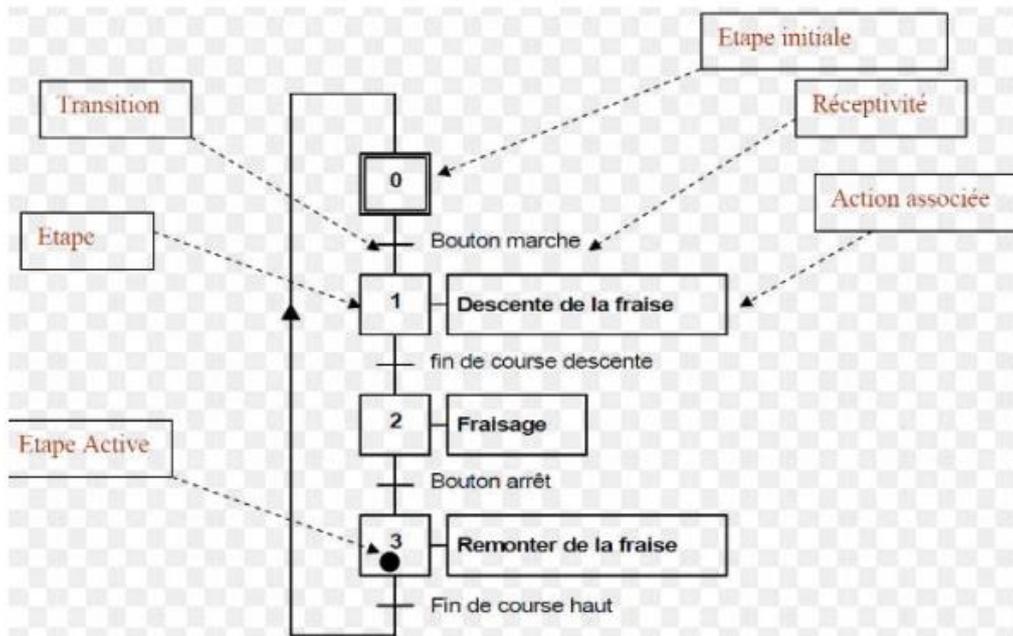


Figure II-8 Représentation de langage Graphe 7.

IV.2. Langage CONT :

Le schéma à contacts (CONT) est un langage de programmation graphique. La syntaxe des instructions fait penser aux schémas de circuits électriques. Le langage CONT permet de suivre facilement le trajet du courant entre les barres d'alimentation en passant par les contacts, les éléments complexes et les bobines.[23]

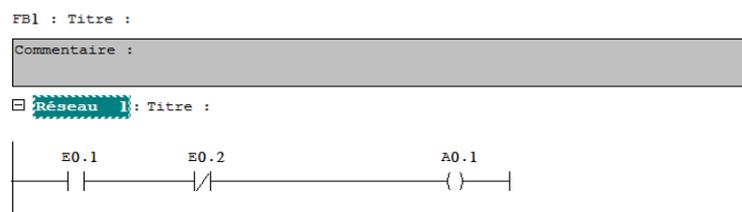


Figure II-9 Représentation de langage CONT.

IV.3. Langage LOG :

Le logigramme (LOG) est un langage de programmation graphique qui utilise les boites de l'algèbre de Boole pour représenter les opérations logiques. Les fonctions complexes comme par exemple les fonctions mathématiques, peuvent être représentées directement combinées avec les boites logiques. [23]

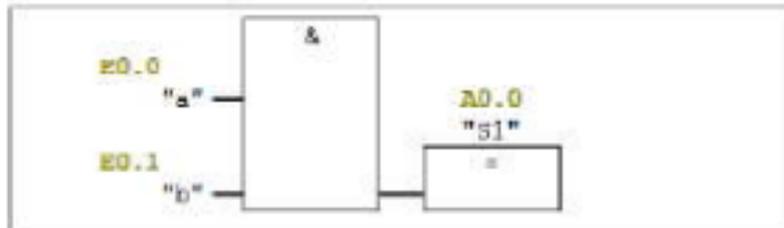


Figure II-10 Représentation de langage LOG.

IV.4. Langage LIST :

La liste d'instructions (LIST) est un langage de programmation textuel proche de la machine. Dans un programme LIST, les différentes instructions correspondent, dans une large mesure, aux étapes par lesquelles la CPU traite le programme. [23]



Figure II-11 Représentation de langage LIST.

V. Logiciels utilisé pour la programmation et la simulation :

V.1. Logiciel Step7: (SIMATIC MANAGER - STEP7):

STEP 7 est le progiciel de base pour la configuration et la programmation de systèmes d'automatisation SIMATIC. Il fait partie de l'industrie logicielle SIMATIC.



Figure II-12 SIMATIC MANAGER.

- Fonctions du logiciel de base :

Le logiciel de base vous assiste dans toutes les phases du processus de création de vos solutions d'automatisation, comme par exemple :

- La création et la gestion de projets
- La configuration et le paramétrage du matériel et de la communication.
- La gestion des mnémoniques.
- La création de programmes, par exemple pour les systèmes cible S7.
- Le chargement de programmes dans des systèmes cible.
- Le test de l'installation d'automatisation.
- Le diagnostic lors de perturbations de l'installation.

Avec SIMATIC Manager, vous pouvez travailler :

- Hors ligne, c'est-à-dire sans qu'un automate soit raccordé
- En ligne, c'est-à-dire avec un automate raccordé. [23]

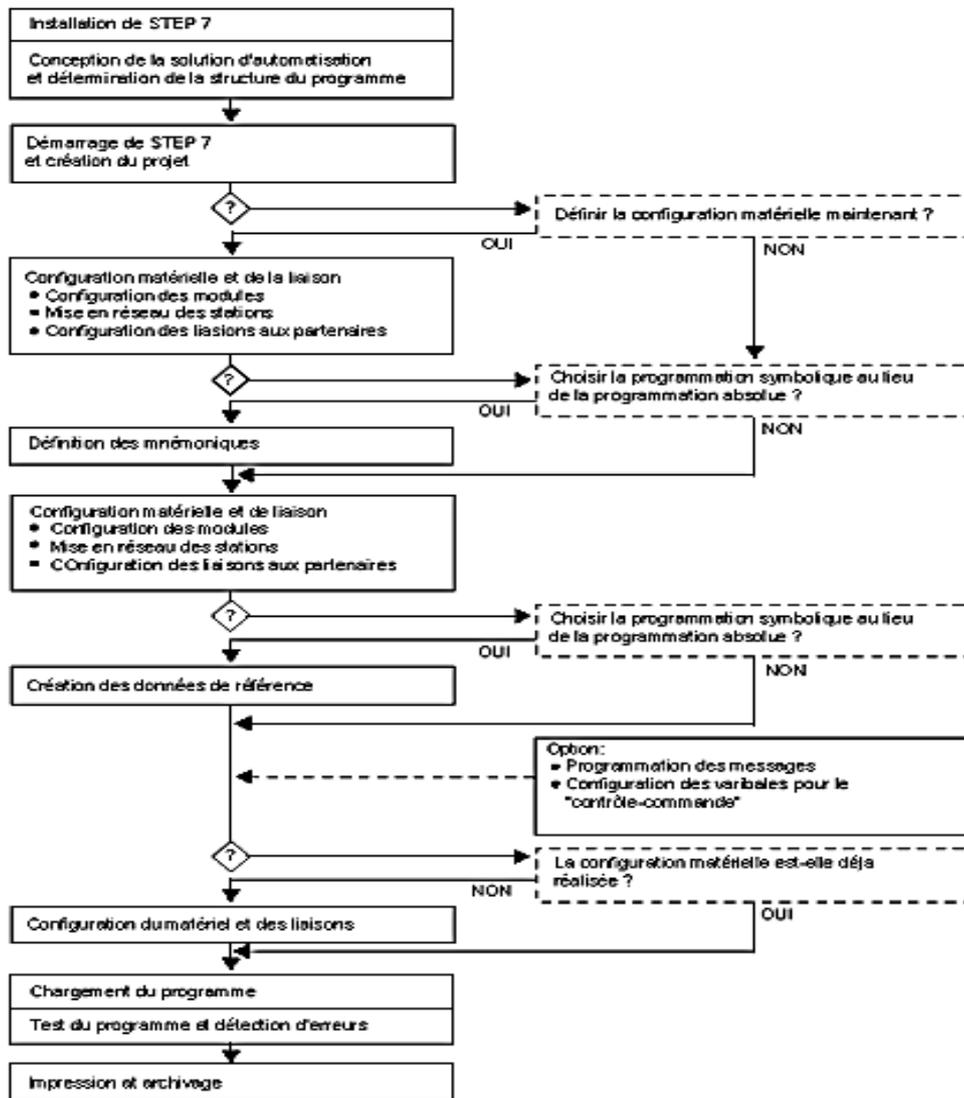


Figure II-13 Les diverses étapes pour réaliser un projet dans Step7.

La mise en place d'une solution d'automatisation avec STEP 7 nécessite la réalisation de tâches fondamentales. La figure au-dessus indique les tâches à exécuter dans la plupart des projets et les classe selon la marche à suivre.

V.2. L'installation du logiciel SIEMENS SIMATIC STEP7 et PLCSIM :

STEP 7 contient un programme SETUP qui exécute l'installation automatiquement. Des messages s'affichant à l'écran vous guident étape par étape tout au long de la procédure

d'installation. Vous l'appellez via la procédure d'installation de logiciel standard sous Windows. Les phases principales de l'installation sont :

- la copie des données dans votre outil de développement,
- l'installation des pilotes pour EPROM et communication,
- l'installation des "License Keys" (clés de licence).

1. Conditions d'installation :

- Système d'exploitation: Windows XP, Windows7, Windows10.
- Matériel de base Ordinateur personnel (PC) ou console de programmation (PG) avec :
 - processeur pentium (600 MHz),
 - mémoire vive : 256 Mo au minimum.
 - moniteur couleur, clavier et souris pris en charge par Microsoft Windows.
 - Mémoire requise Pour l'espace mémoire nécessaire sur le disque dur.

Une console de programmation (PG) est un ordinateur personnel compact tout spécialement conçu pour être utilisé dans un environnement industriel. Elle est équipée en série de tous les programmes nécessaires à la programmation de systèmes d'automatisation SIMATIC.

2. Les étapes d'installations :

- Etape 1 :

Insérez le CD-ROM et lancez le Setup en cliquant sur "setup.exe".

- Etape 2 :

Suivez étape par étape les instructions affichées par le programme d'installation. Ce programme vous guide pas à pas tout au long de la procédure d'installation. Vous avez toujours la possibilité de revenir à l'étape précédente ou d'aller à l'étape suivante.

Pendant l'installation, des questions vous sont posées ou des options proposées dans des boîtes de dialogue. Tenez compte des indications ci-après qui vous permettront de répondre plus rapidement et aisément aux dialogues.

- Etape 3 :

Choisisse la langue et taper suivant

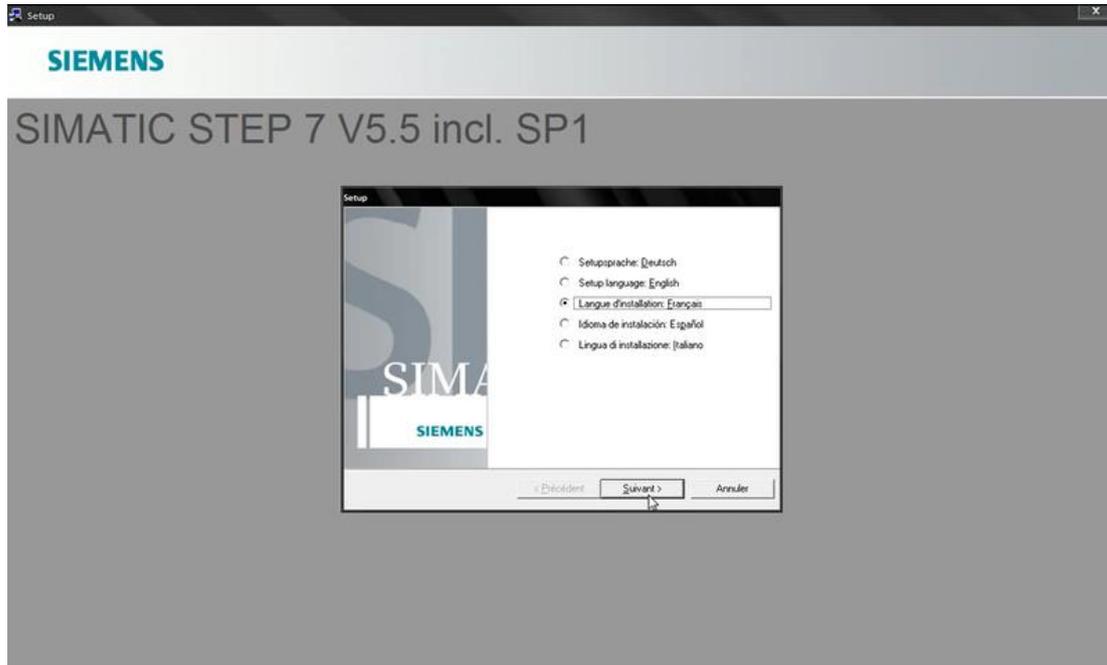


Figure II-14 Le choix de la langue.

- Etape 4 :

L'acceptation de conditions de licence et taper suivant.

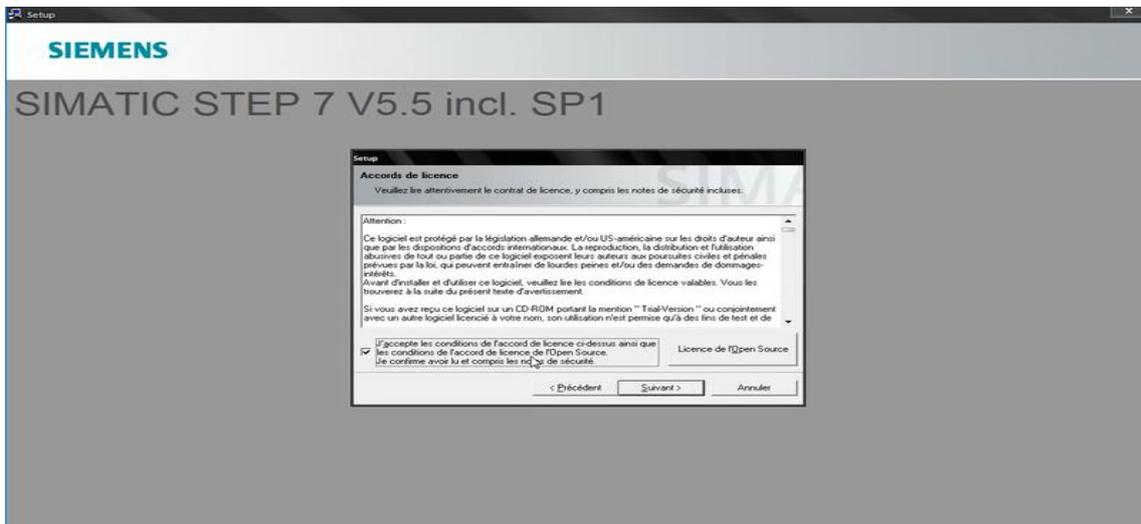


Figure II-15 L'acceptation des conditions de licence.

- Etape 5 :

Cocher toutes les barres qu'on a besoin et taper suivant.

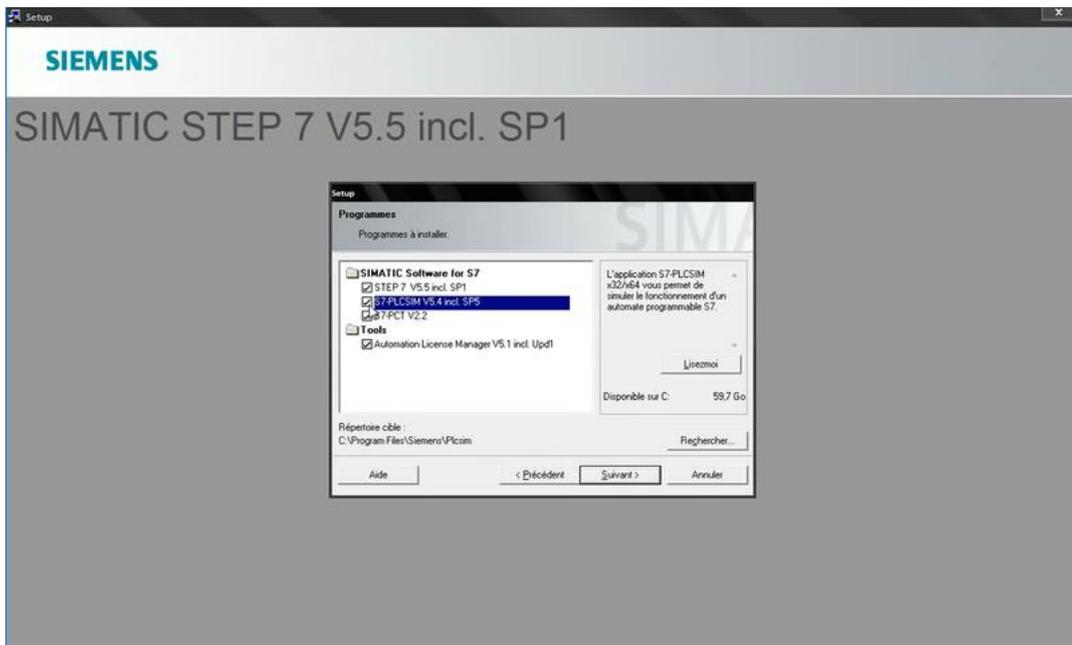


Figure II-16 Le coche des barres qu'on a besoin.

- Etape 6 :

Accepter les modification de système et taper suivant.

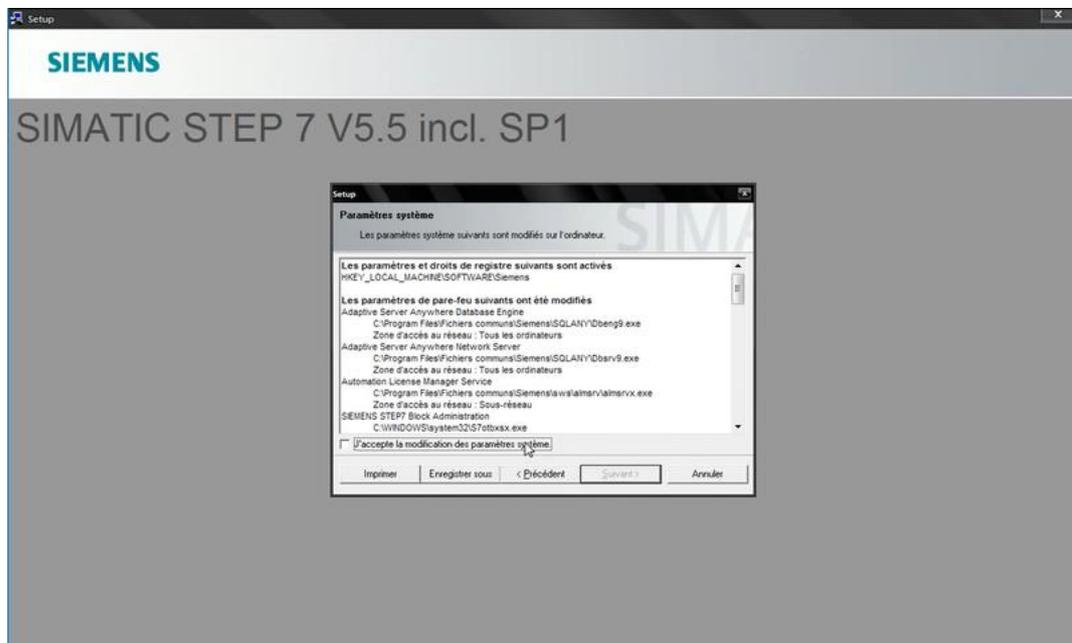


Figure II-17 L'acceptation des modifications des systèmes.

- Etape 7 :

Choisir votre installation, trois variantes d'installation au choix sont possibles :

-Standard : Installation de STEP 7 sur votre ordinateur avec tous ses composants. La boîte de dialogue suivante vous permettra de modifier le choix de la langue.

-Compacte : Installation de STEP 7 sur votre ordinateur avec le minimum de composants nécessaires. La boîte de dialogue suivante vous permettra de modifier le choix de la langue.

- Personnalisée : La boîte de dialogue suivante affiche tous les composants pouvant être installés. Vous pouvez choisir parmi ceux-ci les composants que vous désirez installer.

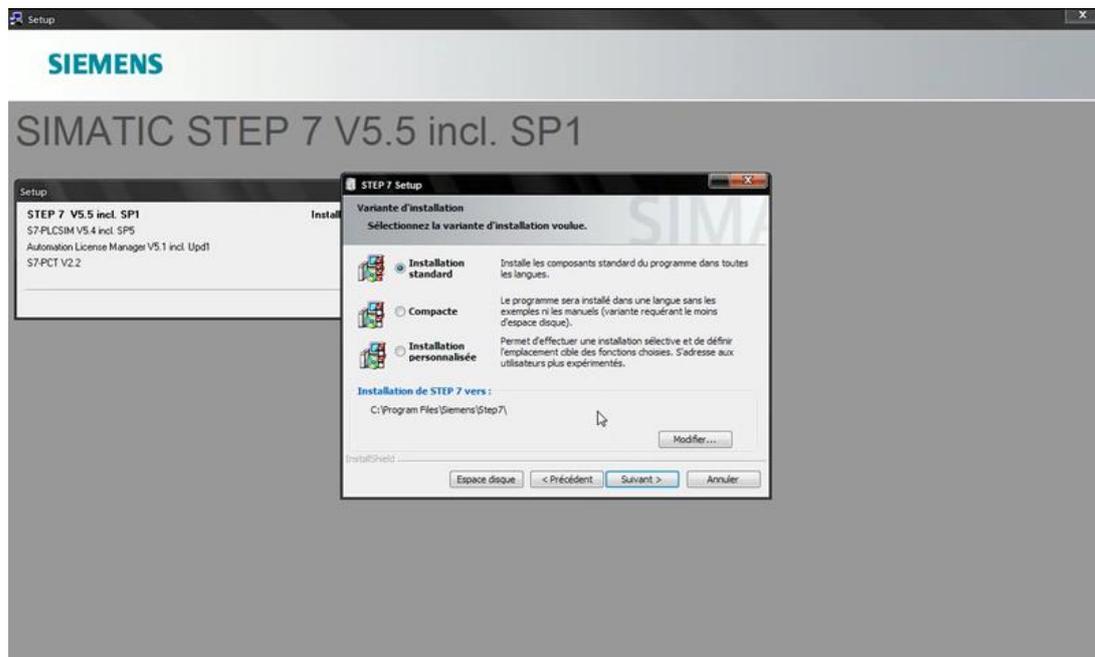


Figure II-18 Le choix de l'installation.

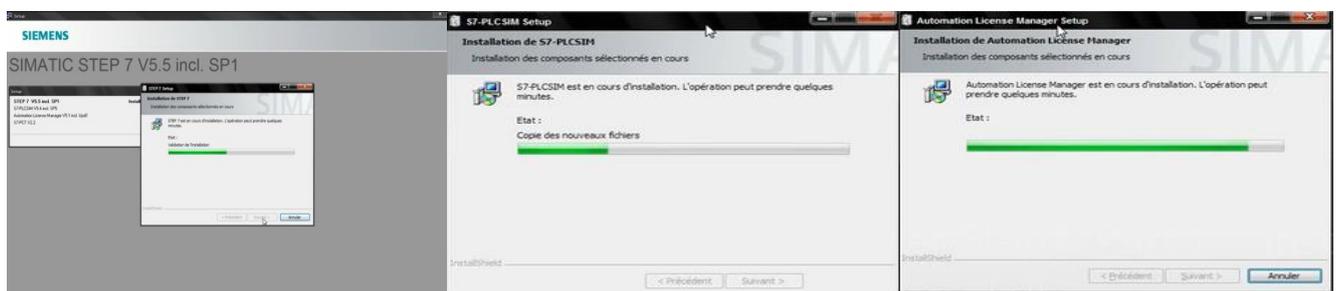


Figure II-19 Le progrès de l'installation.

- Etape 8 : L'installation de "License Keys" :

Le programme vérifie si une "License Key" (clé de licence) correspondante existe sur le disque dur.

- Etape 9 :

Cliquer sur l'icône du « Automation Licence Manager » pour afficher les licences keys.

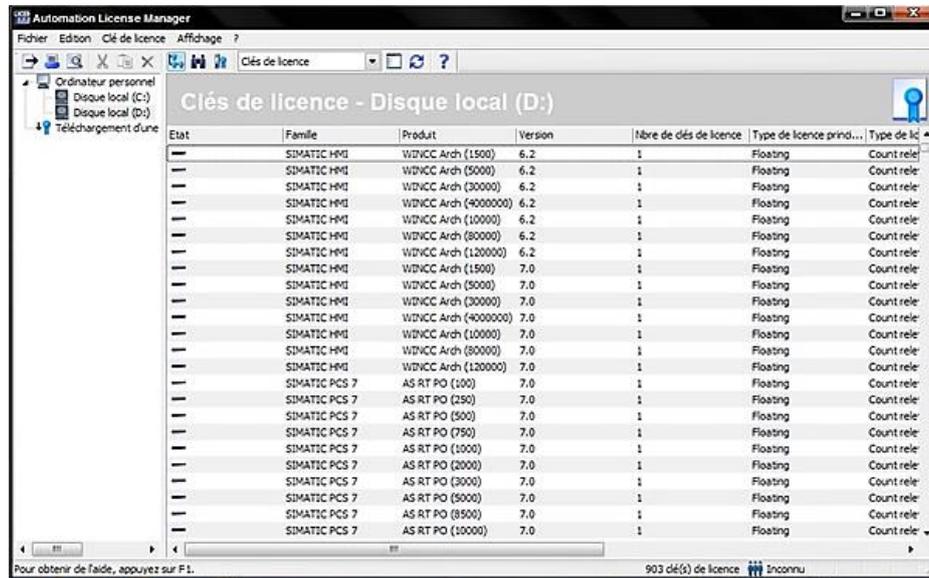


Figure II-22 Automation Licence Manager.

