

الجمهورية الجزائرية الديمقراطية الشعبية
Republique Algerienne Democratique et populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
جامعة تلمسان – أبي بكر بلقايد –
Université de Tlemcen – Aboubakr Belkaïd –
كلية التكنولوجيا
Faculté de Technologie
قسم الهندسة الصناعية
Département de Génie Industriel



Memoire de Projet de fin d'Etudes

Présenté pour l'obtention du **diplôme** de **Master**

En : Génie industriel ,**Spécialité** : ingénierie de la production

Conception, simulation, programmation d'API et supervision d'un système industriel automatisé moderne : intégration de FACTORY IO et TIA Portal

Présenté Par : MEKELLECHE Malika et SEMMACH Nouria

soutenu le 25/06/2025 devant le jury composé de :

Présidente : Mlle. GHOMRI Latefa , Professeur, Université de Tlemcen
Examineur : Mr.GUEZZEN Amine Hakim , M.C.A , Université de Tlemcen
Examinatrice : Mlle. HOUBAD Yamina, M.C.B, Université de Tlemcen
Encadrant : Mr.Ahmed HASSAM M.C.B, Université de Tlemcen
Co-Encadrant : Mr.Mohammed EL Amin MKEDDER Ingénieur de recherche principal,
Université de Tlemcen

Année universitaire 2024/2025

DÉDICACE

Nous dédions ce travail à nos familles respectives, sources d'amour, de soutien et de patience.

À nos parents, pour leurs sacrifices inestimables, leur confiance et leur encouragement permanent tout au long de notre parcours académique.

À nos frères, sœurs et proches, pour leur affection et leur présence réconfortante.

Nous dédions également ce projet à tous nos enseignants, qui nous ont transmis leur savoir avec dévouement.

Enfin, nous dédions ce mémoire à tous ceux qui, de près ou de loin, ont contribué à notre formation, à notre épanouissement personnel et à la réussite de ce projet.

REMERCIEMENTS

Nous remercions tout d'abord le bon Dieu de nous avoir donné la foi, la force et la persévérance nécessaires pour mener à bien ce travail.

Nous tenons à exprimer notre profonde gratitude à Dr. Hassam Ahmed, notre encadreur, ainsi qu'à Dr. Mkedder Mohammed El Amin, notre co-encadreur, pour nous avoir proposé ce sujet de projet de fin d'études. Leur accompagnement, leur rigueur, leurs conseils pertinents et leur disponibilité constante ont grandement contribué à la qualité de notre travail.

Nous adressons également nos remerciements les plus sincères à tous les enseignants du département Genie industriel de l'Université de Tlemcen, qui nous ont transmis, tout au long de notre formation, un savoir précieux et une méthodologie rigoureuse.

Nos remerciements vont aussi à notre collègue et ami Sedeud Aimane, pour son aide, sa motivation et son soutien moral tout au long de ce parcours.

Nous pensons également à toutes les personnes qui, de près ou de loin, ont contribué à la réalisation de ce projet.

Enfin, nous remercions chaleureusement les membres du jury qui nous font l'honneur d'évaluer notre travail.

TABLE DES MATIÈRES

Dédicace	i
Remerciements	ii
Introduction générale	1
1 Présentation des systèmes industriels modernes et de l'industrie 4.0	3
1.1 Introduction	3
1.2 Présentation de l'industrie 4.0	3
1.2.1 Les objectifs de l'industrie 4.0	5
1.2.2 Les systèmes industriels modernes dans le contexte de l'Industrie 4.0	5
1.3 Cadre d'analyse pour l'Industrie 4.0	6
1.3.1 Déclinaison des processus, produits et services	6
1.3.2 Capacités de surveillance, contrôle, optimisation et autonomie	6
1.3.3 Technologies de l'industrie 4.0	8
1.4 Éléments des systèmes industriels modernes	13
1.4.1 Composants matériels : Fondements et Structure des Systèmes Automatisés	13
1.4.2 Composants logiciels	24
1.5 Conclusion	28

2	Factory I/O : Présentation, Modélisation et Programmation avec Contrôle I/O du système de production	29
2.1	Introduction	29
2.2	Présentation factory io	29
2.2.1	Fonctionnalités générales	29
2.2.2	Objectifs et rôle de Factory I/O dans la conception des systèmes automatisés	33
2.2.3	Avantages de Factory I/O	35
2.3	Modélisation du système industriel	35
2.3.1	Description du processus industriel choisi	35
2.3.2	Les caractéristiques d'élément utilisé	38
2.4	Création de l'environnement 3D dans FACTORY IO	42
2.5	Présentation de CONTROL I/O	48
2.5.1	Bases de CONTROL I/O	49
2.6	Description du fonctionnement du programme	52
2.6.1	Station de distribution	52
2.6.2	Acheminement des pièces vers les centres d'usinage	53
2.6.3	Fonctionnement du centre usinage :	54
2.6.4	Fonctionnement de la station d'assemblage	56
2.6.5	Suivi de production :	59
2.7	Conclusion	60
3	PROGRAMMATION, CONFIGURATION ET SIMULATION DU SYSTEME DE PRODUCTION AUTOMATISE VIA TIA PORTAL	61
3.1	Introduction	61
3.2	Présentation de logiciel TIA Portal	61
3.2.1	Description du logiciel TIA Portal	62
3.2.2	Langages de programmation	65
3.2.3	Présentation de S7 PLCSIM	67
3.2.4	Caractéristiques de TIA Portal	68
3.3	Présentation d'automate S7-300	68

3.3.1	Modularité du S7-300 :	69
3.3.2	Caractéristiques de la CPU 314C-2 PN/DP	71
3.4	CAHIER DE CHARGE	72
3.4.1	Démarrage et distribution des marchandises	72
3.4.2	Acheminement des pièces vers les centres d'usinage et la fonctionnement de centre d'usinage	73
3.4.3	Assemblage	74
3.5	Tableau des Entrées/Sorties du Système	76
3.6	Établissement de la communication entre Factory I/O et TIA Portal	79
3.7	La programmation de l'API	85
3.7.1	Programmation et Architecture des Blocs Fonctionnels du Système Automatisé avec GRAFCET	85
3.7.2	Test, Simulation et Supervision avec S7 PLCSIM et Factory I/O	89
3.8	Conclusion	91
	Conclusion générale	92
	Annex	94
	Références	102

TABLE DES FIGURES

1.1	De la première révolution industrielle a l'industrie 4.0 [2]	4
1.2	Classification des capacités des systèmes [1]	7
1.3	Schéma général d'un système automatisé [6]	14
1.4	Architecture fonctionnelle d'un système automatisé. [7]	15
1.5	Types d'automates programmables (compact et modulaire) [8] [9]	17
1.6	Structure interne d'un automate programmable [7]	18
1.7	Cycle de fonctionnement d'un automate programmable[7]	19
1.8	Principe de fonctionnement d'un capteur.[10]	20
1.9	Exemple de programme en langage IL (Instruction List)[14]	24
1.10	Programme en langage ST (Structured Text)[14]	25
1.11	Programme en langage Ladder (LD)[14]	25
1.12	Exemple de bloc fonctionnel en langage FBD (Function Block Diagram)[14]	26
1.13	structure d'un GRAFCET[16]	26
2.1	factory io	30
2.2	barre de menu	31
2.3	menu de palette	32
2.4	barre de drivers	33
2.5	Menu du pilote (Driver menu)	33

2.6	Station de distribution	36
2.7	station d'usinage	37
2.8	Station d'assemblage des pièces	38
2.9	Lancement de Factory I/O : icône sur le bureau	42
2.10	Création d'une nouvelle scène	43
2.11	Interface de travail de Factory I/O	43
2.12	Palette – sélection du convoyeur 6 m	44
2.13	Placement du convoyeur dans la scène	44
2.14	Palette – sélection de la matière première	45
2.15	Affichage du nom du convoyeur et accès au contrôle manuel	46
2.16	Passage en mode Run : bouton « Play »	47
2.17	Déplacement de la pièce du point A au point B	47
2.18	Interface de Control I/O avec un exemple de programme FBD appliqué dans Factory I/O	48
2.19	Interface de Control I/O avec un exemple de programme FBD appliqué dans Factory I/O [22]	49
2.20	Représentation des blocs fonctionnels dans Control I/O et leurs connexions logiques[23]	50
2.21	Logique de lancer le système et de détection et de poussée des pièces	53
2.22	Logique de tri des pièces selon le comptage pair/impair	54
2.23	logique de serrage et de libération de la pièce par le positionneur	55
2.24	conditions de démarrage et gestion des erreurs du centre d'usinage	56
2.25	Logique de controle des positionneurs du couvercle et de base	57
2.26	programmation des mouvements verticaux (axe Z) du robot d'assemblage	58
2.27	utilisation du compteur pour le suivi des mouvements sur les axes X et Z	58
2.28	commande des déplacements horizontaux (axe X) du robot d'assemblage	59
2.29	logique de redémarrage du cycle du processus d'assemblage	59
3.1	Logo de TIA Portal version 15 (Siemens) [26]	62
3.2	Vue du Portail	63
3.3	Vue du Projet	63

3.4	Fenêtre de création d'un nouveau bloc	64
3.5	Tableau présente trois Langages de programmation des API [27]	65
3.6	Les éléments de base du GRAFCET [30]	67
3.7	: Automate Siemens SIMATIC S7-300 – CPU 314C-2 PN/DP [32]	69
3.8	Création d'un nouveau projet dans TIA Portal	79
3.9	Choix de l'option "Écrire un programme API"	80
3.10	Sélection de l'automate S7-314C-2PN/DP	80
3.11	Création de la table des variables (entrées et sorties)	81
3.12	Démarrage de la simulation avec S7-PLCSIM	81
3.13	Choix du type d'interface PG/PC et chargement vers PLCSIM	82
3.14	Chargement du programme dans l'automate	82
3.15	Ouverture du menu des drivers pour la configuration de la connexion	83
3.16	Sélection du driver Siemens S7-PLCSIM dans Factory I/O	83
3.17	Vérification de la configuration du driver S7-PLCSIM	84
3.18	Association des E/S aux adresses du programme et test de la connexion	85
3.19	MAIN BLOC programme principale	85
3.20	bloc fonctionnel FB1	86
3.21	GRAFCET de gestion de l'armoire de commande	87
3.22	GRAFCET d'orientation des pièces selon la parité vers les centres d'usinage	88
3.23	GRAFCET de contrôle du positionneur de bases dans la station d'assemblage	89
3.24	Simulation du système automatisé	90

LISTE DES TABLEAUX

1.1	Grandeurs d'entrée et de sortie et effet utilisé pour les capteurs actifs [11]	21
1.2	comparaison des types d'actionneur	23
2.1	Avantages de Factory I/O	35
2.2	caractéristiques d'élément utilisé [20]	39
2.3	bloc Arithmétique [21]	50
2.4	bloc Supplémentaire(Extra)[21]	51
2.5	bloc Logique[21]	51
2.6	bloc Mathématique [21]	52
2.7	bloc Relationnel et Égalité[21]	52
3.1	caractéristiques de la CPU 314C-2PN/DP. [33]	72
3.2	Tableau représente des Entrées/Sorties du Système	76

INTRODUCTION GÉNÉRALE

L'industrie moderne est en pleine transformation et mutation sous l'impulsion de la révolution numérique et l'industrie 4.0 représente une évolution majeure qui redéfinit les méthodes de production traditionnelles en y intégrant les technologies numériques, l'automatisation avancée, la connectivité, la robotisation, l'intelligence artificielle, et le traitement massif des données (Big Data). Cette nouvelle ère industrielle repose sur la convergence entre le monde physique et le monde numérique, notamment grâce à l'introduction des systèmes cyber-physiques (CPS), de l'Internet des objets industriels (IIoT) et de la communication machine-à-machine. Ces technologies visent à rendre les processus industriels plus intelligents, plus autonomes, plus flexibles, tout en améliorant la qualité des produits et en réduisant les coûts de production.

Le présent projet de fin d'études s'inscrit dans une approche moderne et intégrée de l'automatisation industrielle, en s'appuyant sur des outils numériques avancés. L'objectif principal de ce travail est de concevoir, modéliser, programmer et simuler un système de production automatisé dans un environnement virtuel, en reproduisant les conditions et les contraintes d'un système industriel réel.

Pour mener à bien ce projet, trois outils majeurs ont été mobilisés, chacun jouant un rôle spécifique et complémentaire dans le processus de développement :

- Factory I/O est un logiciel de simulation 3D interactif qui permet de modéliser de manière réaliste un système industriel automatisé. Il offre un environnement graphique immersif où l'on peut visualiser, tester et faire fonctionner un système de production complet (poste de distribution, d'usinage, d'assemblage, etc.). Cet outil facilite la compréhension du comportement dynamique des processus et permet de valider visuellement la logique de commande mise en œuvre.
- Le langage GRAFCET (Graphe Fonctionnel de Commande Étape/Transition) qui est un outil puissant et une méthode normalisée (CEI 60848) de modélisation des séquences de fonctionnement des automatismes industriels. Il permet de représenter de manière claire, structurée et hiérarchisée les enchaînements logiques des actions dans un processus automatisé. L'utilisation du GRAFCET dans ce projet a facilité la transition entre la modélisation fonctionnelle et la programmation dans TIA Portal.
- TIA Portal (Totally Integrated Automation Portal) est la plateforme logicielle de Siemens dédiée à la programmation et à la configuration des automates programmables industriels (API). Elle permet de développer des programmes de commande, de configurer les matériels (API, IHM, entrées/sorties, réseaux de communication),

et d'effectuer des simulations grâce à l'outil intégré S7-PLCSIM. C'est dans cet environnement que le cœur du contrôle logique du système a été élaboré.

L'association de ces trois outils a permis de créer un jumeau numérique fonctionnel du système automatisé, allant de la modélisation physique à la simulation logique du comportement. Elle offre un cadre de travail idéal pour comprendre l'interaction entre la partie commande (automate) et la partie opérative (système physique modélisé), tout en permettant de tester, valider et corriger les algorithmes dans un environnement sécurisé et pédagogique. Ce mémoire est structuré en trois chapitres, chacun correspondant à une étape clé du développement du projet :

- Le premier chapitre est consacré à la présentation des systèmes industriels modernes et aux fondements de l'Industrie 4.0. Il expose les nouvelles tendances en automatisation, les technologies émergentes, ainsi que les exigences en matière de connectivité, de performance et de flexibilité.
- Le deuxième chapitre porte sur l'utilisation de Factory I/O. Il présente l'outil en détail, décrit la modélisation 3D du système de production automatisé, les différentes stations mises en œuvre, ainsi que la programmation par contrôle I/O pour simuler les réactions du système en boucle ouverte ou fermée.
- Le troisième chapitre s'intéresse à la modélisation, la programmation, la configuration et la simulation du système de production automatisé via TIA Portal. Il décrit la mise en œuvre du GRAFCET, l'implémentation dans l'environnement Siemens, la configuration du matériel, ainsi que les tests réalisés grâce à la simulation intégrée.

Ce projet se veut une réponse concrète aux défis de l'ingénierie moderne. Il illustre la capacité à intégrer des outils numériques de pointe dans une approche de conception complète, allant de la modélisation à la simulation en passant par la programmation. Il offre également un cadre d'apprentissage riche pour développer des compétences transversales en automatisation, ingénierie de la production et des systèmes, informatique industrielle et simulation virtuelle, dans une optique alignée avec les exigences de l'industrie du futur.

CHAPITRE 1

PRÉSENTATION DES SYSTÈMES INDUSTRIELS MODERNES ET DE L'INDUSTRIE 4.0

1.1 Introduction

L'Industrie 4.0 marque une transformation profonde du secteur industriel grâce à l'intégration de technologies de pointe telles que l'intelligence artificielle, l'Internet des objets (IoT), le cloud computing et l'analyse des données massives (Big Data). Cette révolution industrielle vise à rendre les systèmes de production plus intelligents, flexibles et autonomes grâce à la connectivité et à la décentralisation des processus décisionnels.

Les systèmes industriels modernes se caractérisent par l'automatisation poussée, La mise en réseau des machines et l'exploitation en temps réel des données collectées par divers capteurs et équipements connectés. L'objectif principal est d'optimiser les moyens et les ressources, d'améliorer la productivité et de renforcer la sûreté et la cybersécurité des infrastructures industrielles.

Dans ce contexte, la supervision industrielle joue un rôle fondamental en permettant le suivi et le contrôle des opérations en temps réel à l'aide de solutions SCADA (Supervisory Control and Data Acquisition). Ces outils, tels que Wonderware (AVEVA System Platform), Siemens WinCC, Ignition, GE iFIX et Schneider Electric EcoStruxure SCADA, offrent aux opérateurs une meilleure visibilité sur les processus, facilitent la détection des anomalies et contribuent à l'optimisation des performances industrielles.

1.2 Présentation de l'industrie 4.0

L'expression 'Industrie 4.0' englobe un bouquet de technologies et d'idées associées à la restructuration de la chaîne de valeur [Hermann et coll., 2015]. Ainsi, la perception de l'industrie 4.0 s'appuie sur la communication en temps réel pour surveiller et intervenir sur les systèmes physiques. Les systèmes interagissent et collaborent non seulement entre eux, mais aussi avec les individus, dans le but de décentraliser le processus décisionnel. L'industrie 4.0 se concentre donc sur la connectivité, encourageant ainsi l'élaboration de nouveaux procédés, produits et services. Son déploiement

nécessite l'intégration de diverses compétences spécifiques aux technologies numériques.

Le terme 'Industrie 4.0', qui a vu le jour en 2011 [Drath & Horch 2014] grâce à une initiative allemande, fait référence à la quatrième révolution industrielle. La première révolution industrielle se réfère à la transition de la production artisanale vers la production mécanisée, survenue dans la seconde moitié du XVIII^e siècle. La seconde révolution industrielle à la fin du 19^e siècle se distingue principalement par l'électrification des systèmes de production et le passage à la production en série. La troisième révolution, pour sa part, se distingue par l'automatisation de la production via l'électronique et les technologies de l'information dans les années 1970.

La quatrième révolution industrielle est désignée par divers noms en fonction des contributeurs et des régions géographiques. Nous rencontrons donc des expressions comme Industry 4.0 dans sa version anglaise, Usine intelligente (Smart Factory), Industrie intelligente (Smart Industry), Usine du futur (Factory of the Future), Industrie du futur (Industry of the Future), Usine numérique (Digital Factory), entre autres. Ces initiatives explorent diverses facettes concernant l'apparition de nouveaux procédés, produits et services. Effectivement, l'Industrie 4.0 ne se limite pas à l'excellence opérationnelle et aux processus de production, elle transforme aussi radicalement la portée des produits et des services.

À l'opposé de diverses méthodes et technologies qui se basent sur une centralisation des données et des décisions, Industrie 4.0 suggère une décentralisation des décisions en répartissant l'information au sein de chaque composant du système global. Cette décentralisation stimule la souplesse et la réactivité des systèmes en renforçant leur capacité de réponse et d'autonomie. Cette décentralisation permet une mise en place progressive des technologies sur lesquelles se fonde l'Industrie 4.0.[1]

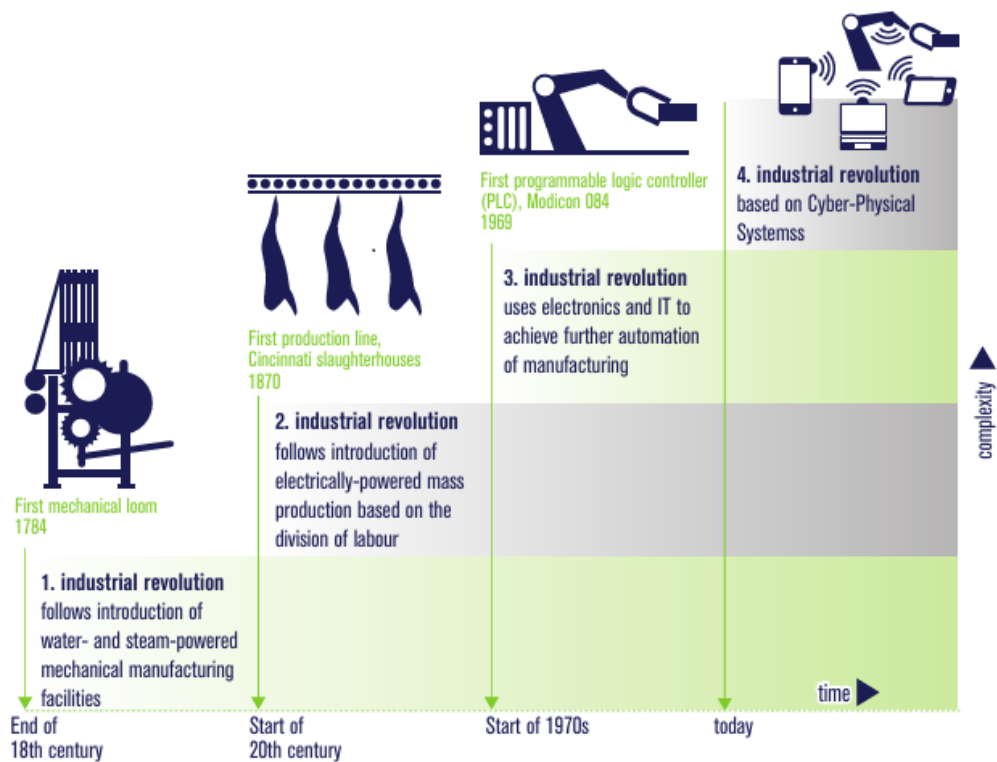


FIG. 1.1 : De la première révolution industrielle à l'industrie 4.0 [2]

1.2.1 Les objectifs de l'industrie 4.0

[2] Le but premier de l'industrie 4.0 est de générer de la valeur ajoutée dans le secteur manufacturier, indépendamment des frais de personnel. Cet objectif peut être divisé en plusieurs sous-objectifs

- **optimisation des moyens et des ressources**

Le processus de fabrication doit être optimisé en ayant un équipement de production polyvalent et une chaîne de production flexible. Cela permet une adaptation rapide aux changements de propriété ou de maintenance, en évitant le temps mort et en optimisant les coûts. Les autres éléments clés comprennent la gestion des matières premières, la simulation des niveaux de stock, le transport, la logistique et l'amélioration de la gestion de l'énergie, en particulier le week-end et les vacances.

- **Vers plus de flexibilité**

La flexibilité des processus de fabrication est un élément clé de l'industrie 4.0 . L'industrie 4.0 permet la création de valeur par la personnalisation du bien par le client . L'objectif est que chaque client puisse demander un produit sur - mesure . Pour cela , il faut donc pouvoir adapter le processus de fabrication en fonction des besoins

- **Sûreté de fonctionnement**

La sûreté est un élément des plus importants dans l'industrie . Il est nécessaire de protéger les biens et les personnes des éventuels risques . L'industrie 4.0 réaffirme cette priorité et l'étend à l'environnement au sens large . Les problèmes de sûreté sont apparus dès le début de l'industrie . Les accidents sont potentiellement plus graves avec des machines La plupart d'entre eux ont heureusement été résolus notamment ceux liés aux protections des équipements

- **Sécurité informatique**

Les premiers virus comme le ver Morris ont montré que la sécurité ne devait plus être négligée . Ces menaces s'attaquent principalement aux systèmes informatiques des entreprises, Dans l'industrie 4.0 , la sécurité est donc considérée comme cruciale , au même titre que la sûreté .Pour les systèmes cyber - physiques , la sécurité doit être intégrée dès la conception (security by design) .

- **L'humain dans l'industrie 4.0**

Les objectifs de l'industrie 4.0 ne se limitent pas aux aspects financiers et techniques. Il insiste sur l'importance de former les employés tout au long de leur parcours professionnel. La formation devrait permettre de favoriser les reconversions internes, par exemple. En pratique, il s'agit de favoriser le développement des employés, leur permettant d'assumer davantage de responsabilités, de contribuer à l'organisation du travail et d'avoir une carrière prolongée au sein de la société [2]

1.2.2 Les systèmes industriels modernes dans le contexte de l'Industrie 4.0

Les systèmes industriels modernes, dans le contexte de l'Industrie 4.0, représentent une évolution majeure par rapport aux systèmes traditionnels. Ils se caractérisent par l'intégration poussée de technologies numériques et de l'information au sein des processus de production, de la conception à la maintenance, en passant par la logistique et la gestion. Ils se caractérisent par l'utilisation de L'automatisation, de l'intelligence artificielle (IA), de l'Internet des objets industriel (IIoT), du big data, et des jumeaux numériques.

1.3 Cadre d'analyse pour l'Industrie 4.0

[1] Suite à une brève introduction de l'Industrie 4.0, nous vous proposons désormais un cadre d'analyse basé sur trois axes : La déclinaison des processus, produits et services, la capacités de surveillance, contrôle, d'optimisation ou d'autonomie et la liste des technologies.

1.3.1 Déclinaison des processus, produits et services

Le terme « Industrie 4.0 » laisse entendre que l'effort se focalise sur l'amélioration des processus de fabrication. Cependant, l'accroissement de la présence de capteurs et des échanges en temps réel ouvre également de nouvelles opportunités dans la définition de produits communicants, de même que dans la « servicisation » des produits [Kohler & Weisz, 2015].

En effet, les apports du numérique peuvent se décliner selon trois axes principaux :

- **En ce qui concerne les procédés** : l'Industrie 4.0 promet une évolution de la production de masse vers une fabrication individualisée (à l'unité). Les processus deviennent plus flexibles et adaptables afin de répondre aux exigences des clients et ainsi optimiser la création de valeur. Avec l'émergence des machines autonomes et la communication entre les machines et les systèmes cyber-physiques (systèmes combinant des éléments physiques et numériques), les décisions de production sont prises en temps réel. Les modules d'usine intelligente (Smart Factory) illustrent cette vision de l'Industrie 4.0 par leur capacité à reconfigurer automatiquement la chaîne de production et les machines afin de s'adapter aux besoins d'une production à l'unité.
- **Les objets connectés** : facilitent la collecte de données en direct. Ces informations peuvent être examinées sur-le-champ, ce qui permet au système de s'ajuster de façon autonome à son environnement ou d'être exploitées plus tard pour la création de nouveaux produits ou services. Le cas de la voiture autonome démontre la possibilité de s'adapter à son environnement extérieur.
- **La disponibilité des données et les options d'analyse** : créent des opportunités pour développer de nouveaux services basés sur les données (data-based services - [PWC, 2016]). L'émergence de services de dernière génération permet l'expansion vers de nouveaux marchés. Dans le cas du gant de golf lié au smartphone, un service est proposé à l'utilisateur grâce à l'interprétation des données. L'utilisateur reçoit des conseils techniques et des informations pour perfectionner son jeu.

1.3.2 Capacités de surveillance, contrôle, optimisation et autonomie

L'abondance des fonctionnalités de connectivité offre la possibilité d'implémenter de nouvelles aptitudes pour les processus, les produits et les services. Ces compétences récentes peuvent être classées en quatre catégories : la surveillance, le contrôle, l'optimisation et l'autonomie, allant ainsi du plus élémentaire au plus avancé [Porter & Heppelmann, 2014]. Les capacités de surveillance servent de fondement pour établir le contrôle. De même, le contrôle et la surveillance sont indispensables pour favoriser l'optimisation.