- Fin de l'installation

Un message s'affiche à l'écran pour vous signaler que l'installation a réussi.

Si l'installation a entraîné l'actualisation des fichiers système, vous êtes invité à relancer Windows. Une fois Windows redémarré (démarrage à chaud), vous pouvez cliquer sur l'icône du SIMATIC Manager pour lancer l'interface utilisateur de STEP7.

Une installation sans erreur s'achève par la création d'un groupe de programmes pour STEP7.



Figure II-23 Fin de l'installation

3. Erreurs pendant l'installation :

Les erreurs suivantes entraînent l'interruption de l'installation :

-Si une erreur d'initialisation se produit immédiatement après le démarrage du SETUP, vous avez certainement lancé l'installation dans un environnement autre que Windows.

- L'espace mémoire est insuffisant : selon l'option d'installation choisie, vous avez besoin d'environ 100 Mo d'espace libre sur votre disque dur pour le logiciel de base.

-CD défectueux: adressez-vous à votre agence Siemens si vous constatez qu'un CD est défectueux.

-Erreur de manipulation : recommencez l'installation en observant rigoureusement les instructions.

VI. Conclusion :

D'après ce chapitre, Nous nous sommes familiariser avec l'installation, le démarrage et l'utilisation des logiciels STEP7 et PLCSIM, ces outils logiciels sont importants pour continuer notre travaille vers la phase suivante de notre but qui est la programmation.

Chapitre III : La supervision

I. Introduction :

Dans ce chapitre nous allons parler de la supervision et ses outils. La supervision est aujourd'hui une fonction indispensable dans la plupart des domaines de la technologie, elle nous permet la surveillance, la collecte et la récolte des données en temps réelle. Les interfaces Homme-Machine au niveau des API ont un but de superviser les systèmes. Nous avons utilisé le logiciel WinCC pour le développement de notre interface Homme-Machine avec un écran tactile.

II. La supervision :

Superviser est une fonction importante dans l'accomplissement des activités d'un programme de prestations de services en général et de Planification Familiale en particulier. Cette importance a fait que beaucoup d'organisations qu'elles soient publiques, parapubliques ou privées ont mis en place des structures officielles de supervision avec des superviseurs chargés du suivi des activités de leur programme. [24]

II.1. Définition :

La supervision est un processus qui vise à s'assurer que le personnel accomplit ses activités de façon satisfaisante et à le rendre plus performant dans la réalisation de son travail. La supervision est une assistance et une action de guidage du personnel dans l'accomplissement de sa tâche.

La supervision est une technique industrielle de suivi et de pilotage informatique de procédés de fabrication automatisés.

La supervision va permettre de surveiller, en termes de disponibilité et de performance de l'ensemble de l'infrastructure et applications du système d'information d'une entreprise.

Elle assure la surveillance du bon fonctionnement des du système.

Elle permet de surveiller, rapporter et alerter en cas de fonctionnement anormal d'un élément du parc informatique. [25]

II.2. L'activité de la supervision :

La supervision permet de surveiller, visualiser, analyser, piloter d'agir et d'alerter en cas de problèmes. Elle indique et commande l'état d'un appareil, système ou réseau. [25]

II.3. Supervision industrielle :

La supervision industrielle permet de suivre en temps réel une installation ou une machine industrielle. Elle permet d'avoir un affichage dynamique du processus avec les différentes alarmes, défauts et événements survenant pendant l'exploitation de la machine. De nos jours, de nouveaux procédés de supervision commencent à voir le jour se basant sur les architectures de systèmes distribués permettant la surveillance ou le monitoring à distance. [25]

II.4. Architecture de la supervision :

La supervision est définie comme étant la surveillance et le contrôle de l'exécution d'une opération ou d'un travail accompli par un homme ou par une machine. En présence de défaillance, la supervision prendra toutes les démarches nécessaires pour le retour du système vers un mode de fonctionnement normal. [25]

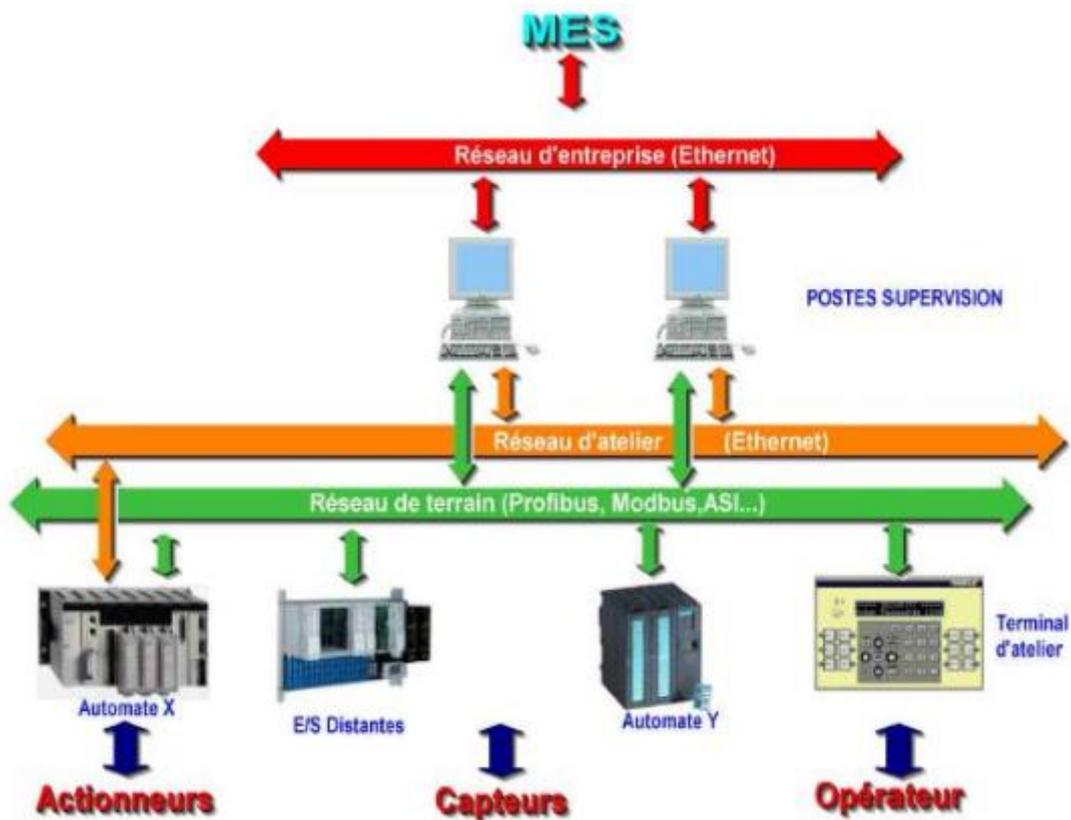


Figure III-1 Architecture de la supervision

II.5. Comment va agir la supervision :

La supervision va agir en fonction des règles, seuil et des points de contrôle préalablement établis.

Elle va aussi analyser les fichiers log (qui regroupe les événements), récupérer les résultats des commandes et des scripts. Elle peut être active ou passive.

Tous ces éléments d'information vont être remontés dans un outil de monitoring qui, en fonction des seuils établit pourra alerter l'administrateur de tout problèmes ou dysfonctionnement de manière précise et en temps réel. Ces seuils vont permettre à l'admin de constater si un point de contrôle est critique (intervention), à surveiller de près car en alerte, ou bien s'il ne mérite pas de surveillance particulière. [26]

II.6. Les fonctions de supervision :

II.6.1. Synoptique :

Fonction essentielle de la supervision, fournit une représentation synthétique, dynamique et instantanée de l'ensemble des moyens de production de l'unité.

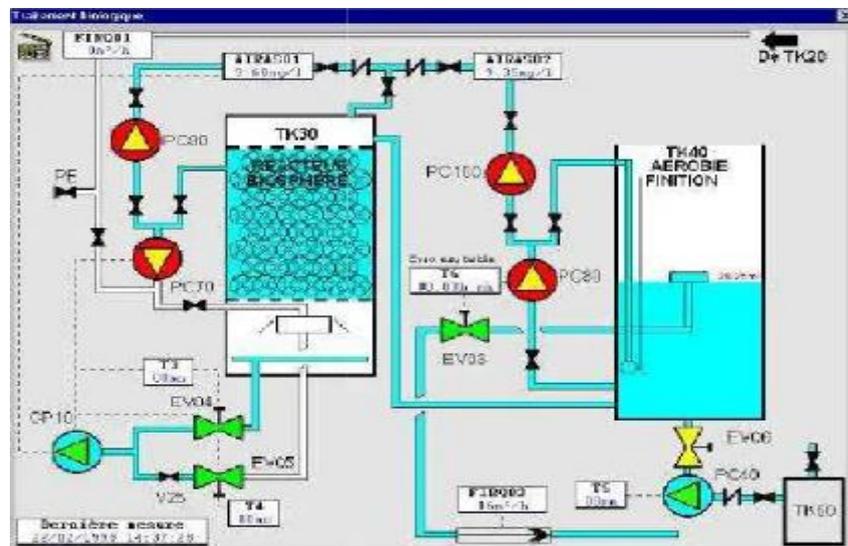


Figure III-2 Représentation d'un synoptique.

II.6.2. Des courbes :

- Donne une représentation graphique de différentes données de processus.
- Donne les outils d'analyses des variables historiées.

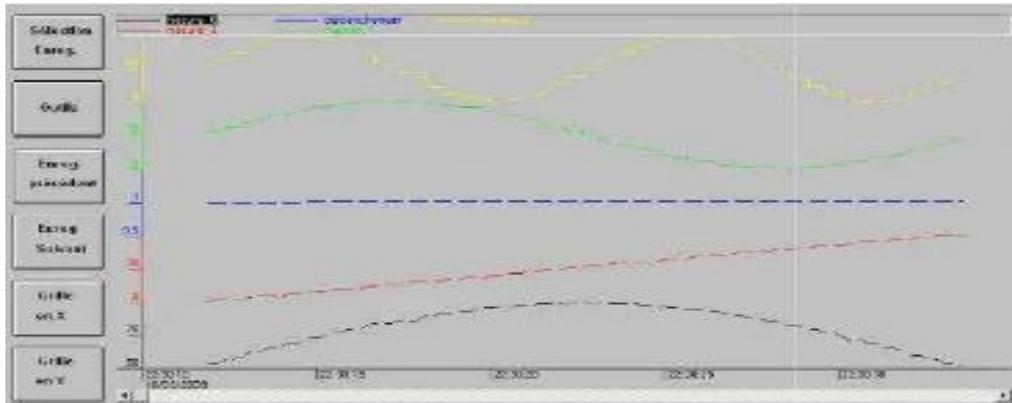


Figure III-3 Représentation d'une courbe.

II.6.3. Les alarmes :

- Calculer en temps réel les conditions de déclenchement des alarmes.
- Affiche l'ensemble des alarmes selon les règles de priorité.
- Assure l'enregistrement de toutes les étapes de traitement de l'alarme.



Figure III-4 Représentation des alarmes.

II.6.4. Historisation des procédés :

- Permet la sauvegarde périodique de grandeurs.
- Fournit les outils de recherche dans les données archivées.
- Permet de garder la traçabilité de données de production.

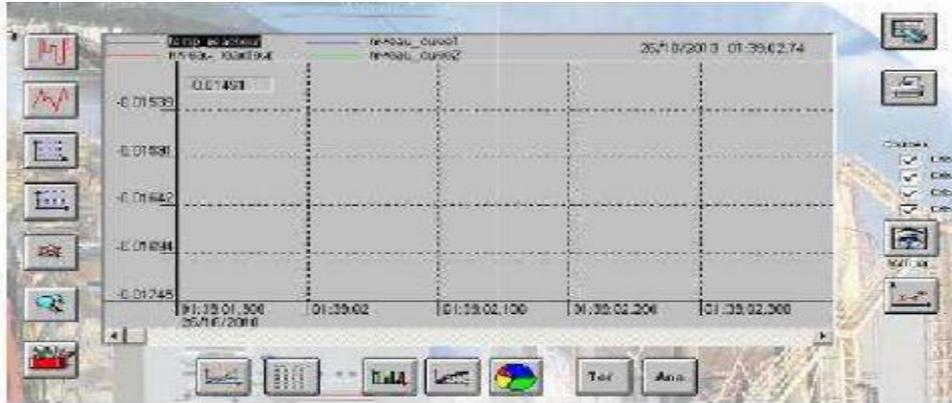


Figure III-5 Représentation d'une historisation.

II.6.5. Gestion des gammes de fabrication :

- Donne un outil de gestion des lots de fabrication.
- Gere les paramètres de réglages des machines pour chacun des lots.

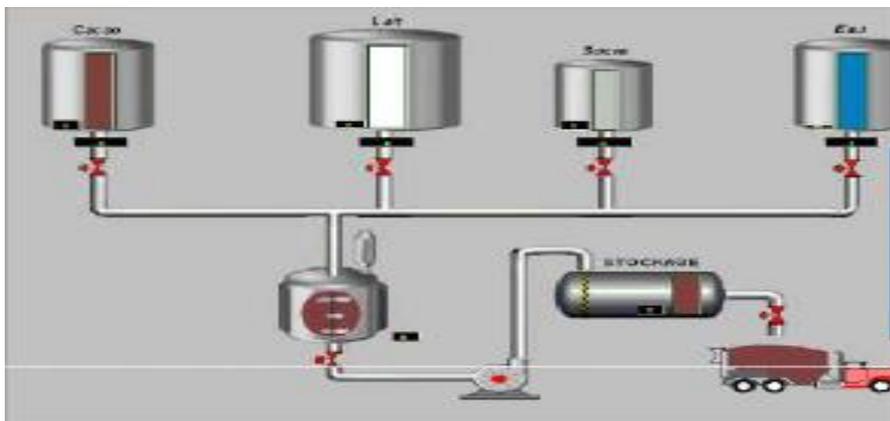


Figure III-6 Représentation d'une gamme de fabrication.

II.7. Pourquoi superviser ?

- Contrôler la disponibilité des services /fonctions.
- Contrôler l'utilisation des ressources.
- Vérifier qu'elles sont suffisantes.
- Détecter et localiser des défauts.
- Diagnostic des pannes.
- Prévoir l'évolution.

-Suivie des variables.

II.8. Quelque organe de supervision :

II.8.1. Les voyants :

Sont des indicateurs lumineux transmettant une information.



Figure III-7 Des voyants.

II.8.2. Les afficheurs avec divers types :



Figure III-8 Exemples d'afficheurs.

II.8.3. Les organes d'émission des consignes :

C'est-à-dire l'opérateur qui va donner les consignes à l'automate pour transmettre l'information aux pré actionneurs pour effectuer l'action.

Les manipulateurs



Les sélecteurs lumineux



Les claviers



Les boutons lumineux



Figure III-9 Exemples des organes d'émission des consignes.

Les boutons poussoirs



Les sélecteurs



Les sélecteurs à clé



Les boutons coup de poing



Figure III-10 Les boutons poussoirs et les sélecteurs.

II.8.4. Les terminaux d'exploitation :

Un terminal est une variété de périphérique réseau placé à l'extrémité d'un nœud .Le terminal est un point d'accès de communication entre l'homme et la machine. Il y a 2 types de terminaux : Les terminaux d'exploitation texte et graphique.

Les terminaux d'exploitation texte



Les terminaux d'exploitation graphiques



Figure III-11 Les terminaux d'exploitation.

III. L'interface homme machine :

III.1. Introduction :

L'interaction avec les ordinateurs est aussi vieille que les ordinateurs eux-mêmes. En effet, un ordinateur est une machine programmable, il faut donc pouvoir y entrer les données et programmes et visualiser les résultats. Les premiers ordinateurs disposaient typiquement de lecteurs de cartes perforées et d'imprimantes, mais ces dispositifs ne permettaient pas une réelle interaction, pendant l'exécution du programme. Aussi l'histoire de l'interaction Homme-machine (IHM) débute réellement au début des années 1960 avec les travaux pionniers d'Ivan Sutherland sur SketchPad, qui ont montré comment un opérateur pouvait interagir en temps réel avec une machine exécutant un logiciel complexe.

Les systèmes de contrôle industriels continuent d'évoluer et, dans le monde d'aujourd'hui, les tâches que les opérations doivent accomplir peuvent changer fréquemment. Pour gérer cette complexité, vos contrôles doivent être flexibles et ergonomiques. C'est l'avantage de l'IHM. Avec une IHM, vous pouvez facilement communiquer avec les machines et obtenir des données opérationnelles sur l'ensemble de vos équipements et vos installations. [27]

III.2. C'est quoi l'IHM ?

- Interface homme-machine.
- Interaction homme-machine.

Et aussi :

- Communication homme-machine.
- Dialogue homme-machine.
- Interaction personne-machine. [34]

III.3. Définition :

III.3.1. Interface homme-machine :

L'IHM est un ensemble de dispositifs matériel et logiciels permettant à un utilisateur de contrôler et gérer un système interactif.

III.3.2. Interaction homme-machine :

Ensemble des actions permettant la communication entre un système interactif et son utilisateur humain.

IHM, c'est l'abréviation d'Interactions Humain-Machine ou Interactions Homme-Machine. C'est un domaine de recherche et de pratique qui fait partie de l'informatique, mais qui est à l'interface avec d'autres disciplines, comme par exemple les sciences cognitives, l'ergonomie, le design, l'électronique, mais aussi d'autres disciplines. [33]

III.3.3. Système informatique interactif :

- Il fournit à l'utilisateur, lors de son exécution, une représentation perceptible d'une partie de son état interne, afin que ce dernier puisse le modifier en fournissant des entrées.
- Les entrées permettent de modifier l'état interne du système, et il y a ainsi interaction: les entrées fournies par l'utilisateur dépendent des sorties produites par le système et inversement.
- Le système est ouvert : les dépendances entre entrées et sorties sont inaccessibles au système. [34]

III.4. Adapter l'IHM :

III.4.1. Caractéristiques de l'utilisateur :

- Différences physiques (âge, handicap).
- Connaissances et expériences (dans le domaine de la tâche: novice, expert, professionnel, en informatique, sur le système: usage occasionnel, quotidien).
- Caractéristiques psychologiques (visuel / auditif, logique / intuitif, analytique / synthétique).
- Caractéristiques socioculturelles (format de date, langue, signification des icônes, des couleurs, ...).[33]

III.4.2. Contexte :

- Grand public (proposer une prise en main immédiate).
- Loisirs (rendre le produit attrayant).
- Industrie (augmenter la productivité, systèmes critiques: sécurité). [33]

III.4.3. Caractéristiques de la tâche :

– répétitive, régulière, occasionnelle, sensible aux modifications de l'environnement, contrainte par le temps. [33]

III.4.4. Contraintes techniques :

- Plateforme.
- Taille mémoire.
- Ecran, capteur, effecteur.
- Réutilisation de code ancien.[33]

III.5. Historique :

III.5.1. 1945-1970 : Les prémisses :

-Dispositifs d'entrée sortie limités :

- Lecteurs de carte :

Un lecteur de carte est un appareil connecté ou indépendant qui sert à accéder aux données contenues dans une carte à puce ou un badge électronique. [34]

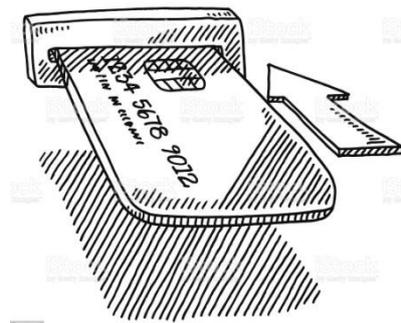


Figure III-12 Lecteurs de carte.

-Tableau de bord (voyant) :

Le tableau de bord de gestion est un outil d'évaluation de l'organisation d'une entreprise ou d'une institution constitué de plusieurs indicateurs de sa performance à des moments donnés ou sur des périodes données.

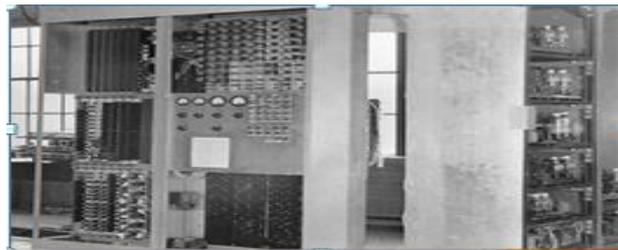


Figure III-13 Tableau de bord.

-Imprimant :

Une imprimante est un engin permettant d'obtenir un document sur papier à partir d'un modèle informatique du document. Par exemple, un texte écrit via un logiciel de traitement de texte sur ordinateur pourra être imprimé pour en obtenir une version papier.



Figure III-14 Imprimant.

III.5.2. Langages de commandes :

Ensemble de commandes et syntaxes permettant d'effectuer une action sur un système informatisé. Les langages d'interrogation sont des langages de commandes.

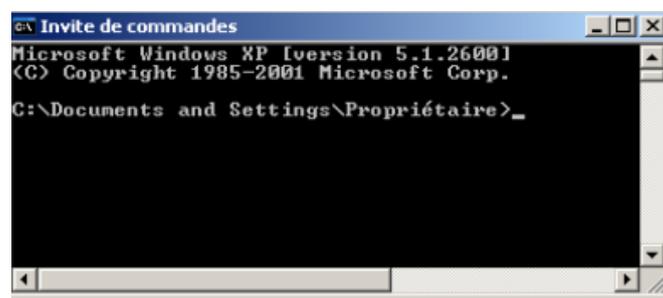


Figure III-15 Invite de commande.

III.5.3. 1970s : Les ordinateurs modernes :

-Nouveaux dispositifs d'entrée-sortie :

-1963 : écran graphique et stylo optique :

La définition d'écran, couramment appelée la résolution d'écran, est le nombre de points ou pixels que peut afficher un écran. La définition est le produit du nombre de points selon l'horizontale par le nombre de points selon la verticale de l'affichage.

Le stylo optique est un stylet informatique, permettant de dessiner ou sélectionner des options directement sur les écrans à tubes cathodiques. [34]

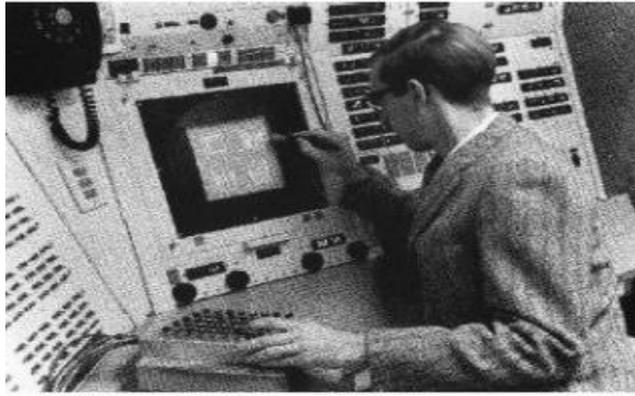


Figure III-16 Un opérateur écrit sur un écran graphique.

-Première souris :

Une souris est un dispositif de pointage pour ordinateur. Elle est composée d'un petit boîtier fait pour tenir sous la main, sur lequel se trouvent un ou plusieurs boutons



Figure III-17 Première souris.

III.5.4. 1980 : applications grand publiques :

-Apparition des micro-ordinateurs :

L'ordinateur, qui par définition est un système informatique, a d'abord été un instrument de calcul pour des applications purement scientifiques (1945). Il est devenu à la fin des années 60 un instrument de gestion pour les entreprises. Avec l'avènement, dans les années 80, de l'ordinateur personnel (le PC : Personal Computer) est né le micro-ordinateur, c'est à dire un ordinateur individuel fonctionnant de manière autonome grâce à son microprocesseur, à son disque dur et sa mémoire interne. [28]



Figure III-18 Micro-ordinateurs.

III.5.5. Evolution des interfaces :

Systèmes plus conviviaux, faciles à comprendre et à utiliser.

Interface graphique : manipulation directe, action directe sur les objets représentés à l'écran.



Figure III-19 Interface graphique

a. Dispositifs de sorties :

-Ecran :

Un écran d'ordinateur est un périphérique de sortie vidéo d'ordinateur. Il affiche les images générées par la carte graphique de l'ordinateur. Il permet de donner l'impression de mouvement.



Figure III-20 Ecran.