FIG. 1.2 : Classification des capacités des systèmes [1]

- **La surveillance :** Les compétences numériques et de connectivité sont mises en application pour veiller à l'état ou au fonctionnement du système ainsi qu'à son environnement externe. Ainsi, via des capteurs ou des données externes, le système est en mesure de déclencher des alertes pour notifier les utilisateurs ou d'autres systèmes, sans toutefois entreprendre d'actions par lui-même. Cette supervision offre la possibilité de surveiller le développement des propriétés du système et de maintenir un registre d'exploitation pour apporter des éclaircissements lors d'une possible prise de décision. Un exemple de surveillance est le réfrigérateur General Electric, qui notifie son propriétaire sur son téléphone lorsque la porte est laissée ouverte.
- **Le contrôle :** Des algorithmes dédiés supervisent le système. Ces algorithmes définissent une action simple que le système doit entreprendre en réponse à des modifications de son état ou de son environnement. Cette fonctionnalité offre également à l'utilisateur la possibilité d'interagir avec le système afin de personnaliser son comportement. Un exemple de mesure de contrôle est l'activation d'un ventilateur qui fournit de l'air frais dès que la teneur en polluants dépasse un certain niveau. Un autre exemple est celui des voitures Tesla [Porter & Heppelmann, 2014], qui intègrent un système pour contrôler l'intégrité du véhicule et télécharger une mise à jour si besoin.
- **L'optimisation :** Grâce à l'abondance des données de surveillance disponibles, couplée à la capacité de gestion, il est possible d'optimiser les performances du système. Celui-ci est donc capable d'effectuer des études sur les données opérationnelles ou d'usage et de mettre en œuvre des algorithmes pour améliorer son exploitation et ses performances. Par exemple, WTC ajuste l'angle des pales de ses turbines éoliennes pour contrôler la puissance produite.
- **L'autonomie :** Le système acquiert de l'indépendance grâce à la fusion des capacités de surveillance, de contrôle et d'optimisation. Un système autonome a la capacité d'apprendre de son contexte, de diagnostiquer lui-même ses besoins et de s'adapter aux préférences de l'utilisateur. Ces systèmes autonomes ont la capacité d'interagir de manière coordonnée et en temps réel avec d'autres produits ou systèmes présents dans leur environnement. Un exemple est l'aspirateur autonome Roomba d'iRobot, qui ajuste son parcours en fonction de l'évaluation de son

environnement, optimise la force d'aspiration selon le type de surface et revient à sa station de charge lorsque les batteries nécessitent une recharge.[1]

1.3.3 Technologies de l'industrie 4.0

Cette section présente les outils technologiques qui peuvent aider à mettre en œuvre la stratégie identifiée, organisés en huit groupes technologiques. Il s'agit notamment des systèmes cyber-physiques (CPS), de l'Internet des objets, des données massives (Big Data), de l'analyse des données en temps réel, de l'intelligence artificielle, de la cybersécurité, de la communication intermachines (Machine-to-Machine, M2M), des systèmes de simulation et de la réalité augmentée. Leur combinaison permet de mettre en œuvre efficacement la stratégie numérique de l'entreprise.[1]

A_ Systèmes cyber-physiques (CPS)

Les Systèmes Cyber-Physiques (CPS) sont les piliers de l'Industrie 4.0, représentant une convergence révolutionnaire entre le monde physique et le monde numérique. Ils intègrent de manière transparente des composants physiques tels que des machines, des capteurs et des actionneurs, avec des éléments cybernétiques sophistiqués, notamment des logiciels, des réseaux et l'intelligence artificielle. Cette fusion permet une interaction et une coordination sans précédent, transformant les processus industriels traditionnels.

L'intelligence artificielle joue un rôle central dans l'optimisation des CPS, permettant une prise de décision autonome et une adaptation en temps réel. Grâce à cette intégration, les CPS facilitent une automatisation avancée, une surveillance continue et une maintenance prédictive, réduisant ainsi les temps d'arrêt et améliorant l'efficacité globale. Ils ouvrent la voie à des usines intelligentes, flexibles et hautement connectées, capables de s'adapter dynamiquement aux demandes du marché.

Les CPS permettent également une personnalisation de masse, où les produits peuvent être adaptés aux besoins spécifiques des clients tout en conservant l'efficacité de la production de masse. Cette technologie favorise également le développement de chaînes d'approvisionnement intelligentes, où les données sont partagées en temps réel pour optimiser la logistique et la gestion des stocks. Les CPS permettent :

- **Collecte et traitement de données** : Les capteurs intégrés aux machines jouent un rôle crucial dans la collecte de données en temps réel. Ils enregistrent des informations sur l'état de la production et les performances des équipements. Ces données brutes sont ensuite transmises à des systèmes informatiques pour être traitées et analysées. L'analyse de ces données permet d'obtenir des informations précieuses pour optimiser les processus de production. Cette approche basée sur les données est essentielle pour l'Industrie 4.0.
- **Communication et contrôle** : Les Systèmes Cyber-Physiques (CPS) établissent une communication fluide entre les machines et les systèmes de contrôle. Cette interconnexion permet une coordination efficace des processus de production. Les machines peuvent échanger des données et des instructions en temps réel. Cette communication optimise l'utilisation des ressources, réduisant le gaspillage et améliorant l'efficacité. Les CPS permettent une automatisation avancée, où les machines s'adaptent et collaborent de manière autonome. Cette approche conduit à une production plus flexible et réactive aux demandes du marché.
- **Automatisation et autonomie** L'intelligence artificielle et l'apprentissage automatique transforment les Systèmes Cyber-Physiques (CPS). Grâce à ces technologies, les CPS gagnent en autonomie décisionnelle. Ils s'adaptent

dynamiquement aux variations de l'environnement de production. Les décisions sont prises en temps réel, optimisant les processus. Cette adaptation permet une flexibilité et une réactivité accrues. Les CPS deviennent ainsi des éléments clés de l'Industrie 4.0.

- **Surveillance et maintenance prédictive** Les Systèmes Cyber-Physiques (CPS) offrent une surveillance continue de l'état des équipements. Ils détectent les anomalies en temps réel, permettant une intervention rapide. Cette capacité rend possible la mise en œuvre d'une maintenance prédictive. Les interventions sont planifiées avant les pannes, réduisant les temps d'arrêt. Cela minimise les coûts de maintenance et optimise la production. Les CPS améliorent ainsi la fiabilité et l'efficacité des équipements. [3]

B_ Internet des objets industriels (IIoT)

L'Internet industriel des objets (IIoT) représente l'évolution de l'IoT spécifiquement adaptée au secteur industriel. Il consiste en un réseau complexe d'appareils interconnectés, comprenant machines, capteurs et actionneurs. Ces dispositifs collectent et partagent des données en temps réel, créant un flux d'informations continu. L'objectif principal de l'IIoT est d'optimiser l'efficacité des processus de production. Il vise également à accroître la productivité et à améliorer la qualité des produits finis. Grâce à l'analyse des données, l'IIoT permet une prise de décision éclairée et une automatisation avancée.

□ Composants clés de l'IIoT

- **Capteurs** : Les capteurs sont des dispositifs qui mesurent des grandeurs physiques (température, pression, vibration, etc.) et les convertissent en signaux électriques. Ils sont intégrés aux machines et aux équipements pour collecter des données sur l'état de la production, les performances des équipements, etc.
- **Actionneurs** : Les actionneurs sont des dispositifs qui convertissent des signaux électriques en actions mécaniques (mouvement, ouverture/fermeture, etc.). Ils sont utilisés pour contrôler les machines et les équipements en fonction des données collectées par les capteurs.
- **Réseaux de communication** : Les réseaux de communication (sans fil ou filaires) sont utilisés pour transmettre les données des capteurs vers les plateformes de gestion de données.
- **Plateformes de gestion de données** : Les plateformes de gestion de données sont essentielles dans l'Industrie 4.0. Elles constituent des systèmes logiciels sophistiqués. Leur rôle principal est de stocker les vastes quantités de données collectées par les capteurs. Ces plateformes ne se contentent pas de stocker, elles traitent également ces données brutes. Elles effectuent des analyses approfondies pour extraire des informations pertinentes. Ces analyses permettent d'optimiser les processus industriels et de prendre des décisions éclairées
- **Applications et services** : Les applications et les services sont des logiciels qui utilisent les données collectées par les capteurs pour fournir des informations et des analyses aux utilisateurs. Ils peuvent être utilisés pour surveiller la production, optimiser les processus, prévoir les pannes, etc.

□ Comment fonctionne l'IIoT ?

- **Collecte de données** : Des capteurs intégrés aux machines et aux équipements collectent des données sur l'état de la production, les performances des équipements, etc.
- **Transmission des données** : Les données sont transmises via des réseaux (sans fil ou filaires) vers des plateformes de gestion de données.

- **Analyse des données** : Les données sont analysées pour identifier les tendances, les anomalies et les opportunités d'amélioration.
- **Actions et contrôle** : Les informations issues de l'analyse des données sont utilisées pour prendre des décisions, optimiser les processus de production et contrôler les équipements.[4]

C_ Big data

□ Big Data : une plongée dans l'océan des données

Le Big Data transcende la simple notion de volume. Il représente une transformation fondamentale dans notre approche des données. Imaginez un océan immense d'informations provenant de sources multiples et variées. Ces données sont générées par des capteurs, des réseaux sociaux, des transactions en ligne et bien d'autres sources. Elles se caractérisent par leur volume massif, leur vitesse de génération élevée et leur diversité. L'analyse de ces données permet de découvrir des tendances et des informations cachées. Le Big Data offre ainsi des opportunités inédites pour optimiser les processus et prendre des décisions éclairées. Il révolutionne la façon dont les entreprises et les organisations exploitent l'information :

- **Données structurées** : ce sont les données organisées dans des tableaux, avec des lignes et des colonnes (par exemple, les données clients dans une base de données).
- **Données semi-structurées** : elles ont une structure, mais moins rigide que les données structurées (par exemple, les fichiers JSON ou XML).
- **Données non structurées** : ce sont les données brutes, sans organisation particulière (par exemple, les textes, les images, les vidéos).

□ Le Big Data se caractérise par les 5V, voire plus :

- **Volume** : la quantité de données est massive, souvent de l'ordre de téraoctets, voire de péta octets.
- **Vélocité** : les données sont générées et collectées à un rythme effréné, parfois en temps réel.
- **Variété** : les données sont de types très différents, ce qui complexifie leur traitement.
- **Véracité** : la qualité et la fiabilité des données sont cruciales pour obtenir des résultats pertinents.
- **Valeur** : les données doivent être transformées en informations utiles pour l'entreprise.[5]

D_ Analyse des données en temps réel : surfer sur la vague de l'information

L'analyse des données en temps réel, c'est comme surfer sur une vague d'informations en mouvement constant. Il s'agit de traiter et d'analyser les données au moment précis de leur génération, sans attendre le moindre délai. Cette approche permet de détecter instantanément les anomalies et les tendances émergentes. Elle offre la possibilité de prendre des décisions éclairées et réactives. Les entreprises peuvent ainsi optimiser leurs processus et anticiper les problèmes potentiels. L'analyse en temps réel est essentielle pour les applications critiques, telles que la surveillance de la production ou la détection de fraudes. Elle transforme la manière dont les organisations utilisent les données pour gagner en agilité et en efficacité :

- **Réagir instantanément** : détecter une anomalie dans un processus de production et agir immédiatement pour la corriger.
- **Personnaliser l'expérience client** : proposer une offre promotionnelle à un client qui vient de visiter un site web.
- **Prendre des décisions éclairées** : baser ses décisions sur des informations à jour et pertinentes.

L'analyse en temps réel repose sur des technologies spécifiques, capables de traiter des flux de données importants et rapides :

- **Plateformes de streaming** : Apache Kafka, Apache Flink.
- **Bases de données NoSQL** : Cassandra, MongoDB.
- **Outils de visualisation** : Tableau, Kibana.[5]

E_ Intelligence artificielle

L'intelligence artificielle (IA) regroupe des techniques permettant aux machines de simuler l'intelligence humaine. Elle ne relève plus de la science-fiction, mais est une réalité tangible. L'IA transforme notre monde, des applications quotidiennes aux avancées industrielles. Elle permet aux machines d'apprendre, de raisonner et de prendre des décisions. L'IA ouvre la voie à de nouvelles possibilités dans de nombreux domaines. Son impact continue de croître, redéfinissant notre interaction avec la technologie :

- **Apprentissage automatique** : les machines apprennent à partir des données, sans être explicitement programmées. Elles peuvent ainsi identifier des motifs, faire des prédictions et prendre des décisions.
- **Apprentissage profond** : les machines utilisent des réseaux neuronaux complexes, inspirés du cerveau humain, pour analyser les données. C'est cette technique qui permet aux voitures autonomes de "voir" et de "comprendre" leur environnement.
- **Traitement du langage naturel** : les machines comprennent et interagissent avec le langage humain. C'est ce qui permet aux assistants virtuels comme Siri ou Alexa de nous répondre.
- **Vision par ordinateur** : les machines "voient" et interprètent les images. C'est ce qui permet aux systèmes de reconnaissance faciale de nous identifier.[5]

F_ L'IA au service du Big Data et de l'analyse en temps réel : un mariage puissant

L'intelligence artificielle (IA) joue un rôle crucial dans l'exploitation du Big Data. Elle permet d'analyser d'immenses volumes de données en un temps record. L'IA facilite la détection de tendances et d'anomalies en temps réel. Elle optimise ainsi l'analyse des données, rendant l'information exploitable. L'IA est un atout indispensable pour tirer pleinement parti du Big Data et de l'analyse en temps réel. Elle permet de :

- **Collecter et prétraiter les données** : l'IA peut identifier et extraire automatiquement les informations pertinentes à partir de sources de données variées, même les plus complexes.

- **Analyser les données** : l'IA peut identifier des schémas, des tendances et des anomalies dans les données, même les plus complexes.
- **Prendre des décisions** : l'IA peut prendre des décisions en temps réel en fonction des données analysées, ce qui est crucial dans de nombreux domaines (par exemple, la finance, l'industrie, la santé).
- **Automatiser les tâches** : l'IA peut automatiser des tâches répétitives et complexes, libérant ainsi les humains pour des tâches plus créatives et stratégiques.[5]

G_ cybersécurité

Le risque de cyber sécurité s'accroît avec la connectivité qui est au cœur de l'Industrie 4.0. En effet, le partage de données industrielles et la supervision à distance des systèmes de production exigent une intensification des mesures de sécurité. On peut utiliser des protocoles de communication sûrs et testés pour gérer les accès aux systèmes de commande des machines. Il est possible de citer certaines technologies prises en charge : VPN, P2P, PIR, TLS, DNSSEC et les étiquettes RFID [Weber & Weber, 2010].[1]

H_ Communication intermachines (M2M)

Au fur et à mesure que le nombre de systèmes et de machines autonomes augmente, les technologies de communication entre machines évoluent. Contrairement à l'Internet des objets (IoT), qui est lié à un réseau mondial, les technologies M2M (Machine-to-Machine) génèrent des réseaux locaux composés de machines autonomes. Le M2M repose sur des protocoles et des normes tels que TCP/IP, IGRS, UPnP, DLNA et OPC-UA.

I_ Simulation

la simulation soit déjà présente dans divers domaines de l'ingénierie, pour l'Industrie 4.0, elle vise à reproduire toutes les opérations de production. Ainsi, les jumeaux numériques représentent un reflet virtuel du monde physique, englobant les machines et les indicateurs associés (tels que les vitesses, les forces...), ainsi que les produits et les individus (Grievens, 2014). L'objectif de la simulation est alors de vérifier et d'améliorer principalement la configuration des machines et les flux avant d'appliquer ces modifications dans la réalité. Cela contribue à réduire les temps de mise au point et à accroître

J_ Réalité augmentée

Les dispositifs de réalité augmentée projettent des informations sur le monde réel en temps réel. Il est ainsi possible d'ajouter et de visualiser virtuellement un meuble issu d'un catalogue dans son propre salon à l'aide d'une tablette, comme le propose Ikea.

Dans le cadre de l'Industrie 4.0, la réalité augmentée facilite la transmission de directives pour l'assemblage ou la réparation, en fonction du contexte actuel. Elle permet également d'assister les conducteurs de chariots élévateurs en les guidant, en temps réel, vers la palette à saisir au sein d'un entrepôt.[1]

1.4 Éléments des systèmes industriels modernes

1.4.1 Composants matériels : Fondements et Structure des Systèmes Automatisés

A_ Matières d'œuvre

Un matériau peut se présenter sous différentes formes. Il peut être :

- **Un produit** : correspondant à une substance solide, liquide ou gazeuse, à un état plus ou moins transformé. Il peut inclure :
 - **Éléments techniques** : barre, roulement, moteur, véhicule...
 - **Produits chimiques** : pétrole, éthylène, matière plastique...
 - **Produits textiles** : fibre, tissu, vêtement...
 - **Produits électroniques** : transistor, puce, microprocesseur, automate programmable...
 - **Une source d'énergie** : disponible sous différentes formes, telles que l'énergie électrique, thermique ou hydraulique.
- **Une information** : exprimée sous divers formats, notamment écrit, physique, audiovisuel, etc.
- **Des individus** : considérés de manière individuelle ou collective

B_ Valeur ajoutée

L'objectif fondamental de ce système, depuis sa conception jusqu'à ses éventuelles modifications, est de conférer une valeur ajoutée aux matières premières. Cette valeur peut être générée par une altération physique des matériaux utilisés, notamment à travers :

- **Un traitement mécanique** : tournage, façonnage, broyage, impression, etc.
- **Un procédé chimique ou biologique** : transformation de la matière première.
- **Une conversion énergétique** : transformation d'une forme d'énergie en une autre.
- **Un traitement thermique** : cuisson, congélation, etc.
- **Une finition en surface** : application de peinture, teinture, etc.

C_ Automatisation

L'automatisation de la production implique le transfert total ou partiel des tâches de coordination, précédemment réalisées par des opérateurs humains, vers un système intégré, d'éléments techniques désignés sous le nom de PARTIE COMMANDE. La Partie Commande enregistre l'EXPERTISE des opérateurs pour déterminer la série d'actions à entreprendre sur les matières premières afin de créer la valeur ajoutée. Elle utilise un ensemble de données extraites de la Partie Opérative pour Établir la séquence d'instructions requises pour réaliser les actions désirées.

□ Objectifs de l'automatisation

L'automatisation offre des compléments à la valeur ajoutée produite par le système. On peut exprimer ces éléments en termes d'objectifs par :

- Améliorer la productivité du système, c'est-à-dire accroître la quantité de produits fabriqués sur une période déterminée. Cet accroissement de productivité traduit un ajout de valeur
- Optimiser la rentabilité,
- Accroître la compétitivité.
- Renforcer la flexibilité de production ;
- Améliorer la qualité du produit en assurant une répétabilité supérieure de sa valeur ajoutée.
- s'ajuster à des situations spécifiques -Adaptation à des milieux inhospitaliers pour l'être humain (environnement salin, spatial, etc.). Nucléaire...),
- Adaptation à des tâches ardues, que ce soit sur le plan physique ou intellectuel (manipulation de charges lourdes, travaux répétitifs parallèles...),
- améliorer la sécurité, etc.
- D'autres buts, de nature sociale, financière...

□ Composition d'un système automatisé

Tout système automatisé comprend :

- **une section opérationnelle (S.O.)** qui s'occupe du traitement des matières premières pour de créer la valeur ajoutée
- **une partie commande (P.C.)** qui coordonne la séquence des opérations sur la Partie Opérative. dans le but de réaliser cette valeur ajoutée.

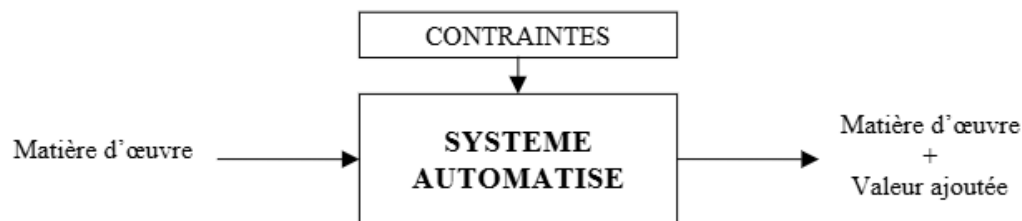


FIG. 1.3 : Schéma général d'un système automatisé [6]

Tout système automatisé est en pleine interaction : en tenant compte du contexte ou de l'environnement physique et humain externe au système

La Partie de Contrôle d'un système autonome est un regroupement d'éléments et de composants pour le traitement de l'information, prévu pour :

- à orchestrer le déroulement des actions sur la Partie Opérative,
- en veillant à son fonctionnement optimal,
- à orchestrer les échanges avec les parties prenantes,
- pour administrer les échanges avec d'autres systèmes,
- en charge du traitement des informations et des résultats liés au processus, aux matériaux utilisés, aux durées de production, à la consommation d'énergie... (gestion technique) [6]

D_ API

Un automate programmable est un dispositif conçu pour superviser une machine ou un processus industriel, composé d'éléments électroniques, possédant une mémoire que l'utilisateur non spécialiste en informatique peut programmer à l'aide d'un langage spécifique. Autrement dit, un automate programmable est une sorte de calculateur logique, ou ordinateur, doté d'un jeu d'instructions délibérément restreint, conçu pour la gestion et le contrôle en temps réel de processus industriels. Trois caractéristiques fondamentales distinguent totalement l'Automate Programmable Industriel (API) des ordinateurs (PC industriel ou autres)

Grâce à ses interfaces industrielles, il peut être directement relié aux capteurs et pré-actionneurs.

est conçu pour opérer dans des environnements industriels hostiles (températures élevées, vibrations, coupures momentanées de l'alimentation, interférences, etc.) et sa programmation utilise des langages spécialement adaptés au traitement des fonctions d'automatisme qui fait en sorte que sa mise en oeuvre et son exploitation ne nécessitent aucune connaissance en informatique

□ Structure d'un système automatisé

Tout système automatisé peut se décomposer selon le schéma ci-dessous :

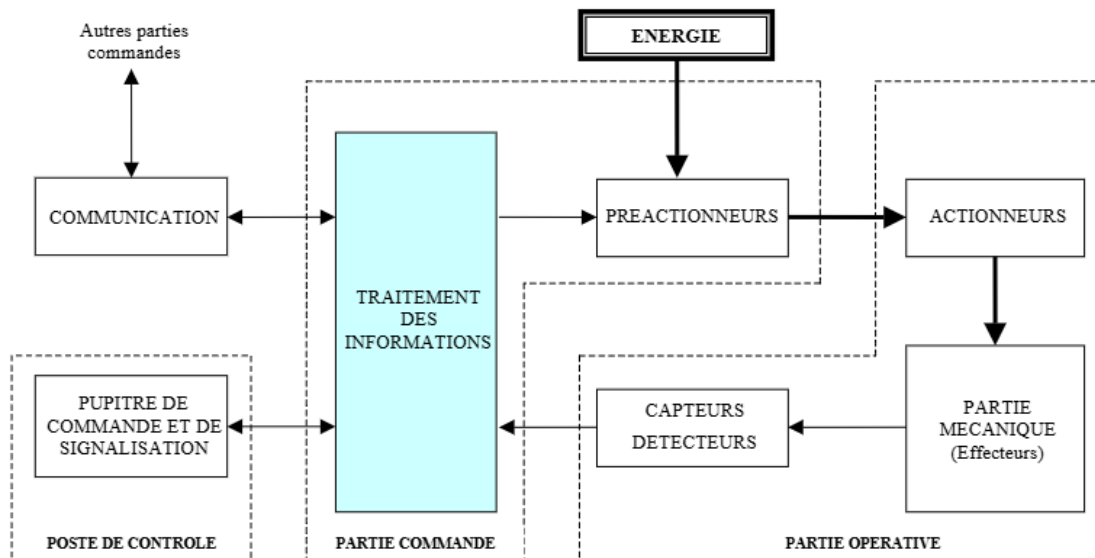


FIG. 1.4 : Architecture fonctionnelle d'un système automatisé. [7]

- **Partie opérationnelle :**

La partie opérative intervient directement sur le matériau de base afin de lui conférer une valeur ajoutée. Ce processus repose sur plusieurs composants essentiels qui assurent la transformation de la matière d'œuvre. Les actionneurs, tels que les moteurs et les vérins, jouent un rôle central dans la composante mécanique du système en exerçant une action physique sur le matériau. En parallèle, les détecteurs et capteurs permettent de surveiller et d'analyser les différents états du système, garantissant ainsi un fonctionnement optimal et une adaptation en temps réel aux conditions de production.

- **Partie commande :**

Elle transmet les instructions opérationnelles à la section exécutive. Les préactionneurs offrent la possibilité de contrôler les actionneurs ; ils garantissent le passage d'énergie entre la source de puissance (réseau électrique, pneumatique...) et les actionneurs. Paraphrase : contacteur, distributeur... Le bloc de traitement des informations, à son tour, contrôle ces préactionneurs. Il reçoit les instructions du tableau de bord (opérateur) ainsi que les données de la section opérative fournies par les capteurs/détecteurs. Selon ces instructions et son programme de gestion des tâches (intégré dans un automate programmable ou réalisé par des relais, ce que l'on appelle la logique câblée), Elle se chargera de l'acquisition des préactionneurs et de la transmission d'informations au panneau de signalisation ou à d'autres dispositifs de commande et/ou de supervision, en employant un réseau ainsi qu'un protocole de communication.

- **Poste de contrôle :**

Il offre à l'opérateur la possibilité de contrôler le système (démarrer, arrêter, débiter un cycle...) grâce aux panneaux de commande et de signalisation qui le composent. Il offre aussi la possibilité d'observer les divers états du système par le biais de témoins lumineux, de consoles de communication ou d'interfaces utilisateur.

□ Applications des automates

Les API sont employées dans divers domaines industriels pour la gestion des équipements (convoyeurs, emballages...) ou des lignes de production (automobile, agroalimentaire...), tout autant qu'elles peuvent garantir des fonctions de contrôle de processus (métallurgie, chimie...). L'automate programmable est de plus en plus utilisé dans le secteur de la construction, qu'il soit tertiaire ou industriel, notamment pour la supervision du chauffage, de l'éclairage, de la sécurité ou encore des systèmes d'alarme.

Il traite différents types d'informations :

- **Tout ou rien (T.O.R.) :** une information binaire qui ne peut prendre que deux états (vrai/faux, 0 ou 1). Ce type de signal provient généralement de dispositifs comme des détecteurs ou des boutons poussoirs...
- **Analogique :** une information continue dont la valeur se situe dans une plage définie. Elle est généralement issue de capteurs mesurant des grandeurs physiques telles que la pression ou la température...
- **Numérique :** une information encodée sous forme binaire ou hexadécimale, transmise par des ordinateurs ou des modules intelligents.

□ Architecture des automates

- **Aspect extérieur**

Les automates se divisent en deux grandes catégories : compacts et modulaires. Parmi les modèles compacts, on distingue les modules de programmation tels que le LOGO (Siemens), le ZELIO (Schneider) ou le MILLENIUM

(Crouzet), des microautomates. Il comprend le processeur, l'alimentation électrique, ainsi que les entrées et sorties. En fonction des modèles et des producteurs, il pourrait accomplir certaines tâches additionnelles (rapidité de comptage, entrées/sorties analogiques...) et accepter un nombre restreint d'extensions. Ces machines, dotées d'un fonctionnement simplifié, sont habituellement conçues pour la Commande de petits automates.

Le processeur, l'alimentation et les interfaces d'entrée/sortie sont disposés dans des unités distinctes (modules) et sont montés sur un ou plusieurs supports contenant le « fond de panier » (bus et connecteurs). Ces automates sont incorporés dans des systèmes automatisés complexes où la puissance, la capacité de traitement et la flexibilité sont indispensables.

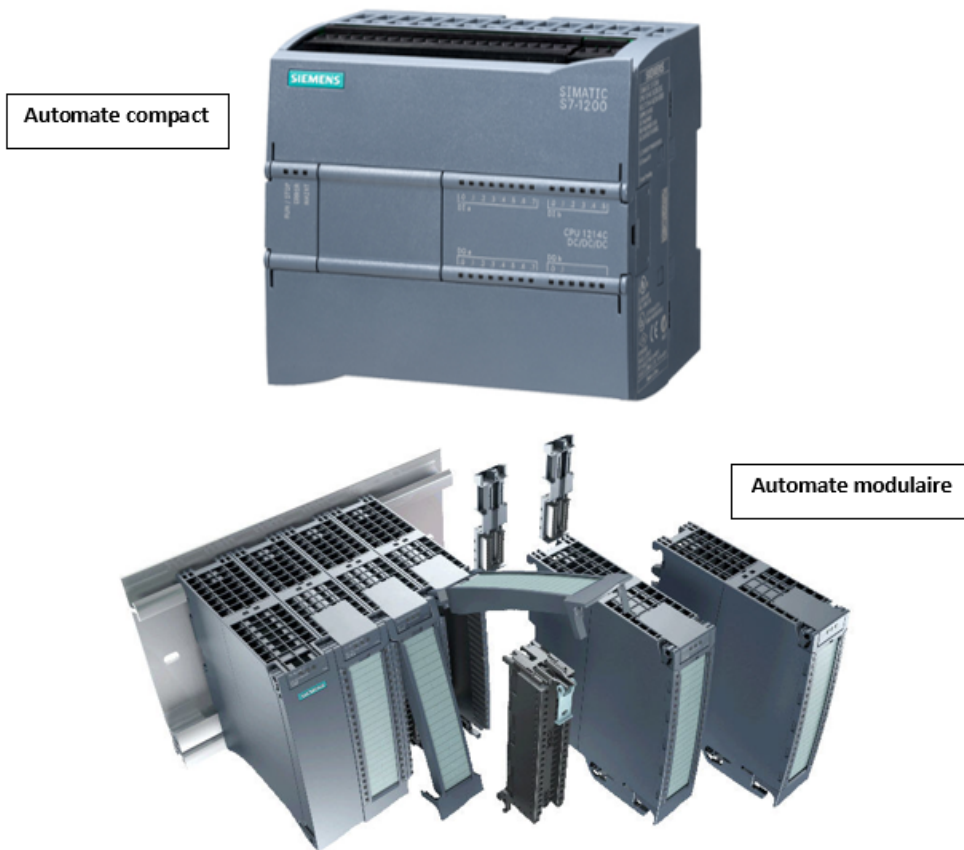


FIG. 1.5 : Types d'automates programmables (compact et modulaire) [8] [9]

- **Structure interne**

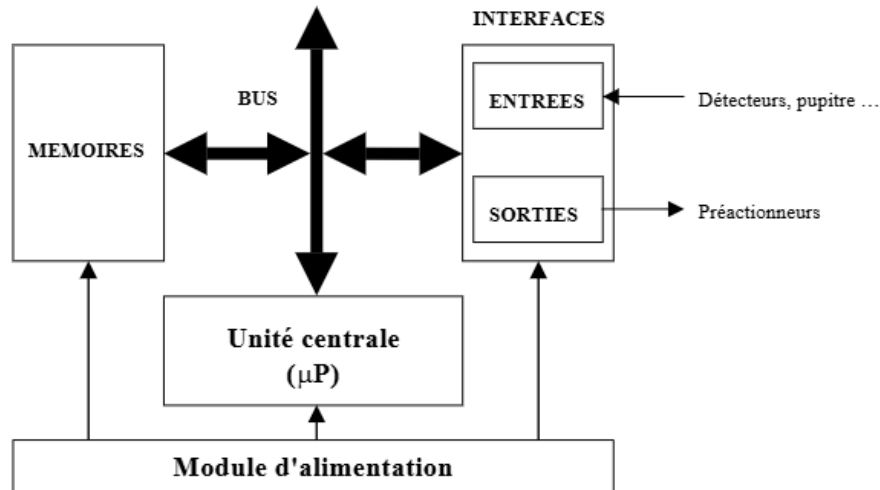


FIG. 1.6 : Structure interne d'un automate programmable [7]

- **Module d'alimentation** : il est chargé de fournir de l'énergie aux différents modules.
- **Unité centrale** : dotée d'un microprocesseur, elle exécute toutes les opérations logiques, arithmétiques et numériques (transfert, comptage, temporisation...).
- **Le bus interne** : il assure la communication entre tous les composants de l'automate ainsi que les extensions potentielles. Mémoires : elles sont utilisées pour la mémorisation le système d'exploitation (ROM ou PROM), le programme (EEPROM) ainsi que les données système pendant l'exécution (RAM). Cette dernière est généralement alimentée par une pile ou une batterie. Il est habituellement possible d'accroître la capacité mémoire en ajoutant des barrettes de type PCMCIA.
- **Interfaces d'entrées / sorties**
 - * Interface d'entrée : elle a pour fonction de collecter les données du S.A.P. ou du pupitre, de traiter (filtrer,...) ce signal tout en garantissant son isolement électrique. (Optocouplage).
 - * Interface de sortie : elle offre la possibilité de piloter les différents préactionneurs et composants de signalisation du S.A.P. tout en garantissant l'isolement électrique

☐ Fonctions réalisées

Les automates compacts sont capables de piloter des sorties tout ou rien (T.O.R.) et intègrent parfois des fonctions de comptage ou de traitement analogique. En revanche, les automates modulaires offrent une plus grande flexibilité grâce à l'ajout de modules intelligents montés sur un ou plusieurs racks. Ces modules présentent l'avantage de soulager la CPU principale, car ils intègrent généralement leur propre processeur.

☐ Traitement du programme automate

Tous les automates suivent un mode de fonctionnement identique :

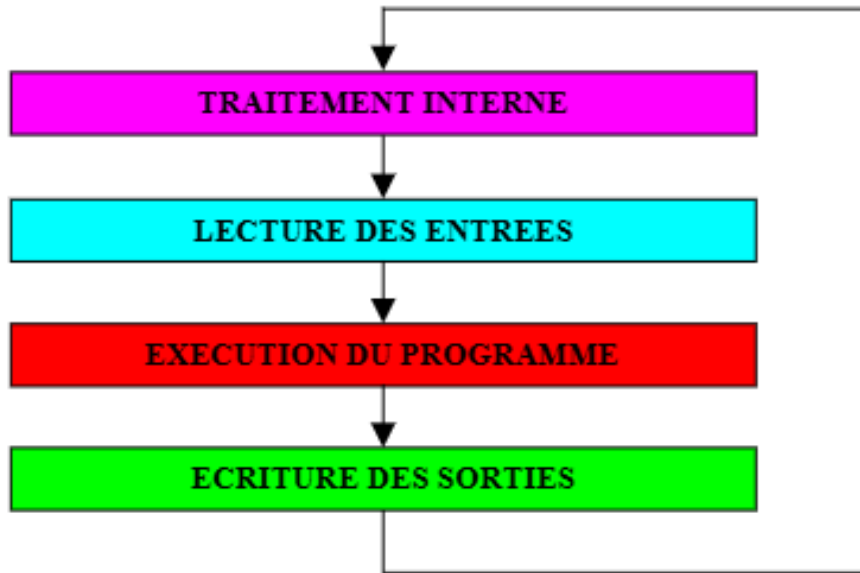


FIG. 1.7 : Cycle de fonctionnement d'un automate programmable[7]

- **Traitement interne** : L'automate réalise des actions de vérification et actualise certains réglages système (identification des transitions RUN / STOP, réactualisation des valeurs de l'horodateur, ...).
- **Lecture les entrées** : L'automate procède à la lecture des données (de manière synchronisée) et les enregistre dans la mémoire d'image des entrées.
- **Exécution du programme** : le dispositif exécute chaque instruction successivement et stocke les résultats dans la mémoire image des sorties
- **Ecriture des sorties** : L'automate commute les diverses sorties (de manière synchronisée) vers les positions déterminées dans la mémoire d'image des sorties.

L'automate exécute constamment ces quatre actions (opération cyclique).

La scrutation désigne l'ensemble des quatre opérations exécutées par l'automate, et le temps de scrutation correspond à la durée nécessaire pour que l'automate traite la même portion du programme. Pour les applications courantes, ce délai est généralement de quelques dizaines de millisecondes.

□ Critères de choix d'un API

Le choix d'un automate programmable repose avant tout sur l'orientation stratégique d'une entreprise ou d'un groupe industriel. L'expérience acquise avec un fabricant, ainsi que les relations commerciales établies, constituent un point de départ essentiel dans cette sélection. Les grandes entreprises optent généralement pour deux fournisseurs afin de favoriser la concurrence et de disposer d'une alternative en cas de déclin de l'un d'entre eux.

La formation du personnel de maintenance est un élément déterminant, car une trop grande diversité de matériels peut entraîner des difficultés d'exploitation et de maintenance. Il est préférable d'opter pour un automate utilisant un langage de programmation standardisé tel que le GRAFCET, afin de faciliter la mise au point des systèmes et les interventions en cas de panne.

L'acquisition d'un logiciel de programmation adapté constitue également un levier d'optimisation, tant en termes de coûts (achat du logiciel, formation du personnel) que d'efficacité. De plus, l'intégration d'outils de simulation permet d'anticiper les erreurs et d'optimiser le développement des programmes.

Le choix d'un automate doit être guidé par une évaluation précise des besoins techniques, notamment :

- **Nombre d'entrées/sorties (E/S) :** Un nombre élevé de cartes d'entrées/sorties peut influencer la nécessité d'ajouter des racks d'extension, impactant ainsi l'architecture du système.
- **Type de processeur :** La capacité mémoire, la vitesse de traitement et les fonctionnalités spécifiques du processeur doivent être prises en compte pour garantir une réponse optimale aux exigences du processus industriel.
- **Modules et fonctions spécifiques :** Certains modules spécialisés (commande d'axe, pesage, etc.) permettent de décharger le processeur principal et doivent offrir les caractéristiques adaptées aux besoins de précision et de performance.
- **Fonctions de communication :** L'automate doit être en mesure de communiquer avec d'autres systèmes de contrôle, tels que d'autres automates programmables industriels (API) ou des logiciels de supervision. Il est essentiel qu'il intègre des protocoles de communication standardisés (comme Profibus) pour assurer une compatibilité optimale avec les équipements existants[7]

E_ Les capteurs :

Un capteur est un instrument qui transforme une mesure physique en une mesure électrique quantifiable. Lorsqu'il est exposé à l'influence d'un mesurande qui n'est pas électrique, il produit une réaction électrique sous forme de charge, tension, courant ou impédance. Cette réponse, notée s , est fonction du mesurande m , selon la relation : $s=f(m)$ où :

- s représente la grandeur de sortie ou la réponse du capteur (charge, tension, courant, impédance).
- m correspond à la grandeur d'entrée ou excitation appliquée au capteur.[10]

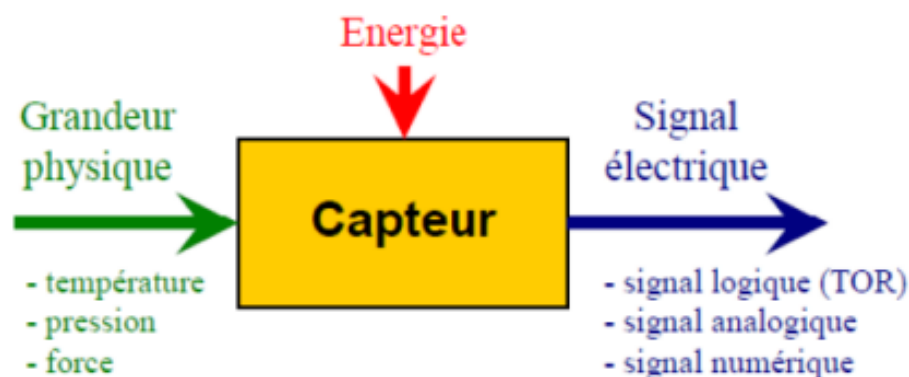


FIG. 1.8 : Principe de fonctionnement d'un capteur.[10]

□ capteur actif

Un capteur actif, qui fonctionne comme un générateur, se base généralement sur un phénomène physique garantissant la transformation d'une forme d'énergie en énergie électrique. Cette énergie peut provenir de sources thermiques, mécaniques ou d'un rayonnement. Les effets physiques fréquemment observés en instrumentation sont :

- **Effet thermoélectrique** : Un circuit composé de deux conducteurs chimiquement distincts, avec des jonctions à des températures T_1 et T_2 , est à l'origine d'une force électromotrice d'origine thermique $e(T_1, T_2)$.
- **Effet piézoélectrique** : L'application d'une contrainte mécanique à certains matériaux dits piézoélectriques (comme le quartz) entraîne une déformation ainsi que l'apparition de charges électriques de signes opposés sur les faces opposées du matériau.
- **Effet d'induction électromagnétique** : La variation du flux d'induction magnétique dans un circuit électrique induit une tension électrique (utilisée notamment pour la détection du passage d'un objet métallique)
- **Effet photoélectrique** : La libération de charges électriques au sein d'une substance sous l'action d'une lumière ou, plus largement, d'une onde électromagnétique. Effet Hall : L'interaction entre un champ magnétique B et un courant électrique I provoque une différence de potentiel U_h dans le matériau
- **Effet photovoltaïque** : L'éclairement d'une jonction PN engendre la libération d'électrons et de trous, dont le mouvement entraîne une variation de la tension à ses bornes.[11]

Grandeur physique Mesurée	Effet utilisé	Grandeur de sortie
Température	Thermoélectricité	Tension
Flux de rayonnement Optique	Photoémission Effet photovoltaïque Effet photo-électrique	Courant Tension Tension
Force Pression	Piézo-électricité	Charge électrique
Accélération Vitesse	Induction électromagnétique	Tension
Position (Aimant) Courant	Effet Hall	Tension

TAB. 1.1 : Grandeurs d'entrée et de sortie et effet utilisé pour les capteurs actifs [11]

□ capteur passifs

Un détecteur passif est incorporé dans un circuit électrique qui reçoit de l'énergie. C'est un élément dont l'un des paramètres électriques (tels que la résistance, l'inductance ou la capacité) est réactif à la mesure effectuée. Soit la variation d'impédance découle d'une :

- **Changement des dimensions du capteur** : Mécanisme opérationnel de nombreux capteurs.
- **Déformation** : Résultant d'une force ou d'une grandeur physique associée (pression, accélération). Exemples : Une armature de condensateur soumise à une variation de pression, ou une jauge d'extensométrie fixée à une structure déformable.

□ Types de capteurs passifs

- **Capteurs de position** : Un détecteur de position est un appareil qui collecte des données concernant la localisation et le déplacement d'un objet. On peut obtenir ces informations :
 - Par interaction directe avec l'objet
 - À distance, en utilisant des détecteurs magnétiques.
- **Potentiomètre** : Un potentiomètre est un type de résistance ajustable à trois connexions, dont l'une est connectée à un curseur qui glisse sur une surface résistante. Ce dispositif facilite la collecte, entre le point de connexion au curseur et l'un des autres points de connexion, d'une tension qui correspond à la position du curseur et à la tension appliquée à la résistance.
- **Inductance à noyau mobile** : L'inductance à noyau mobile offre la possibilité de mesurer l'inductance et l'auto-induction d'une bobine alimentée en fonction de la position du noyau ferreux inséré. Elle sert à l'analyse des circuits en courant alternatif. Elle est généralement constituée de :
 - Une bobine de cuivre contenue dans un boîtier en plastique résistant aux chocs muni de poignées.
 - Un noyau mobile en fer est placé dans la bobine grâce à une vis à broche.
 - Un système gradué en centimètres est utilisé pour déterminer la longueur du noyau inséré dans La bobine.
- **Condensateur à armature mobile** : Un composant, appelé condensateur à armature mobile, est employé en radio pour obtenir une capacité variable. Il se compose de :
 - Deux armatures sont présentes, l'une d'elles étant mobile (R : le rotor) et contrôlée par un axe (A), alors que l'une est fixe (S : le stator).

Sa capacité fluctue constamment entre un niveau minimal (capacité résiduelle) et un niveau maximal (capacité nominale), en suivant une loi de variation précise. Dans les circuits oscillateurs des radioamateurs (comme les oscillateurs, amplificateurs, boîtes d'accord, etc.), l'utilisation principale se fait avec des condensateurs variables.[10]

F_ Les actionneurs

Un actionneur est un élément constitutif de la composante opérationnelle d'un système automatisé. Sa fonction est de convertir l'énergie qu'il reçoit en un phénomène physique qui facilite l'exécution d'une tâche, le changement de comportement ou l'altération de l'état d'un système. On identifie principalement trois catégories d'actionneurs :

- **Actionneur électrique** : Un actionneur électrique exploite l'énergie électrique afin de générer un mouvement. Il est généralement constitué d'un moteur électrique (qu'il soit à courant continu ou alternatif) et peut être contrôlé par un automate ou un microcontrôleur.
- **Actionneur hydraulique** : Un actionneur hydraulique emploie un fluide pressurisé (huile ou autre fluide) pour produire un mouvement linéaire ou rotatif. Il comprend une pompe hydraulique, des vérins et des vannes de régulation.
- **Actionneur pneumatique** : Un actionneur pneumatique opère à l'aide d'un gaz pressurisé (généralement de l'air comprimé). Il se compose essentiellement de vérins pneumatiques, de compresseurs et de vannes de régulation.

G_ Pré-actionneur

La distribution de l'énergie du pré-actionneur vers l'actionneur se fait sur ordre de la partie commande (A.P.I).

□ Types de pré-actionneurs selon l'énergie

Type d'actionneur	Énergie utilisée	Avantages	Inconvénients
Électrique	Électricité	Précis, facile à contrôler, peu d'entretien	Moins puissant, sensible aux pannes électriques
Hydraulique	Fluide sous pression	Très puissant, robuste	Entretien fréquent, fuites possibles
Pneumatique	Air comprimé	Rapide, économique, sûr	Moins puissant, bruit du compresseur

TAB. 1.2 : comparaison des types d'actionneur

- **Pré-actionneurs électriques** : Ils assurent la distribution de l'énergie électrique de puissance aux actionneurs électriques. Les circuits électriques utilisent généralement des relais ou des contacteurs comme pré-actionneurs.
- **Pré-actionneurs pneumatiques** : Ils ont pour fonction essentielle de distribuer l'air sous pression aux différents orifices des actionneurs pneumatiques. En général, le pré-actionneur pneumatique est un distributeur associé à un vérin pneumatique.
- **Pré-actionneurs hydrauliques** : Ils jouent le rôle d'interface entre le circuit et les actionneurs. Leur fonction principale est assurer l'orientation du fluide hydraulique vers les différents actionneurs tout en facilitant son retour vers le réservoir. Les pré-actionneurs hydrauliques sont généralement des distributeurs associés à des actionneurs hydrauliques.[12]

H_ réseau industriel

□ le réseau local industriel (RLI)

Un RLI est une technologie de communication qui permet l'échange d'informations entre divers équipements industriels (tels que capteurs, actionneurs, automates, etc.) dans une zone géographique délimitée. Bien que parfois appelés à tort réseaux de terrain, les réseaux locaux industriels se positionnent à différents niveaux de la pyramide CIM (Computer Integrated Manufacturing). On recense plus de 2000 types de bus de terrain distincts, mais parmi ceux-ci, certaines technologies sont particulièrement courantes : Modbus, Profibus-FMS/PA, Profibus-DP Interbus-S, ASI, WorldFIP, Lonworks et bus CAN.

□ Architecture d'un réseau local industriel

- **Les réseaux de terrain** : Les réseaux de terrain englobent et relient une multitude de nœuds, tels que des détecteurs, des actionneurs, des automates, des régulateurs, ainsi que tout équipement supportant des applications qui requièrent l'accès à des dispositifs de terrain. Ils peuvent également être reliés à travers différents réseaux locaux.
- **Les réseaux d'atelier ou de cellule** : Les réseaux d'atelier englobent et relient un nombre moins conséquent de dispositifs tels que les commandes de robots, de machines-outils, ou encore les contrôles de qualité (lasers). On trouve surtout ces réseaux dans le secteur de la fabrication.

□ Classement des réseaux en fonction de leur type

- **Réseaux homogènes** : Proposé par le producteur de l'équipement d'automatisation. Par exemple, dans le cas des automates télémechaniques, seules leurs capteurs et actionneurs peuvent être intégrés pour automatiser un processus.
- **Réseaux hétérogène** : Les réseaux hétérogènes sont des réseaux multi-fournisseurs, où l'on peut observer une variété d'automates programmables (Schneider, Siemens, Omron,...), de types d'entrées et sorties ainsi que de protocoles de communication (Profibus, Ethernet, Modbus,...). À l'heure actuelle, la plupart des réseaux sont de nature hétérogène.[13]

1.4.2 Composants logiciels

A_ Langages de programmation

la norme internationale CEI 61131-3 définit quatre langages normalisés pour la programmation des automates.

Chaque automate est programmé soit à l'aide d'une console spécifique au fabricant, soit via un ordinateur équipé du logiciel fourni par le constructeur.

□ Liste d'instructions (IL : Instruction list)

Il s'agit d'un langage textuel similaire à l'assembleur utilisé pour la programmation des microcontrôleurs.

mais il est rarement employé par les automaticiens.

```
! %L0: LD      %I1.0
      ANDN   %M12
      OR (   %TM4.Q
      AND   %M17
      )
      AND   %I1.7
      ST    %Q2.5
! %L5: LD      %I1.10
      ANDN   %Q2.3
      ANDN   %M27
      IN     %TM0
      LD     %TM0.Q
      AND    %M25
      AND    %M00.X5
      [%M00.S := %M00.S+500]
```

FIG. 1.9 : Exemple de programme en langage IL (Instruction List)[14]

□ Langage littéral structuré (ST : Structured Text)

C'est un langage informatique de type Pascal, basé sur l'utilisation de structures conditionnelles telles que (if... then... else si... alors... sinon...).

Il est peu utilisé par les automaticiens

```

IF %M0 THEN
  FOR %MOV99 := 0 TO $1 DO
    IF %MOV100 [%MOV99] > 0 THEN
      %MOV10 := %MOV100 [%MOV99];
      %MOV11 := %MOV99;
      %M1 := TRUE;
      EXT;          (*Sortie de la boucle FOR*)
    ELSE
      %M1 := FALSE;
    END_IF;
  END_FOR;
ELSE
  %M1 := FALSE;
END_IF;

```

FIG. 1.10 : Programme en langage ST (Structured Text)[14]

□ Langage à contacts (LD : Ladder diagram)

Langage graphique initialement conçu pour les électriciens, il repose sur l'utilisation de symboles tels que les contacts, relais et blocs fonctionnels, organisés sous forme de réseaux(labels).

Il s'agit du langage le plus couramment utilisé.

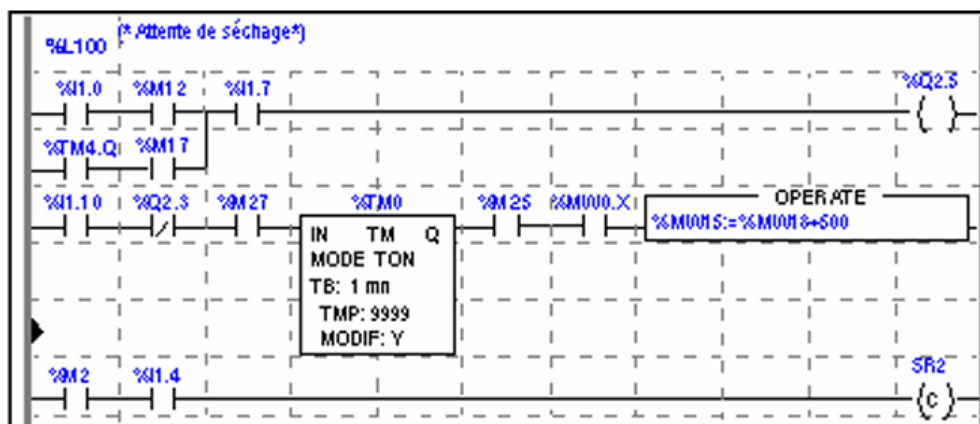


FIG. 1.11 : Programme en langage Ladder (LD)[14]

□ Blocs Fonctionnels (FBD : Function Bloc Diagram)

Langage graphique dans lequel les fonctions sont représentées par des rectangles, avec les entrées situées à gauche et les sorties à droite. Les blocs peuvent être préprogrammés (bibliothèque) ou programmables

Utilisé par les automaticiens[14]

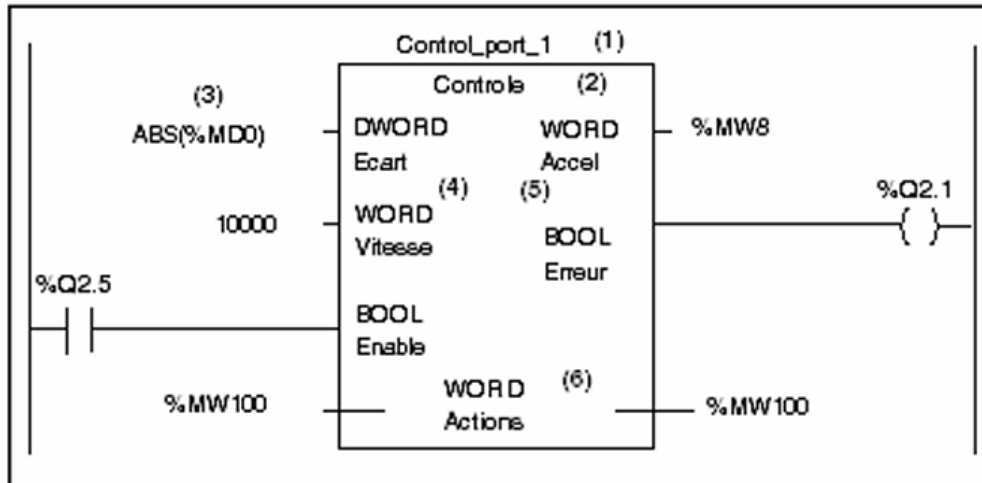


FIG. 1.12 : Exemple de bloc fonctionnel en langage FBD (Function Block Diagram)[14]

□ GRAFCET

Le GRAFCET (Steps/Transitions Command Functional Graphic) est un langage graphique standard utilisé pour décrire le comportement des systèmes automatisés, notamment ceux qui fonctionnent de manière séquentielle. Il modifie les différentes étapes d'un processus, les transitions entre ces étapes et les conditions associées, fournissant ainsi une représentation claire et structurée du fonctionnement de l'automatisation industrielle.[15]

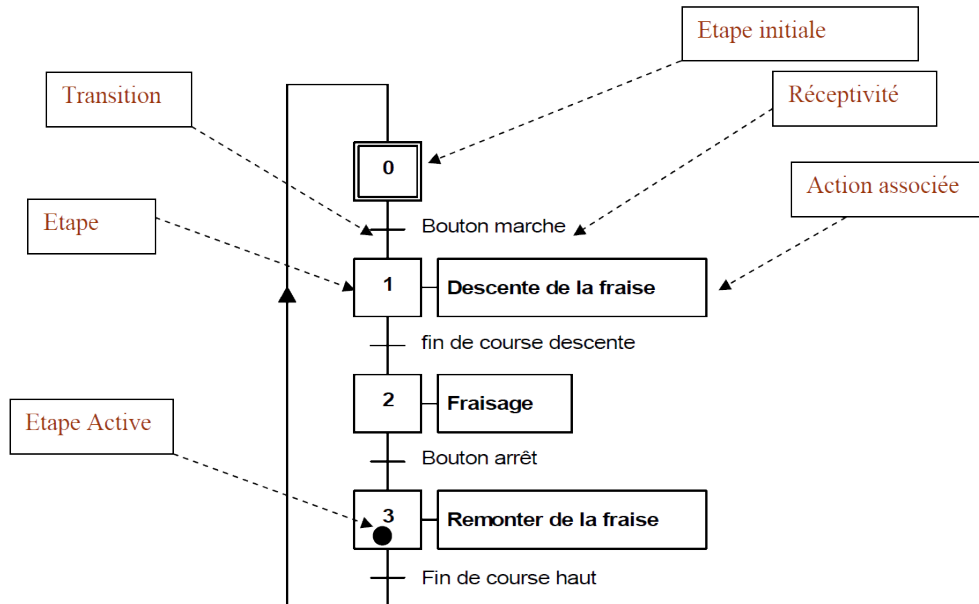


FIG. 1.13 : structure d'un GRAFCET[16]

B_ Simulation

Les outils de simulation des systèmes automatisés ont évolué grâce aux avancées technologiques et à la démocratisation des logiciels performants. Ils jouent un rôle clé dans la mise au point des programmes d'automates programmables

industriels (API), la formation des opérateurs et l'optimisation des processus industriels.