-Imprimant 3D :

L'impression 3D est une technique de fabrication dite additive qui procède par ajout de matière, contrairement aux techniques procédant par retrait de matière comme l'usinage. L'impression 3D permet de réaliser des objets usuels, des pièces détachées ou encore des

prototypes destinés aux essais. Le point de départ est un fichier informatique représentant l'objet en trois dimensions, décomposé en tranches. Ces informations sont envoyées à une imprimante 3D qui va réaliser la fabrication par ajout de couches successives. [29]

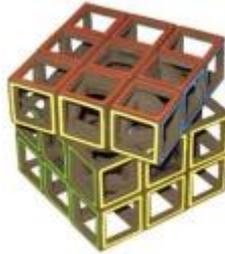


Figure III-21 Représentation 3D

-Son et synthèse vocale :

La synthèse vocale est une technique informatique de synthèse sonore qui permet de créer de la parole artificielle à partir de n'importe quel texte.



Figure III-22 Vocal.

-Représentation de fichiers et le contexte :

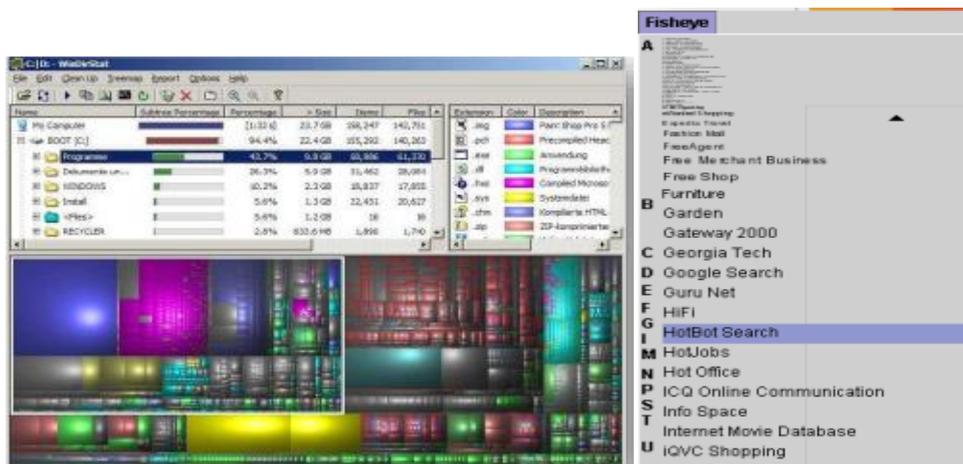


Figure III-23 Différente représentations de fichier.

-Visualisation d'information en 2D :

Bidimensionnel ou 2D sont des expressions qui caractérisent un espace conçu en termes de largeur et de hauteur. Il ne comporte pas de profondeur.

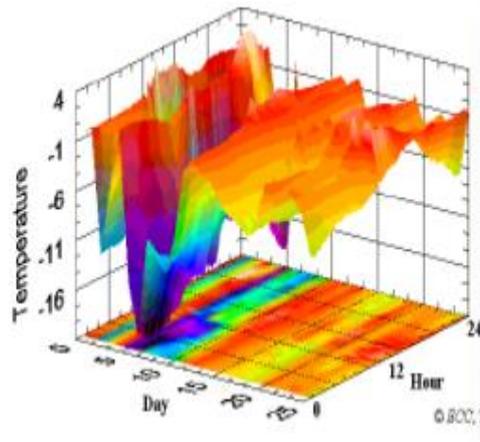


Figure III-24 Représentation en 2D.

-Visualisation d'information en 3D :

Trois dimensions ou tridimensionnel ou 3D sont des expressions qui caractérisent l'espace qui nous entoure, tel que perçu par notre vision, en termes de largeur, hauteur et profondeur.

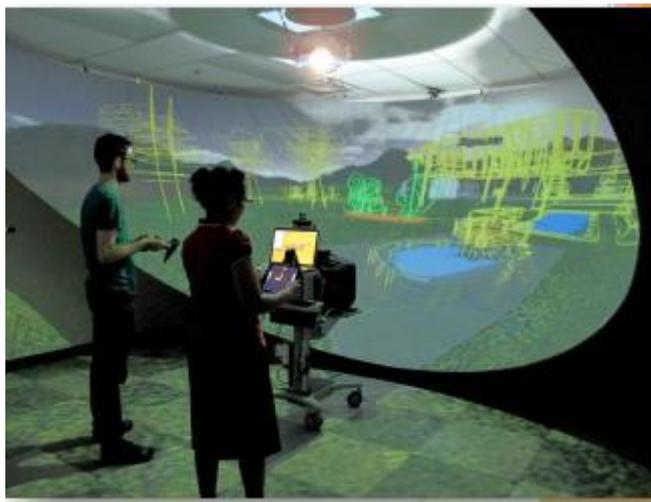


Figure III-25 Représentation en 3D.

b. Dispositifs d'entrés :

-Clavier :

Un clavier d'ordinateur est une interface Homme-machine qui permet la saisie de textes, de numéros, de symboles et de commandes à l'aide d'un certain nombre de touches. En général, il s'agit d'une série de boutons en plastique disposés selon des normes régionales, chacune étant reliée à un interrupteur électronique. Chaque frappe émet un code qui est transmis à l'ordinateur et qui déclenche une fonction comme la saisie textuelle ou le déplacement à l'écran.



Figure III-26 Clavier.

-Dispositif de pointage :

Un dispositif de pointage est un périphérique d'entrée permettant à un utilisateur d'entrer des données spatiales comme la souris et joystick.



Figure III-27 Dispositifs de pointage.

-Son :

Reconnaissance vocale, reconnaissance de son.

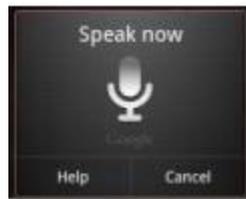


Figure III-28 Synthèse vocale.

-Code barre 2D :

Le code barre 2D est l'héritier du code barre classique à une dimension. Alors que le code barre classique ne permet qu'un codage horizontal, le code barre 2D joue sur les deux dimensions et peut donc transmettre plus d'informations.

Le code barre 2D présente de nombreuses application comme le domaine du marketing mobile, des contenus presse, des affiches ou un packaging.[30]



Figure III-29 Code barre 2D.

-Code tactile :

Système de communication non oral permettant de s'exprimer en utilisant le toucher selon un code compris par le récepteur. [31]



Figure III-30 Code tactile.

-Lecteur d'empreinte :

Les empreintes digitales sont une signature que nous laissons derrière nous à chaque fois que nous touchons un objet. [32]

La lecture des empreintes digitales d'une personne est une technique faisant usage des empreintes digitales d'un individu comme moyen de sécurisation, d'identification ou de protection de donnée.



Figure III-31 Lecteur d'empreinte.

-Reconnaissance de visage :

La reconnaissance faciale est un logiciel permettant d'identifier un individu d'après une photo ou d'après un modèle 3D créé à partir de plusieurs images de la personne. [35]



Figure III-32 Reconnaissance visage.

IV. WINCC FLEXIBLE:

IV.1. Définition :

WinCC est le logiciel phare de Siemens permettant de créer des interfaces homme-machine sur pupitre tactile (IHM : interface homme machine) ou sur l'écran. La dernière version de WinCC est intégrée à TIA Portal, pour vous familiariser avec le logiciel WinCC.[36]

IV.2. Fonctionnalités:

Le WinCC flexible offre, pour les applications au niveau machine (couvert jusqu'à présent par la famille ProTool), un considérable gain d'efficacité dans la configuration ainsi que des concepts d'automatisation innovateurs.

Dans les secteurs proches du procès, de la construction d'installations et de machines ainsi que de la construction de machines de série, SIMATIC WinCC flexible permet en outre :

- d'améliorer la productivité (efficacité de la configuration) lors de la création de projets IHM.
- de réaliser des concepts d'IHM et d'automatisation innovants dans le cadre de réseaux TCP/IP et du Web.
- d'accroître la disponibilité des machines et installations par de nouveaux concepts de maintenance.
- d'accéder facilement, en toute sécurité aux données de procès à partir de n'importe quel endroit du globe. [36]

IV.3. Caractéristiques techniques:

- Intégration dans les automates programmables.
- Manipulation du projet.

- Editeurs de tableau.
- Signalisation et acquittement d'événements.
- Gestion de données orientée objet avec possibilités d'édition et de recherche confortables.
- Bibliothèques d'objets de configuration prédéfinis ou confectionnés par l'utilisateur.
- Prise en charge linguistique.
- Visual Basic Script Support.
- Runtime.
- Test et assistance à la mise en service.
- Communication ouverte entre systèmes HMI et systèmes de niveau supérieur. [36]

IV.4. Avantage:

- La cohérence du logiciel de configuration assure une réduction des coûts de maintenance et d'entretien tout en étant une garantie d'évolutivité du produit.
- Outils intelligents pour une configuration simple et efficace.
- Prise en charge exhaustive de configurations multilingues pour une mise en œuvre globale.
- Fonctionnalité de Runtime flexible grâce à des scripts Visual Basic.
- Des concepts de maintenance innovateurs avec commande à distance, le diagnostic, l'administration via intranet/Internet et la communication par courrier électronique améliorent la disponibilité.
- Prise en charge de solutions d'automatisation distribuées simples sur la base de réseaux TCP/IP au niveau machine.[36]

IV.5. Différence entre WinCC et WinCC Flexible :

La différence entre WinCC et WinCC flexible est :

- WinCC est un logiciel de supervision.
- WinCC Flexible est un logiciel pour pupitre opérateur.

IV.6. WinCC flexible Runtime :

WinCC flexible Runtime est un logiciel performant et facile à utiliser pour la visualisation du processus des projets créés avec le logiciel de configuration WinCC flexible Advanced.

Il se distingue par son interface utilisateur entièrement graphique, basée sur la technique des fenêtres. Il permet grâce à des temps de réaction rapides une conduite de processus sûre, le mode manuel à vue sur la machine ainsi qu'une collecte sûre des données. [36]

IV.7. Intégration dans SIMATIC STEP 7 :

Les variables du processus représentent la liaison pour la communication entre le système d'automatisation et le système IHM. Sans les avantages de Totally Integrated Automation, vous devriez définir chaque variable à deux reprises : une fois pour le système d'automatisation et une fois pour le système IHM.

L'intégration de SIMATIC STEP 7 dans l'interface de configuration permet de diminuer la fréquence des erreurs et de réduire les tâches de configuration nécessaires. Durant la configuration, vous accédez directement à la table des mnémoniques de STEP 7 ainsi qu'aux paramètres de communication :

- La table des mnémoniques de STEP 7 contient la définition des points de données (p. ex. adresses ou types de données) que vous avez paramétré lors de la création du programme de commande.
- Les paramètres de communication contiennent les adresses de bus ainsi que les protocoles de commande. Vous définissez les paramètres de communication avec NetPro, p. ex.[36]

IV.8. Fonctions des différents éditeurs :

Il existe les éditions suivantes de WinCC flexible :

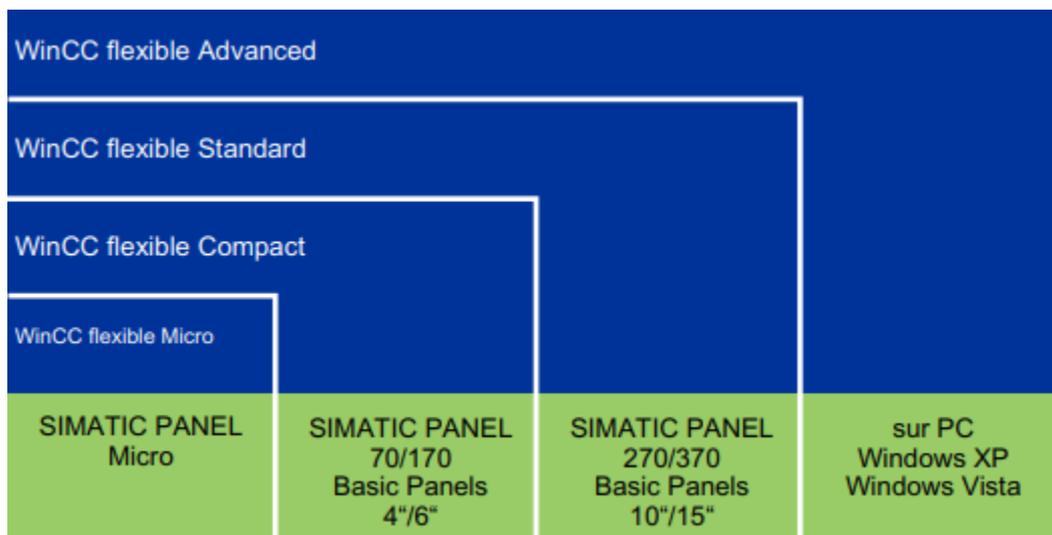


Figure III-33 Les éditions de WinCC flexible.

IV.9. L'installation du WINCC Flexible 2008 :

IV.9.1. Les conditions d'installation :

- Mémoire principale : RAM 1 Go min. 2 Go recommandé
- Processeur : Pentium IV min. ou processeur comparable avec 1,6 GHz
- Graphique : XGA 1024 x 768, WXGA pour Notebooks, Prise en charge des couleurs 16 bits.
- Espace mémoire libre sur le disque dur : 2 Go min. pour l'installation d'une langue. 200 Mo supplémentaires pour chaque langue supplémentaire.
Si WinCC flexible n'est pas installé sur la partition système, les besoins en mémoire se répartissent comme suit :
 - env. 700 Mo sur la partition système.
 - env. 200 Mo par langue supplémentaire sur la partition d'installation.

IV.9.2. Marche à suivre :

- Etape 1 :

Insérez le DVD produit. Le programme d'installation démarre automatiquement. Si le programme d'installation ne démarre pas automatiquement, lancez-le manuellement en effectuant un double clic sur le fichier "setup.exe" sur le DVD produit.

- Etape 2 :

Choisissez la langue du programme d'installation. Les boîtes de dialogue sont affichées dans la langue du programme d'installation.

- Etape 3 :

Dans la boîte de dialogue suivante, ouvrez les informations produit et lisez-les attentivement.

- Etape 4 :

Lisez et acceptez le contrat de licence.



Figure III-34 Acceptation les conditions de licence.

- **Etape 5 :**

Choisissez les langues d'interface a installé. Vous pouvez commuter l'interface de configuration entre les langues sélectionnées.



Figure III-35 Le choix de la langue.

- **Etape 6 :**

Pour une installation complète, choisissez installation complète.

Selon l'état des différents composants, de petites icônes s'affichent à la gauche de ceux-ci.

Vous trouverez des informations complémentaires sur cette icône en utilisant le bouton "Aide".

Démarrez l'installation avec "Suivant" L'installation de WinCC flexible est démarrée.

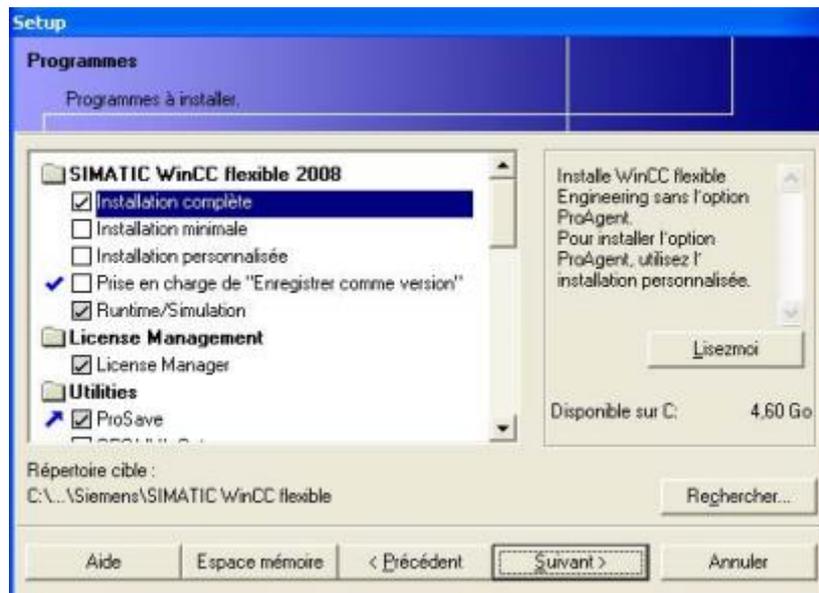


Figure III-36 Le choix de l'installation complète.

- Etape 7 :

S'il n'existe sur le PC aucune clé de licence pour les composants installés, vous êtes invité à transférer les clés de licence une fois l'installation terminée. Suivez les instructions des boîtes de dialogue pour la concession de licence et transférez les clés de licence du lieu d'archivage sur le disque dur de votre PC. Vous pouvez également transférer les clés de licence ultérieurement à l'aide d'Automation License Manager.

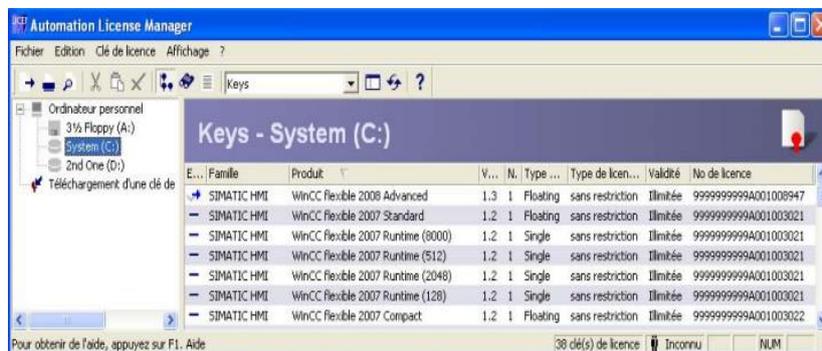


Figure III-37 Automation Licence Manager.

V. Conclusion :

Dans ce chapitre nous avons présenté des généralités sur la supervision ainsi que ces outils, l'interface Homme-Machine et ces fonctionnalités, ainsi que le logiciel WinCC Flexible, Nous avons bien compris comment installer et bien maîtriser comment réaliser une interface Homme-Machine dans ce logiciel.

Ce travail est la deuxième phase de notre but à réaliser qui est la création d'une interface Homme-Machine.

Chapitre IV : La programmation et la création d'IHM

I. Introduction :

Dans ce chapitre nous allons programmer la sous station de contrôle sous le logiciel STEP7 après l'étude du cahier de charge adéquat. Nous allons aussi réaliser une simulation du fonctionnement de la station sur PLCSIM. En deuxième partie nous allons concevoir l'interface Homme-Machine sous WinCC de cette sous station pour faire la supervision de son fonctionnement.

II. Création du projet dans l'éditeur SIMATIC Manager :

Une fois Windows démarré, vous trouverez dans l'interface Windows une icône pour SIMATIC Manager qui permet d'accéder au logiciel STEP 7.

II.1. Les étapes pour créer un projet :

- Etape 1 :

Vous démarrez rapidement STEP 7 en effectuant un double clic sur l'icône "SIMATIC Manager". La fenêtre de SIMATIC Manager s'ouvre alors. De là, vous pouvez accéder à toutes les fonctions que vous avez installées aussi bien du logiciel de base que des logiciels optionnels.

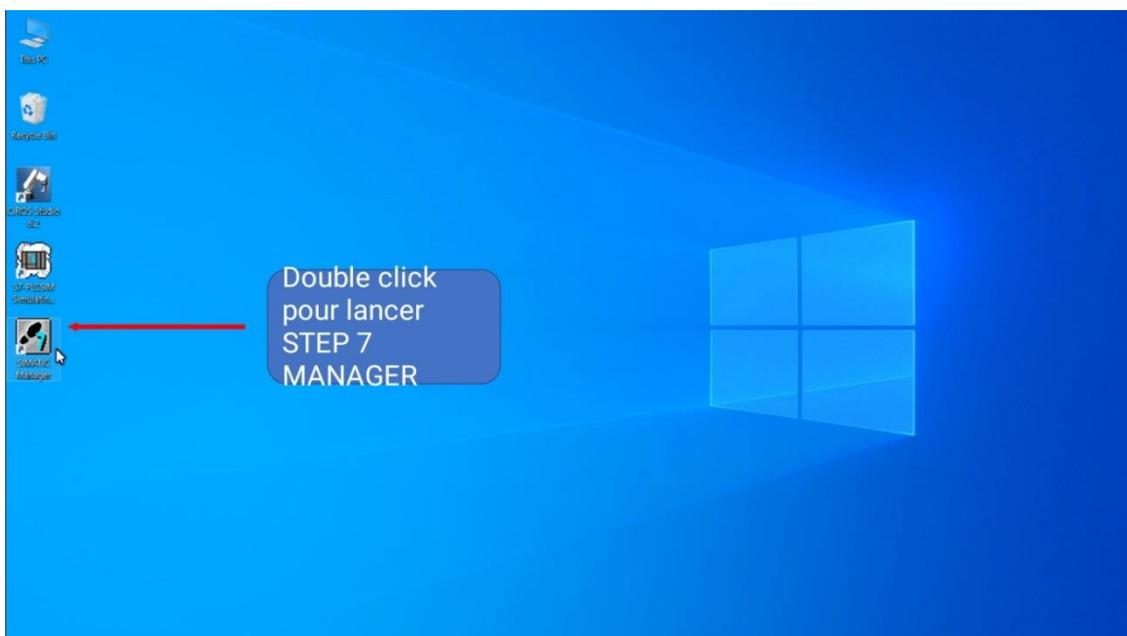


Figure IV-1 Clic sur l'icône de STEP7 MANAGER.

- Etape 2 :

Création de projet :

Pour réaliser votre tâche d'automatisation au sein d'un gestionnaire de projets, vous devez d'abord créer un nouveau projet, Il sera créé dans le répertoire que vous avez sélectionné pour les projets.

Insertion de stations :

La station représente la configuration matérielle de l'automate programmable et contient les données pour la configuration et le paramétrage des divers modules.

Vous devez créer la station en choisissant la commande Insertion > Station.

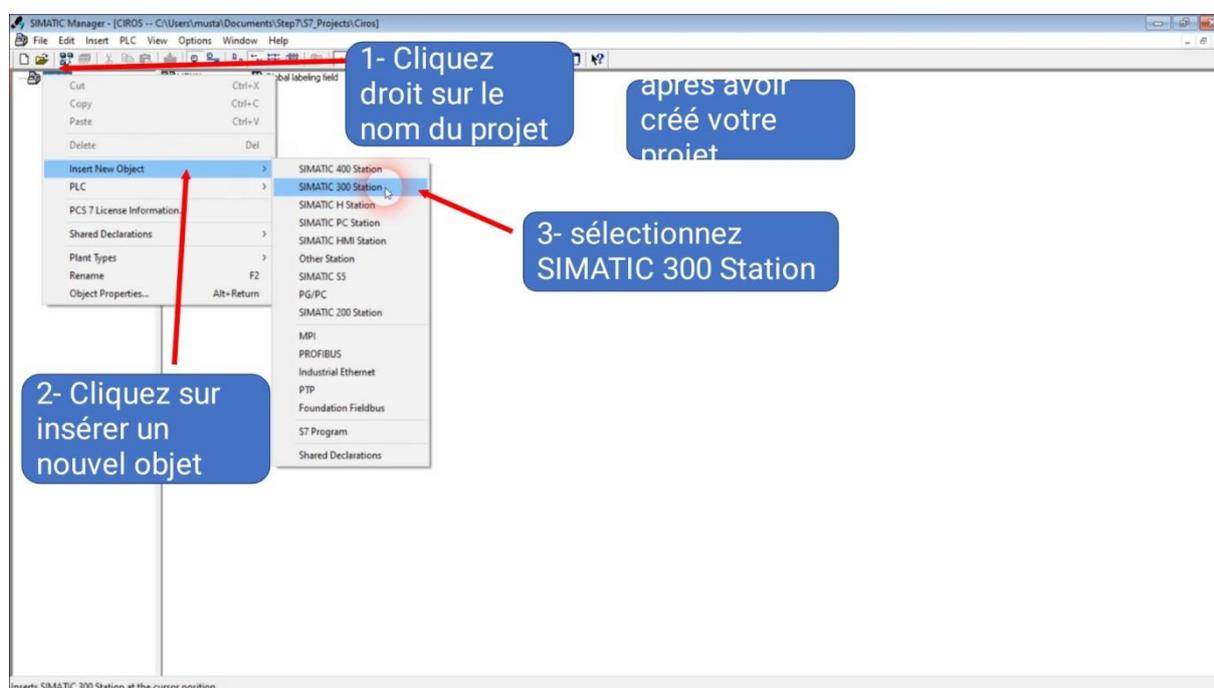


Figure IV-2 La création et l'insertion de projet.

- Etape 3 :

Réalisation de la configuration matérielle :

Dans la configuration matérielle, vous utilisez un catalogue des modules pour définir la CPU et tous les modules contenus dans sa commande.

Une fois que vous avez sauvegardé et quitté la configuration matérielle, un programme S7 est automatiquement créé comme dossier du logiciel ainsi qu'une table de liaisons pour chaque module programmable que vous avez créé lors de la configuration.