La simulation permet d'anticiper et de corriger les erreurs avant la mise en service réelle, réduisant ainsi les coûts et les délais d'installation. Elle repose sur plusieurs composants essentiels : un moteur de rendu pour la visualisation, un moteur physique pour simuler les lois mécaniques, et un module de dialogue permettant aux automates, réels ou virtuels, de piloter le système simulé.

Grâce à ces outils, les entreprises peuvent optimiser le développement de leurs systèmes automatisés tout en améliorant la sécurité et la performance des équipements industriels.

Exemples PLCSimulate (Siemens), PProSim Studio (Schneider Electric), Emulate 3D (Rockwell Automation).[17]

C_ Outils de supervision Industrielle

La supervision industrielle est un ensemble de techniques et de systèmes permettant de surveiller, contrôler et optimiser les processus industriels en temps réel. Elle repose principalement sur les systèmes SCADA (Supervisory Control and Data Acquisition), qui assurent la collecte, l'analyse et l'affichage des données issues des équipements industriels, tels que les capteurs et les automates programmables.

L'objectif principal de la supervision est d'assurer le bon fonctionnement des installations industrielles en fournissant aux opérateurs une visibilité complète sur les processus. Grâce à ces systèmes, il est possible d'anticiper les anomalies, d'améliorer la productivité et de réduire les temps d'arrêt des équipements.

□ Principaux Logiciels de Supervision

Divers outils SCADA sont disponibles sur le marché, chacun ayant des spécificités adaptées aux exigences industrielles. Parmi les plus utilisés, on retrouve :

- **Wonderware (AVEVA System Platform)** : Cette plateforme modulaire permet une gestion centralisée des équipements industriels et est largement utilisée dans des secteurs comme l'énergie, l'agroalimentaire et l'automobile. Elle offre une compatibilité avancée avec les bases de données et le cloud.
- **Siemens WinCC** : Développé par Siemens, ce logiciel s'intègre parfaitement dans l'écosystème TIA Portal. Il est reconnu pour son interface intuitive, ses capacités avancées de gestion des alarmes et des tendances, ainsi que sa compatibilité avec les équipements industriels Siemens.
- **Ignition (Inductive Automation)** : Ce logiciel se distingue par son architecture basée sur le web et son approche modulaire. Il repose sur des technologies modernes comme HTML5, SQL et MQTT, facilitant ainsi son intégration avec des solutions cloud et IIoT.
- **GE iFIX** : Développé par General Electric, il est apprécié pour sa robustesse et ses capacités avancées d'analyse et de reporting. Il est couramment utilisé dans les secteurs de l'énergie, des transports et des infrastructures nécessitant une supervision en temps réel.
- **Schneider Electric EcoStruxure SCADA** : Intégré à l'écosystème EcoStruxure, ce logiciel est idéal pour la gestion intelligente des infrastructures, en particulier dans les secteurs de l'eau, des bâtiments intelligents et des réseaux électriques.

Grâce à ces outils, les entreprises peuvent surveiller les performances de leurs machines, gérer les alarmes et événements critiques, commander des équipements à distance et analyser les données collectées pour optimiser leurs processus industriels.[18]

1.5 Conclusion

L'Industrie 4.0 transforme le secteur industriel en intégrant des technologies avancées comme l'intelligence artificielle, l'IoT, le cloud computing et le Big Data. La supervision industrielle, à travers les systèmes SCADA, joue un rôle clé en permettant de surveiller, contrôler et optimiser les opérations en temps réel.

Des solutions comme Wonderware, Siemens WinCC, Ignition, GE iFIX et Schneider Electric EcoStruxure SCADA facilitent la gestion efficace des infrastructures industrielles et l'amélioration des performances. L'IIoT et les systèmes cyberphysiques (CPS) renforcent l'intelligence des machines et favorisent une production plus autonome et flexible.

Cependant, cette transformation pose des défis en matière de cybersécurité, d'interopérabilité et de formation. L'industrie doit s'adapter pour exploiter pleinement ces innovations tout en garantissant la sécurité et l'efficacité des infrastructures.

CHAPITRE 2

FACTORY I/O : PRÉSENTATION, MODÉLISATION ET PROGRAMMATION AVEC CONTRÔLE I/O DU SYSTÈME DE PRODUCTION

2.1 Introduction

Dans un contexte industriel en constante évolution, l'optimisation des processus de production et la maîtrise des systèmes automatisés deviennent des compétences incontournables pour les ingénieurs et techniciens. Ce chapitre s'inscrit dans cette dynamique en explorant l'environnement de simulation Factory I/O, un outil pédagogique puissant pour la modélisation, la conception et le test de systèmes automatisés. Ce logiciel permet de reproduire avec un grand réalisme les différentes étapes d'un processus industriel, de la distribution des matières premières à l'assemblage final, en passant par l'usinage robotisé. Dans ce cadre, un exemple appliqué a été mis en œuvre et brièvement présenté, illustrant concrètement le fonctionnement d'une ligne de production automatisée. L'objectif est de présenter en détail les fonctionnalités de Factory I/O, son interface, les composants utilisés, ainsi que l'intégration avec le logiciel Control I/O, afin de démontrer la pertinence de la simulation dans la formation et l'optimisation des systèmes industriels.

2.2 Présentation factory io

2.2.1 Fonctionnalités générales

Le logiciel de simulation industrielle Factory I/O est proposé en plusieurs éditions, chacune adaptée à des besoins spécifiques en matière de formation et de compatibilité avec différents automates programmables industriels (API). Ces éditions permettent aux utilisateurs de simuler des systèmes automatisés en 3D et d'interagir avec divers types de contrôleurs. Voici un aperçu détaillé des éditions disponibles, basé sur les informations fournies par le site officiel de Factory I/O :

- **Starter Edition** : Conçue pour les débutants, cette version permet de créer des contrôleurs en utilisant des blocs

fonctionnels avec un automate virtuel intégré.

- **Allen-Bradley Edition** : Permet la connexion via Ethernet avec les automates Allen-Bradley tels que ControlLogix, CompactLogix, Micro800, MicroLogix 1100 & 1400, SLC 5/0X et SoftLogix.
- **Siemens Edition** : Offre une connexion Ethernet avec les automates Siemens, y compris LOGO!, S7-200/200 SMART/300/400, S7-1200/1500, ainsi que S7-PLCSIM.
- **Modbus & OPC Edition** : Facilite l'intégration avec diverses technologies d'automatisation en utilisant Modbus TCP/IP (serveur et client) et OPC Client DA/UA.
- **MHJ Edition** : Conçue pour une utilisation avec les automates virtuels WinSPS-S7 et le simulateur Siemens WinPLC-Engine, cette édition est idéale pour les environnements de formation à faible coût.
- **Automgen Edition** : Permet la conception de contrôleurs en utilisant des langages standard tels que Grafcet et SysML, avec une intégration facile au logiciel Automgen.
- **Ultimate Edition** : Il s'agit de la version la plus complète, offrant toutes les fonctionnalités des autres éditions, y compris un SDK (Software Development Kit) pour le développement de pilotes de communication personnalisés, ainsi que la possibilité de câbler des API via des cartes d'interface.[19]

Factory I/O permet à l'utilisateur de créer des systèmes automatisés en utilisant une bibliothèque de pièces industrielles basée sur les parties réelles utilisées dans l'environnement usine. Aucune autre application n'a atteint un tel degré de réalisme dans l'apprentissage de l'automatisation. Comme une installation de modélisation en 3D, l'utilisateur peut assembler des systèmes en utilisant des variétés de composants dans la scène. Cette application émule le vrai processus, et l'utilisateur peut tester immédiatement les actionneurs et les capteurs en exécutant la simulation ou les programmes. Et, Factory I/O est essentiel en aide à la recherche et développement puisqu'il est un outil important pour les chercheurs et développeurs qui travaillent sur l'innovation en automatisation et en toute technologie de l'Industrie 4.0, lui conférant un rôle plus que fondamental.[20]

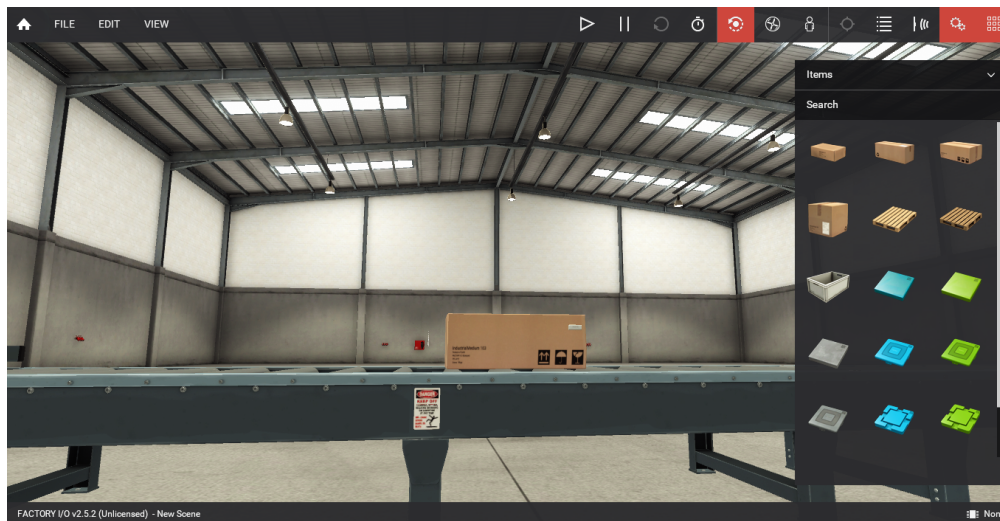


FIG. 2.1 : factory io

- **L'interface du logiciel**

Le logiciel dispose d'une interface conviviale qui inclut trois menus, et plusieurs boutons pour contrôler les caméras, afficher la palette, les capteurs et actionneurs, ou encore lancer une simulation ...etc [20]

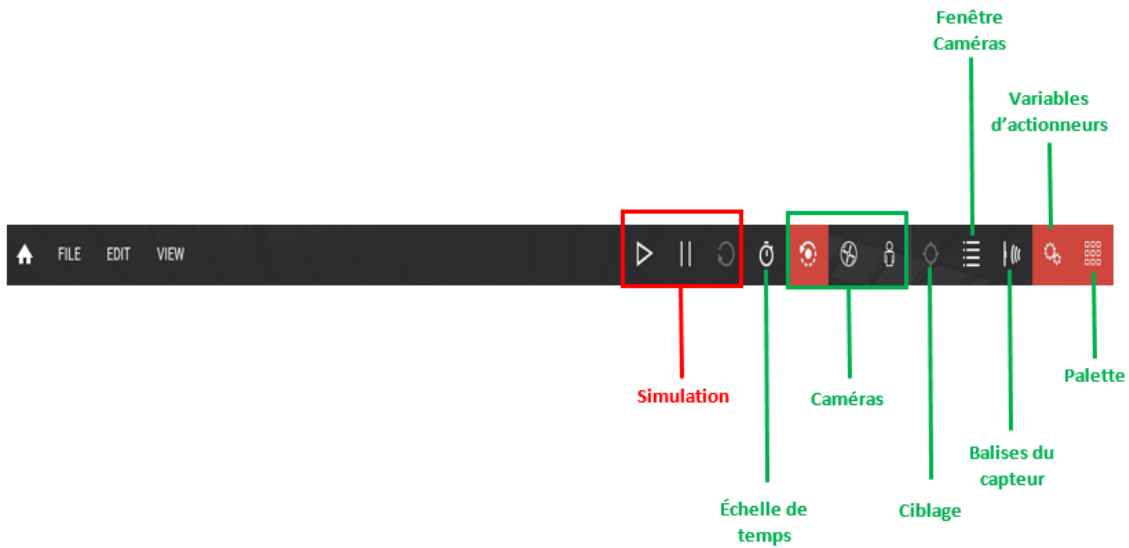


FIG. 2.2 : barre de menu

- **Caméras**

Il existe principalement trois types de caméras :

- Fly Camera
- Orbit Camera
- First Person Camera

Pour manipuler ces différentes caméras, vous devez utiliser les deux boutons de la souris, la molette et les touches directionnelles du clavier. Pour être à l'aise avec leur utilisation, un entraînement régulier est recommandé

- **La palette**

La palette contient les différents composants nécessaires à la conception d'un système automatisé, élément par élément. On y trouve notamment des tapis roulants, des boutons-poussoirs, des lumières, etc.[20]

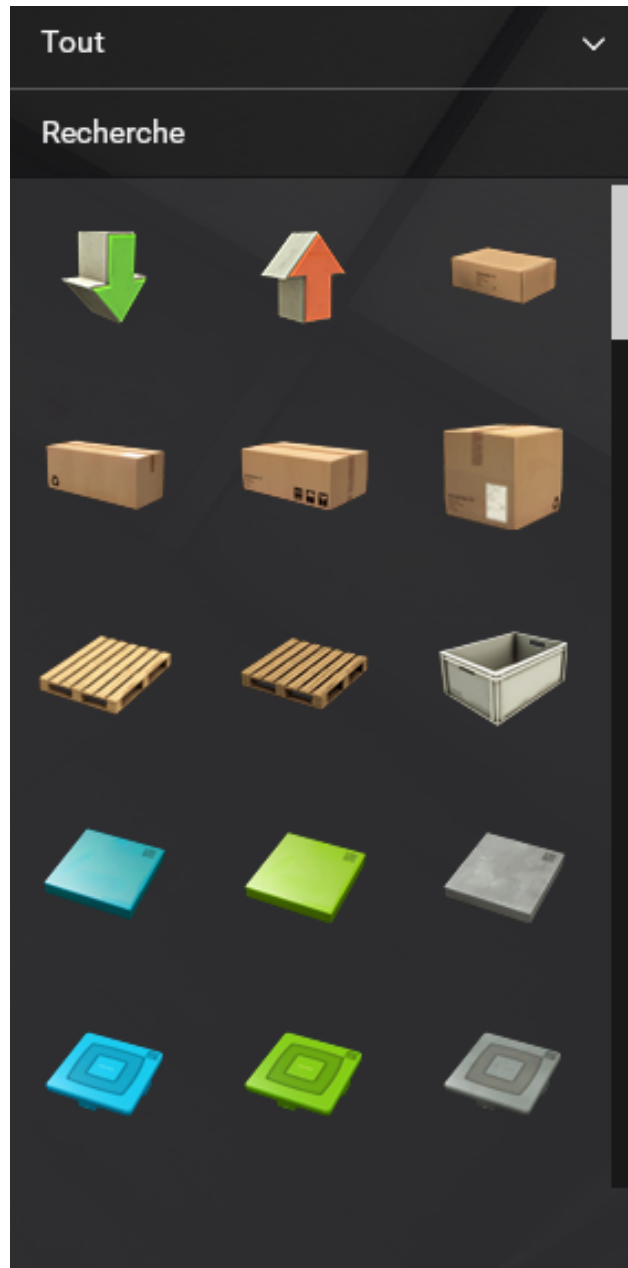


FIG. 2.3 : menu de palette

Vous pouvez concevoir plusieurs types de systèmes en glissant et déposant simplement le composant dans l'interface d'édition.

- **Les drivers**

Les drivers permettent de contrôler le système automatisé conçu via l'interface d'édition. Il en existe plusieurs types, notamment pour les automates Siemens, Schneider, Rockwell, etc.

Factory I/O intègre également Control I/O, un Soft PLC conçu pour faciliter l'apprentissage de l'automatisation. Il permet de programmer et de contrôler un système automatisé sans nécessiter d'automate physique, en utilisant le langage FBD (Function Block Diagram).

Pour activer un driver spécifique, cliquez sur l'icône située dans le coin inférieur droit du logiciel, puis sélectionnez le pilote souhaité dans la liste des options disponibles.[20]



FIG. 2.4 : barre de drivers



FIG. 2.5 : Menu du pilote (Driver menu)

2.2.2 Objectifs et rôle de Factory I/O dans la conception des systèmes automatisés

Afin d'illustrer concrètement l'apport des outils de simulation dans le domaine de l'automatisation industrielle, il est pertinent d'examiner les principaux objectifs et usages du logiciel Factory I/O. Les informations qui suivent sont issues d'une synthèse réalisée à partir de la documentation officielle disponible en ligne. Cette présentation vise à situer le rôle de cet outil dans le cycle de développement, de formation et de validation des systèmes automatisés.

Factory I/O est en effet un logiciel de simulation 3D conçu spécifiquement pour l'apprentissage et la conception de systèmes automatisés. Il propose un environnement virtuel immersif et interactif, permettant aux ingénieurs, techniciens et étudiants de concevoir, tester et optimiser des systèmes automatisés avant leur déploiement physique. Grâce à une interface intuitive et à une large compatibilité avec différents types d'automates industriels (Siemens, Allen-Bradley, Schneider Electric, etc.), Factory I/O s'impose comme un outil polyvalent, particulièrement adapté à la formation, au prototypage, ainsi qu'à l'amélioration continue des processus industriels.

A_ Objectifs de Factory I/O

Factory I/O vise à répondre aux besoins de l'industrie et de la formation en automatisation en offrant une plateforme de simulation complète et flexible. Ses objectifs principaux sont les suivants :

- **Formation et pédagogie** : Factory I/O permet aux étudiants et aux professionnels d'acquérir des compétences pratiques en automatisation sans nécessiter d'équipements physiques coûteux. Il est compatible avec plusieurs langages de programmation, notamment le Ladder Diagram (LD), le Function Block Diagram (FBD) et le Structured Text (ST), utilisés dans l'industrie pour programmer les automates programmables industriels (API).
- **Prototypage et validation** : L'un des principaux défis en ingénierie d'automatisation est de s'assurer qu'un système fonctionne correctement avant son déploiement sur une ligne de production réelle. Factory I/O permet aux ingénieurs de simuler des scénarios complexes, de tester différents algorithmes de contrôle et de corriger les erreurs potentielles avant la mise en œuvre.
- **Optimisation des processus industriels** : Factory I/O offre la possibilité d'analyser les flux de production et de tester des modifications dans un environnement virtuel. Les entreprises peuvent ainsi identifier les points d'amélioration, réduire les temps d'arrêt et optimiser l'efficacité énergétique de leurs systèmes automatisés.
- **Recherche et développement** : Avec l'avènement de l'Industrie 4.0, la simulation joue un rôle clé dans le développement de nouvelles solutions d'automatisation. Factory I/O permet aux chercheurs d'expérimenter des concepts avancés tels que la communication entre machines (M2M), l'intégration de l'Internet des objets industriels (IIoT) et l'intelligence artificielle appliquée à l'automatisation.

B_ Rôle de Factory I/O dans la conception des systèmes automatisés

Factory I/O joue un rôle fondamental dans la conception et la validation des systèmes d'automatisation industrielle. Son utilisation couvre plusieurs aspects essentiels :

- **Modélisation et simulation** : Factory I/O propose une bibliothèque riche en composants industriels, tels que des convoyeurs, des capteurs, des actionneurs et des robots. Ces éléments peuvent être assemblés facilement via une interface glisser-déposer, permettant aux utilisateurs de concevoir des systèmes automatisés en quelques minutes.
- **Programmation et interaction avec les automates** : Le logiciel est compatible avec plusieurs marques d'automates programmables (Siemens, Allen-Bradley, Schneider Electric, etc.). Il permet de tester les programmes développés sur ces automates dans un environnement simulé avant leur implémentation réelle. Pour les utilisateurs ne disposant pas d'un API physique, Factory I/O intègre un Soft PLC, Control I/O, qui permet de programmer et de simuler un automate directement dans le logiciel.
- **Validation et diagnostic** : Factory I/O facilite la détection des erreurs et des incohérences dans les programmes d'automatisation avant leur mise en œuvre réelle. Les utilisateurs peuvent tester la réactivité des capteurs, vérifier le bon fonctionnement des actionneurs et analyser les signaux d'entrée et de sortie en temps réel. Cette capacité à simuler les interactions entre les composants améliore considérablement la fiabilité des systèmes avant leur déploiement.
- **Amélioration continue** : Dans un contexte industriel, les entreprises cherchent constamment à améliorer la productivité et la qualité de leurs processus. Factory I/O permet d'expérimenter différentes configurations sans interrompre la production réelle, ce qui aide à identifier les meilleures stratégies d'optimisation. De plus, il peut être utilisé pour former les opérateurs sur des scénarios spécifiques avant leur application sur le terrain.

2.2.3 Avantages de Factory I/O

Factory I/O offre de nombreux avantages pour les étudiants, ingénieurs et industriels cherchant à améliorer leurs compétences et leurs systèmes d'automatisation :

Avantage	Description
Réduction des coûts	Factory I/O permet d'éviter l'achat de matériel coûteux pour tester des concepts en automatisation, réduisant ainsi les investissements en équipements physiques.
Sécurisation des tests	La simulation permet d'expérimenter des scénarios à risque sans compromettre la sécurité des machines et des opérateurs.
Flexibilité	Factory I/O est compatible avec divers automates, langages de programmation et protocoles de communication, offrant une grande adaptabilité aux besoins des utilisateurs.
Gain de temps	En testant et en optimisant les systèmes en environnement simulé, les ingénieurs peuvent accélérer la mise en service des équipements réels.
Accessibilité	Grâce à son interface intuitive, Factory I/O est accessible aussi bien aux débutants qu'aux professionnels expérimentés.

TAB. 2.1 : Avantages de Factory I/O

2.3 Modélisation du système industriel

2.3.1 Description du processus industriel choisi

Le système industriel automatisé est composé de deux lignes de production parallèles et identiques, installées dans un atelier moderne. Ce système adopte une configuration de type "flow shop" (atelier à flux continu), dans laquelle chaque produit suit une séquence fixe d'étapes à travers les postes de travail. Chaque ligne est structurée en trois stations principales : une station de distribution, un centre d'usinage robotisé, station d'assemblage et une zone de sortie.

A_ Station de distribution : acheminement des pièces brutes vers les stations d'usinage :

Deux stations de distribution alimentent les deux lignes de production avec deux types de pièces distinctes, différenciées par leur nature : une pièce métallique et une pièce en plastique verte. Avant leur acheminement vers les stations d'usinage, des capteurs de vision (caméras) analysent et trient les pièces en fonction de leur couleur.

Les pièces sont ensuite déposées sur des convoyeurs motorisés, dont la vitesse peut être réglée à 0,6 m/s, pour être transportées vers les quatre stations robotisées correspondantes.

Des capteurs de présence, notamment des capteurs photoélectriques, détectent l'arrivée des pièces et transmettent un signal au PLC (Automate Siemens sous TIA Portal). Ce dernier gère le flux de production en fonction de la disponibilité des stations d'usinage, permettant ainsi d'éviter les blocages et d'assurer une synchronisation efficace des opérations.

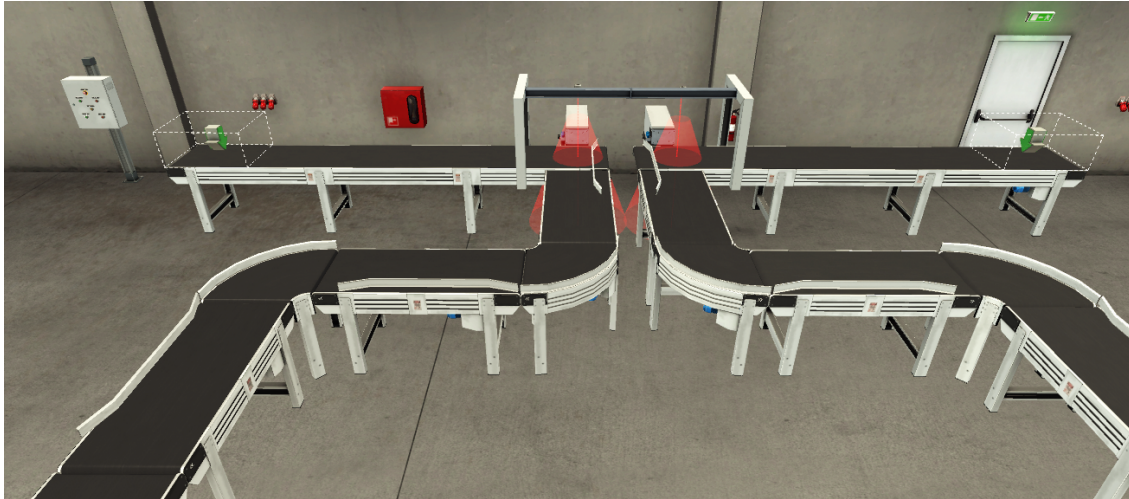


FIG. 2.6 : Station de distribution

B_ Stations d'usinage : Transformer les pièces brutes en pièces semi-finies prêtes pour l'assemblage :

Chaque station de distribution alimente deux stations d'usinage robotisées, soit un total de quatre centres d'usinage : deux dédiés aux pièces métalliques et deux aux pièces vert. Chaque centre d'usinage est équipé d'un bras anthropomorphe et d'une machine CNC, permettant la fabrication de couvercles et de bases à partir des matières premières.

Le processus débute lorsque le robot articulé détecte la présence d'une pièce au poste d'entrée. Une fois la pièce identifiée, le bras anthropomorphe la charge dans la machine CNC, qui réalise l'usinage en fonction de la nature du produit semi-fini détecté. Le temps d'usinage varie selon le type de pièce : 6 secondes pour un couvercle et 3 secondes pour une base. Une fois l'usinage terminé, le bras récupère la pièce transformée et la dépose sur le poste de sortie. Elle est ensuite transférée via un convoyeur vers la station d'assemblage.