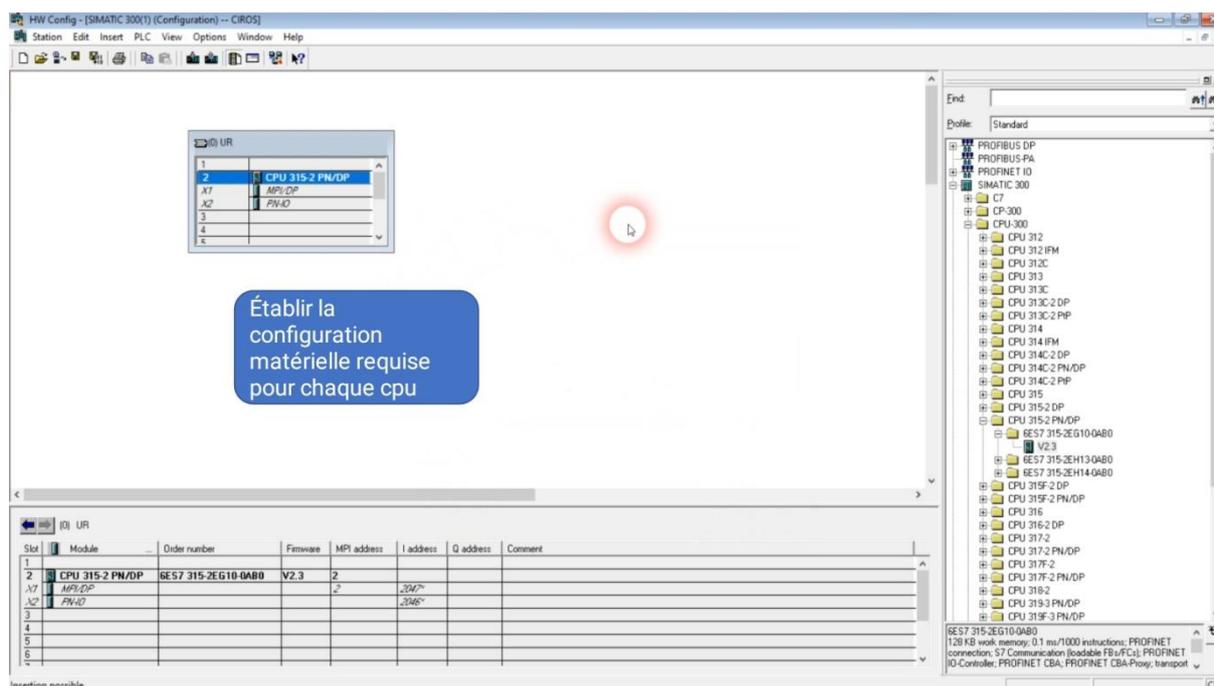


Figure IV-3 La configuration matérielle.

- Etape 4 :

Création de la table mnémotechnique :

La table mnémotechnique vous permet d'utiliser des désignations parlantes à la place d'adresses absolues. En combinant l'usage de mnémoniques courts et de commentaires explicites, vous répondez à la fois aux besoins d'une programmation concise d'une programmation bien documentée.

	Etat	Mnémotechnique	Opérande	Type de do	Commentaire
1		1B1	E 0.4	BOOL	Vérin de levage en haut
2		1B2	E 0.5	BOOL	Vérin de levage en bas
3		1M1	A 0.0	BOOL	Vérin de levage vers le bas
4		1M2	A 0.1	BOOL	Vérin de levage vers le haut
5		2B1	E 0.6	BOOL	Vérin d'éjection rentré
6		2M2	A 0.2	BOOL	Vérin d'éjection vers l'avant
7		3M1	A 0.3	BOOL	Glissière à coussin d'air activée
8	B	B2	E 0.1	BOOL	pièce autre que noire
9		B4	E 0.2	BOOL	Barrière lumineuse de sécurité
10		B5	E 0.3	BOOL	Hauteur de pièce à usiner correcte
11		coul1/haut0	M 0.3	BOOL	la sélection par couleur ou bien par hauteur
12		G7_STD_3	FC 72	FC 72	
13		IP_FI	E 0.7	BOOL	Station en aval libre
14		IP_N_FO	A 0.7	BOOL	Station occupée
15		P1	A 1.0	BOOL	Voyant START allumé
16		P2	A 1.1	BOOL	Voyant position de repos (Reset)
17		P3	A 1.2	BOOL	Voyant matériau allumé
18		Part_AV	E 0.0	BOOL	Pièce présente
19		RESET	M 0.0	BOOL	RESET HMI
20		S_Auto	E 1.2	BOOL	Automatic/manuel
21		S1	E 1.0	BOOL	Touche START
22		S2	E 1.1	BOOL	Touche STOP (contact à ouverture)
23		S3	E 1.3	BOOL	Touche mise en référence/RESET
24		START	M 0.1	BOOL	START HMI
25		STOP	M 0.2	BOOL	STOP HMI
26		TIME_TCK	SFC 64	SFC 64	Read the System Time

Figure IV-4 La table mnémotechnique.

- Etape 5 :

Création de la table des liaisons :

La table de liaisons vide est automatiquement créée pour chaque module programmable. Elle est utilisée pour la définition de liaisons de communication entre modules programmables au sein d'un réseau. A son ouverture apparaît une fenêtre contenant une table pour la définition de liaisons entre modules programmables. Des informations détaillées sont données dans Mise en réseau de stations au sein d'un projet.

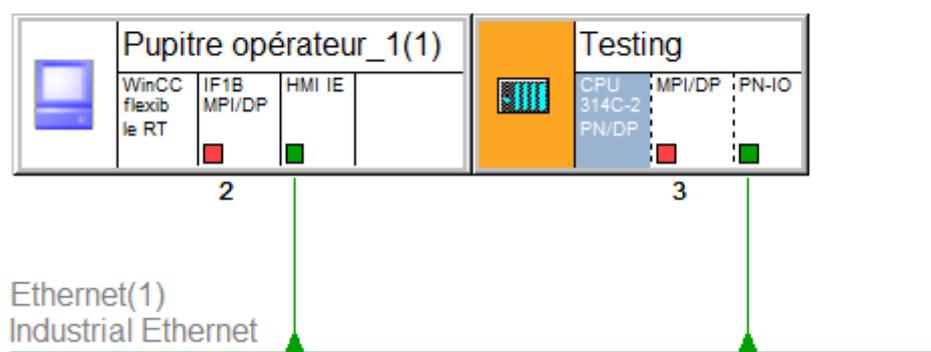


Figure IV-5 La liaison Ethernet.

- Etape 6 :

Insertion d'un programme S7 :

Le logiciel destiné aux modules programmables est stocké dans des dossiers d'objets. Pour un tel dossier d'objets s'appelle "ProgrammeS7", pour les modules SIMATIC S7".

Notre langage de programmation graphique permettant l'interconnexion graphique de fonctions existantes. Ces fonctions couvrent un large éventail allant de combinaisons logiques simples à des régulations et commandes complexes. Un grand nombre de ces fonctions est disponible sous la forme de blocs dans une bibliothèque. La programmation se fait en copiant des blocs sur un diagramme et en reliant les connecteurs de blocs par des lignes.

La figure suivante montre notre programme S7 dans un module programmable d'u station SIMATIC 300.

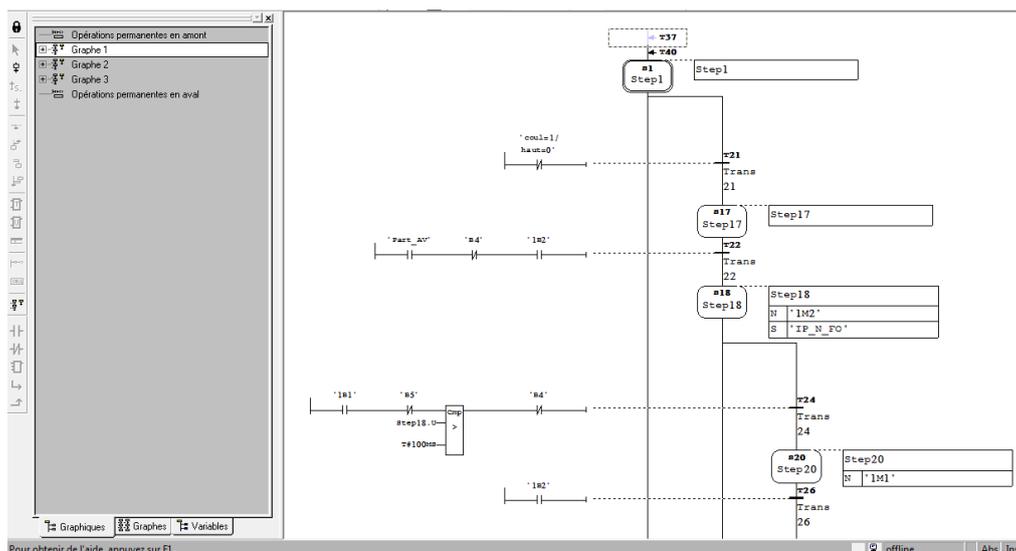


Figure IV-6 La programmation de notre station Testing selon le cahier de charge.

- Etape 7 :

Paramétrage de l'interface PG/PC :

Dans SIMATIC Manager, ouvrez la boîte de dialogue "Paramétrage de l'interface PG/PC" via le menu "Outils > Paramétrage de l'interface PG/PC..." et choisissez l'interface souhaitée dans la liste.

- Etape 8 :

La simulation via S7-PLCSIM :

Le simulateur PLCSIM de l'outil step7 pour l'api de Type Siemens nous a permis d'exécuter et de tester le programme selon le langage de modélisation ainsi le cahier de charge de notre station à automatiser.

Les étapes à suivre afin de lancer le PLCSIM sont détaillées dans les figures suivantes :

a. L'activation de PLCSIM.

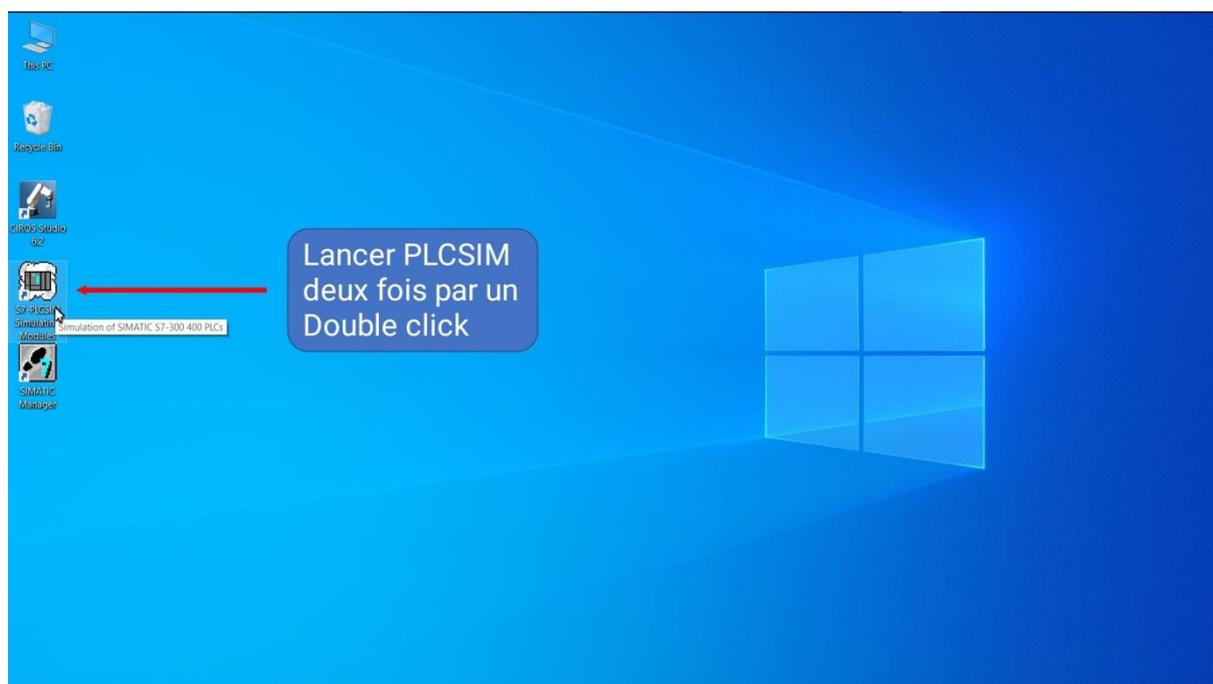


Figure IV-7 Le lancement de PLCSIM.

b. Le chargement du bloc dans PLCSIM :

Dans ce cas, on va sélectionner tous les blocs puis on charge comme premier temps ce dernier dans le PLCSIM comme il est détaillé dans **Figure 0-8**.

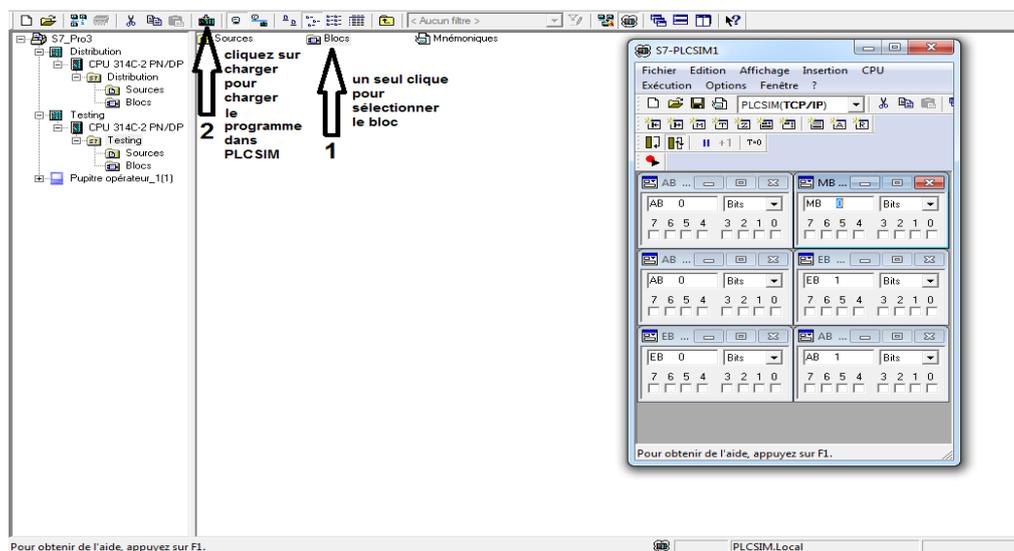


Figure IV-8 Le chargement du bloc dans PLCSIM.

c. L'activation de la visualisation dans le S7-Graph :

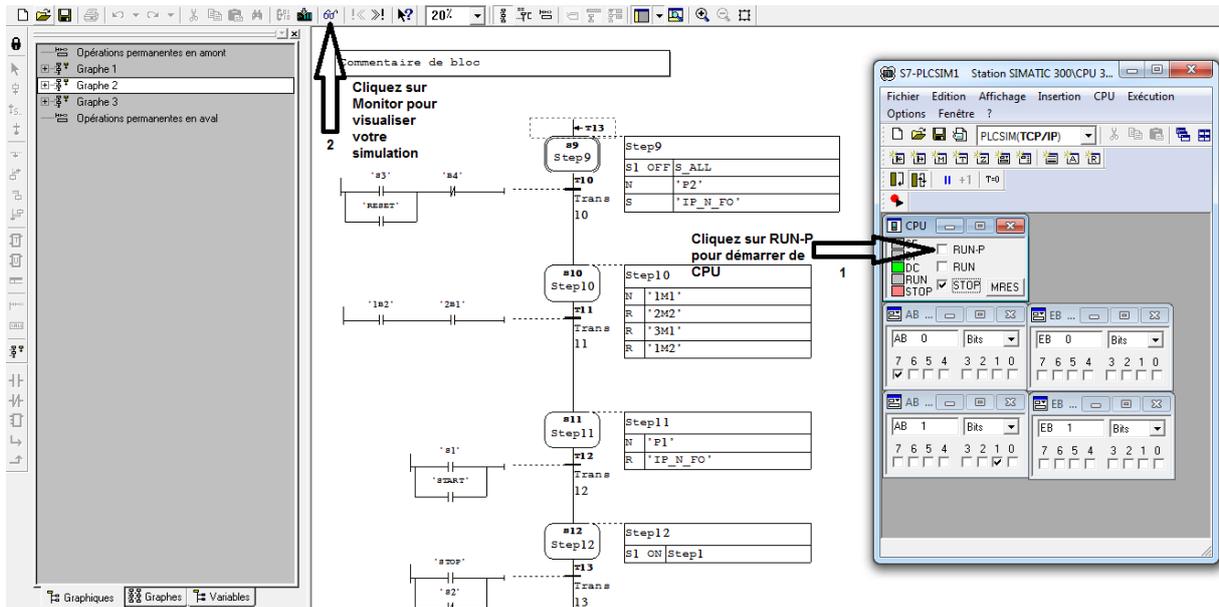


Figure IV-9 La visualisation.

d. Le lancement de notre programme avec RUN-P :

Selon notre cahier de charge, on suivra notre séquence de programme par la mise à 1 ou 0 des bits des entrées au niveau de PLCSIM et on teste les erreurs.

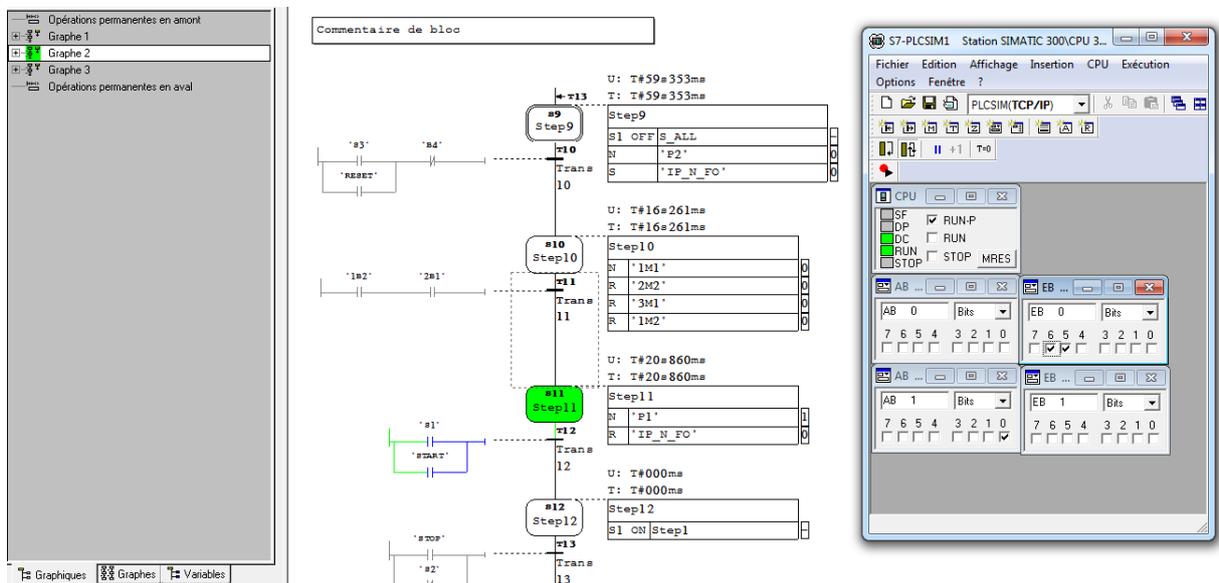


Figure IV-9 La simulation de notre programme.

II.2. Le résultat obtenu :

- les figures au-dessous représentent la simulation de notre programme (avec le langage graphique) de la station **MPS-testing** avec le simulateur **PLCSIM** qui est dans le côté droit des figures. Les petits rectangles arrondis représentent les étapes (le rectangle arrondi blanc représente l'étape non-active/le rectangle arrondi vert représente l'étape active dans la séquence de notre programme). Les grands rectangles affectés à ces étapes représentent les actions associées à cette étape.

- Nous allons présenter quelque étape de notre programme de la station MPS testing :

- 1- La première figure (**Figure IV-11**) nous dit que le vérin de levage déplace la pièce de poste1 vers le poste de contrôle de la hauteur avec l'actionneur **1M2** jusqu'à la détection de capteur électromagnétique **1B1** (La mise à 1 de bit **E0.4** d'entrée **1B1**, et la mise à 0 des bits E0.1 et E0.5 des entrées **B2** ET **1B2** avec **PLCSIM**).

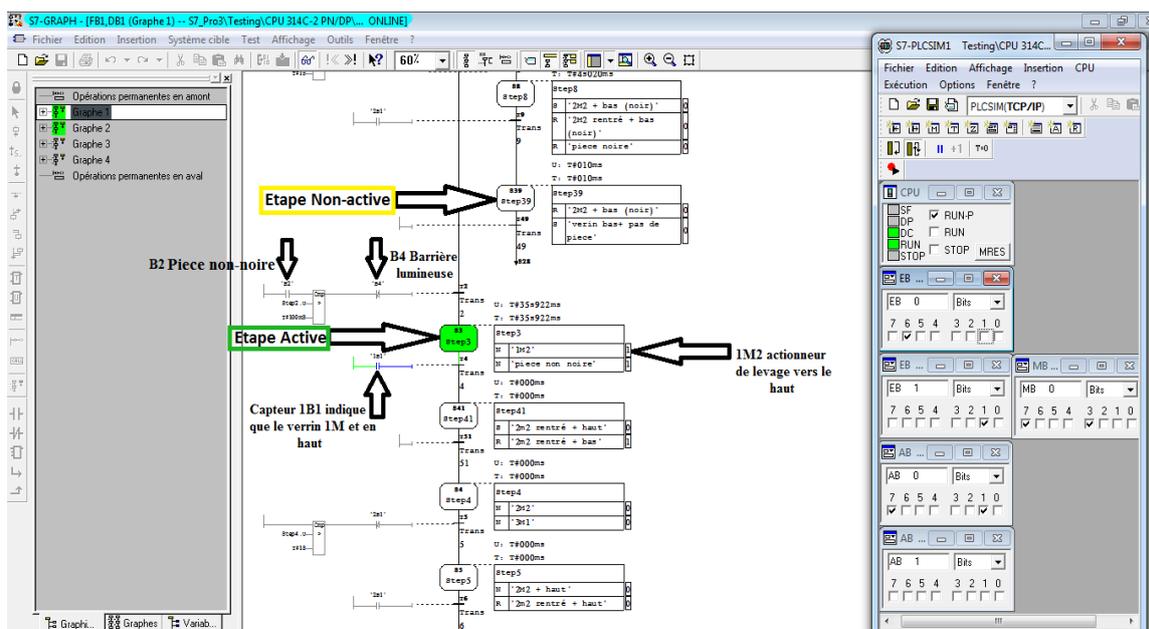


Figure IV-11 La pièce est au niveau de poste de contrôle de hauteur

- 2- La deuxième figure (**Figure IV-12**) nous dit que le vérin de levage est en haut (**1B1** est activé) et que le vérin d'éjection éjecte la pièce vers le poste 2 jusqu'à la mise à 0 de bit E0.6 d'entrée **2B1** (qui dit que le vérin déjection est rentré), et que le Glissière à coussin d'air est activée.

Chapitre IV: La programmation et la création de HMI

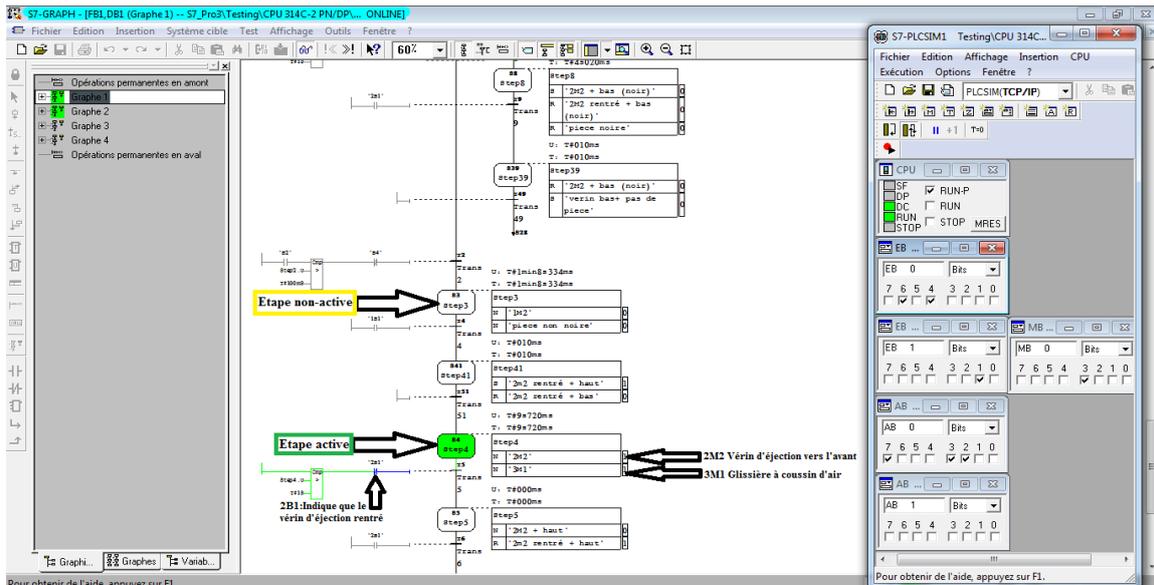


Figure IV-12 La pièce est éjecté vers le poste 2

- 3- La troisième figure (Figure IV-13) nous dit que le vérin de levage est en bas (1B2 est activé par la mise à 1 de son bit via PLCSIM) et que le vérin d'éjection éjecte la pièce vers le poste 3 jusqu'à la mise à 0 de bit E0.6 d'entrée 2B1 (qui dit que le vérin déjection est rentrée).