Les produits usinés sont les suivants :

- **Pièce en plastique vert :**
 - Couvercle du produit – Poids : 5 kg
 - Base du produit – Poids : 7 kg
- **Pièce métallique :**
 - Couvercle du produit – Poids : 7 kg
 - Base du produit – Poids : 9 kg

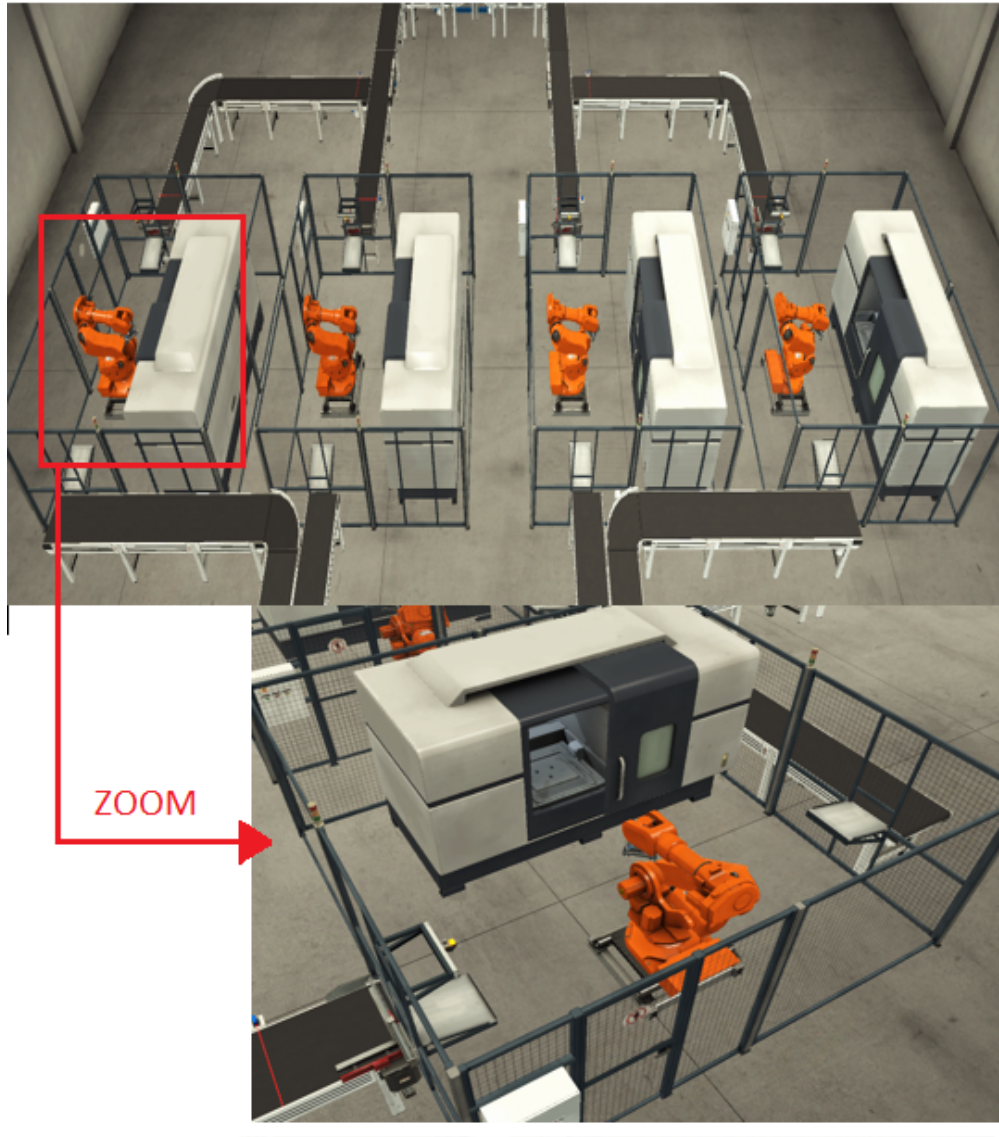


FIG. 2.7 : station d'usinage

• **Gestion des situations d'urgence :**

En cas de dysfonctionnement ou de problème de sécurité, un arrêt d'urgence peut être déclenché, interrompant immédiatement les opérations du robot articulé et de la CNC. Après un arrêt d'urgence, la station ne peut être réinitialisée qu'après désarmement du bouton d'urgence et activation du bouton de réinitialisation.

Deux colonnes lumineuses situées au-dessus du dispositif de sécurité fournissent des indications sur l'état de fonctionnement du centre d'usinage :

- **Voyant vert :** Station non occupée, prête à fonctionner.

- • **Voyant jaune** : Station occupée, en cours d'usinage.
- • **Voyant rouge** : Erreur détectée (élément incorrect à l'entrée).[20]

C_ Station d'assemblage des pièces : Combiner deux pièces semi-finies de même couleur pour créer un produit assemblé :

Le processus d'assemblage s'effectue dans deux stations distinctes, chacune étant dédiée à un type de pièce, l'une pour les pièces métalliques, l'autre pour les pièces en plastique vert. Les pièces usinées sont acheminées vers ces stations via un convoyeur, assurant une synchronisation efficace du flux de production.

L'assemblage est réalisé à l'aide d'un robot Pick & Place à deux axes (robot cylindrique RPP), chargé de positionner et de fixer les couvercles sur les bases correspondantes. Afin de garantir un alignement précis, des positionneurs sont utilisés pour maintenir les pièces en place et corriger toute éventuelle déviation lors de l'assemblage.

Une fois le processus terminé, le produit assemblé passe par un compteur à la sortie de la station afin d'identifier et de surveiller le nombre de produits finis assemblés.



FIG. 2.8 : Station d'assemblage des pièces

2.3.2 Les caractéristiques d'élément utilisé

Le tableau ci-dessous présente les différents éléments utilisés dans la simulation Factory I/O, en les classant par catégorie, en précisant leur nom, leur description fonctionnelle ainsi que leurs principales caractéristiques techniques.

tab. 2.2 : caractéristiques d'élément utilisé [20]

Catégorie	Élément	Description	Caractéristiques principales
Émetteur (Emitter)	Émetteur	Émet un article à utiliser dans une scène (par exemple, boîte en carton, palette, etc.)	<ul style="list-style-type: none"> • Type d'article à émettre • Temps entre les émissions • Nombre d'articles à émettre • Position et orientation aléatoires
Articles (Items)	Matière première	Utilisée pour produire des couvercles et des bases dans le Centre d'Usinage. Disponible en plastique bleu, plastique vert et métal.	<ul style="list-style-type: none"> • Plastique bleu : 8 kg • Plastique vert : 8 kg • Métal : 10 kg
	Couvercle de produit	Pièce pouvant être assemblée avec une base pour produire un produit final.	<ul style="list-style-type: none"> • Plastique bleu : 5 kg • Plastique vert : 5 kg • Métal : 7 kg
	Base du produit	Pièce pouvant être assemblée pour former un couvercle et produire un produit final.	<ul style="list-style-type: none"> • Plastique bleu : 7 kg • Plastique vert : 7 kg
	Boîte empilable	Utilisée pour transporter des articles tels que la matière première, la base du produit et le couvercle de produit.	Non spécifié
Pièces à charge légère(Light Load Parts)	Convoyeurs à bande (Belt Conveyors)	Les convoyeurs à bande sont utilisés pour le transport de charges légères(boîtes et palette) . Ils peuvent être contrôlés par des valeurs numériques ou analogiques	<ul style="list-style-type: none"> • Longueurs disponibles :2, 4 et 6 m • Vitesse de transport maximale : 0,6m/s(numérique) 3 m/s (analogique)

	Convoyeur à bande courbe (Curved Belt Conveyor)	Les convoyeurs à bande sont utilisés pour le transport de charges légères. Ils modifient la direction du flux des articles (courbé).	<ul style="list-style-type: none"> • Contrôle : numérique ou analogique • Rotation : sens horaire/antihoraire
	Aligneurs (Aligners)	Structures métalliques fines, fixables sur un convoyeur pour empêcher la chute des pièces lors du transport à grande vitesse.	-Il existe plusieurs types et couleurs différentes.
	Pousseur(Pusher)	Trieur pneumatique à poussoir équipé de deux capteurs Reed indiquant les limites avant et arrière. Il comprend également une servovalve permettant de régler et de mesurer la position de la tige. Le contrôle peut être effectué par des valeurs numériques ou analogiques selon la configuration sélectionnée.	<ul style="list-style-type: none"> • Vitesse par défaut : 1 m/s • Course : 0,9 m • Vitesse rapide : 4 m/s
	Lame d'arrêt (Stop Blade)	La lame d'arrêt est un dispositif pneumatique utilisé pour arrêter ou accumuler des matériaux.	<ul style="list-style-type: none"> • Normalement abaissée • Course : 0,12 m
	Barres de positionnement (Positioning Bars)	Dispositif permettant de positionner consécutivement des articles au même endroit par serrage. Disponible en deux configurations : gauche et droite. Couramment utilisé avec les systèmes de prélèvement et de placement.	<ul style="list-style-type: none"> • Course verticale : 0,373 m • Course de la pince : 0,48 m
Capteur (sensor)	Capteur diffus (Diffuse sensor)	Capteur photoélectrique diffus capable de détecter tout objet solide.	<ul style="list-style-type: none"> • LED : rouge (détection) • Matériaux détectables : solides • Portée de détection : 0 - 1,6 m

	Capteur de vision (Vision Sensor)	Le capteur de vision reconnaît les matières premières, les couvercles et les bases des produits, ainsi que leurs couleurs respectives	<ul style="list-style-type: none"> • LED : rouge (détection) • Matériaux détectables : matières premières, bases des produits, couvercles des produits • Portée de détection : 0,3 à 2 m
Panneau de contrôle (Operators)	Arrêt d'urgence (Emergency Stop)	Gâchette rouge à deux positions et bouton poussoir non éclairé (tête de champignon).Généralement utilisé dans les situations d'urgence	-Type de contact normalement fermé
	Boutons-poussoirs (Push Buttons)	Boutons-poussoirs lumineux disponibles en différentes couleurs, Utilisables en mode momentané ou alternatif, selon la configuration sélectionnée. La couleur du bouton-poussoir peut être modifiée dans le menu contextuel.	<ul style="list-style-type: none"> • Boutons-poussoirs : Marche, Réinitialisation et Arrêt (normalement fermés) • Deux boutons-poussoirs génériques : Bouton-poussoir NO (normalement ouvert) et Bouton-poussoir NC (normalement fermé) • Couleurs conformes à la norme IEC 60204-1:2016
	Affichage numérique(Digital Display)	Permet d'afficher des valeurs numériques pendant la simulation	Non spécifié
Stations	Centre d'usinage (Machining Center)	Le centre d'usinage est une station de fabrication de couvercles et de fonds à partir de matières premières. Il utilise un robot articulé pour charger la matière dans une machine CNC. L'opération peut être arrêtée, redémarrée ou réinitialisée.	<ul style="list-style-type: none"> • Temp de fabrication : Lid (couvercle) : 6 sec Base : 3 sec • Commandes (Emergency,Start,Stop) • Indicateurs lumineux(Vert,Jaune,Rouge)

	<p>Système de prélèvement et de placement à deux axes (Two-Axis Pick & Place)</p>	<p>permet d'assembler des couvercles sur des socles ou de déplacer des objets, en assurant un alignement correct grâce à des barres de positionnement</p>	<ul style="list-style-type: none"> • Course axe X :1,125 m • Course axe Z :0,625 m • Vitesse du bras et du préhenseur :2 m/s • Rotation axe Z par incréments de 90
--	---	---	--

2.4 Création de l'environnement 3D dans FACTORY IO

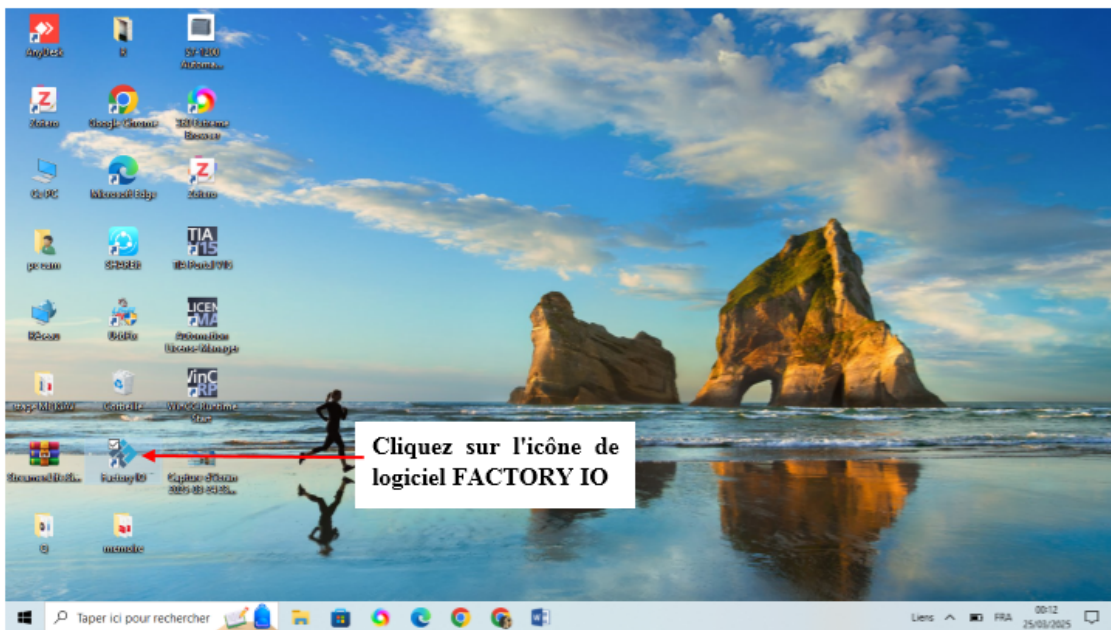


FIG. 2.9 : Lancement de Factory I/O : icône sur le bureau

Capture de l'icône Factory I/O dans l'environnement Windows ; l'utilisateur double clique pour démarrer le logiciel.

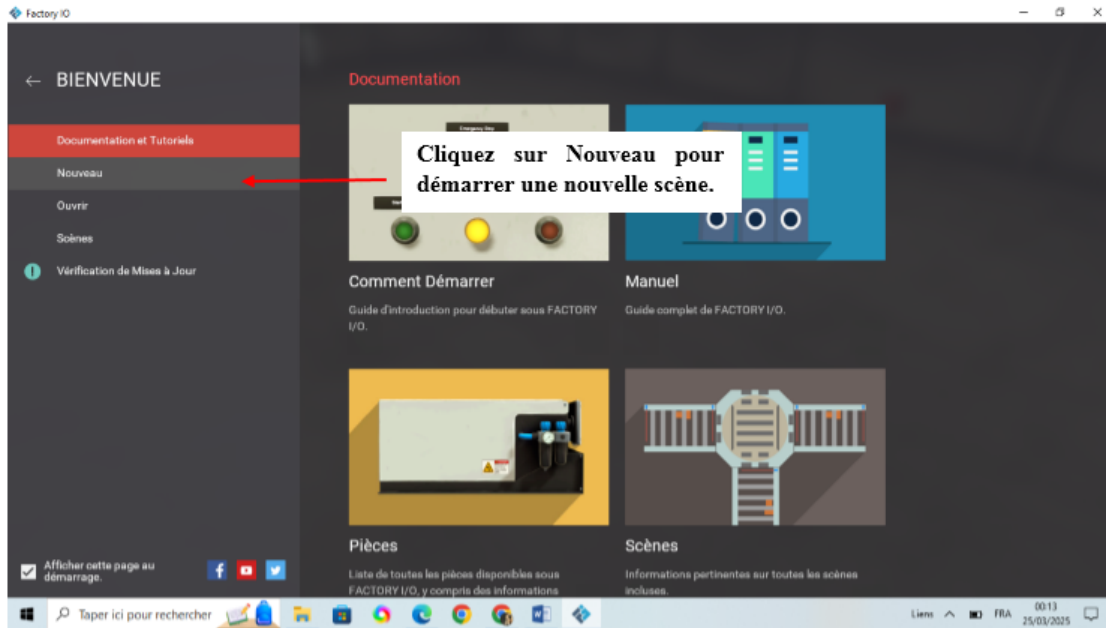


FIG. 2.10 : Création d'une nouvelle scène

Depuis le menu File, sélection de New afin d'obtenir une scène vierge (Ctrl + N)



FIG. 2.11 : Interface de travail de Factory I/O

Vue générale de l'éditeur 3D : barre d'outils, fenêtre de la scène et panneaux latéraux permettant d'ajouter ou de configurer les objets.



FIG. 2.12 : Palette – sélection du convoyeur 6 m

Ouverture de la Palette et clic sur la catégorie «Light Load Parts» pour choisir un convoyeur à bande de 6 mètres.

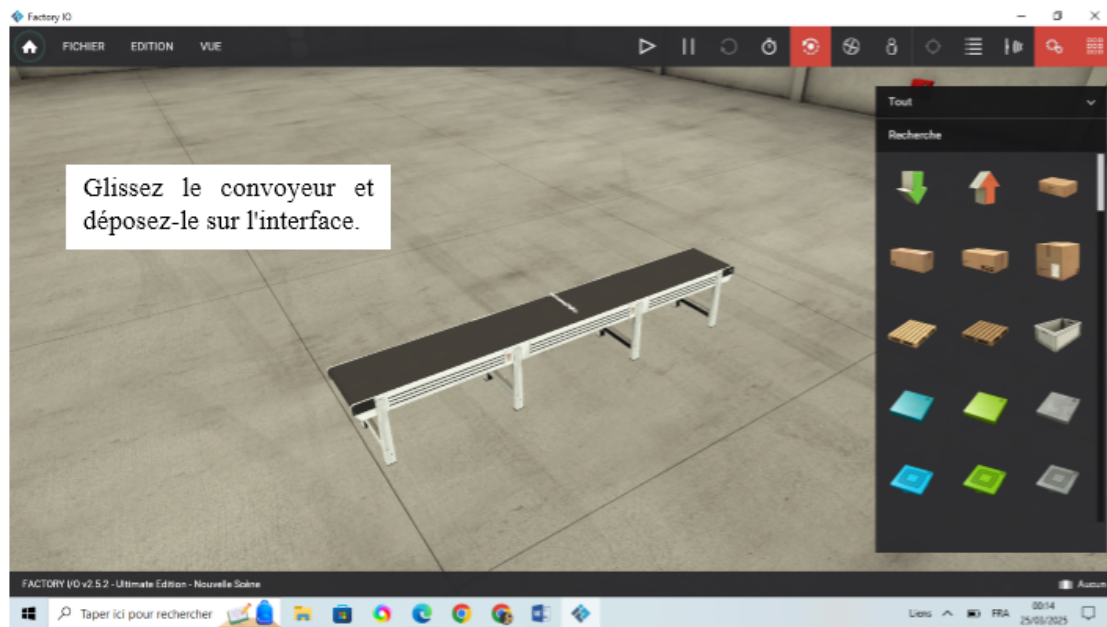


FIG. 2.13 : Placement du convoyeur dans la scène

Le convoyeur est glissé déposé dans la zone de travail ; un cadre blanc indique qu'il est correctement sélectionné et positionné.



FIG. 2.14 : Palette – sélection de la matière première

Dans la palette, l'élément «Raw Material» est sélectionné comme exemple de pièce à transporter. Il est ensuite déposé sur le convoyeur afin de constituer le flux de production virtuel.

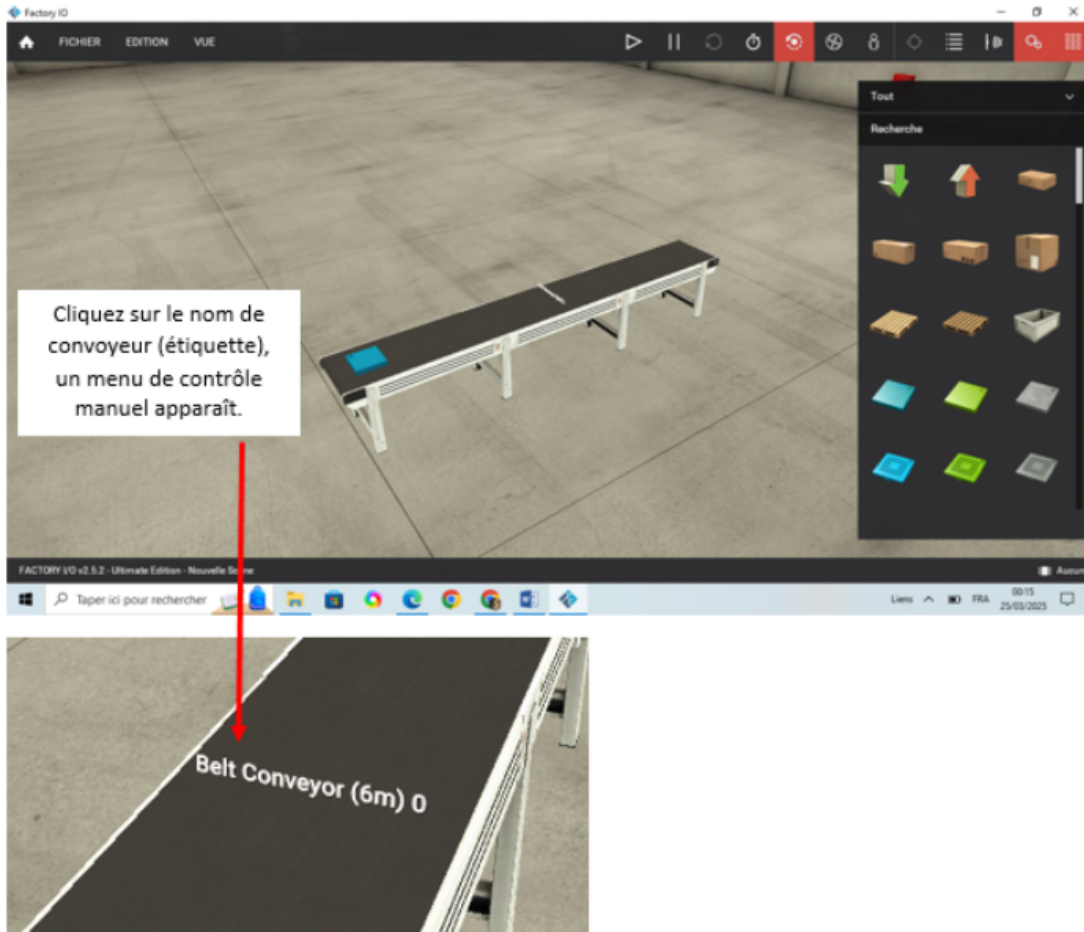


FIG. 2.15 : Affichage du nom du convoyeur et accès au contrôle manuel

En cliquant sur l'étiquette affichant le nom du convoyeur (Belt Conveyor (6m)), un menu contextuel s'ouvre. Ce menu permet d'accéder aux commandes manuelles de l'élément, telles que la configuration de la vitesse ou l'activation du mouvement en mode simulation.

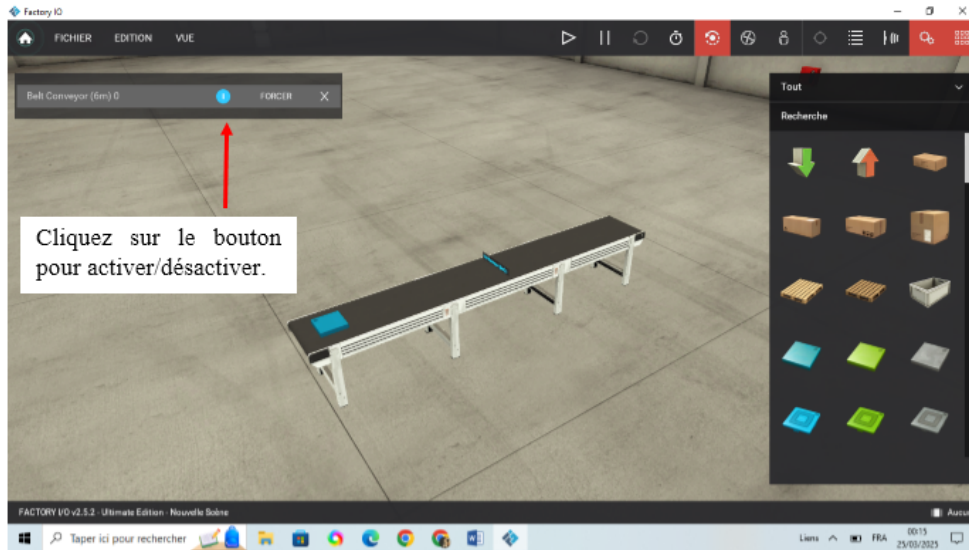


FIG. 2.16 : Passage en mode Run : bouton « Play »

Clic sur le bouton Play (ou touche F5) pour basculer de Edit vers Run ; la simulation s'exécute en temps réel.



FIG. 2.17 : Déplacement de la pièce du point A au point B

Résultat de l'exécution : la matière première se déplace le long du convoyeur, illustrant la validation du scénario simple « From A to B ».

2.5 Présentation de CONTROL I/O

Control I/O est un automate programmable simple et puissant, conçu exclusivement pour être utilisé avec Factory I/O. Son objectif principal est de fournir un outil indépendant, facile à prendre en main par toute personne débutant dans le monde de l'automatisation. Avec Control I/O, vous développez des programmes avec des diagrammes de blocs fonctionnels utilisant les fonctions les plus courantes disponibles sur les automates réels.[21]

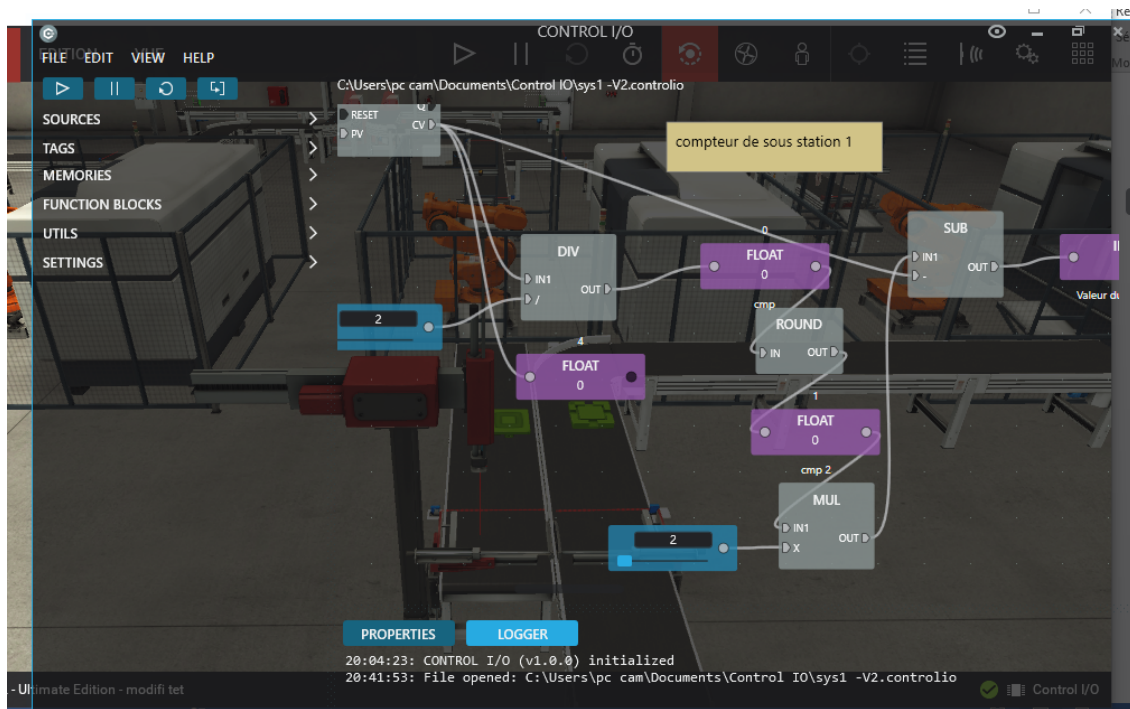


FIG. 2.18 : Interface de Control I/O avec un exemple de programme FBD appliqué dans Factory I/O

2.5.1 Bases de CONTROL I/O

A_ Interface de control io

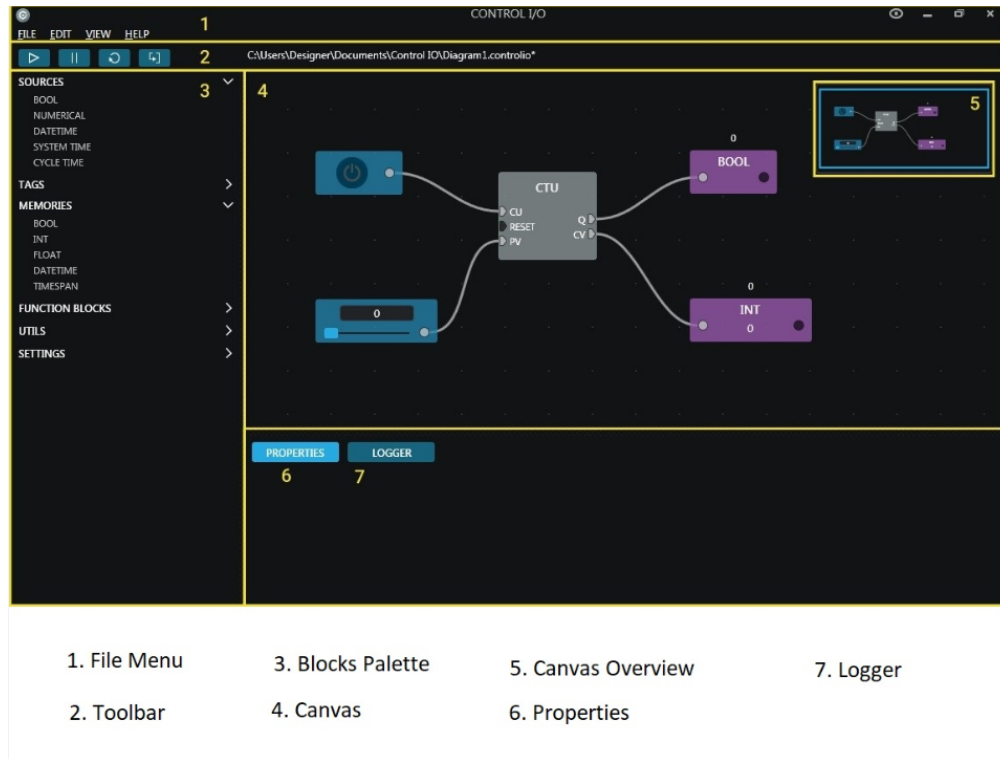


FIG. 2.19 : Interface de Control I/O avec un exemple de programme FBD appliqué dans Factory I/O [22]

B_ Blocs

Un bloc représente une valeur (booléen, entier, flottant, etc.) ou une fonction entre des entrées et des sorties (compteur, minuteur, etc.). Les blocs sont constitués de sockets permettant de relier différents blocs entre eux. En reliant des blocs, vous définissez le flux de données du bloc le plus en amont au bloc le plus en aval (de gauche à droite). Les sockets de sortie ne peuvent être reliés qu'aux sockets d'entrée; il est impossible de créer un retour d'information en reliant directement des sockets de sortie à des sockets d'entrée. Pour créer des boucles dans vos diagrammes, vous devrez utiliser des mémoires.[21]

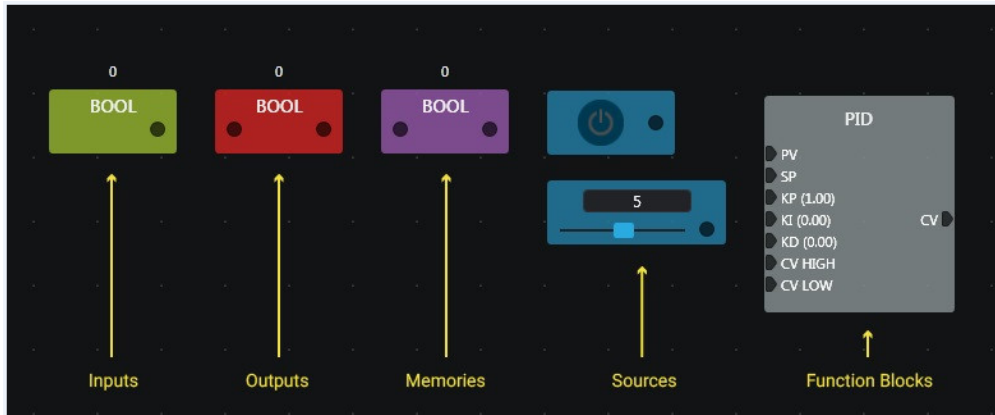


Fig. 2.20 : Représentation des blocs fonctionnels dans Control I/O et leurs connexions logiques[23]

C_ Comment Créer un diagramme

- Un diagramme est composé de blocs reliés par des liens. Les blocs sont constitués de sockets d'entrée et de sortie, et les liens relient toujours les sockets de sortie d'un nœud aux entrées d'un autre. Pour créer un diagramme, faites glisser des blocs de la palette Blocs vers le canevas
- Après avoir placé et organisé les blocs, créez des liens entre eux en cliquant sur le socket de sortie d'un bloc et en le faisant glisser vers le socket d'entrée cible.
- Boucles Il est impossible de créer des boucles en reliant des sockets de sortie à des sockets d'entrée. Pour créer une rétroaction, vous devez utiliser des blocs mémoire de même adresse. Lorsque plusieurs blocs mémoire partagent la même adresse, une mémoire en amont prendra la valeur de la mémoire en aval lors du prochain cycle de mise à jour du diagramme.²¹

D_ Fonctionne bloc diagramme

• Arithmétique :

Blocs effectuant des opérations arithmétiques. Vous pouvez utiliser n'importe quel type de données numériques avec ces blocs.

Bloc	Description
ADD2	Additionne deux nombre
ADD3	Additionne trois nombres
DIV	Divise deux nombres.
MUL	Multiplie deux nombres.
SUBS	Soustrait deux nombres.

TAB. 2.3 : bloc Arithmétique [21]

• Compteurs

- **CTD(compteur décroissant)** :Compte à rebours un nombre et définit la sortie (Q) sur True lorsqu'elle atteint 0. Utilisez PV pour définir la valeur
- **CTU (compteur progressif)** :Compte un nombre en définissant la sortie (Q) sur True lorsqu'elle atteint la valeur prédéfinie (PV). Utilisez RESET pour réinitialiser le compteur et CV pour obtenir la valeur actuelle.
- **CTUD(compteur ascendant/descendant)** :Implémente les compteurs CTU et CTD.
- **Supplémentaire(Extra)**

Bloc	Description
ASSIGN	OUT est égal à PT2 si IN est vrai ; sinon, OUT est égal à PT1.
Datetime to numerical	Convertit une valeur DateTime en composants numériques : année, mois, jour, heure, minute, seconde, etc.
Numerical to datetime	Assemble des composants numériques (année, mois, jour, etc.) en une valeur DateTime.
PID	Régulateur proportionnel-intégral-dérivé.

TAB. 2.4 : bloc Supplémentaire(Extra)[21]

• Logique

Bloc	Description
AND2	Effectue un ET logique entre deux valeurs booléennes.
AND4	Effectue un ET logique entre quatre valeurs booléennes
OR2	Effectue un OU logique entre deux valeurs booléennes.
OR4	Effectue un OU logique entre quatre valeurs booléennes.
FTRIG	Détecte la transition d'une valeur booléenne de Vrai à Faux (détection de front descendant).
JK	Bascule JK. (ou flip-flop JK)
NOT	inverse l'état logique d'une valeur booléenne.
ROL	Effectue une rotation à gauche bit à bit sur un nombre.
ROR	Effectue une rotation à droite bit à bit sur un nombre.
RS	Mémoire avec priorité de réinitialisation
RTRIG	Détecte la transition d'une valeur booléenne de Faux à Vrai (détection de front montant).
SHL	Effectue un décalage binaire vers la gauche sur un nombre.
SHR	Effectue un décalage binaire vers la droite sur un nombre.
SR	Mémoire avec priorité définie.
XOR2	Effectue un OU exclusif logique entre deux valeurs booléennes.
XOR3	Effectue un OU exclusif logique entre trois valeurs booléennes.

TAB. 2.5 : bloc Logique[21]

• Mathématique

Bloc	Description
ABS	Calcule la valeur absolue du nombre saisi.
CLAMP	Fixe le nombre d'entrée entre une valeur minimale et une valeur maximale.
EXP	Calcule e élevé à la puissance d'entrée.
SIN	Calcule le sinus de l'angle saisi(en radians).
MIN	Calcule le plus petit des deux nombres saisis.
MAX	Calcule le plus grand des deux nombres saisis.
LOG	Calcule le logarithme en base e de la valeur d'entrée
ROUND	Arrondit la valeur saisie à l'entier le plus proche.
SQRT	Calcule la racine carrée du nombre saisi.

TAB. 2.6 : bloc Mathématique [21]

• **Relationnel et Égalité**

Bloc	Description
Inférieur à (<)	Compare si le premier opérande (IN1) est inférieur au second (IN2).
Inférieur ou égal à (<=)	Compare si le premier opérande (IN1) est inférieur ou égal au second (IN2).
Égal à (=)	Compare si le premier opérande (IN1) est égal au second (IN2).
Différent de (≠)	Compare si le premier opérande (IN1) est différent du second (IN2).
Supérieur à (>)	Compare si le premier opérande (IN1) est supérieur au second (IN2).
Supérieur ou égal à (>=)	Compare si le premier opérande (IN1) est supérieur ou égal au second (IN2).

TAB. 2.7 : bloc Relationnel et Égalité[21]

• **Temporisateurs**

• **TOF (temporisateur désactivé)**

Temporisateur qui définit la sortie (Q) sur Faux après un temps défini par PT. Il démarre sur une transition de front descendant (Vrai vers Faux) de IN et se réinitialise sur une transition de front montant (Faux vers Vrai).

• **TON (temporisateur activé)**

Temporisateur qui définit la sortie (Q) sur Vrai après un temps défini par PT. Il démarre sur une transition de front montant de IN (Faux vers Vrai) et se réinitialise sur une transition de front descendant (Vrai vers Faux).[21]

2.6 Description du fonctionnement du programme

2.6.1 Station de distribution

Lorsque le système est démarré via le bouton Start de l'armoire de commande, les matières premières sont introduites dans la station de distribution, et les convoyeurs se mettent en marche.

Dès qu'un capteur de type caméra détecte la présence d'une pièce, un pusher (poussoir) est activé afin de transférer la pièce détectée vers un autre convoyeur. Ce convoyeur achemine ensuite la pièce vers la station suivante du processus.

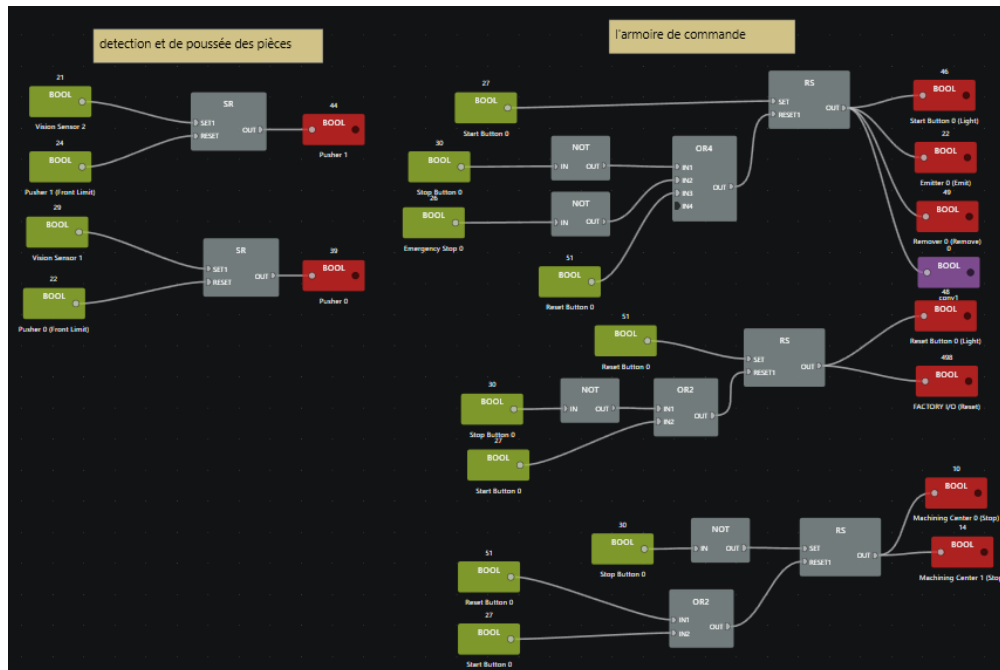


FIG. 2.21 : Logique de lancer le système et de détection et de poussée des pièces

Rôle de cette partie du programme :

Cette partie est responsable de :

- Vérifier l'état du bouton Start pour l'activation initiale du système.
- Détecter la présence d'une pièce grâce au capteur caméra.
- Activer le poussoir pour déplacer la pièce détectée vers un convoyeur secondaire.
- Assurer la continuité du flux vers la station suivante.

2.6.2 Acheminement des pièces vers les centres d'usinage

Lorsque la pièce est détectée par un capteur de présence, un compteur est activé afin de suivre le nombre de pièces

Si le nombre du compteur est pair, le poussoir reste inactif et la pièce continue automatiquement son chemin sur le convoyeur principal vers le premier centre d'usinage, destiné à la fabrication de la couvercle .

En revanche, si le nombre est impair, le poussoir est activé pour dévier la pièce vers un deuxième centre d'usinage, destiné à la production de la base du produit.

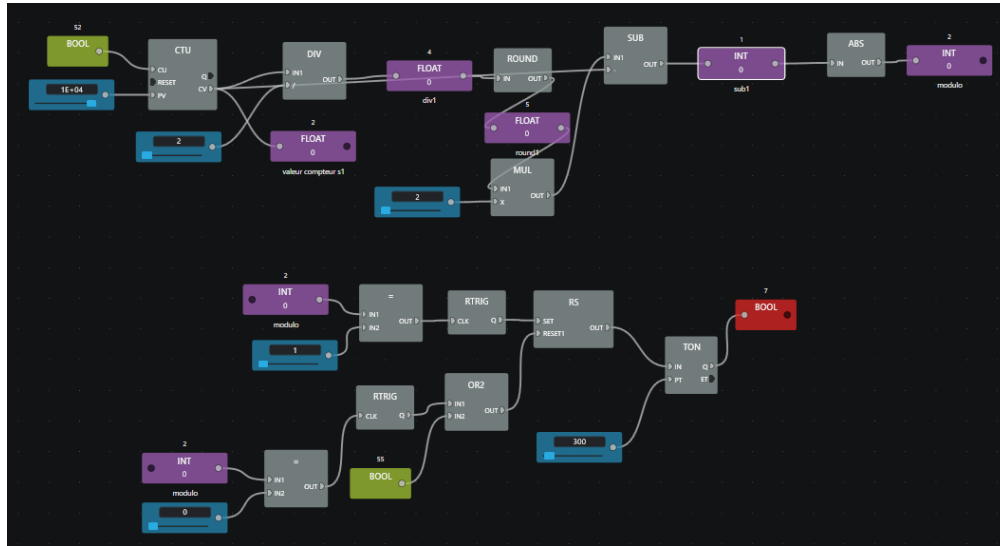


FIG. 2.22 : Logique de tri des pièces selon le comptage pair/impair

Rôle de cette partie du programme :

Cette section du programme permet :

- La décision conditionnelle (paire ou impaire) sur le trajet de chaque pièce.
- Le tri automatique des pièces :
 - Les paires sont envoyées vers le centre d’usinage principal.
 - Les impaires sont déviées via un poussoir vers une station secondaire.
- L’optimisation du flux de production en fonction du type de pièce à fabriquer (base ou couvercle).

2.6.3 Fonctionnement du centre usinage :

Lorsqu’une pièce est détectée par le capteur photoélectrique, le positionneur la serre automatiquement pour la maintenir en place.

Ensuite, si la machine est vide, que la porte est ouverte et qu’aucune erreur n’est détectée, le positionneur libère la pièce et effectue un mouvement vertical vers le haut afin de permettre son entrée dans le centre d’usinage.

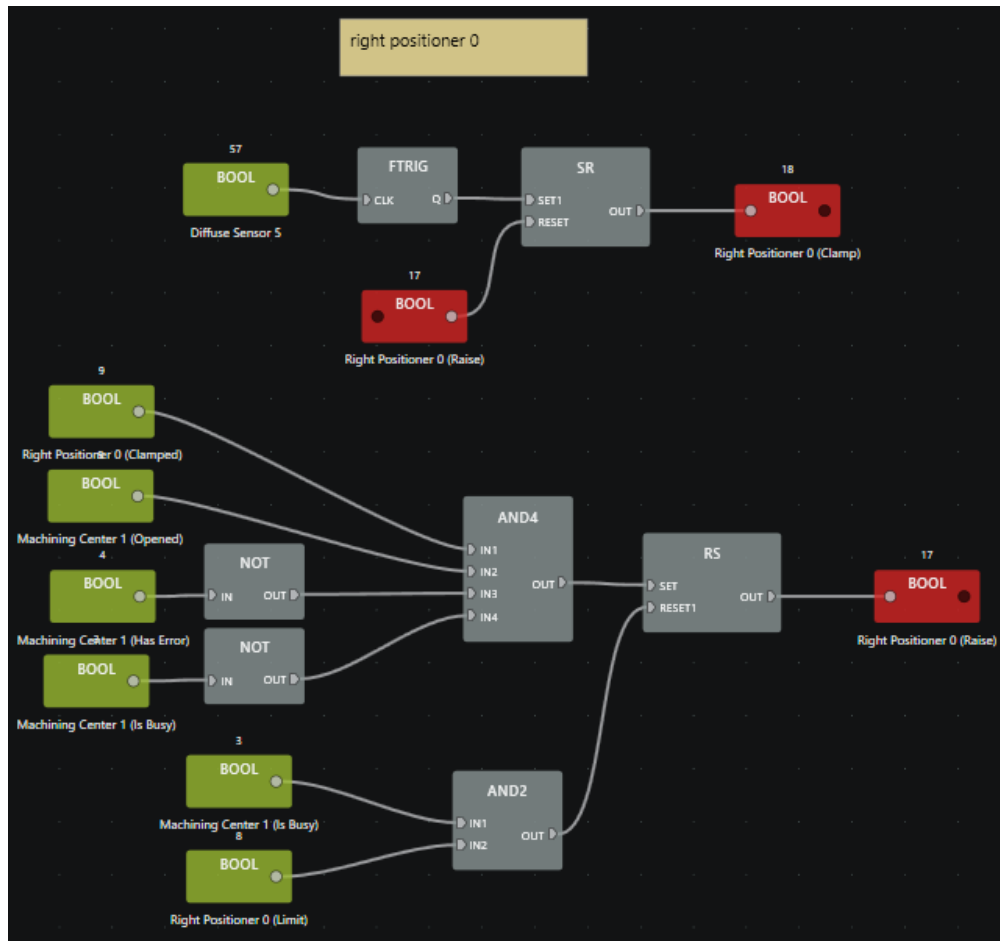


Fig. 2.23 : logique de serrage et de libération de la pièce par le positionneur

Si la pièce est ensuite détectée par le capteur de vision (caméra) intégré au centre d'usinage, la machine démarre automatiquement le processus d'usinage.

En cas de défaut détecté (état d'erreur), la machine doit être réinitialisée afin de reprendre un fonctionnement normal.

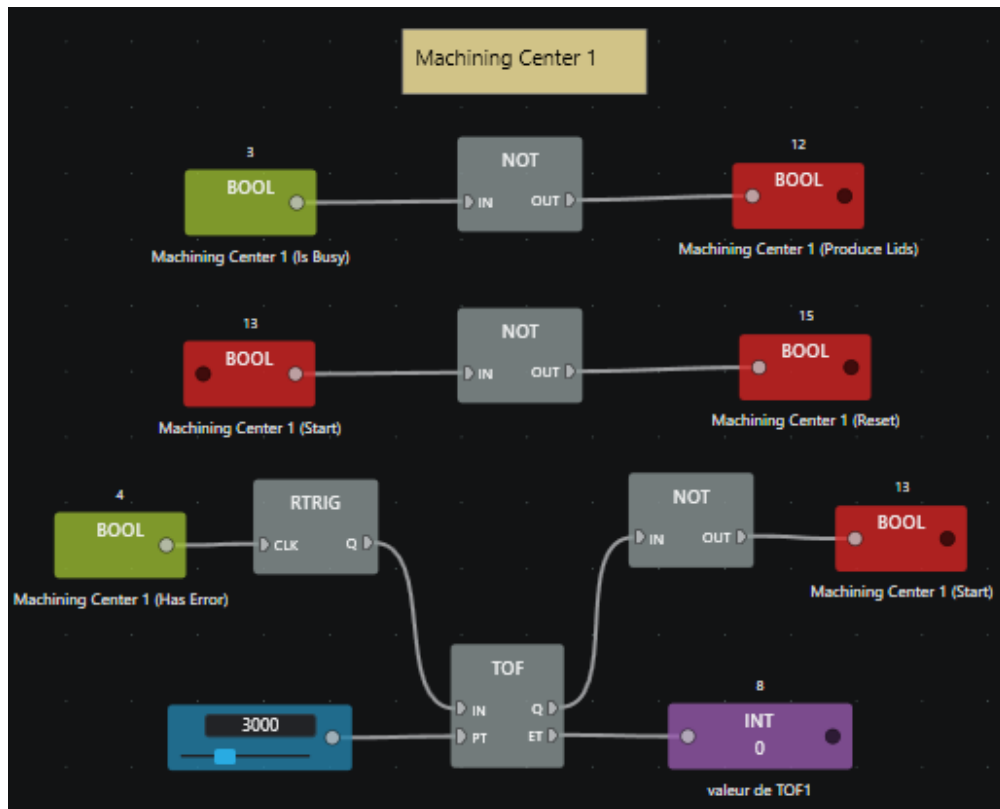


FIG. 2.24 : conditions de démarrage et gestion des erreurs du centre d'usinage

Rôle de cette partie du programme :

Cette section de la logique permet de :

- Assurer la stabilité de la pièce grâce au serrage contrôlé par le positionneur.
- Coordonner le mouvement vertical du positionneur pour transférer la pièce dans la machine.
- Contrôler les conditions de sécurité : porte fermée, machine disponible et sans erreur.
- Gérer les pannes par une procédure de réinitialisation automatique du système.

2.6.4 Fonctionnement de la station d'assemblage

Lorsque le capteur détecte une pièce, le positionneur serre celle-ci pour la maintenir en place. Lorsque les deux pièces (le couvercle et la base) sont correctement positionnées, le robot cylindrique (Two-Axis Pick & Place) effectue un mouvement vertical (axe Z) vers le couvercle. Si la pince détecte la présence d'une pièce par un capteur, elle la saisit à l'aide d'un système pneumatique. Une fois la pièce saisie :

- Le positionneur libère la pièce.
- Le robot remonte verticalement, puis effectue un déplacement horizontal (axe X) vers la deuxième pièce semi fini (la base), qui est déjà maintenue par un second positionneur.

- Le robot descend verticalement pour poser le couvercle sur la base (opération d'assemblage).

Ensuite, le robot revient à sa position initiale, et le positionneur libère le produit fini en effectuant un mouvement vers le haut.

Remarque :

Un compteur a été ajouté pour faciliter et organiser les mouvements du robot et s'assurer du bon déroulement du cycle d'assemblage.

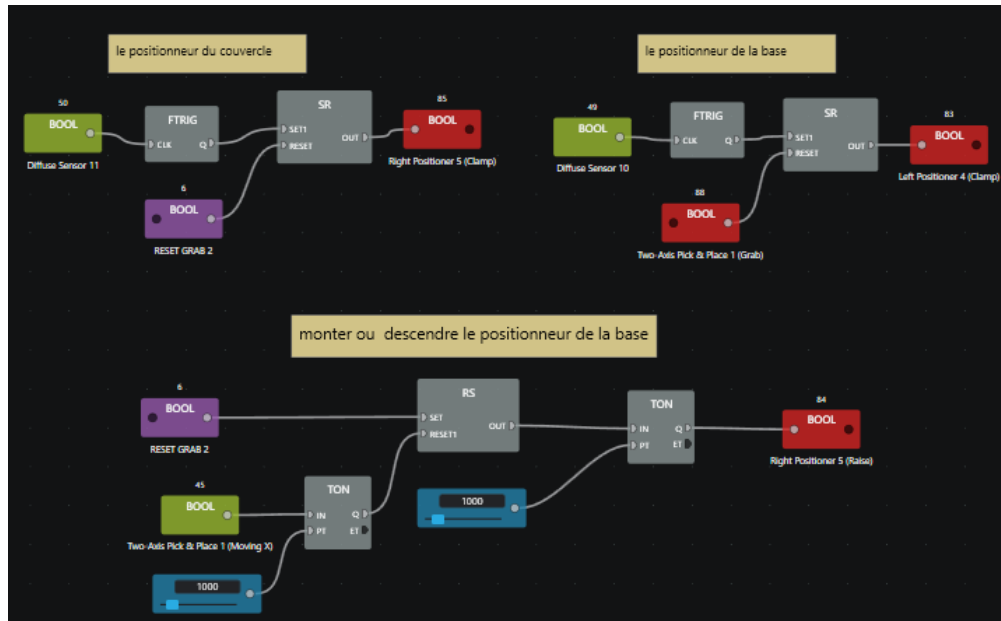


FIG. 2.25 : Logique de controle des positionneurs du couvercle et de base

Cette figure montre comment les capteurs et les positionneurs assurent le maintien précis des pièces semi-finies avant assemblage. Elle illustre une synchronisation efficace des actions mécaniques.

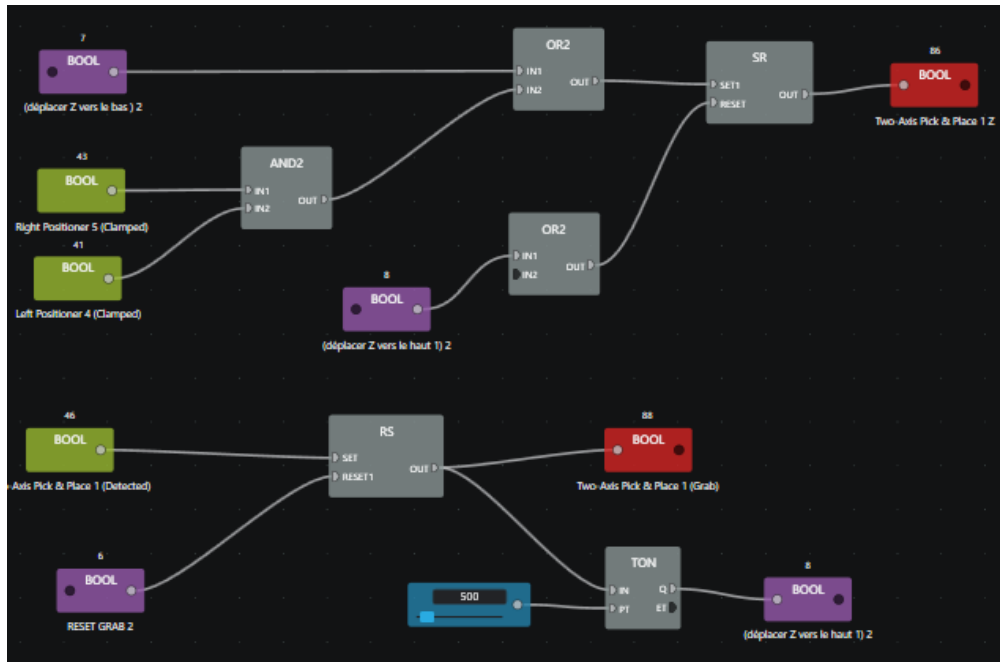


FIG. 2.26 : programmation des mouvements verticaux (axe Z) du robot d'assemblage

Cette figure est divisée en deux parties : la première montre le déplacement vertical du robot (axe Z) vers la pièce à saisir, tandis que la seconde illustre l'activation de la ventouse (fonction « Grab ») lorsque le capteur intégré détecte la présence d'une pièce. Cette logique garantit une prise précise et sécurisée avant le transfert vers la zone d'assemblage.

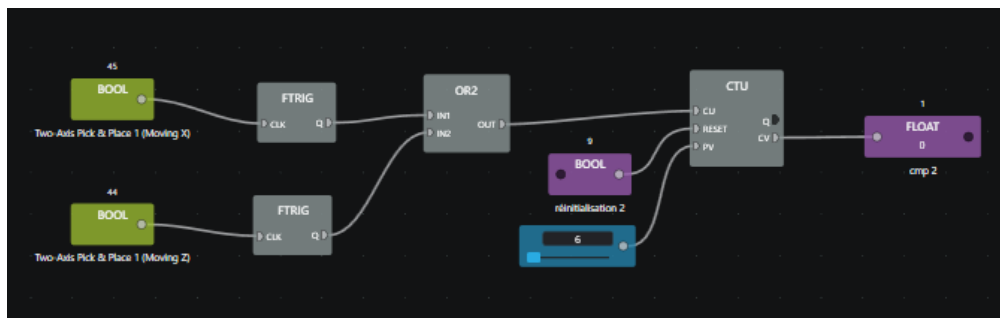


FIG. 2.27 : utilisation du compteur pour le suivi des mouvements sur les axes X et Z

Cette figure détaille l'utilisation d'un compteur afin de suivre les cycles du robot. Cela permet une automatisation séquentielle fiable et un contrôle précis du nombre d'opérations effectuées.