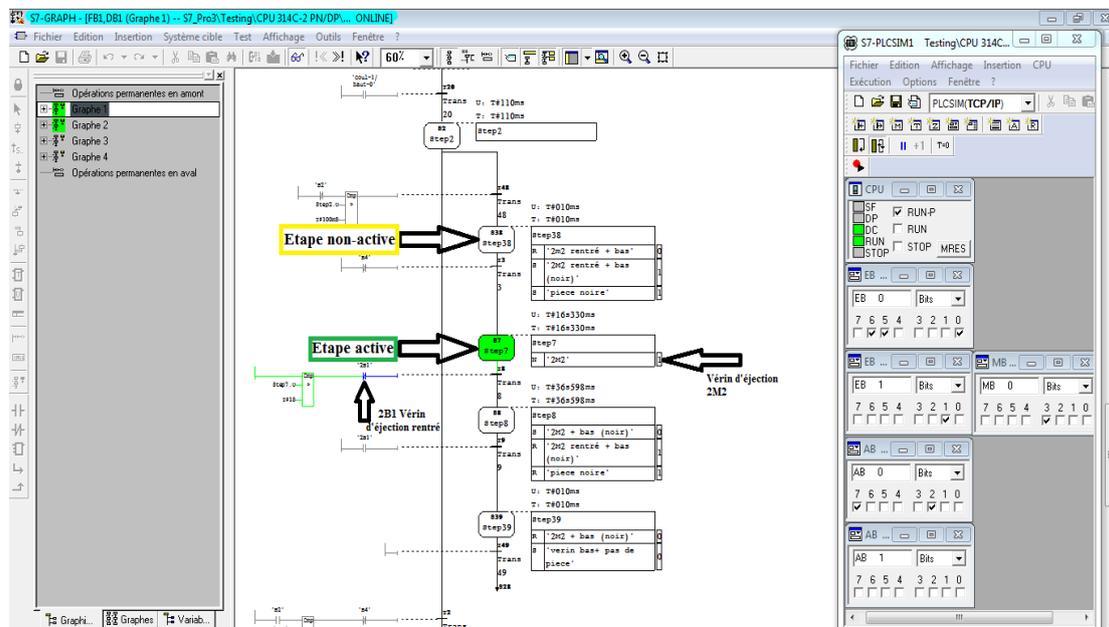


Figure IV-13 La pièce est éjecté vers le poste 3

III. La création de notre IHM :

III.1. Introduction :

WinCC flexible est le logiciel qui nous permet de réaliser notre IHM de supervision, par des moyens d'ingénierie simples et efficaces, de concepts d'automatisation évolutifs, au niveau machine. Vous avez accès à toutes les fonctionnalités prises en charge par le pupitre opérateur connecté.

Pour créer un projet vous devez suivre les étapes suivantes :

- Vous démarrez WINCC Flexible depuis l'interface utilisateur de Windows.
- Déterminer le type de projet.
- Déterminer le type de pupitre.
- L'intégration du projet dans Step7.
- Configuration de liaisons de communication.
- Définition de la table variable.
- La création de l'Interface Homme-Machine.
- Sauvegarder et lancer le système WINCC Runtime.

III.2. Les étapes à suivre :

- Etape 1 :

Vous démarrez WinCC flexible, soit par l'icône placée sur le bureau de votre ordinateur de configuration, soit par le menu Démarrer de Windows.



Figure IV-14 L'icône de WinCC Flexible.

- Etape 2 :

Lorsque vous démarrez WinCC flexible, un Assistant vous guide tout au long des étapes nécessaires pour créer un nouveau projet. L'Assistant vous demande p. ex. d'entrer un nom pour le projet et de sélectionner un pupitre opérateur. Si vous avez déjà ouvert WinCC flexible, choisissez la commande "Nouveau" pour créer un projet. Vous êtes ensuite également guidé par un Assistant. Pour charger un projet existant, choisissez la commande "Ouvrir" du menu "Projet".

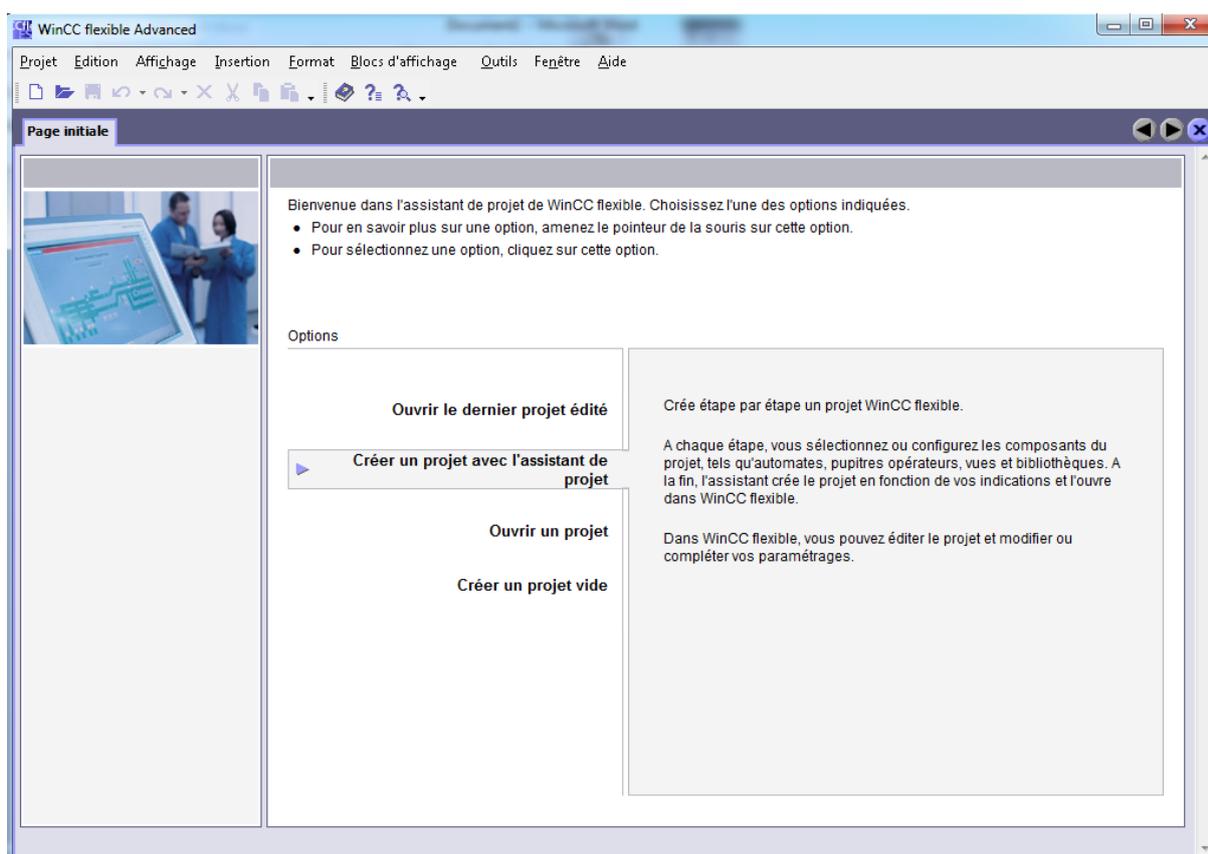


Figure IV-15 La création d'un projet.

- Etape 3 :

Pour commander notre station de Testing, il suffit d'un pupitre et d'un automate. Sélectionnez pour cela «Petite machine»:

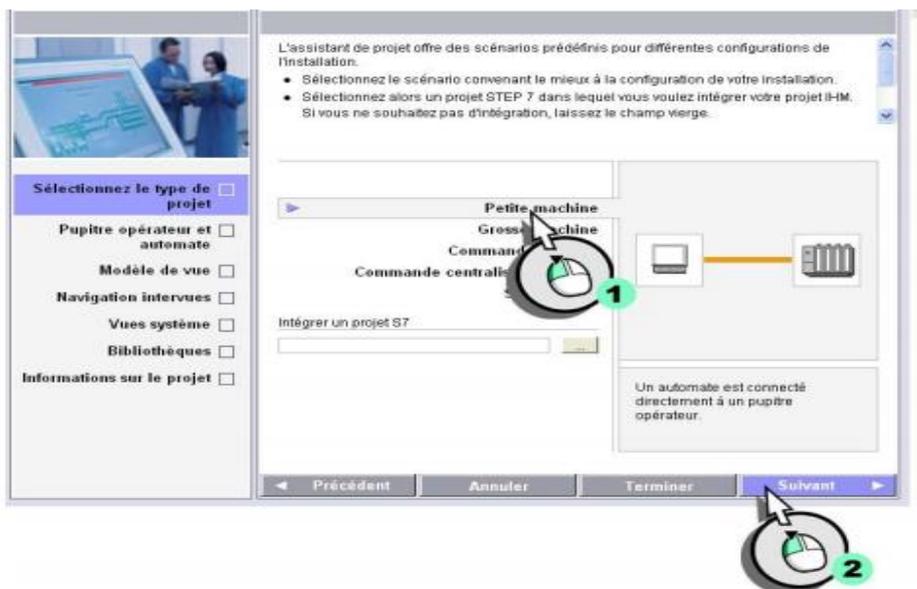


Figure IV-16 Sélectionner petite machine.

- Etape 4 : Le choix du pupitre.

Dans notre projet on a travaillé avec le type du pupitre PC :

Un système IHM est relié avec un PC via Ethernet.

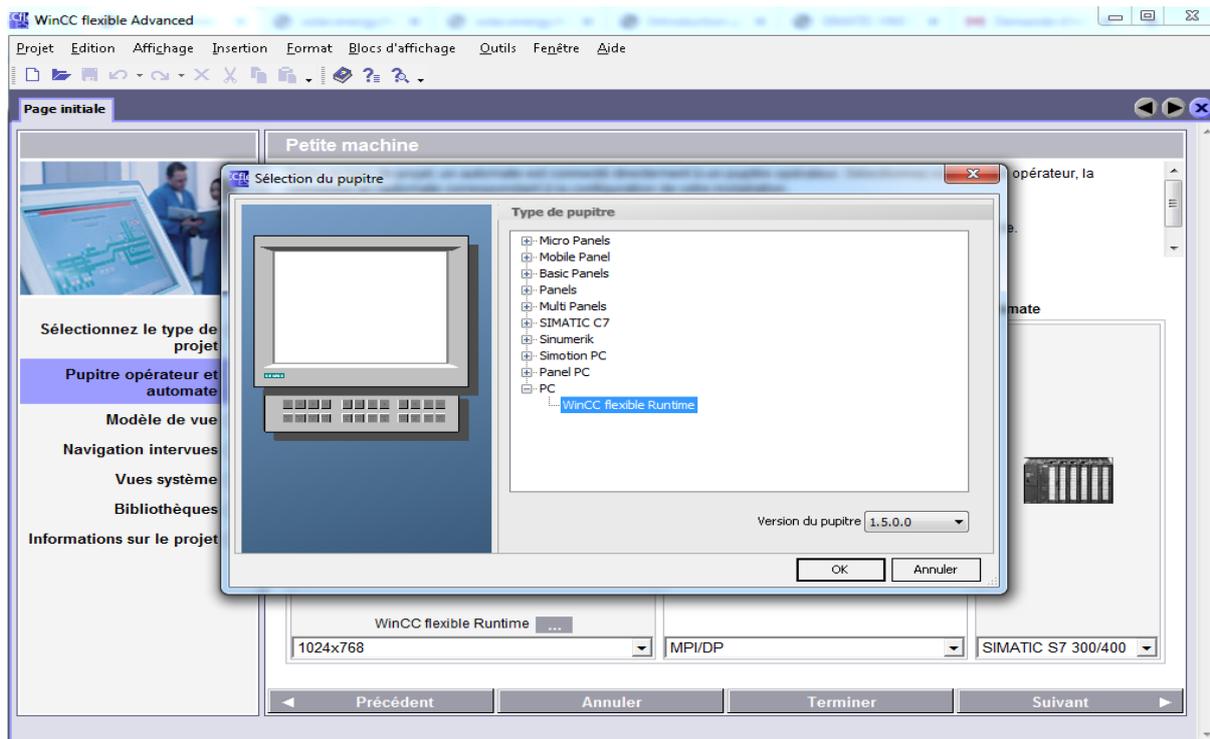


Figure IV-17 Sélection du pupitre.

- Etape 5 :

- Validez les paramètres par défaut sur la page «Modèle de vue» avec «Suivant».
- Validez les paramètres par défaut sur la page «Bibliothèques» avec «Suivant».
- Entrez ensuite les informations relatives au projet :

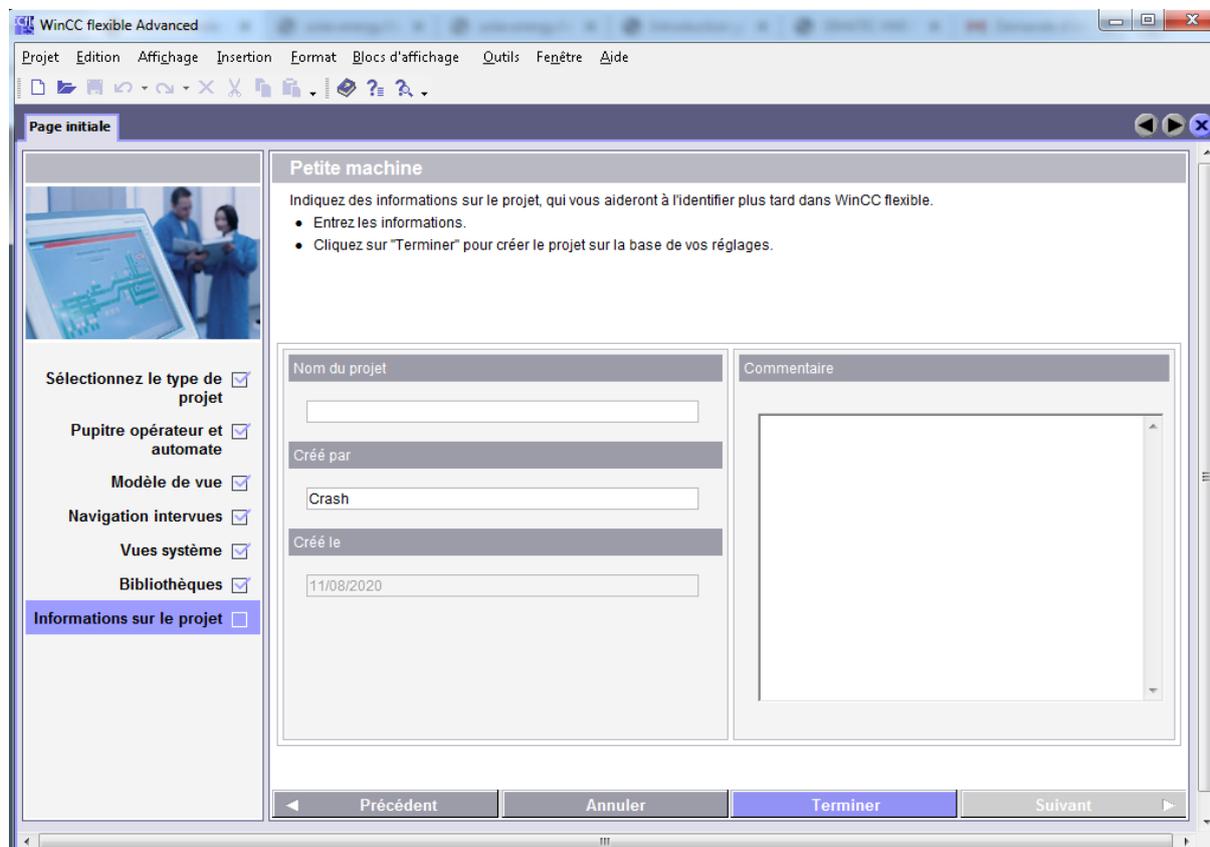


Figure IV-18 Informations sur le projet.

- Etape 6 :

- L'Intégration dans SIMATIC STEP 7 :

Les variables du processus représentent la liaison pour la communication entre le système d'automatisation et le système IHM. Sans les avantages de Totally Integrated Automation, vous devriez définir chaque variable à deux reprises : une fois pour le système d'automatisation et une fois pour le système IHM.

L'intégration de SIMATIC STEP 7 dans l'interface de configuration permet de diminuer la fréquence des erreurs et de réduire les tâches de configuration nécessaires. Durant la configuration, vous accédez directement à la table des mnémoniques de STEP 7 ainsi qu'aux paramètres de communication :

-La table des mnémoniques de STEP 7 contient la définition des points de données (p. ex. adresses ou types de données) que vous avez paramétrée lors de la création du programme de commande.

- Les paramètres de communication contiennent les adresses de bus ainsi que les protocoles de commande.

Pour intégrer un projet WinCC flexible dans STEP 7, procédez comme suit :

1-Ouvrez la configuration WinCC flexible.

2-Sélectionnez la commande de menu "Projet> Intégrer dans STEP 7 ...".

La boîte de dialogue "Intégrer dans les projets STEP 7" s'ouvre.

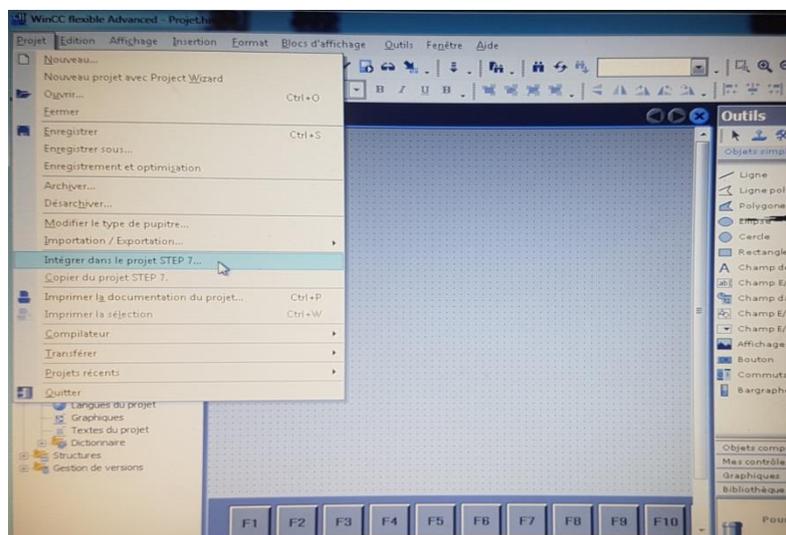


Figure IV-19 L'intégration dans le projet STEP7.

- **Etape 7 :**

Sélectionnez le projet STEP 7 correspondant dans la boîte de dialogue.

Si le projet souhaité ne figure pas dans la liste, naviguez via le champ de recherche jusqu'au dossier dans lequel se trouve le projet STEP 7.

L'intégration est exécutée une fois que vous avez sélectionné le projet STEP 7.

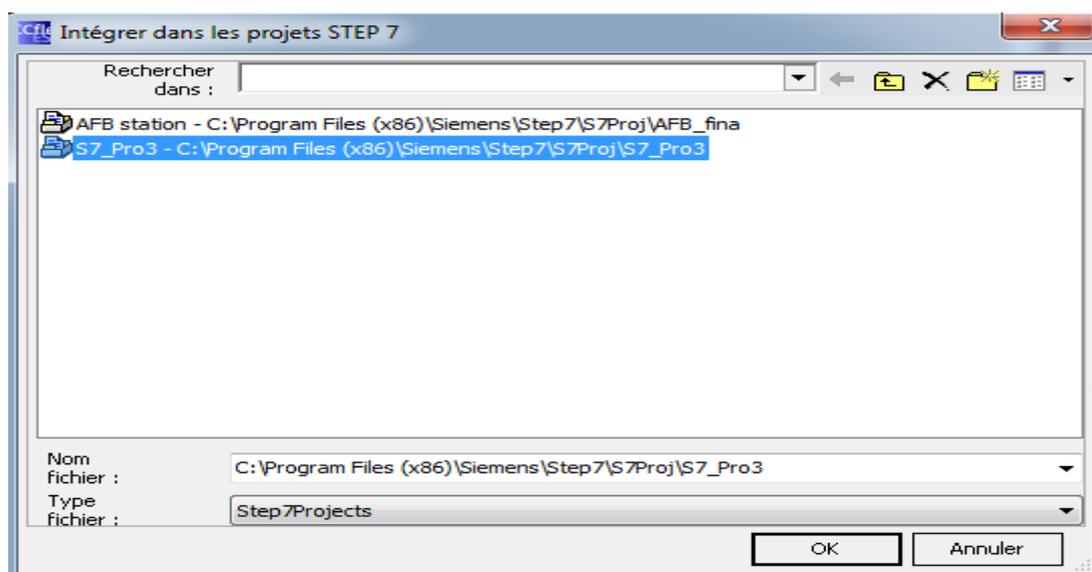


Figure IV-20 Le sélection du projet STEP 7.

- **Etape 8 :**
- **La configuration réseau :**

Nous pouvons configurer la communication entre notre IHM et la Sous station étudié selon plusieurs moyens qui sont les suivants :

- a-** Communications via MPI.
- b-** Communication via CP Ethernet Industriel.
- c-** Communication via PROFIBUS.

Les deux dernières communications et d'après les utilisateurs et la documentationsont très fiable pour une longue duré de travail par rapport au MPI.

Nous avons choisi pour notre cas communication Processing CP Ethernet Industriel 'CP1411'.

- 1- La figure (Figure0-21) est considérée comme une première étape pour configurer le réseau Ethernet.

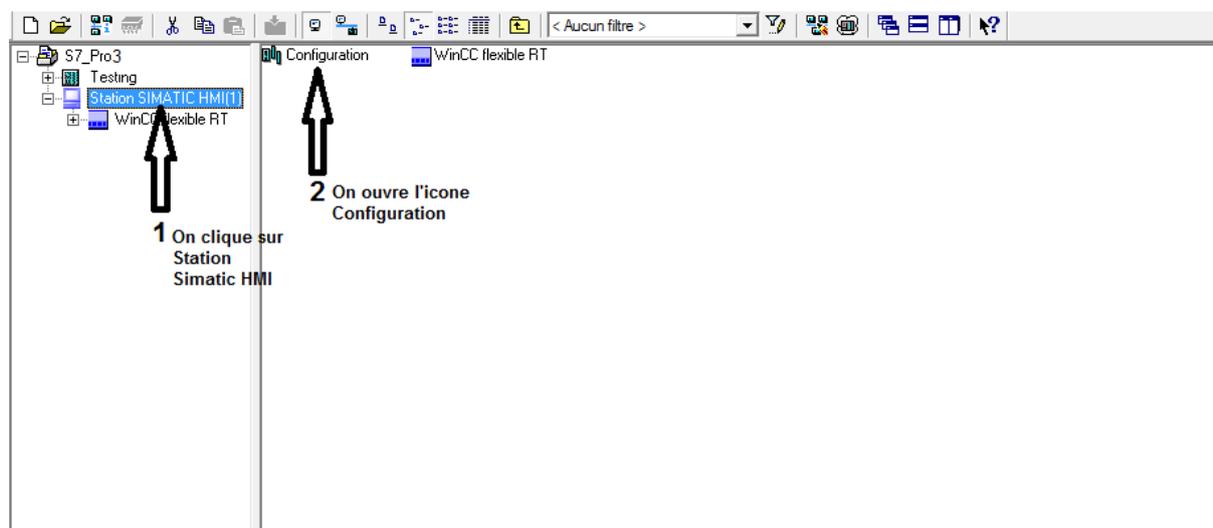


Figure IV-21 L'icône de configuration.

- 2- On ouvre le « HW Config », on sélectionne le « CP1411 » et on fait l'enregistrement et la compilation.

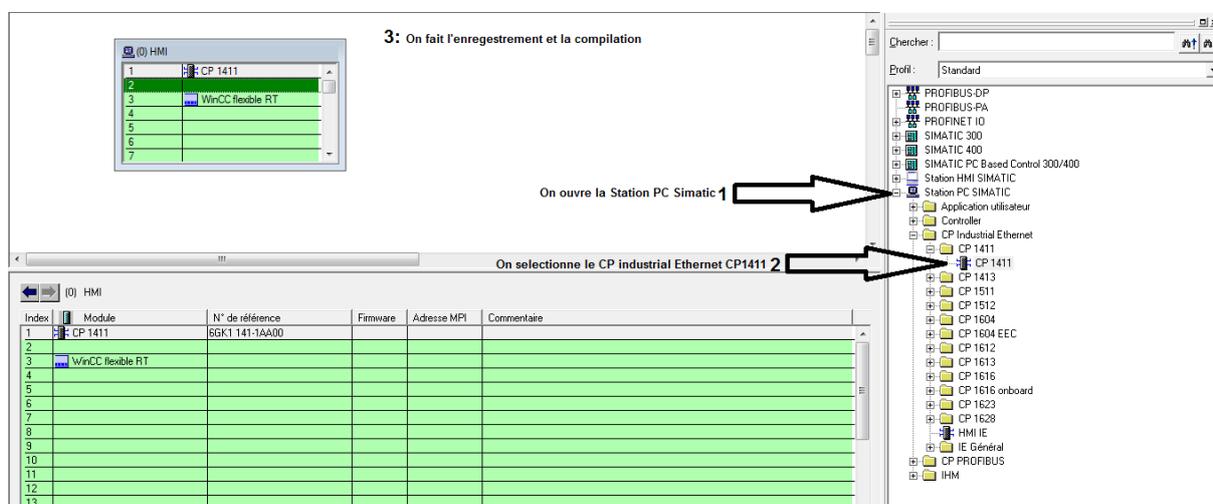


Figure IV-22 La configuration sur « HW Config ».

3- On fait la configuration finale sur« Netpro ».

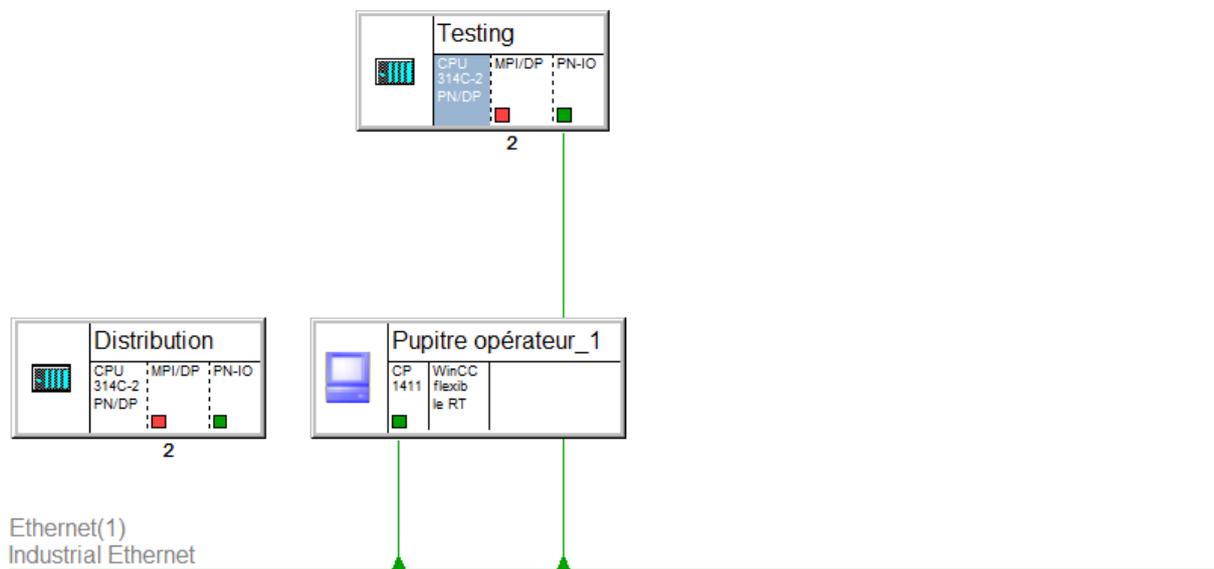


Figure IV-23 La configuration sur Netpro.

III.3. Les modifications dans notre programme pour la réalisation d'IHM :

Les modifications sont l'ajoute des entrées (sous formes mémoire Bit) et sorties afin d'établir la possibilité de la commande via IHM.

- Pour les boutons :

Les entrées que nous avons ajouté sert à nous donner la main de commander le système à partir du l'interface IHM réalisée, mais à condition que le bouton sera sous l'adressage mémoire bit M (Mémento).

- Pour l'affichage graphique :

Les sorties que nous avons ajouté dans notre programme ont le but de nous permet de configurer les afficheurs graphiques (donner à chaque afficheur graphique un variable) afin qu'ils nous donnent la main de voir des animations dans notre interface IHM.

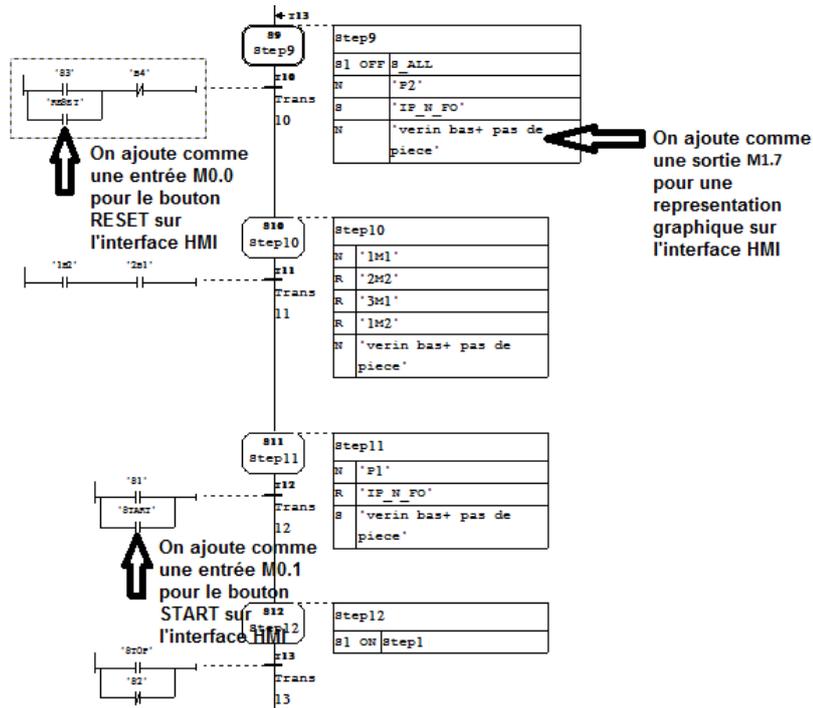


Figure IV-24 Les modifications dans notre programme.

- L'interface de travail sur WINCC Flexible :

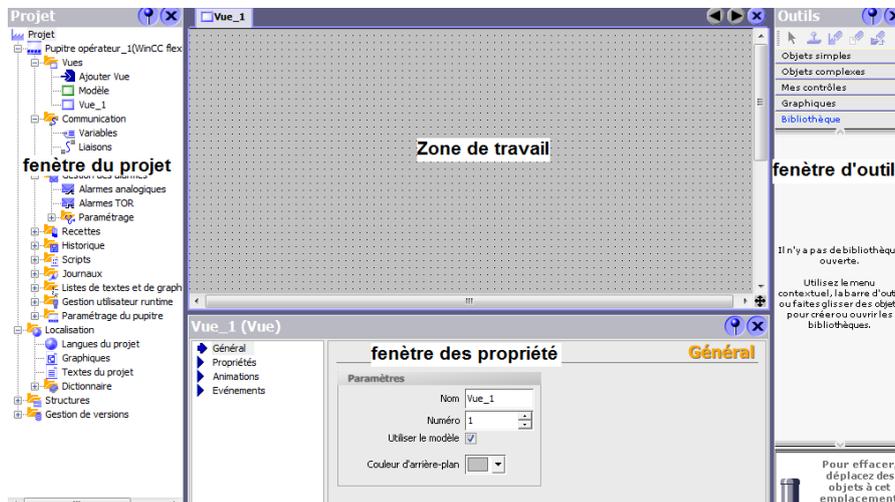


Figure IV-25 L'interface de travail sur WinCC.

- La zone de travail sert à éditer les objets du projet. Tous les éléments de WinCC flexible sont disposés autour de la zone de travail. A l'exception de la zone de travail, vous pouvez disposer et configurer, déplacer ou masquer p. ex. tous les éléments comme vous le souhaitez.

- Dans la fenêtre du projet tous les éléments et tous les éditeurs disponibles d'un projet sont affichés dans l'arborescence et peuvent y être ouverts. Dans la fenêtre de projet, vous pouvez de plus accéder aux propriétés du projet et au paramétrage du pupitre utilisateur.
- La fenêtre d'outils vous propose une sélection d'objets que vous pouvez insérer dans vos vues, p. ex. des objets graphiques et des éléments de commande. La fenêtre d'outils contient en outre des bibliothèques d'objets et collections de blocs d'affichage prêts à l'emploi.
- La fenêtre des propriétés permet de modifier les propriétés d'un objet sélectionné dans la zone de travail. Le contenu de la fenêtre des propriétés dépend de l'objet sélectionné.

Si vous voulez configurer par exemple une variable pour un objet graphique, cliquez sur la zone de liste. La zone de liste permet d'ouvrir la liste d'objets, qui propose toutes les variables disponibles dans le projet de type de données valide

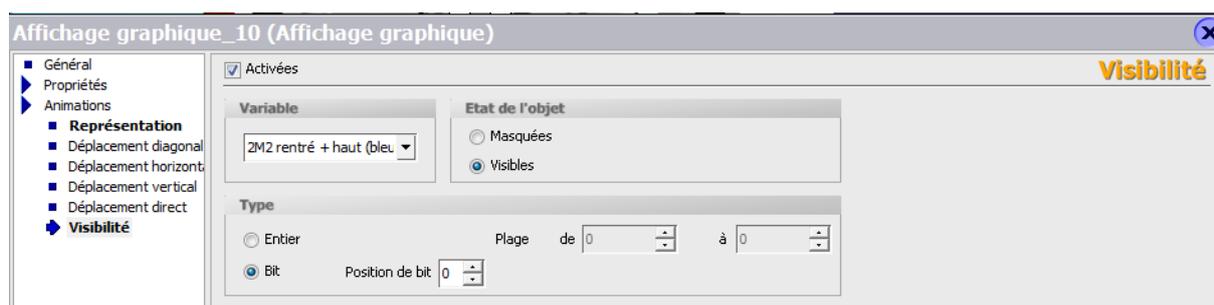


Figure IV-26 Fenêtre des propriétés.

- L'assistant projet a déjà créé certains objets :

a. Liaisons :

Les liaisons entre le pupitre et l'automate sont également déjà définies :

- Nous devons vérifier si la liaison est active.

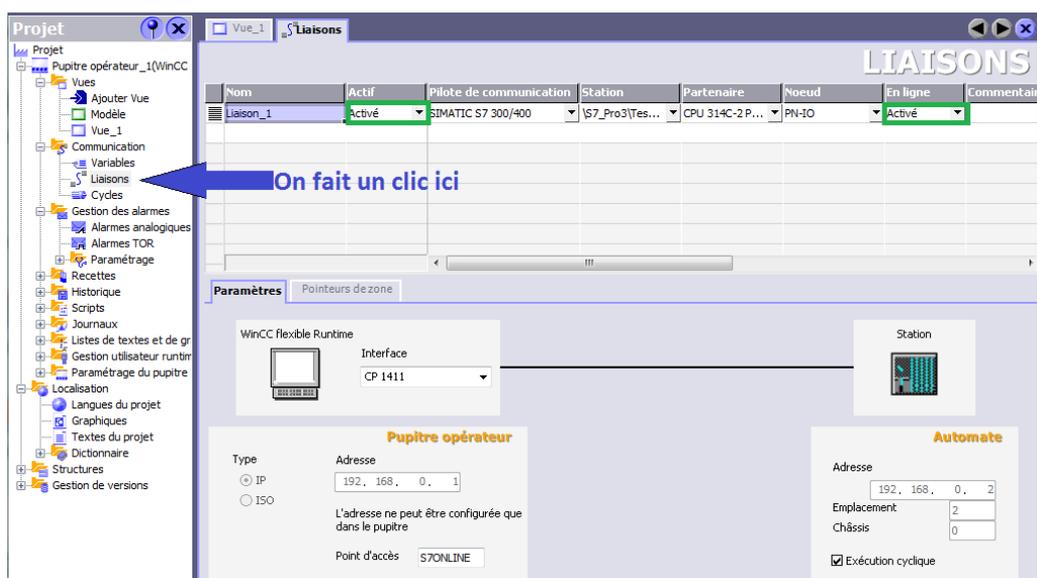


Figure IV-27 L'interface de liaison.

b. Zone de travail:

Dans la zone de travail, nous avons ajouté :

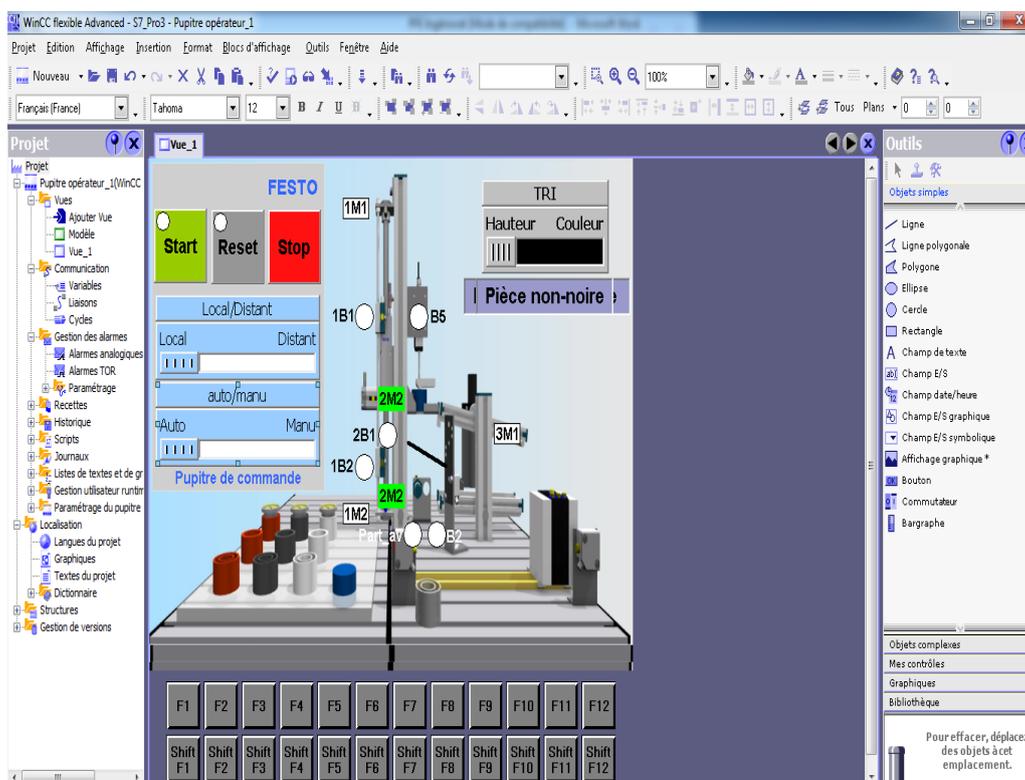


Figure IV-28 La vue de notre IHM.

- Bouton **START** avec le Mémento **M0.1** pour le lancement du système avec notre HMI.
- Bouton **STOP** avec le Mémento **M0.2** pour l'arrêt total du système avec notre HMI.
- Bouton **RESET** avec le Mémento **M0.0** pour le redémarrage du système avec notre HMI.
- Commutateur **TRI** avec le Mémento **M0.3** pour le choix de tri des pièces avec notre IHM.
- Commutateur **Auto/Manu** avec le Mémento **M0.4** pour le choix de mode automatique ou manuel.
- Pour le bon fonctionnement de notre IHM concernant la commande de mode Auto/Manu, On a ajouté un commutateur qui nous permet de choisir le mode de commande (à distance pour la commande par IHM et localement pour la commande au niveau de PLCsim).

Si on met le mode distant sur l'interface IHM le commutateur Auto/Manu s'affiche sinon elle ne s'affiche pas.



Mode Distant



Mode Local

c. L'éditeur Variables:

La zone de travail affiche toutes les variables sous forme de tableau. Vous éditez les attributs des variables dans les cellules du tableau.

Tous les attributs des variables et des éléments du tableau peuvent être définies dans les tableaux et dans les fenêtres des propriétés correspondantes.

Nom	Nom d'affichage	Connexion	Type de données	Mnémonique	Adresse	Éléments du...	Cycle d'acqui...
Vérin haut + pas pieces		Liaison_1	Bool	Vérin haut + pas pieces	Q 2.7	1	100 ms
verin bas+ pas de piece		Liaison_1	Bool	verin bas+ pas de piece	Q 3.1	1	100 ms
STOP		Liaison_1	Bool	STOP	M 0.2	1	100 ms
START		Liaison_1	Bool	START	M 0.1	1	100 ms
S_Auto		Liaison_1	Bool	S_Auto	I 1.2	1	100 ms
RESET		Liaison_1	Bool	RESET		1	100 ms
piece non noire		Liaison_1	Bool	piece non noire	Q 3.5	1	100 ms
piece non correcte		Liaison_1	Bool	piece non correcte	Q 3.6	1	100 ms
piece noire		Liaison_1	Bool	piece noire	Q 3.4	1	100 ms
piece correcte		Liaison_1	Bool	piece correcte	Q 3.0	1	100 ms
Part_AV		Liaison_1	Bool	Part_AV	I 0.0	1	100 ms
P3		Liaison_1	Bool	P3	Q 1.2	1	100 ms
P2		Liaison_1	Bool	P2	Q 1.1	1	100 ms
P1		Liaison_1	Bool	P1	Q 1.0	1	100 ms
coul=1/haut=0		Liaison_1	Bool	coul=1/haut=0	M 0.3	1	100 ms
B5		Liaison_1	Bool	B5	I 0.3	1	100 ms
B4		Liaison_1	Bool	B4	I 0.2	1	100 ms
B2		Liaison_1	Bool	B2	I 0.1	1	100 ms
auto/manu		Liaison_1	Bool	auto/manu	M 0.4	1	100 ms
3M1		Liaison_1	Bool	3M1	Q 0.3	1	100 ms
2M2 rentré + haut (bleu)		Liaison_1	Bool	2M2 rentré + haut (bleu)	Q 2.2	1	100 ms
2m2 rentré + haut		Liaison_1	Bool	2m2 rentré + haut	Q 3.3	1	100 ms
2M2 rentré + bas (noir)		Liaison_1	Bool	2M2 rentré + bas (noir)	Q 2.4	1	100 ms
2M2 rentré + bas (bleu)		Liaison_1	Bool	2M2 rentré + bas (bleu)	Q 2.6	1	100 ms
2m2 rentré + bas		Liaison_1	Bool	2m2 rentré + bas	Q 1.7	1	100 ms
2M2 + haut		Liaison_1	Bool	2M2 + haut	Q 1.4	1	100 ms

Figure IV-29 La table de variable.

IV. Le résultat obtenu :

Dans cette partie on va présenter la simulation de notre projet via PLCSIM et Runtime avec quelques figures des IHM réalisés on observant les différents changements par la mise à 1 ou 0 des différents bits des entrées au niveau de simulateur PLCSIM et d'autre part l'appuis au différents boutons (Start, Reset, Stop) et au commutateurs pour le choix de tri (par couleur/par hauteur) et aussi pour le mode auto/manu dans l'interface IHM.

Dans notre interface on a utilisé des cercles pour représenter les différents capteurs et des rectangles pour les Actionneurs qu'on a dans notre station.

-Après l'activation du simulateur PLCSIM (PLCSIM en mode RUN), nous remarquons que tous les capteurs et actionneurs sont alimentés et leurs états de bit est à 0 mise à part le voyant RESET qui est aussi alimenté est au niveau de bit 1et qui nous informe elle-même que le Bouton Reset prêt à l'appuyer afin d'initialiser la station.

-Après l'appuis au bouton RESET ou la mise à 1 de bit d'entrée S3 au niveau du PLCSIM, nous avons obtenu le changement de bit au niveau de l'IHM et de visualisation S7-Graph pour l'actionneur 1M1 qui mise à 1 et pour les actionneurs 2M2, 3M1 et 1M2 qui misent à 0 après on fait la mise à 1 des bits responsables sur l'évolution du grafcet de cette sous station

voions les capteurs 1B2, 2B1 au niveau du PLCsim pour obtenir l'état initial et la mise à 1 de voyant START qui nous informe que le Bouton START est prêt pour démarrer notre station. Avant de lancer notre station en marche, nous choisissons le tri des pièces soit par hauteur ou par couleur dans l'interface HMI :

IV.1. Le cas de tri pas hauteur:

- Etape 1 :

Après la sélection du cas hauteur par le commutateur TRI au niveau de HMI ou la mise à 0 de bit d'entrée M0.3 au niveau du PLCSIM, nous obtenons La figure (**Figure IV-30**) au-dessous qui représente une IHM de la station testing en mode simulateur Runtime de WinCC Flexible et mode Simulateur PLCSIM de Step7 à l'état initial.

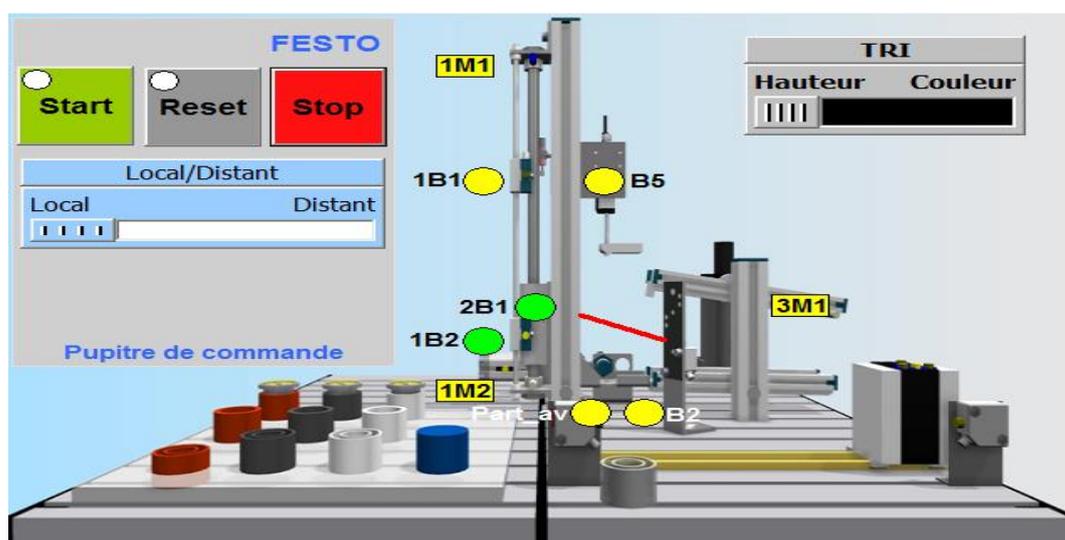


Figure IV-30 Le tri par hauteur en état initial.

- Etape 2 :

Après la mise en marche de notre système avec le bouton Start au niveau de Runtime ou avec la mise à 1 de bit d'entrée S1 au niveau du PLCSIM, nous obtenons la figure (**Figure IV-31**) au-dessous qui représente une IHM de la sous-station testing en mode simulateur Runtime et PLCSIM, nous faisons aussi la mise à 1 de bit de capteur Part_av au niveau du PLCSIM. Nous remarquons la présence de la pièce et aussi le changement de bit au niveau de l'IHM pour le capteur Part_av avec un changement de couleur du jaune à la couleur vert et la mise à 1 de l'actionneur 1M2 (pour le transfert de la pièce vers le poste de contrôle de la hauteur) avec le changement de couleur du jaune à vert.

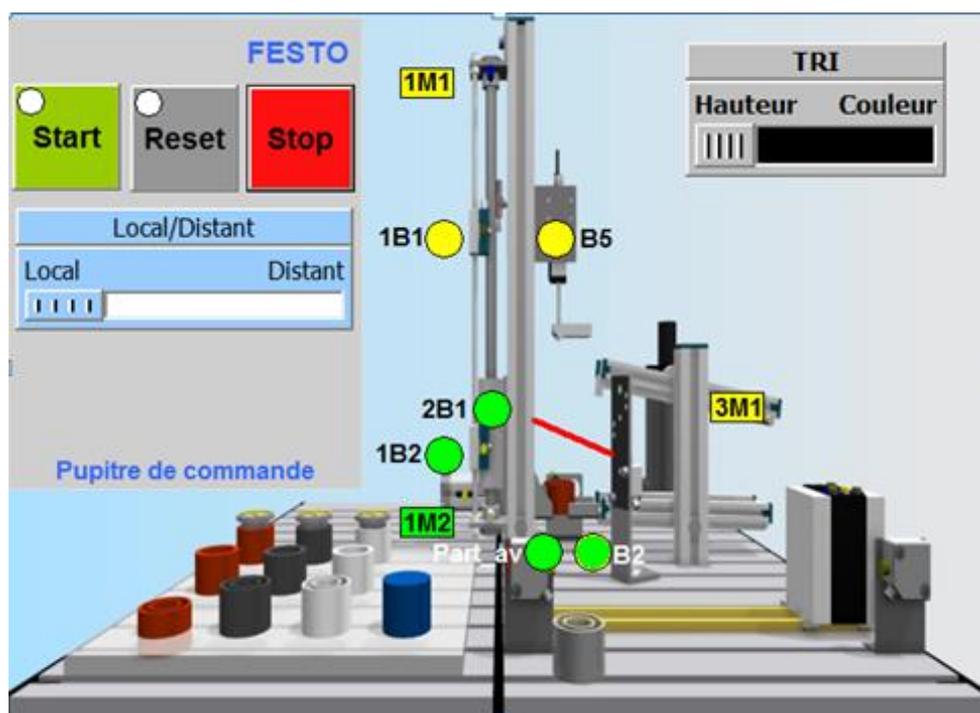


Figure IV-31 La présence de la pièce rouge.

- Etape 3 :

Après la mise à 1 de l'actionneur 1M2, nous obtenons la figure (**Figure IV-32**) au-dessous qui représente une IHM de la sous-station testing en mode simulateur Runtime et PLCSIM, nous faisons aussi la mise à 1 de bit de capteur 1B1 et la mise à 0 des bits des capteurs 1B2 et Part_av au niveau du PLCSIM.

Nous remarquons la présence de la pièce bleu au niveau de poste de contrôle de la hauteur à cause de capteur B5 qui indique que la pièce est non-correcte (le cercle qui représente le capteur B5 est en jaune au niveau de l'IHM) et aussi le changement de bit au niveau de l'IHM pour :

Le capteur 1B1 qui indique que le vérin est en haut avec un changement de couleur du jaune à la couleur vert et aussi pour les capteurs 1B2 et Part_av avec des changements de couleur du vert à la couleur jaune.