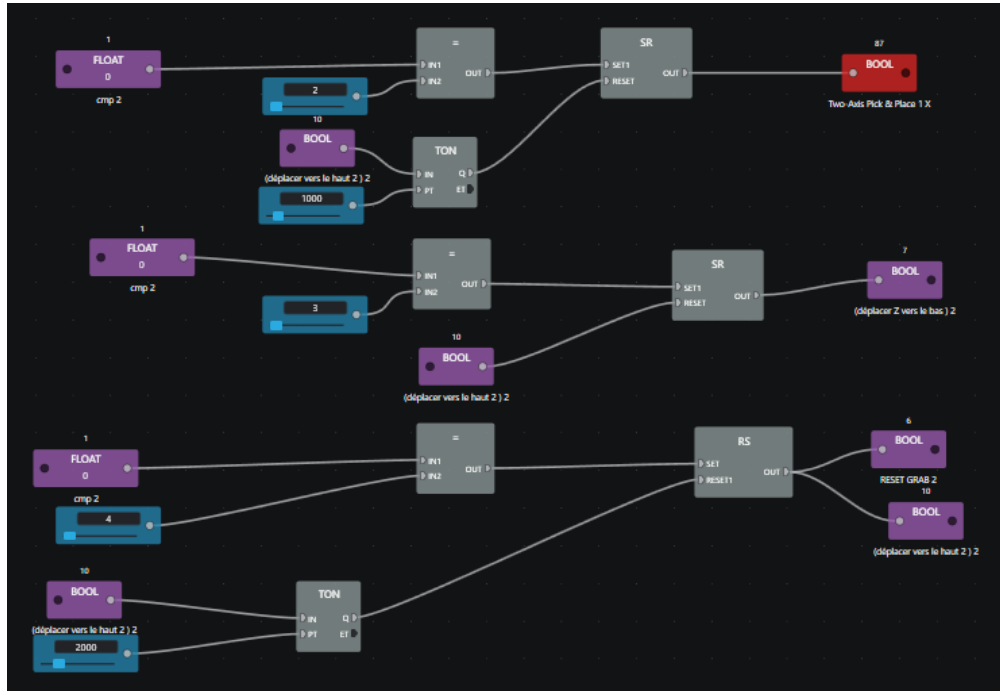


FIG. 2.28 : commande des déplacements horizontaux (axe X) du robot d'assemblage

La figure présente la logique programmée pour déplacer le robot horizontalement entre les deux pièces. Ce déplacement est crucial pour aligner le couvercle avec la base sans désynchronisation.

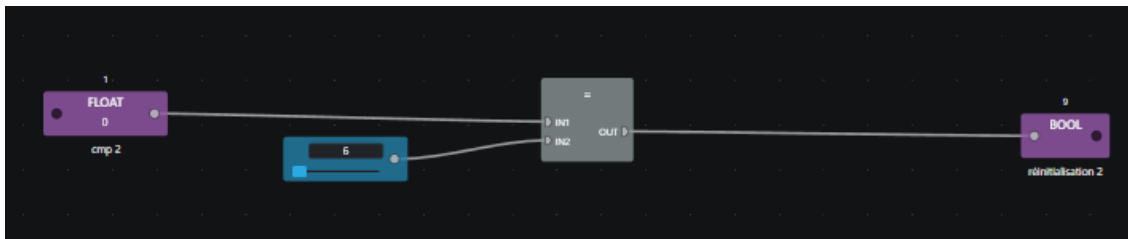


FIG. 2.29 : logique de redémarrage du cycle du processus d'assemblage

Cette figure illustre le redémarrage du cycle après l'assemblage d'un produit. Elle garantit que le système est prêt pour le cycle suivant.

2.6.5 Suivi de production :

Un compteur a été ajouté à la sortie de la station afin de déterminer le nombre de produits finis assemblés. Cela permet de suivre la production en temps réel.

2.7 Conclusion

Ce chapitre a permis de mettre en lumière le potentiel considérable de Factory I/O dans l'apprentissage, le prototypage et l'optimisation des systèmes automatisés. Grâce à une modélisation réaliste et interactive, ce logiciel offre une plateforme idéale pour concevoir et tester virtuellement des lignes de production complètes. L'intégration avec Control I/O facilite également la programmation des automates via des diagrammes de blocs fonctionnels, sans avoir besoin d'un matériel physique. À travers l'exemple concret d'une station de fabrication composée de modules de distribution, d'usinage et d'assemblage, nous avons pu démontrer l'efficacité de la simulation dans le suivi et le contrôle de la production. Ce travail souligne ainsi l'importance croissante des outils numériques dans l'ingénierie de l'automatisation et dans la préparation aux défis de l'industrie 4.0.

CHAPITRE 3

PROGRAMMATION, CONFIGURATION ET SIMULATION DU SYSTEME DE PRODUCTION AUTOMATISE VIA TIA PORTAL

3.1 Introduction

Dans le domaine de l'automatisation industrielle, la maîtrise des outils logiciels modernes constitue un facteur essentiel de réussite pour le développement, le test et la supervision des systèmes automatisés. Le présent chapitre s'intéresse à la plateforme TIA Portal, développée par Siemens, qui offre un environnement d'ingénierie intégré permettant la programmation, la configuration et la simulation des automates programmables industriels.

Ce chapitre met particulièrement en lumière l'implémentation d'une ligne de production automatisée à l'aide de TIA Portal, en intégrant la simulation avec S7-PLCSIM et la supervision visuelle via Factory I/O. À travers une démarche progressive – de la description du matériel à la réalisation des programmes en GRAFCET – il illustre comment ces outils s'articulent pour concevoir un système flexible, fiable et performant.

Une attention particulière est également portée à la communication entre Factory I/O et TIA Portal, qui permet une interaction dynamique entre la logique de contrôle et la simulation visuelle du processus industriel.

3.2 Présentation de logiciel TIA Portal

Le TIA Portal de Siemens s'impose comme une référence dans le domaine de l'automatisation industrielle grâce à son interface intuitive et son intégration complète. Conçu pour optimiser la productivité, il facilite à la fois la conception et la maintenance des systèmes automatisés. Il représente ainsi un choix stratégique pour des opérations performantes et un investissement durable.

TIA Portal constitue une solution idéale pour les entreprises souhaitant accroître l'efficacité de leurs processus d'automatisation. Son ergonomie et sa capacité à réduire significativement les temps de développement en font un outil précieux. Pour toute organisation en quête d'une solution fiable, évolutive et pérenne, le TIA Portal de Siemens s'impose comme une option incontournable. [24]

3.2.1 Description du logiciel TIA Portal

La plateforme Totally Integrated Automation Portal (TIA Portal) représente le nouvel environnement de développement unifié proposé par Siemens pour concevoir des solutions d'automatisation. Elle regroupe de manière harmonieuse les principaux outils d'ingénierie, notamment les logiciels SIMATIC Step7 pour la programmation des automates et SIMATIC WinCC pour la supervision. [25]



FIG. 3.1 : Logo de TIA Portal version 15 (Siemens) [26]

- **SIMATIC STEP 7 :**

SIMATIC STEP 7 est l'un des outils d'ingénierie les plus reconnus et utilisés mondialement dans le domaine de l'automatisation industrielle. La version V15 s'intègre dans l'environnement de développement TIA Portal, permettant à l'utilisateur de configurer, programmer, tester et diagnostiquer les automates programmables de la gamme SIMATIC.

- **SIMATIC WinCC :**

SIMATIC WinCC, intégré dans le TIA Portal, est un logiciel dédié à la configuration des interfaces homme-machine (IHM), qu'il s'agisse de pupitres SIMATIC, de PC industriels ou d'ordinateurs standards. Il propose un environnement unifié pour la programmation, la supervision et la visualisation des processus industriels. WinCC couvre un large éventail d'applications, allant des IHM de base sur Basic Panels jusqu'à des systèmes SCADA multipostes sur PC. [25]

A_ Vue de TIA Portal

Lors du lancement de TIA Portal, l'environnement de travail se divise en deux modes principaux :

- **La vue Portail :** orientée vers les différentes tâches à accomplir, elle est conçue pour une prise en main rapide. Chaque portail est dédié à une catégorie d'actions (programmation, configuration matérielle, etc.). L'interface affiche les différentes opérations disponibles en fonction de la tâche sélectionnée.[25]

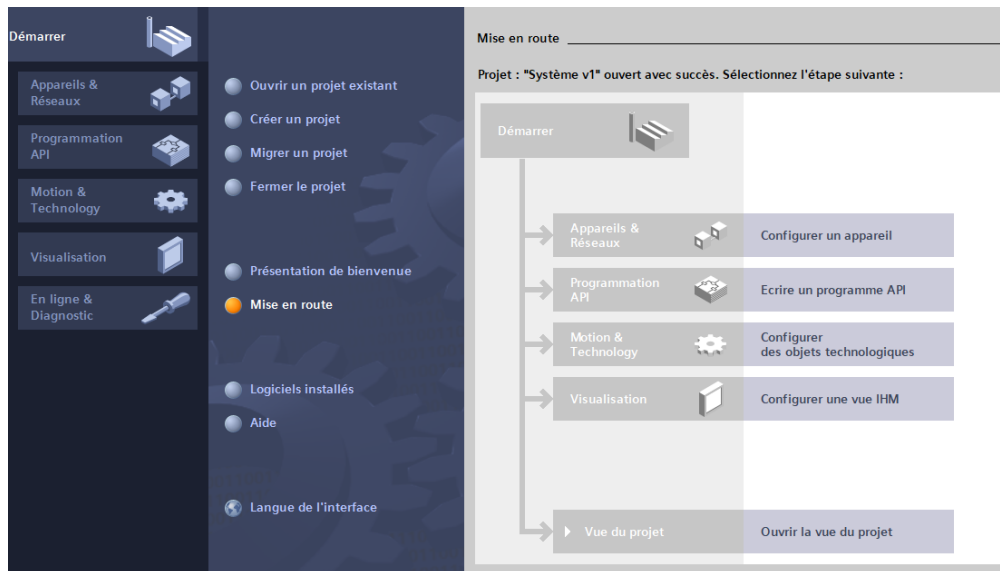


FIG. 3.2 : Vue du Portail

- **La vue Projet** : cette vue présente une arborescence complète des composants du projet (automates, HMI, réseaux, etc.). Elle permet l'accès direct aux éditeurs nécessaires selon les actions en cours. Les paramètres, les données et les éditeurs sont regroupés dans une interface unifiée facilitant la navigation et la modification du projet. [25]

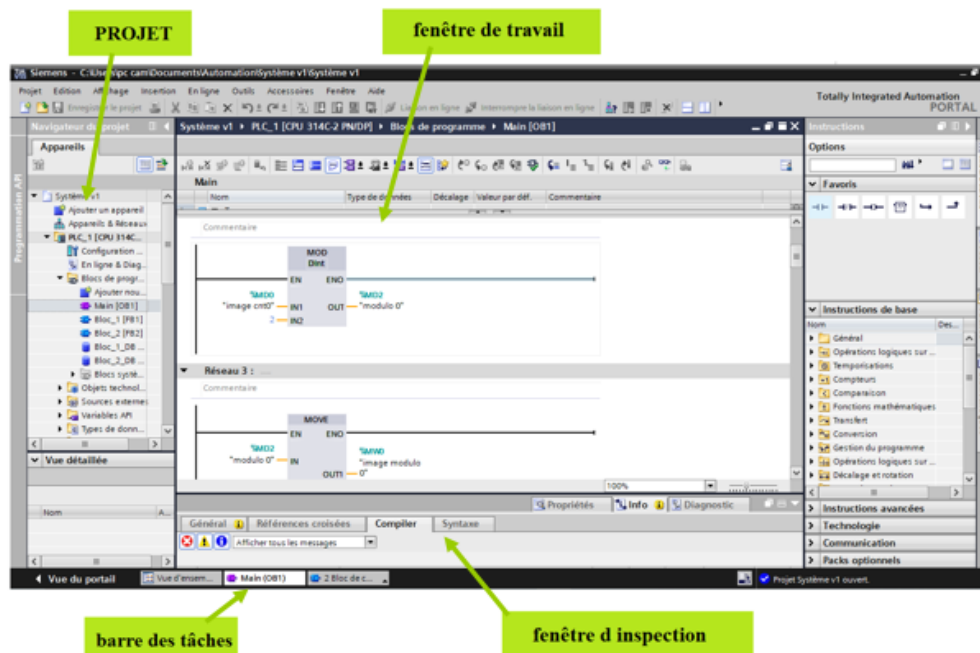


FIG. 3.3 : Vue du Projet

B_ Blocs de programmation

Les automates proposent différents types de blocs permettant de structurer et d'organiser le programme ainsi que les données associées. En fonction des besoins et de la complexité du processus, le programme peut être divisé en plusieurs catégories de blocs : OB, FB, FC et DB.

- **Blocs d'organisation (OB)** : Ils assurent la gestion du programme utilisateur. Ce sont les points d'entrée du programme, déclenchés par des événements spécifiques (cycle, interruption, démarrage, etc.)
 - **Blocs fonctionnels (FB)** : Ce sont des blocs de code capables de mémoriser des valeurs via des blocs de données d'instance. Ils conservent les données entre deux exécutions, ce qui permet une gestion d'états persistants.
 - **Fonctions (FC)** : Ces blocs contiennent des séquences d'instructions réutilisables. Contrairement aux FB, les FC ne conservent pas de données entre les appels ils ne disposent pas de mémoire propre.
 - **Blocs de données (DB)** : Ce sont des espaces de stockage pour les données utilisateur. Ils peuvent être associés à des FB (blocs de données d'instance) ou utilisés de manière globale pour stocker des informations partagées.
- [25]

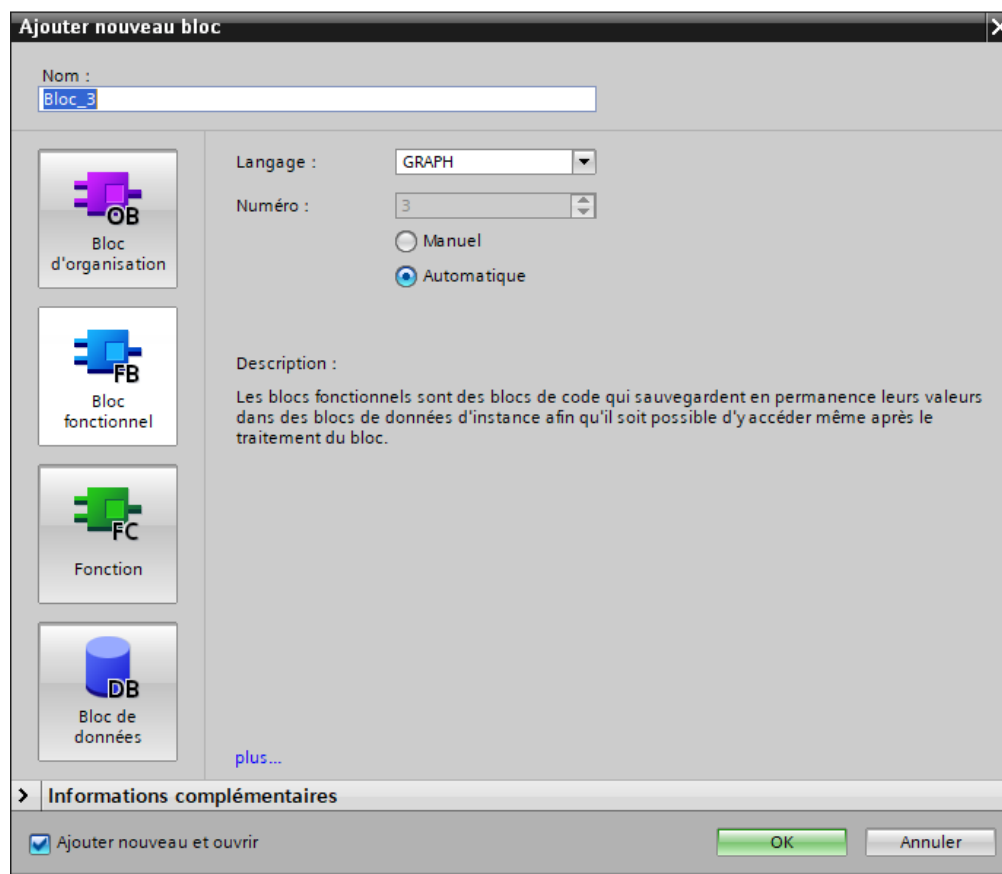


FIG. 3.4 : Fenêtre de création d'un nouveau bloc

3.2.2 Langages de programmation

Le TIA Portal offre de puissants éditeurs pour la programmation des automates SIMATIC S7. Il prend en charge différents langages conformes à la norme IEC 61131-3, parmi lesquels :

- Schéma à contacts (Ladder Diagram - LD).
- Diagramme de blocs fonctionnels (Function Block Diagram - FBD).
- Liste d'instructions (Instruction List - IL), également appelée STL dans TIA Portal.
- Langage séquentiel (GRAPH) basé sur le GRAFCET.

Ces langages sont disponibles pour tous les types d'automates. L'utilisateur bénéficie d'outils intuitifs facilitant l'ensemble des tâches de programmation. [25]

Le tableau suivant présente quelques fonctions implémentées à l'aide de trois de ces langages :

Désignation	Langage LD	Langage IL	Langage FBD
ET Logique		U	
ET NON		UN	
OU Logique		O	
OU NON		ON	
Affectation résultat		=	
ET d'une expression		U(
OU d'une expression		O(
Exécute l'instruction différée)	
Definer repère (Etiquette)		: <repère>	
Saut		SPA	

FIG. 3.5 : Tableau présente trois Langages de programmation des API [27]

A_ Diagramme de blocs fonctionnels (Function Block Diagram - FBD)

C'est un langage de programmation graphique basé sur l'utilisation de blocs représentant les opérations logiques selon l'algèbre de Boole. Il permet également d'intégrer directement des fonctions complexes, telles que les opérations mathématiques, en les combinant avec les blocs logiques.

B_ Schéma à contacts (Ladder Diagram - LD)

C'est un langage de programmation graphique très répandu chez les automaticiens. Il s'inspire des schémas électriques, ce qui le rend intuitif et facile à comprendre. Il se compose de trois types d'éléments principaux :

- **Les entrées (ou contacts)** : elles servent à lire l'état d'une variable booléenne.
- **Les sorties (ou bobines)** : elles permettent d'écrire une valeur dans une variable booléenne.
- **Les blocs fonctionnels** : ils sont utilisés pour exécuter des fonctions plus complexes.

C_ Liste d'instructions (Instruction List - IL)

C'est un langage textuel de bas niveau, basé sur une instruction par ligne. Il convient particulièrement aux applications de petite taille. Chaque instruction agit sur un registre appelé résultat courant (ou registre IL). L'opérateur spécifie le type d'opération à réaliser entre ce résultat courant et l'opérande. Le résultat obtenu est ensuite stocké de nouveau dans le registre du résultat courant. [25]

D_ GRAFCET

Le GRAFCET (acronyme de GRAPhe Fonctionnel de Commande Étapes/Transitions) est un diagramme fonctionnel destiné à la description précise des systèmes séquentiels. Il permet de représenter graphiquement le fonctionnement d'un système automatisé.

Utilisé comme outil de modélisation du cahier des charges, le GRAFCET est normalisé, sans ambiguïté, et reste facile à comprendre et à utiliser, ce qui le distingue d'autres langages de description plus complexes.

Ce mode de représentation est particulièrement bien adapté aux automatismes à évolution séquentielle, c'est-à-dire aux systèmes pouvant être décomposés en étapes successives. L'un de ses atouts majeurs réside dans la transition fluide entre la modélisation fonctionnelle et l'implémentation technologique dans un automate programmable industriel (API).

Ainsi, le GRAFCET permet de passer aisément du langage de spécification au langage d'implémentation, facilitant ainsi la mise en œuvre concrète de l'automatisme.[28]

□ Eléments de base de Grafcet

Le Grafcet est un graphique structuré composé de séquences d'étapes et de transitions, connectées entre elles par des liaisons orientées.[29]

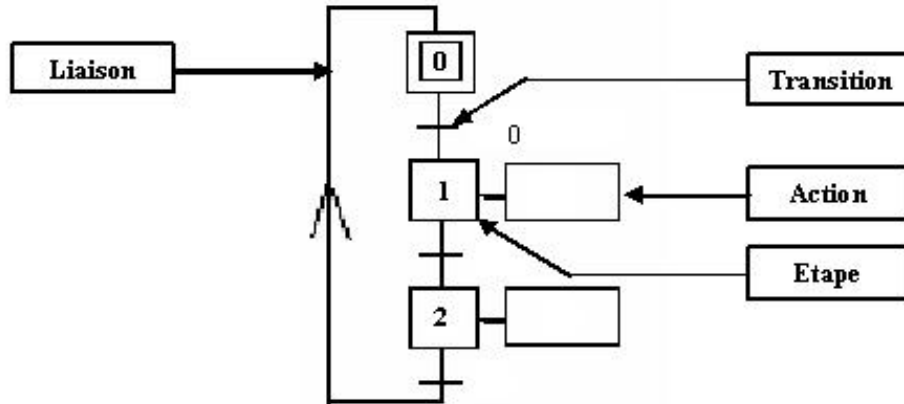


FIG. 3.6 : Les éléments de base du GRAFCET [30]

- **Étape** : Une étape correspond à un état stable de l'automatisme, c'est-à-dire un état qui ne change pas en fonction des entrées/sorties. Elle peut être active ou inactive. À un instant donné, l'état global du Grafcet est défini par l'ensemble des étapes actives.
- **Transition** : La transition représente la possibilité de passer d'un état (étape) à un autre. Ce changement se produit lorsque la transition est franchie. Une transition n'est validée que si toutes les étapes immédiatement précédentes (en amont) sont actives.
- **Liaison orientée** : Il s'agit d'un lien directionnel reliant une étape à une transition (ou inversement), permettant de définir les enchaînements possibles à partir d'un état donné. [29]

3.2.3 Présentation de S7 PLCSIM

Le simulateur S7-PLCSIM est un outil logiciel qui permet d'exécuter et de tester un programme destiné à un automate programmable industriel (API) dans un environnement virtuel, directement sur un ordinateur ou via une console de programmation. En l'absence d'un automate réel, S7-PLCSIM devient un outil indispensable pour simuler les programmes et les concepts de commande automatisée avant leur implantation sur un système réel.

S7-PLCSIM offre une interface simple et conviviale qui permet de visualiser, contrôler et forcer les différents paramètres utilisés par le programme, comme par exemple activer ou désactiver des entrées, tout en exécutant le programme dans l'automate virtuel.

Ce simulateur est intégré au gestionnaire de projet SIMATIC et peut être utilisé dès lors qu'aucune liaison à un automate réel n'est établie. Il facilite grandement la validation du programme, la correction des erreurs, ainsi que la visualisation du comportement des sorties, rendant possible la simulation complète et fiable du fonctionnement de l'unité automatisée.

En résumé, S7-PLCSIM est un outil puissant pour tester, valider et optimiser les programmes d'automates Sie-

mens avant leur mise en service réelle, réduisant ainsi les risques d'erreurs sur le terrain et améliorant l'efficacité du développement.[25]

3.2.4 Caractéristiques de TIA Portal

- **Gain de temps et d'efficacité** : réduction du temps de conception pouvant atteindre 30%.
- **Environnement intégré** : regroupe automate, IHM et systèmes d'entraînement dans une seule plateforme.
- **Logiciel ergonomique** : interface intuitive, accessible même aux utilisateurs débutants.
- **Bibliothèque optimisée** : réutilisation de composants existants pour accélérer le développement.
- **Outils de diagnostic intégrés** : diminuent les temps d'arrêt en facilitant la détection des pannes.
- **Large éventail d'applications** : prise en charge des fonctions de motion, d'entraînement et de sécurité.
- **Courbe d'apprentissage réduite** : interface unifiée pour tous les éditeurs, simplifiant la prise en main.
- **Sécurité renforcée** : intégration de technologies de protection avancées pour les machines et les données.[24]

3.3 Présentation d'automate S7-300

L'automate programmable industriel S7-300, développé par Siemens, fait partie intégrante de la gamme SIMATIC S7. Il représente une plateforme d'automatisation à la fois modulaire et flexible, conçue pour répondre aux besoins des systèmes centralisés comme décentralisés, et parfaitement adaptée aux applications de complexité intermédiaire à élevée.

Cet automate est capable de gérer jusqu'à 512 entrées/sorties numériques et 64 entrées/sorties analogiques. Il permet l'intégration de 32 modules de signaux, répartis entre une unité centrale (CPU) et trois extensions supplémentaires. Cette structure modulaire favorise une grande liberté de configuration, tout en conservant un format compact, sans nécessiter de ventilation forcée.

Les différents modules sont montés sur un rail métallique, garantissant un assemblage rigide et durable. Cette conception offre une excellente résistance aux contraintes typiques du milieu industriel, telles que l'humidité, les interférences électromagnétiques et les chocs mécaniques.

Le S7-300 est conçu pour fonctionner dans des conditions climatiques variées, avec une plage de température opérationnelle allant de -25 °C à +60 °C, ce qui le rend particulièrement adapté aux environnements industriels exigeants.

Sur le plan matériel, la configuration minimale d'un S7-300 comprend un module d'alimentation, une unité centrale (CPU), un coupleur, ainsi que des modules d'entrées/sorties.

Enfin, la programmation et la configuration du système s'effectuent via le logiciel TIA Portal (STEP 7). [31]



FIG. 3.7 : : Automate Siemens SIMATIC S7-300 – CPU 314C-2 PN/DP [32]

3.3.1 Modularité du S7-300 :

Le S7-300 se distingue par sa conception entièrement modulaire, offrant une large gamme de modules adaptables aux besoins spécifiques de chaque application d'automatisation. Cette flexibilité permet de composer librement la configuration du système en combinant différents types de modules selon les exigences techniques du projet.

A_ Module d'alimentation

Le choix d'un module d'alimentation adapté est une étape cruciale lors de la configuration du S7-300. Deux éléments doivent être soigneusement pris en compte :

- La consommation électrique des modules installés.
- La puissance thermique dissipée par ces modules.

Ces deux facteurs influencent directement les performances et la stabilité du système, et doivent être évalués dès la phase de planification. La consommation globale comprend à la fois l'énergie nécessaire au fonctionnement sur le bus interne et celle requise via l'alimentation externe en 24 VCC.

Siemens propose trois références principales pour l'alimentation du S7-300, selon les besoins en intensité :

- PS 307 / 2A
- PS 307 / 5A
- PS 307 / 10A

B_ Unité centrale

L'unité centrale (CPU) constitue le cœur opérationnel de l'automate, assurant le traitement logique et la gestion du système. Elle remplit plusieurs fonctions essentielles :

- Lecture des signaux d'entrée
- Exécution du programme utilisateur et commande des sorties
- Gestion du comportement au démarrage et diagnostic des anomalies

La gamme S7-300 propose un large éventail de CPU aux capacités variées, permettant de répondre à différents besoins en termes de performances et de complexité des tâches.

Dans notre configuration, le choix s'est porté sur la CPU 314C-2 PN/DP, dont les caractéristiques techniques sont détaillées dans le tableau 3.1.

C_ Coupleurs

Les coupleurs sont des cartes électroniques essentielles au fonctionnement modulaire de l'automate. Leur rôle principal est d'assurer la communication entre l'unité centrale (CPU) et les modules d'entrées/sorties, qu'ils soient situés sur le châssis principal ou sur des châssis d'extension.