Le changement de champ de texte (qui nous indique que la pièce est non-correcte) d'invisible à visible.

La mise à 0 de l'actionneur 1M2 (pour le transfert de la pièce vers le poste de contrôle de la hauteur) avec le changement de couleur du vert à jaune et la mise à 1 de l'actionneur 1M1 (pour le transfert de la pièce vers le bas) avec le changement de couleur du jaune à vert.

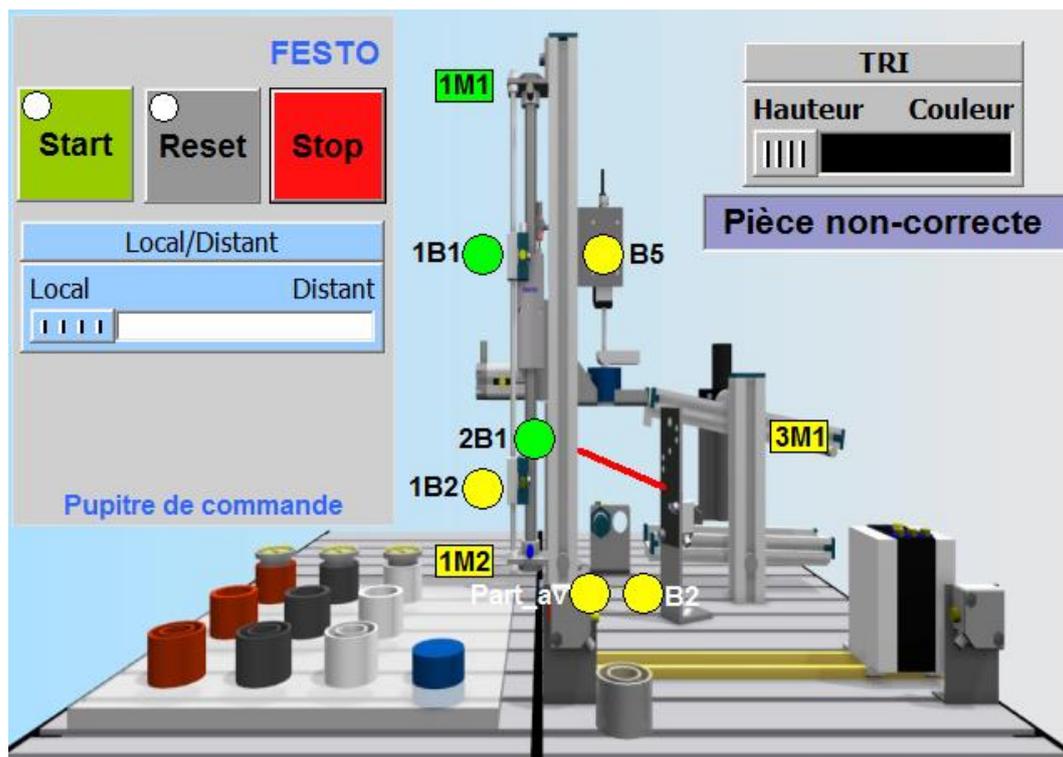


Figure IV-32 La présence de la pièce bleue.

- Etape 4 :

Après la mise à 1 de l'actionneur 1M1, nous obtenons la figure (**Figure IV-33**) au-dessous qui représente une IHM de la sous-station testing en mode simulateur Runtime et PLCSIM, nous faisons aussi la mise à 1 des bits des capteurs 1B2 et Part_av et la mise à 0 de bit de capteur 1B1 au niveau du PLCSIM.

Nous remarquons la présence de la pièce bleue en bas et aussi le changement de bit au niveau de l'IHM pour :

Le capteur 1B2 qui indique que le vérin est en bas avec un changement de couleur du jaune à la couleur vert et aussi pour les capteurs 1B1 et Part_av avec des changements de couleur du vert à la couleur jaune.

Le changement de champ de texte (qui nous indique que la pièce est non-correcte) de visible à invisible.

La mise à 0 de l'actionneur 1M1 (pour le transfert de la pièce vers le bas) avec le changement de couleur du vert à jaune.

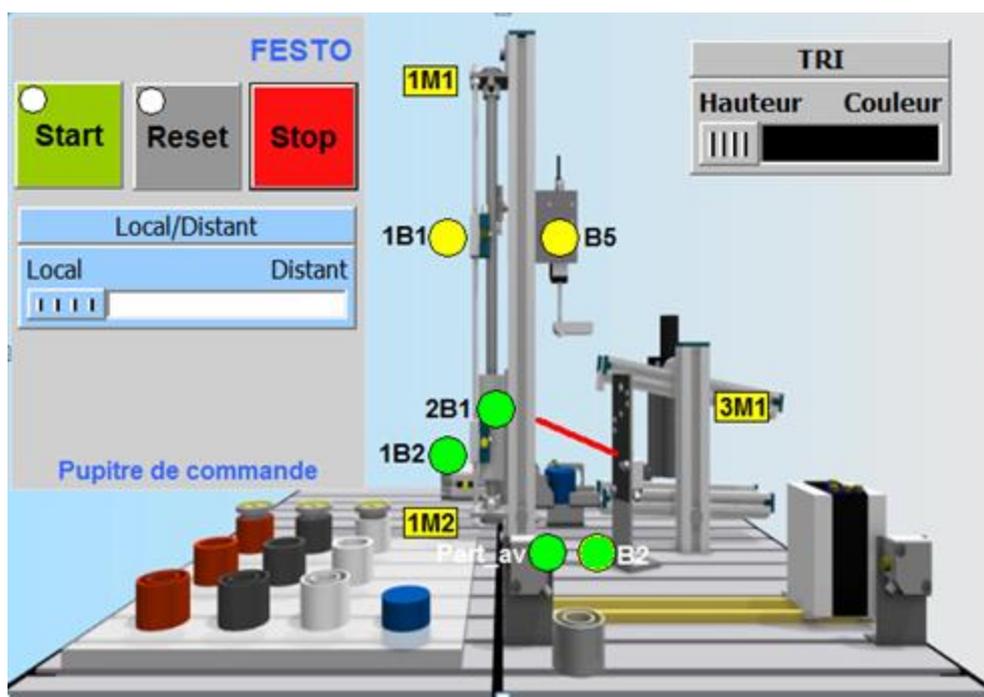


Figure IV-33 La présence de la pièce bleue en bas.

- Etape 5 :

Après la mise à 1 de capteur 1B2, nous obtenons la figure (Figure IV-34) au-dessous qui représente une IHM de la sous-station testing en mode simulateur Runtime et PLCSIM, nous faisons aussi la mise à 0 des bits des capteurs Part_av et 2B1 au niveau du PLCSIM.

Nous remarquons l'éjection de la pièce bleue vers le poste 2 et aussi le changement de bit au niveau de l'IHM pour :

Le capteur Part_av avec le changement de couleur du vert à la couleur jaune et pour le capteur 2B1 qui indique que le vérin d'éjection n'est pas en état rentrée avec le changement de couleur du vert à la couleur blanche.

La mise à 1 de l'actionneur 2M2 (pour l'éjection de la pièce vers le poste 2) avec le changement de couleur en vert.

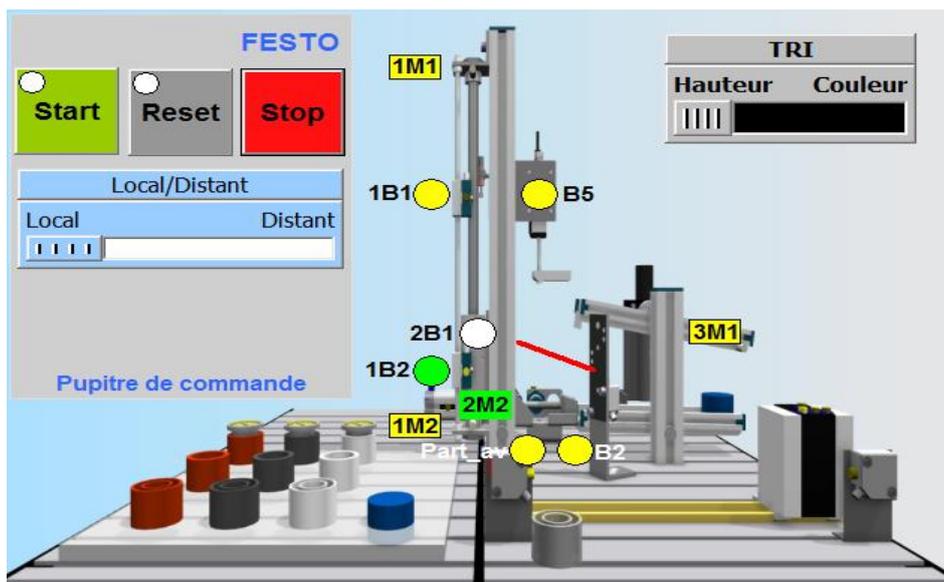


Figure IV-34 L'éjection de la pièce bleue vers le poste 2.

- Etape 6 :

Après l'éjection de la pièce bleue vers le poste 2 et la mise à 0 de l'actionneur 2M2 et la mise à 1 de bit de capteur 2B1 (qui indique que le vérin 2M2 est en état rentrée) au niveau de PLCSIM en retourne à **Etape 1** jusqu'à l'arrivée de la pièce qui vas se transfert vers le poste de contrôle de la hauteur par l'actionneur 1M2 comme il est indiqué dans l'**Etape 2**.

- Etape 7 :

Après la mise à 1 de l'actionneur 1M2, nous obtenons la figure (**Figure IV-35**) au-dessous qui représente une IHM de la sous-station testing en mode simulateur Runtime et PLCSIM, nous faisons aussi la mise à 1 des bits de capteurs 1B1 et B5 et la mise à 0 des bits des capteurs 1B2 et Part_av au niveau du PLCSIM.

Nous remarquons la présence de la pièce rouge au niveau de poste de contrôle de la hauteur et aussi le changement de bit au niveau de l'IHM pour :

Le capteur 1B1 qui indique que le vérin est en haut avec un changement de couleur du jaune à la couleur vert et le capteur B5 qui indique que la pièce est correcte avec un changement de couleur du jaune à la couleur vert aussi et pour les capteurs 1B2 et Part_av avec des changements de couleur du vert à la couleur jaune.

Le changement de champ de texte (qui nous indique que la pièce est correcte) d'invisible à visible.

La mise à 0 de l'actionneur 1M2 (pour le transfert de la pièce vers le poste de contrôle de la hauteur) avec le changement de couleur du vert à jaune et la mise à 1 de l'actionneur 3M1 (qui active le coussin d'air) avec le changement de couleur du jaune à vert.

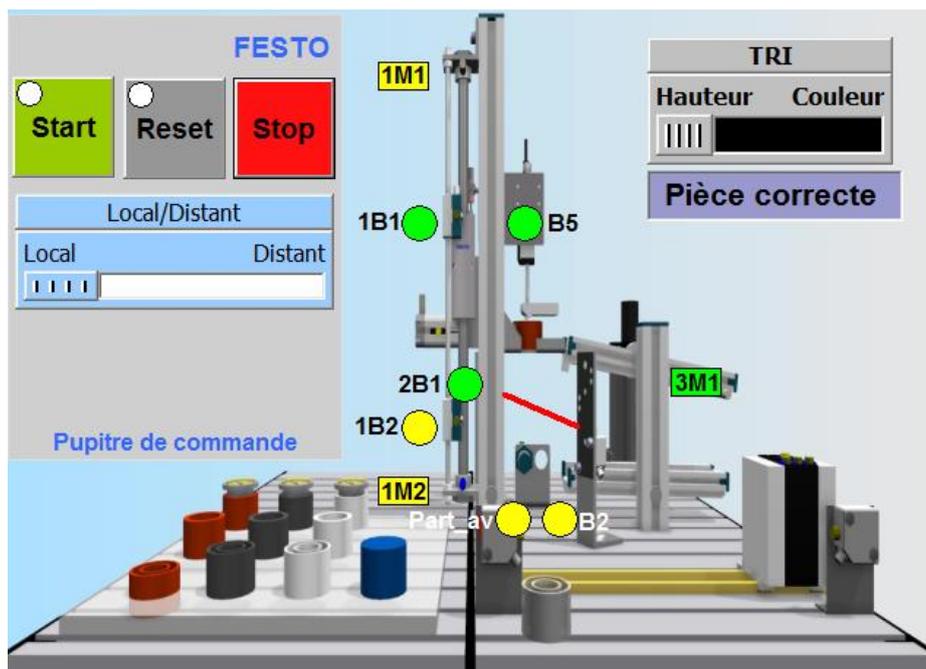


Figure IV-35 La présence de la pièce rouge.

- Etape 8 :

Après la mise à 1 de capteur 1B1, Nous obtenons la figure (**Figure IV-36**) au-dessous qui représente une IHM de la sous-station testing en mode simulateur Runtime et PLCSIM, nous faisons aussi la mise à 0 des bits des capteurs B5 et 2B1 au niveau du PLCSIM.

Nous remarquons l'éjection de la pièce rouge vers le poste 3 et aussi le changement de bit au niveau de l'IHM pour :

Le capteur B5 qui indique que la pièce est correcte avec le changement de couleur du vert à la couleur jaune et pour le capteur 2B1 qui indique que le vérin d'éjection n'est pas en état rentrée avec le changement de couleur du vert à la couleur blanche.

Le changement de champ de texte (qui nous indique que la pièce est correcte) de visible à invisible.

La mise à 1 de l'actionneur 2M2 (pour l'éjection de la pièce vers le poste 2) avec le changement de couleur en vert.

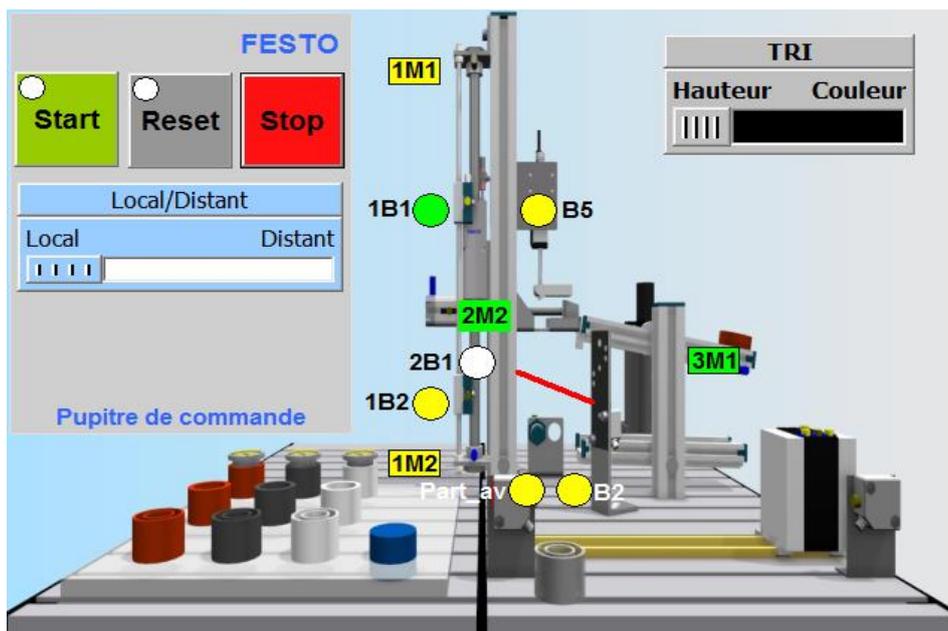


Figure IV-36 L'éjection de la pièce rouge vers le poste 3.

- Etape 9 :

Après la mise à 1 de l'actionneur 2M2, nous obtenons la figure (Figure IV-37) au-dessous qui représente une IHM de la sous-station testing en mode simulateur Runtime et PLCSIM, nous faisons aussi la mise à 1 de bit de capteur 2B1 au niveau du PLCSIM.

Nous remarquons qu'il y a pas de pièce au niveau de poste de contrôle de la hauteur et aussi le changement de bit au niveau de l'IHM pour :

Le capteur 2B1 qui indique que le vérin d'éjection est rentré avec un changement de couleur blanche à la couleur vert.

La mise à 0 de l'actionneur 2M2 (pour l'éjection de la pièce vers le poste 3) et de l'actionneur 3M1 (qui active le coussin d'air) avec des changements de couleur du vert à jaune et la mise à 1 de l'actionneur 1M1 (pour le transfert de vérin vers le bas) pour revenir à l'état initial avec le changement de couleur du jaune à vert.

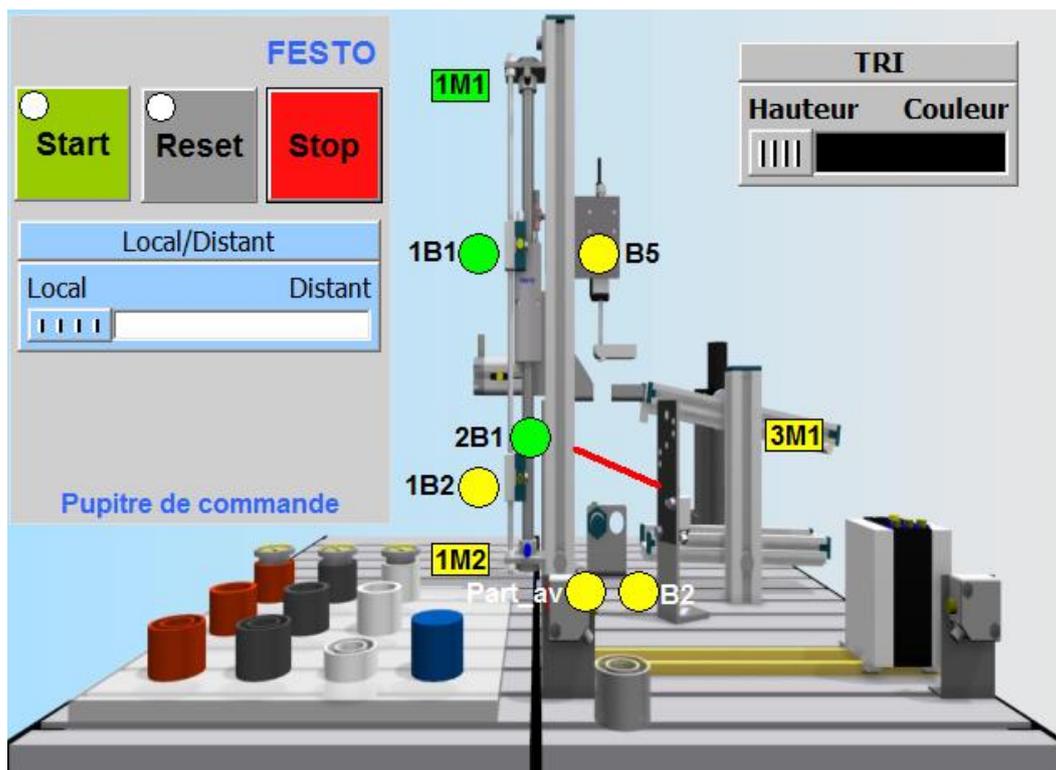


Figure IV-37 Le vérin en haut + Pas de pièce.

IV.2. Le tri par couleur :

- Etape 1 :

Après la sélection du cas couleur par le commutateur TRI au niveau d'IHM ou la mise à 1 de bit d'entrée M0.3 au niveau du PLCSIM, nous obtenons La figure (Figure IV-38) au-dessous qui représente une IHM de la station testing en mode simulateur Runtime de WinCC Flexible et mode Simulateur PLCSIM de Step7 à l'état initial.

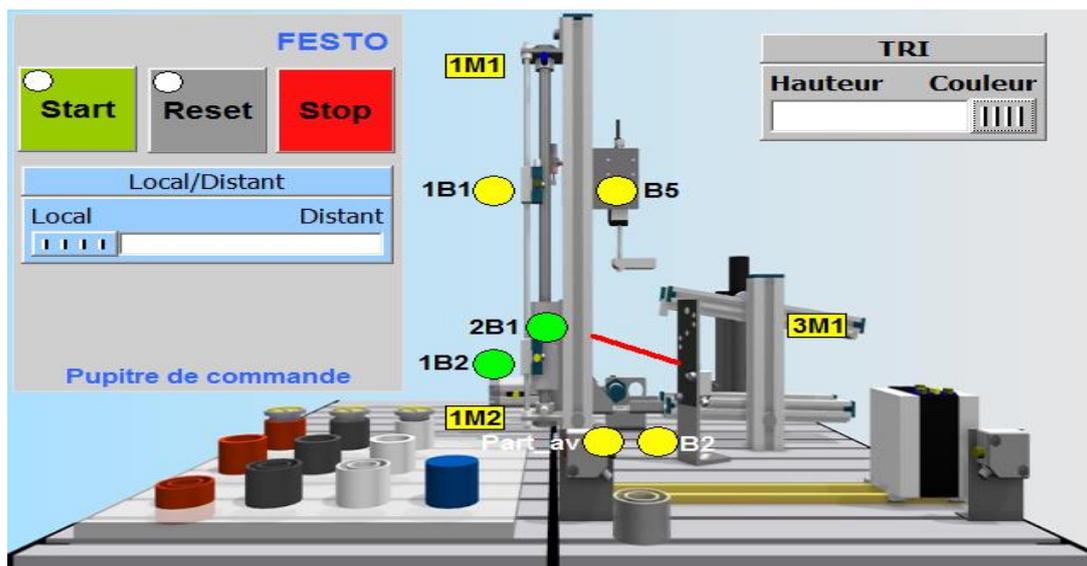


Figure IV-38 Le tri par couleur en état initial.

- Etape 2 :

Après la mise en marche de notre système avec le bouton Start au niveau de Runtime ou avec l'incrémentement de bits d'entrée S1 au niveau du PLCSIM, nous obtenons la figure (Figure IV-39) au-dessous qui représente une IHM de la sous-station testing en mode simulateur Runtime et PLCSIM, nous faisons aussi la mise à 1 de bit responsable sur l'évolution du grafctet de cette sous station voyons le capteur Part_av au niveau du PLCSIM. Nous remarquons la présence de la pièce noire et aussi le changement de bit au niveau de l'IHM pour

Le capteur Part_av avec un changement de couleur du jaune à la couleur vert.

Le capteur B2 est en couleur jaune (il n'y a pas de pièce non-noire).

Le changement de champ de texte (qui nous indique que la pièce est noire) d'invisible à visible.

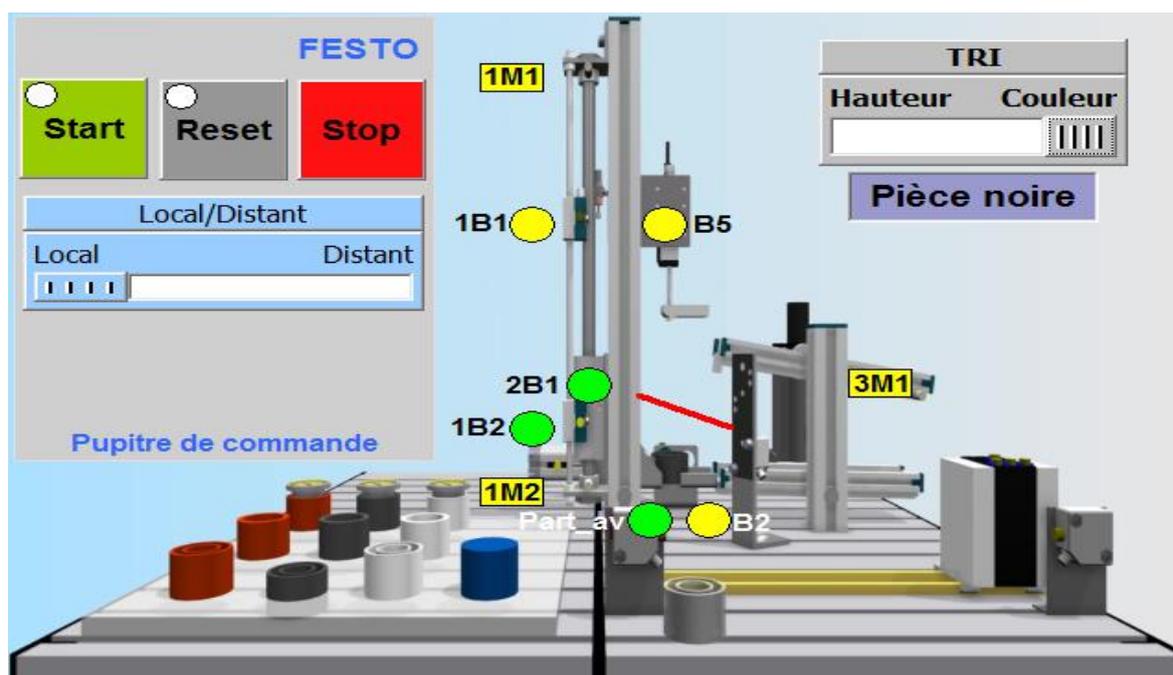


Figure IV-39 La présence de la pièce noire.

- Etape 3 :

Après l'arrivée de la pièce (la mise à 1 de bit de capteur Part_av) et la détection de sa couleur noire par le capteur B2, nous obtenons la figure (Figure IV-40) au-dessous qui représente une IHM de la sous-station testing en mode simulateur Runtime et PLCSIM, nous faisons la mise à 1 de bit de capteur 2B1 qui indique que le vérin d'éjection n'est pas en état rentrée et la mise à 0 de bit de capteur Part_av au niveau du PLCSIM.

Nous remarquons l'éjection de la pièce noire vers le poste 2 et aussi le changement de bit au niveau de l'IHM pour :

Le capteur Part_av avec le changement de couleur du vert à la couleur jaune et pour le capteur 2B1 qui indique que le vérin d'éjection n'est pas en état rentrée avec le changement de couleur du vert à la couleur blanche.

La mise à 1 de l'actionneur 2M2 (pour l'éjection de la pièce vers le poste 2) avec le changement de couleur en vert.

Le changement de champ de texte (qui nous indique que la pièce est noire) de visible à invisible.

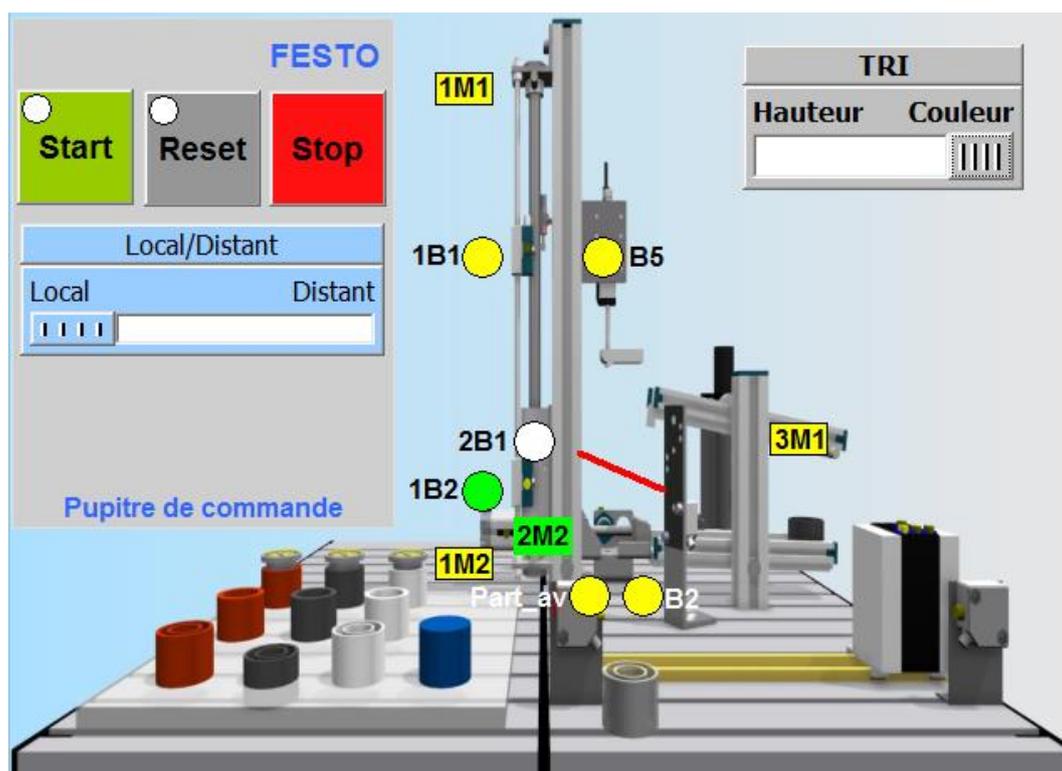


Figure IV-40 L'éjection de la pièce noire vers le poste 2.

- Etape 4 :

Après l'éjection de la pièce noire vers le poste 2 et la mise à 0 de l'actionneur 2M2 et la mise à 1 de bit de capteur 2B1 (qui indique que le vérin 2M2 est en état rentrée) au niveau de PLCSIM en retourne à l'**Etape 1** jusqu'à l'arrivée de la pièce.

- Etape 5 :

Après la mise à 1 des bits des capteurs B2 et Part_av au niveau du PLCsim., nous obtenons la figure (**Figure IV-41**) au-dessous qui représente une IHM de la sous-station testing en mode simulateur Runtime et PLCSIM.

Nous remarquons la présence de pièce non-noire et aussi le changement de bit au niveau de l'IHM pour :

Le capteur Part_av qui indique la présence de la pièce avec un changement de couleur du jaune à la couleur vert et le capteur B2 qui indique que la pièce est non-noire avec un changement de couleur du jaune à la couleur vert.

La mise à 1 de l'actionneur 1M2 (pour le transfert de vérin vers le haut) avec le changement de couleur du jaune à vert.

Le changement de champ de texte (qui nous indique que la pièce est non-noire) d'invisible à visible.

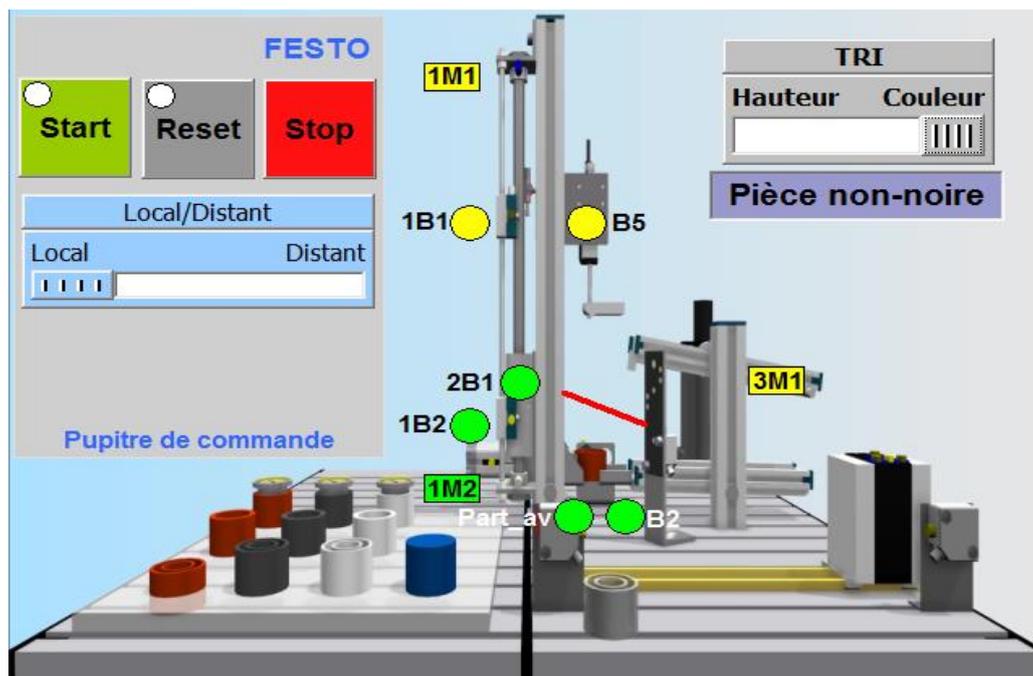


Figure IV-41 La présence de la pièce non noir (rouge).

- Etape 6 :

Après la mise à 1 de l'actionneur 1M2, nous obtenons la figure (Figure IV-42) au-dessous qui représente une IHM de la sous-station testing en mode simulateur Runtime et PLCSIM, nous faisons aussi la mise à 1 de bit de capteur 1B1 et la mise à 0 des bits des capteurs 1B2 et Part_av et B2 au niveau du PLCSIM.

Nous remarquons la présence de la pièce rouge au niveau en haut et aussi le changement de bit au niveau de l'IHM pour :

Le capteur 1B1 qui indique que le vérin est en haut avec un changement de couleur du jaune à la couleur vert et aussi et pour les capteurs 1B2, Part_av et B2 avec des changements de couleur du vert à la couleur jaune.

Le changement de champ de texte (qui nous indique que la pièce est non-noire) de visible à invisible.

La mise à 0 de l'actionneur 1M2 (pour le transfert de la pièce vers le haut) avec le changement de couleur du vert à jaune et la mise à 1 de l'actionneur 3M1 (qui active le coussin d'air) avec le changement de couleur du jaune à vert.

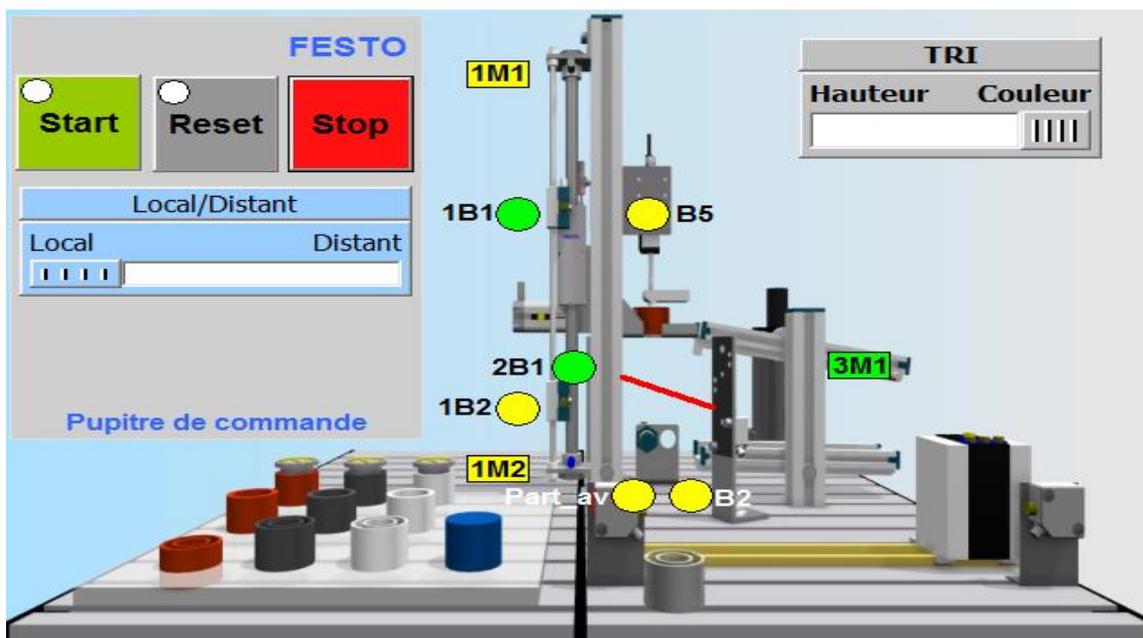


Figure IV-42 La présence de la pièce rouge en haut.

- Etape 7 :

Après la mise à 1 de capteur 1B1, nous obtenons la figure (Figure IV-43) au-dessous qui représente une IHM de la sous-station testing en mode simulateur Runtime et PLCSIM, nous faisons aussi la mise à 0 de bit de capteur 2B1 au niveau du PLCSIM.

Nous remarquons l'éjection de la pièce rouge vers le poste 3 et aussi le changement de bit au niveau de l'IHM pour :

Le capteur 2B1 qui indique que le vérin d'éjection n'est pas en état rentrée avec le changement de couleur du vert à la couleur blanche.

La mise à 1 de l'actionneur 2M2 (pour l'éjection de la pièce vers le poste 2) avec le changement de couleur en vert.

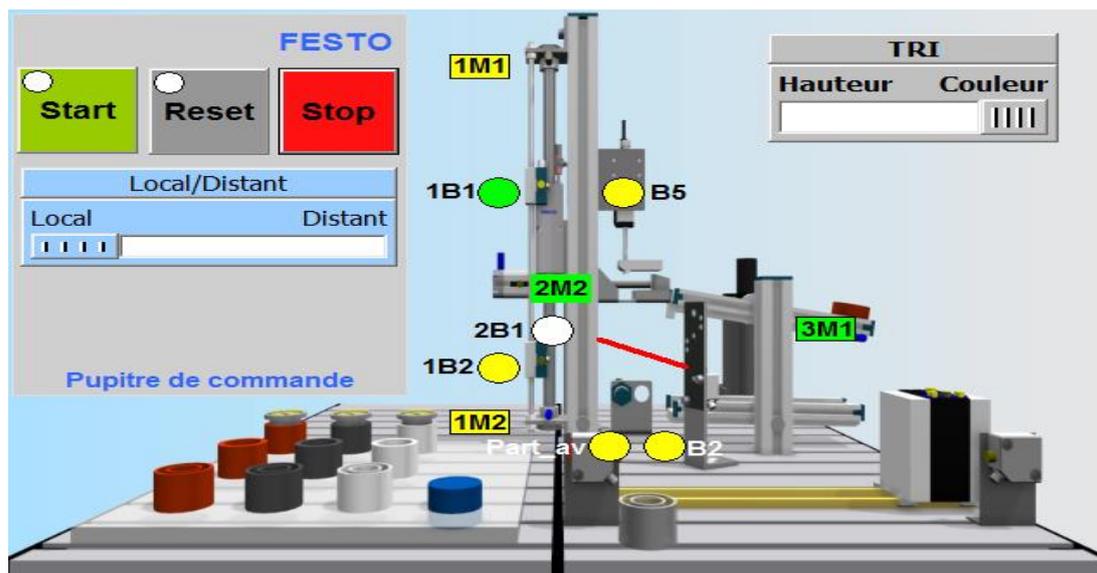


Figure IV-43 Ejection de la pièce rouge vers le poste 3.

- Etape 8 :

Après la mise à 1 de l'actionneur 2M2, Nous obtenons la figure (Figure IV-44) au-dessous qui représente une IHM de la sous-station testing en mode simulateur Runtime et PLCSIM, nous faisons aussi la mise à 1 de bit de capteur 2B1 au niveau du PLCSIM.

Nous remarquons qu'il y a pas de pièce au niveau de haut et aussi le changement de bit au niveau de l'IHM pour :

Le capteur 2B1 qui indique que le vérin d'éjection est rentré avec un changement de couleur de blanche à la couleur vert.

La mise à 0 de l'actionneur 2M2 (pour l'éjection de la pièce vers le poste 3) et de l'actionneur 3M1 (qui active le coussin d'air) avec des changements de couleur du vert à jaune et la mise à 1 de l'actionneur 1M1 (pour le transfert de vérin vers le bas) pour revenir à l'état initial avec le changement de couleur du jaune à vert.

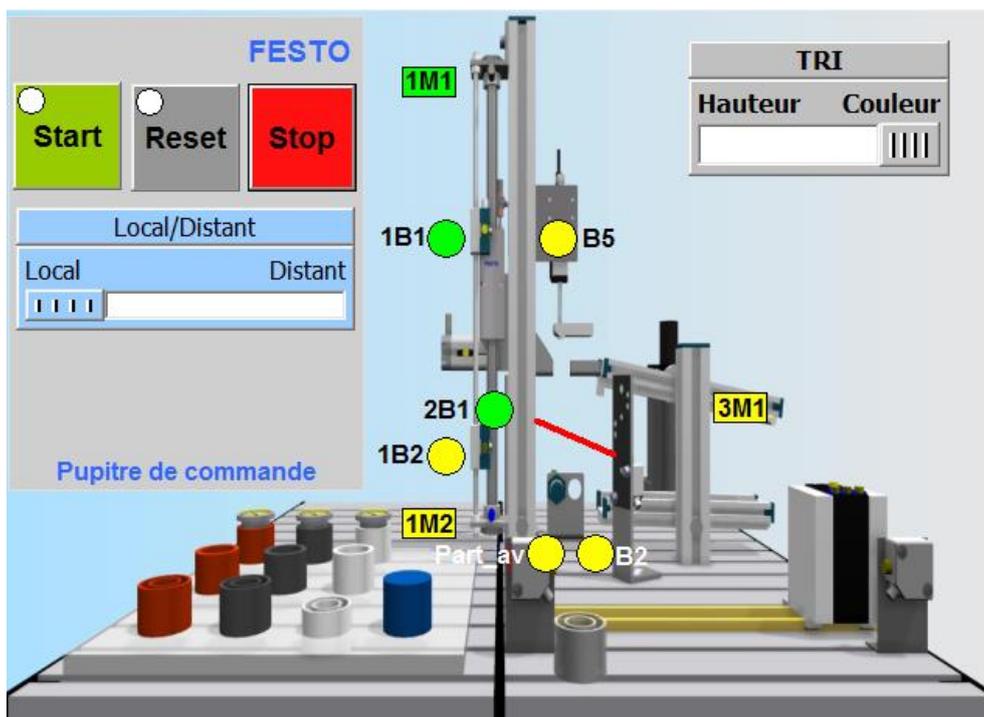


Figure IV-44 Pas de pièces au niveau de haut.

V. Conclusion :

Dans ce dernier chapitre nous avons obtenu le résultat de notre travail qui est la programmation sous STEP7 et la création d'une interface Homme-Machine dans le but de supervision.

Nous avons bien détaillé les étapes pour mieux aider les promotions à venir.

Conclusion générale :

Dans le cadre du projet de fin d'études d'ingénieur « La programmation et la supervision sous les outils Step7 et WinCC flexible de la station MPS-testing » et au regard des résultats obtenus, nous pouvons en tirer que l'objectif fixé a été atteint. En faisant ce mémoire et à l'aide de travail que nous avons fait sur la station MPS testing au niveau de laboratoire de recherche MELT qui se trouve à l'université de Tlemcen, nous avons pu accumuler une expérience et des compréhensions adéquates en automatisation des systèmes industriels, et acquérir des notions sur les systèmes MPS.

L'automatisation est une nécessité pour l'industrie de nos jours, elle permet de réduire le nombre des opérateurs et de minimiser les coûts de fabrication, tout en assurant les équipements et le personnel.

Dans ce travail nous avons étudié en premier le fonctionnement de la station MPS-testing et nous avons aussi récolté les données nécessaires pour accomplir notre objectif.

Nous avons fait dans le premier chapitre une présentation sur les systèmes automatisés et les systèmes MPS et aussi la présentation de notre station à étudier. Pour le deuxième chapitre, nous avons fait une présentation sur les automates programmables industriels (API) et sur le logiciel Step7 ainsi ces langages de programmation et comment l'installer, nous avons enchaîné ce chapitre par le troisième chapitre où nous avons fait une présentation sur la supervision et les HMI ainsi une présentation sur le WinCC flexible 2008 et comment l'installer. Pour le dernier chapitre, nous avons fait la programmation de notre station MPS-testing avec le langage graphique et leur simulation via PLCsim et nous avons aussi créé une interface HMI de supervision par le WINCC Flexible qui nous permet de visualiser le fonctionnement de notre station MPS-testing et de nous donner la possibilité de la commander par l'interface créée,

Finalement nous avons lancé le simulateur Runtime et le PLCsim (après le chargement de notre bloc) et selon la séquence de notre programme de cette station, nous avons incrémenté les bits par PLCsim et nous avons suivi le changement dans l'interface Runtime et aussi nous avons testé le fonctionnement des boutons et Switch que nous avons ajouté dans notre interface HMI.

Cette réalisation met en lumière la possibilité de développer en local des entreprises d'installation des systèmes d'automatisation en vue de la politique de l'état envers

l'encouragement du produit national, sachant que 90% des entreprises d'automatisation en Algérie sont étrangère. Enfin, nous espérons que ce travail puisse apporter un plus et constituer un support supplémentaire pour les promotions à venir. Nous allons aussi poursuivre notre travail dans la partie Master.

Référence :

- [1] : automate programmable industrielle pour GEEA.pdf.
- [2] : M.LEMSYEH, Systèmes automatisés, Dossier référence - 1 -, Maroc: Lycée Technique ER-RAZI El Jadida.
- [3] : <http://technologie-sciarretta.ovh/>
- [4] : http://siitannarelli.com/contenus_opale/chaine_fonctionnelle_gen_auroraW/co/famille_preactionneur.html
- [5] : <https://www.mga-technologies.fr/capteur-vitesse/#:~:text=Le%20capteur%20de%20vitesse%2C%20un,%C3%A0%20la%20vitesse%20de%20rotation.&text=Dans%20le%20secteur%20automobile%2C%20un,vitesse%20de%20roue%20est%20n%C3%A9cessaire.>
- [6] : <https://www.schoolmouv.fr/>
- [7] : Source 3 :A. MTIBAA, Les Systèmes Automatisés de Production, 2012.
- [8] : <http://www.bh-automation.fr/Ressources/Pour-les-automaticiens/Communications/>
- [9] : <https://www.agilicom.fr/tutorial-ASi.html>
- [10] : <https://www.technologuepro.com/cours-systemes-embarques/cours-systemes-embarques-Bus-CAN.htm>
- [11] : <https://le-routeur-wifi.com/ethernet/>
- [12] : <https://uplandsoftware.com/>
- [13] : <https://horustest.io/>
- [14] : <https://www.festo-didactic.com>
- [15] : Automatisation industrielle, Systèmes d'apprentissage et services pour la formation techniqueFESTO
- [16] : https://www.uvt.rnu.tn/resources-uvt/cours/Automates/chap2/co/Module_chap2_6.html
- [17] :Hans Berger, "Automating with SIMATIC", second edition, 2003
- [18]:LP P&M Curie Les Automates – Structure Aulnoye-Aymeries
- [19] :Alain GONZAGA, « Les automates programmables industriels », 2004.
- [20] : <https://www.automation-sense.com>.
- [21] : Grasset, Confucius et les automates; 15/10/2014

- [22] : <https://www.univ-reims.fr/meserp/descriptif-du-materiel/descriptif-du-materiel,9506,27016.htm>
- [23] : https://cache.industry.siemens.com/dl/files/107/45531107/att_91662/v1/S7pr
- [24] : https://www.thecompassforsbc.org/sites/default/files/project_examples/Curriculum%20de%20supervision.pdf
- [25] : https://www.academia.edu/9269456/Supervision_et_S%C3%BBret%C3%A9_des_syst%C3%A8mes_industriels_des_syst%C3%A8mes_industriels_Master_Master_2_2--IIP_IIP_305_305
- [26] : <http://supervisionit.canalblog.com/>
- [27] : 0-IHM.odt.
- [28] : <https://www.futura-sciences.com/tech/definitions/informatique-micro-ordinateur-577/>
- [29] : <https://www.futura-sciences.com/tech/definitions/imprimante-3d-impression-3d-15137/>
- [30] : <https://www.definitions-marketing.com/definition/code-barre-2d/>
- [31] : http://gdt.oqlf.gouv.qc.ca/ficheOqlf.aspx?Id_Fiche=26527250
- [32] : <https://www.biometrie-online.net/technologies/empreintes-digitales>
- [33] : <https://lipn.univ-paris13.fr/~recanati/docs/M2-InHM/Interaction.pdf>
- [34] : <http://perso.liris.cnrs.fr/stephanie.jean-daubias/lifIHM/>
- [35] : <https://www.journaldunet.fr/business/dictionnaire-du-marketing/1208194-reconnaissance-faciale-definition-et-traduction/#:~:text=D%C3%A9finition%20du%20mot%20reconnaissance%20faciale,plusieurs%20images%20de%20la%20personne.>
- [36] : <https://www.univ-reims.fr/meserp/descriptif-des-logiciels/descriptif-des-logiciels,9486,27015.html>

Résumé :

Ce projet d'ingénierie concernant la programmation et la réalisation d'interface Homme-Machine sous les outils STEP7 et WINCC-Flexible est réalisé sur la station MPS-testing au niveau de laboratoire de recherche MELT qui se trouve à l'université Abou Bakr Belkaid à Tlemcen. La première étape de notre travail se caractérisait par la construction du cahier des charges et la récolte des données.

La deuxième étape contient la description des différentes étapes de la création du projet sur le logiciel « STEP7 », ainsi que la création d'une interface Homme – Machine avec le logiciel « WinCC flexible » afin de permettre aux opérateurs de piloter et de superviser en temps réel de la station MPS-testing.

Mots clés : STEP7, SIMATIC Manager, WinCC, WINCC Flexible, Automatisation, API, Supervision, HMI, Interface Homme-Machine, MPS500, MPS testing, MELT

Abstract:

This engineering project concerning the programming and the realization of Man-Machine interface under the STEP7 and WINCC-Flexible tools is realized on the MPS-testing station at the MELT research laboratory which is located at the AbouBakr Belkaid University in Tlemcen. The first step was characterized by the construction of the specifications and the data collection.

The second step contains the description of the different steps of the project creation on the "STEP7" software, as well as the creation of a Man - Machine interface with the "WinCC flexible" software in order to allow the operators to control and supervise in real time the MPS-testing station.

ملخص :

يتم تنفيذ مشروع شهادة مهندس الدولة المتعلق ببرمجة وإنشاء واجهة HMI في إطار الأدوات STEP7 و-WINCC Flexible على محطة testing-MPS على مستوى معمل أبحاث MELT الموجود في جامعة أبو بكر بلقايد بتلمسان. تميزت المرحلة الأولى من عملنا ببناء المواصفات وجمع البيانات اللازمة لإنجاز العمل. نظمنا أطروحتنا على النحو التالي. الفصل الأول المخصص لعرض تقديمي عن الأنظمة الآلية وأنظمة MPS وأيضاً لعرض محطات المراد دراستها، الفصل الثاني الذي يحتوي على عرض تقديمي عن وحدات التحكم المنطقية الصناعية القابلة للبرمجة (API) وحول برنامج Step7 ولغات البرمجة وكيفية تثبيته، ثم الفصل الثالث الذي يحتوي على عرض تقديمي عن الإشراف و HMI وكذلك عرض تقديمي عن WinCC-Flexible 2008 وكيفية تثبيته، وأخيراً الفصل الرابع الذي يحتوي على وصف للمراحل المختلفة لإنشاء المشروع على برنامج "STEP7"، بالإضافة إلى إنشاء واجهة HMI عن طريق برنامج "WinCC-Flexible" للسماح للمشغلين بالتحكم والإشراف في الوقت المناسب بمحطة testing-MPS.