La transmission des données entre la CPU et les E/S s'effectue via un bus interne, auquel les coupleurs servent de liaison physique.

Dans l'architecture du S7-300, les coupleurs permettent le raccordement d'un ou plusieurs châssis d'extension au châssis de base, assurant ainsi une évolutivité du système sans compromettre la fiabilité de la communication.

Trois modèles de coupleurs sont disponibles pour le S7-300 : IM 360, IM 361, IM 365.

D_ Modules de signaux

Les modules de signaux du S7-300 assurent l'interface entre l'automate et les éléments du processus, tels que capteurs et actionneurs. Ils permettent le transfert et l'adaptation des signaux entre le monde extérieur et le système de traitement interne de l'automate.

On distingue plusieurs types de modules :

- **Modules d'entrées TOR (tout ou rien) :** Ils adaptent les signaux discrets provenant des capteurs ou contacts (comme des détecteurs de proximité) au niveau de traitement du S7-300.
- **Modules de sorties TOR :** Ils convertissent les signaux internes de l'automate en signaux de puissance utilisables pour commander des dispositifs tels que électrovannes, lampes, contacteurs ou micromoteurs.
- **Modules d'entrées/sorties analogiques :** Ces modules effectuent la conversion entre les signaux analogiques du processus (ex. tension, courant) et les signaux numériques nécessaires au traitement par l'automate, et inversement.

E_ Module de fonction (FM) :

Les modules de fonction (FM) permettent de délester la CPU des calculs complexes, en prenant en charge certaines tâches spécifiques de manière autonome. Parmi les plus courants, on retrouve :

- **FM 354 et FM 357** : modules de commande d'axes pour servomoteurs,
- **FM 353** : module de positionnement pour moteurs pas à pas,
- **FM 355** : module dédié à la régulation,
- **FM 350-1 et FM 350-2** : modules de comptage.

F_ Module de communication (CP)

Les modules de communication (CP) sont conçus pour gérer les échanges de données via des transmissions série. Ils permettent d'établir des liaisons point à point, notamment avec :

- Des pupitres opérateurs
- D'autres automates, comme les SIMATIC S7, SIMATIC S5, ou des équipements d'autres fabricants.

G_ Module d'extension (UR)

: Le module d'extension (UR) assure la liaison électrique entre les différents modules d'un système réparti. Il comprend un rail support en aluminium et un bus de fond de panier avec connecteurs, généralement installé dans des armoires électriques, garantissant un câblage structuré et sécurisé.

H_ Raccordement avec la console de programmation

Les consoles de programmation Siemens servent à la création, modification, archivage et supervision des programmes d'automatisation. Grâce à la plateforme logicielle TIA Portal V15, l'utilisateur dispose d'une suite complète pour toutes les tâches de développement.

Le raccordement entre l'automate et la console s'effectue via l'interface multipoint (MPI), qui assure une communication fiable et rapide entre l'automate et les outils de développement [31]

3.3.2 Caractéristiques de la CPU 314C-2 PN/DP

La CPU 314C-2 PN/DP fait partie de la gamme des automates programmables SIMATIC S7-300 développée par Siemens. Elle se distingue par ses capacités étendues en matière de traitement, de communication et de contrôle de processus. Le tableau suivant présente les principales caractéristiques techniques de ce module :

Caractéristique	Détail
Référence	6ES7 314-6EH04-0AB0
Famille	SIMATIC S7-300
Version firmware	V3.3
Mémoire de travail	192 Ko
Temps de traitement	0,06 ms / 1 000 instructions
Entrées TOR (DI)	24
Sorties TOR (DO)	16
Entrées analogiques (AI)	5
Sorties analogiques (AO)	2
Sorties d'impulsions	4 (jusqu'à 2,5 kHz)
Voies de comptage avec codeurs incrémentaux	4 voies (24 V / 60 kHz)
Fonctions intégrées	<ul style="list-style-type: none"> • Positionnement (2 axes) • Régulation (2 boucles PID)
Interfaces de communication	<ul style="list-style-type: none"> • MPI/DP intégré • PROFINET IO / PROFINET CBA / Proxy CBA • PROFIBUS DP
Protocoles supportés	TCP/IP
Capacité maximale d'extension	Jusqu'à 31 modules
Fonctionnalités supplémentaires	<ul style="list-style-type: none"> • Émission / réception directe de données • Routage • Équidistance

TAB. 3.1 : caractéristiques de la CPU 314C-2PN/DP. [33]

3.4 CAHIER DE CHARGE

3.4.1 Démarrage et distribution des marchandises

Lorsque le bouton de démarrage (Start Button 0) est activé, son voyant lumineux (Start Button 0 Light) s'allume pour signaler le début du cycle. L'émetteur (Emitter 0) commence alors à distribuer des pièces à des intervalles réguliers compris entre 17 et 20 secondes.

En parallèle, le convoyeur Belt Conveyor (6m) 3 est activé et fonctionne à une vitesse de 0,6 m/s.

Lorsque le capteur de vision (Vision Sensor 1) détecte une pièce et que le poussoir 0 (Pusher 0) est en position arrière (confirmée par le capteur de fin de course Back Limit 0), celui-ci se déclenche pour pousser la pièce. Le mouvement se poursuit jusqu'à l'activation du capteur de fin de course Front Limit 0, indiquant que la position avant est atteinte, ce qui entraîne l'arrêt automatique du poussoir.

Simultanément à l'activation du capteur de vision 1, les convoyeurs suivants sont mis en marche pour orienter la pièce :

- Belt Conveyor (2m) 1
- Belt Conveyor (2m) 2
- Belt Conveyor (2m) 6
- Belt Conveyor (6m) 4

- Curved Belt Conveyor 2 CCW
- Curved Belt Conveyor 6 CW

3.4.2 Acheminement des pièces vers les centres d'usinage et la fonctionnement de centre d'usinage

Lorsqu'une pièce est détectée par le capteur de présence (Diffuse Sensor 4), un compteur est activé pour suivre le nombre de pièces distribuées.

Cas 1 : Nombre pair

Si le nombre est pair et que le poussoir 2 (Pusher 2) est inactif (en position arrière, détectée par Back Limit 2), alors :

- La pièce continue automatiquement son chemin sur le convoyeur vers le positionneur 6 (Right Positioner 6).
- Lorsqu'elle est détectée par le capteur de présence (Diffuse Sensor 0), le positionneur serre la pièce (Clamp 6).
- Une fois la pièce serrée, le capteur (Clamped 6) reste activé, confirmant que la pièce est maintenue.
- Ensuite, le processus dépend de trois conditions de la machine :
 - La machine n'est pas en cours de traitement (In Busy inactif)
 - La porte est ouverte (Opened activé)
 - Aucune erreur n'est détectée (Has Error inactif).
- Si ces conditions sont réunies, le positionneur :
 - Libère la pièce
 - Effectue un mouvement vertical (Raise 6) vers la position haute pour insérer la pièce dans le centre d'usinage.
 - Pendant ce déplacement, le capteur (Limit6) s'active, indiquant que le positionneur est soit en bas, soit en haut.
- **Démarrage du Centre d'Usinage 0** : Depuis l'armoire de commande, l'opérateur doit :
 - Appuyer sur Reset (MC Reset 0) pour réinitialiser les erreurs
 - Puis sur Start (MC Start 0) pour démarrer le centre d'usinage

À ce moment-là, si le centre d'usinage est en mode marche et que la pièce est entrée le centre d'usinage, le robot :

- Saisit la pièce automatiquement
- Insère la pièce dans la machine CNC pour fabriquer un Couvercle.

Du fait que le robot saisit la pièce, le positionneur 6 effectue un mouvement vertical vers le bas (état initial)

Cas 2 : Nombre impair

Si le nombre du compteur est impair et que le poussoir 2 (Pusher 2) est en position arrière (Back Limit 2 activé), alors :

- Le poussoir 2 (Pusher 2) est déclenché pour pousser la pièce sur une voie différente.
 - Le mouvement se poursuit jusqu'à ce que Front Limit 2 soit activé, signalant l'arrêt du poussoir.
 - Dès l'activation de Front Limit 2, les convoyeurs suivants se mettent en marche :
 - Belt Conveyor (6m) 1
 - Curved Belt Conveyor 0 CW
 - Belt Conveyor (4m) 0
 - La pièce est ainsi acheminée vers le positionneur 0 (Right Positioner 0).
 - Lorsqu'elle est détectée par le capteur (Diffuse Sensor 5), le positionneur 0 serre automatiquement la pièce (Clamp 0).
 - Une fois serrée, le capteur (Clamped 0) reste activé pour confirmer la prise de la pièce.
 - Comme précédemment, si :
 - La machine est vide (In Busy 1 inactif)
 - La porte est ouverte (Opened 1 activé)
 - Et aucune erreur n'est présente (Has Error 1 inactif)
 - Alors le positionneur
 - Libère la pièce
 - Effectue un mouvement vertical (Raise 0) pour insérer la pièce dans le centre d'usinage
 - Le capteur Limit 0 valide le déplacement en hauteur ou en bas
 - **Démarrage du Centre d'Usinage 1** : Depuis l'armoire de commande, l'opérateur doit :
 - Appuyer sur Reset (MC Reset 1) pour effacer les erreurs
 - Puis sur Start (MC Start 1) pour mettre en marche le centre d'usinage
 - Ensuite, le robot :
 - Prend la pièce
 - Insère la pièce dans la machine CNC pour produire une Bas
- Du fait que le robot saisit la pièce, le positionneur 0 effectue un mouvement vertical vers le bas (état initial)

3.4.3 Assemblage

- Lorsque le capteur Diffuse Sensor 2 détecte un couvercle, le positionneur 2 serre automatiquement la pièce.
- Le capteur Clamped 2 reste activé pour confirmer la prise du couvercle.
- Lorsque le capteur Diffuse Sensor 3 détecte une base, le positionneur 1 serre automatiquement la pièce.
- Le capteur Clamped 1 reste activé pour confirmer la prise de la base.

Dès que les deux positionneurs ont serré leurs pièces (base + couvercle), le robot à deux axes (Pick & Place 0) commence son cycle :

- Il effectue un mouvement vertical vers le bas (Move Z)
- Une fois en bas, le capteur Detected, intégré à la pince, détecte la présence de couvercle.
- Si le couvercle est bien détecté :
 - La ventouse du robot est activée (Grab) pour saisir le couvercle.
- Une fois la ventouse activée :
 - Le robot effectue un mouvement vertical vers le haut (retour en position haute).
 - Le positionneur 2 libère automatiquement le couvercle (arrêt du serrage).
- En suit :
 - Le robot effectue un mouvement horizontal vers la droite (move X).
 - Une fois en position, il descend à nouveau (move Z activé).
 - S’il descend, la ventouse est désactivée, ce qui libère le couvercle pour l’assemblage avec la base.
- Après avoir déposé l’assemblage :
 - Le robot effectue un mouvement vertical vers le haut (remontée).
 - En suit un mouvement horizontal vers la gauche pour revenir à sa position initiale.
- En parallèle :
 - Le positionneur 1 relâche la base
 - Puis il effectue un mouvement vertical vers le haut (Raise 1) pour placer le produit assemblé sur le convoyeur de sortie
- Comptage des produits finis
 - À la fin du convoyeur de sortie se trouve un capteur de présence Diffuse Sensor 6.
 - Chaque fois qu’une pièce assemblée passe devant ce capteur, un compteur est incrémenté de 1 pour suivre le nombre de produits finis sortis du système.
 - Tableau des Entrées/Sorties du Système

Le tableau ci-dessous récapitule les différentes entrées et sorties utilisées dans le système automatisé de la première ligne de production, spécialisée dans la fabrication des pièces vertes. Il précise les symboles, les adresses associées ainsi que les commentaires décrivant leur fonction respective. La deuxième ligne de production utilise les mêmes entrées et sorties ; seules les adresses diffèrent.

Remarque :

Ce cahier des charges décrit spécifiquement le fonctionnement de la première ligne de production, dédiée à la fabrication des pièces vertes. Une deuxième ligne de production, structurellement identique à la première, est également en place. La seule différence notable réside dans la nature des pièces produites : la deuxième ligne est spécialisée dans la fabrication de pièces métalliques. Les équipements, capteurs, actionneurs et séquences d’automatisation restent les mêmes, à l’exception des adresses de certains composants, adaptées à cette seconde ligne

3.5 Tableau des Entrées/Sorties du Système

Le tableau ci-dessous récapitule les différentes entrées et sorties utilisées dans le système automatisé de la première ligne de production, spécialisée dans la fabrication des pièces vertes. Il précise les symboles, les adresses associées ainsi que les commentaires décrivant leur fonction respective.

La deuxième ligne de production utilise les mêmes entrées et sorties ; seules les adresses diffèrent.

tab. 3.2 : Tableau représente des Entrées/Sorties du Système

Élément	Symbole	Adresse	Commentaire
Start Button 0	Start0	I_0.0	Un bouton poussoir utilisé pour démarrer la ligne de production.
Stop Button 0	Stop0	I_0.1	Un bouton d'arrêt pour interrompre la ligne de production.
Reset Button 0	Reset0	I_0.2	Bouton pour réinitialiser le cycle de production.
Emergency Stop 0	AU	I_0.3	Arrêt d'urgence.
Vision Sensor 1	VS1	I_0.4	Capteur de vision utilisé pour détecter les pièces selon leur couleur, déclenche des actions selon l'identification.
Back Limit 0	BL0	I_0.6	Capteur de fin de course arrière : indique que le pusher (poussoir) est en position de repos.
Back Limit 2	BL2	I_1.0	/
Front Limit 0	FL0	I_1.2	Capteur de fin de course avant : indique que le pusher (poussoir) a atteint sa course maximale (fin de poussée).
Front Limit 2	FL2	I_1.4	/
Diffuse Sensor 0	DV0	I_1.6	Capteur optique de proximité qui détecte une pièce lorsqu'elle passe devant, sans réflecteur. Très utilisé pour compter ou détecter le passage d'objets.
Diffuse Sensor 2	DS2	I_2.0	/
Diffuse Sensor 3	DS3	I_2.1	/
Diffuse Sensor 4	DS4	I_2.2	/
Diffuse Sensor 5	DS5	I_2.3	/
Diffuse Sensor 6	DS6	I_2.4	/
Right Positioner 0 (Limit)	LT0	I_3.2	Capteur de fin de course vertical (haut/bas) du positionneur. Sert à savoir si le bras de positionnement est en position haute ou basse.
Right Positioner 0 (Clamped)	Clamped0	I_3.3	Capteur qui indique que le positionneur a fermé sa pince.
Right Positioner 1 (Limit)	LT1	I_3.4	/
Right Positioner 1 (Clamped)	Clamped1	I_3.5	/
Right Positioner 4 (Limit)	LT4	I_4.2	/
Right Positioner 4 (Clamped)	Clamped4	I_4.3	/

Right Positioner 6 (Limit)	LT6	I_4.6	/
Right Positioner 6 (Clamped)	Clamped6	I_4.7	/
Machining Center 0 (Has Error)	Error0	I_5.2	Signal d'erreur provenant d'un centre d'usinage : indique qu'un défaut ou une anomalie bloque le processus.
Machining Center 0 (Opened)	Opened0	I_5.3	Capteur détectant que la porte du centre d'usinage est ouverte
Machining Center 0 (Progress)	Progress0	I_5.4	Signal numérique qui indique la progression de l'usinage, généralement exprimée de 0 à 100 %, et permet de suivre l'avancement du cycle d'usinage en temps réel.
Machining Center 0 (Is Busy)	By0	I_5.5	capteur montrant que le centre d'usinage est en cours de traitement (pièce en cours).
Machining Center 1 (Has Error)	Error1	I_5.6	/
Machining Center 1 (Opened)	Opened1	I_5.7	/
Machining Center 1 (Progress)	Progress1	I_6.0	/
Machining Center 1 (Is Busy)	By1	I_6.1	/
Moving Z 0	Mz0	I_7.2	Capteur indiquant que l'axe Z du robot est en mouvement vertical (descente ou montée en cours).
Moving X 0	Mx0	I_7.3	Capteur indiquant que l'axe X du robot (Pick & Place) est en mouvement horizontal (gauche/droite).
Detected 0	Detected0	I_7.4	Le capteur Detected est un capteur de proximité optique intégré à la pince du robot, qui détecte la présence d'un objet.
Gripper Rotating 0	GR0	I_7.5	Capteur indiquant que la pince (gripper) est en cours de rotation.
Rotating 0	RT0	I_7.6	Capteur signalant un mouvement rotatif du robot Pick & Place.
Emitter 0	Emitter 0	Q_0.0	Génère les pièces à intervalles réguliers pour lancer le cycle de production.
Remover 0	Remover 0	Q_0.2	Évacue les pièces.
Pusher 0	Pusher 0	Q_0.4	Dispositif utilisé pour pousser une pièce d'un convoyeur vers une autre zone.
Pusher 2	Pusher 2	Q_0.6	/
Right Positioner 0 (Clamp)	Clamp 0	Q_1.0	Assure le serrage de la pièce.
Right Positioner 0 (Raise)	Raise 0	Q_1.1	Élève ou abaisse la pince de positionneur.
Right Positioner 1 (Clamp)	Clamp 1	Q_1.2	/
Right Positioner 1 (Raise)	Raise 1	Q_1.3	/
Right Positioner 4 (Clamp)	Clamp 4	Q_2.0	/

Right Positioner 4 (Raise)	Raise 4	Q_2.1	/
Right Positioner 6 (Clamp)	Clamp 6	Q_2.4	/
Right Positioner 6 (Raise)	Raise 6	Q_2.5	/
Machining Center 0 (Start)	MC start 0	Q_3.0	Démarre l'usinage automatique une fois les conditions réunies.
Machining Center 0 (Reset)	MC Reset 0	Q_3.1	Réinitialise le centre d'usinage.
Machining Center 0 (Stop)	MC Stop 0	Q_3.2	Interrompt le centre d'usinage.
Produce Lids 0	PL0	Q_3.3	Déclenche la fabrication de couvercles dans le cycle du centre d'usinage.
Machining Center 1 (Start)	MC start 1	Q_3.4	/
Machining Center 1 (Reset)	MC Reset 1	Q_3.5	/
Machining Center 1 (Stop)	MC Stop 1	Q_3.6	/
Produce Lids 1	PL1	Q_3.7	/
Move X0	Move X0	Q_5.0	controle le déplacement horizontal du robot (Pick & Place) sur l'axe X.
Move Z0	Move Z0	Q_5.1	Contrôle le mouvement vertical du robot (axe Z), pour monter ou descendre.
Grab 0	Grab 0	Q_5.2	Active la ventouse pour saisir une pièce.
Rotate CW 0	Rotate CW 0	Q_5.3	Rotation horaire du robot.
Rotate CCW 0	Rotate CCW 0	Q_5.4	Rotation antihoraire du robot.
Gripper CW 0	Gripper CW 0	Q_5.5	Rotation horaire de la pince.
Gripper CCW 0	Gripper CCW 0	Q_5.6	Rotation antihoraire de la pince.
Belt Conveyor (2m) 1	Cnv 2/1	Q_6.7	Convoyeur linéaire motorisé d'une longueur de 2 mètres.
Belt Conveyor (2m) 2	Cnv 2/2	Q_7.0	/
Belt Conveyor (2m) 6	Cnv 2/6	Q_7.4	/
Belt Conveyor (4m) 0	Cnv 4/0	Q_7.5	Convoyeur linéaire motorisé d'une longueur de 4 mètres.
Belt Conveyor (4m) 1	Cnv 4/1	Q_7.6	/
Belt Conveyor (4m) 7	Cnv 4/7	Q_8.4	/
Belt Conveyor (6m) 1	Cnv 6/1	Q_8.6	Convoyeur linéaire motorisé d'une longueur de 6 mètres.
Belt Conveyor (6m) 2	Cnv 6/2	Q_8.7	/
Belt Conveyor (6m) 3	Cnv 6/3	Q_9.0	/
Belt Conveyor (6m) 4	Cnv 6/4	Q_9.1	/
Curved Belt Conveyor 0 CW	Cnv CW0	Q_9.5	Convoyeurs courbes pour changer la direction des pièces dans le sens horaire
Curved Belt Conveyor 1 CCW	Cnv CCW1	Q_9.6	Convoyeurs courbes pour changer la direction des pièces dans le sens antihoraire
Curved Belt Conveyor 2 CCW	Cnv CCW2	Q_9.7	/
Curved Belt Conveyor 6 CW	Cnv CW6	Q_10.3	/

FACTORY I/O (reset)	FACTORY I/O (reset)	Q_10.5	Remise à zéro générale du système de simulation dans Factory I/O.
Stop Button 0 (Light)	light stop 0	Q_10.6	Indicateur lumineux associé au bouton d'arrêt pour signaler l'état actif/inactif.
Start Button 0 (Light)	light start 0	Q_10.7	/
Reset Button 0 (Light)	light reset 0	Q_11.0	/

3.6 Établissement de la communication entre Factory I/O et TIA Portal

L'objectif de cette section est de décrire les étapes nécessaires pour établir une communication fonctionnelle entre le logiciel de simulation Factory I/O et l'environnement de programmation TIA Portal, via le simulateur S7-PLCSIM. Cette communication permet de piloter un système virtuel (modélisé dans Factory I/O) à l'aide d'un programme automatique développé dans TIA Portal.

ouvrir le logiciel tia portal et créer un nouveau projet tia portail

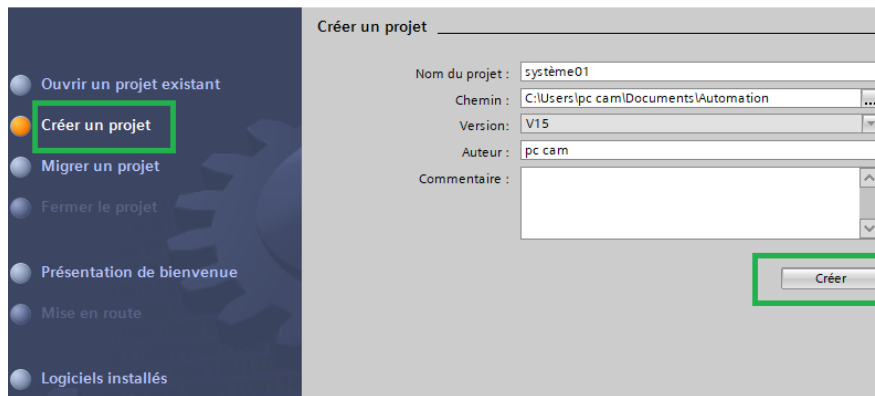


FIG. 3.8 : Création d'un nouveau projet dans TIA Portal

Sélectionner écrire un programme API

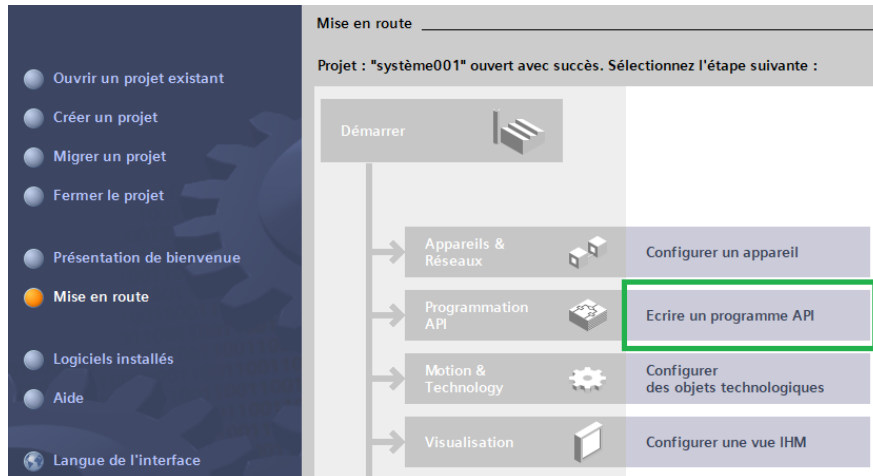


FIG. 3.9 : Choix de l'option "Écrire un programme API"

Cliquez sur «Ajouter un nouveau appareil» dans l'arborescence du contrôleur, développez Simatic S7-300 > CPU > CPU 314 -2PN/DP. Sélectionnez le CPU correspondant en dessous, cliquez sur «Ajouter».

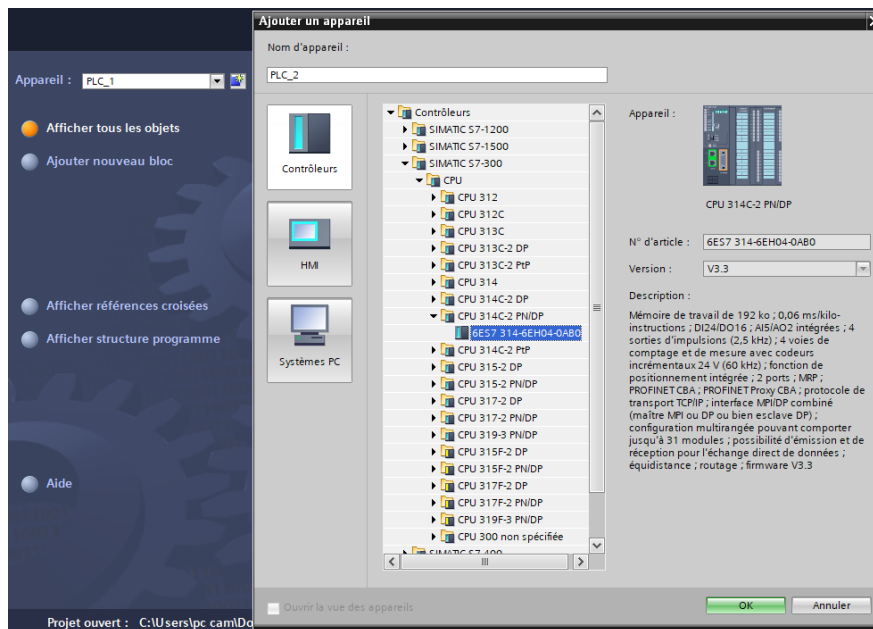


FIG. 3.10 : Sélection de l'automate S7-314C-2PN/DP

Déclarez les variables dans la table des variables, renseignez les entrées (I) et les sorties (Q) correspondant aux capteurs et aux actionneurs de votre système, et assurez-vous que les adresses sont cohérentes et uniques.

Nom	Table de variables s.	Type de données	Adresse	Réma...	Acces...	Visibl...	Commentaire
FL3	Table de variables s.	Bool	%I1.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
DS0	Table de variables s.	Bool	%I1.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
DS1	Table de variables s.	Bool	%I1.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
DS2	Table de variables s.	Bool	%I2.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
DS3	Table de variables s.	Bool	%I2.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
DS4	Table de variables s.	Bool	%I2.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
DS6	Table de variables s.	Bool	%I2.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
LTO	Table de variables s.	Bool	%I3.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
clamped0	Table de variables s.	Bool	%I3.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
LT1	Table de variables s.	Bool	%I3.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
clamped1	Table de variables s.	Bool	%I3.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
LT2	Table de variables s.	Bool	%I3.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
clamped2	Table de variables s.	Bool	%I3.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
LT3	Table de variables s.	Bool	%I4.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
clamped3	Table de variables s.	Bool	%I4.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
LT4	Table de variables s.	Bool	%I4.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
clamped4	Table de variables s.	Bool	%I4.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
LT5	Table de variables s.	Bool	%I4.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
clamped5	Table de variables s.	Bool	%I4.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
LT6	Table de variables s.	Bool	%I4.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
clamped6	Table de variables s.	Bool	%I4.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
LT7	Table de variables s.	Bool	%I5.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
clamped7	Table de variables s.	Bool	%I5.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

FIG. 3.11 : Création de la table des variables (entrées et sorties)

écrire votre programme avec le langage qui choisit (grafcet,ladder,FBD,Structured Text..etc), on suit cliqué sur démarrer la simulation.

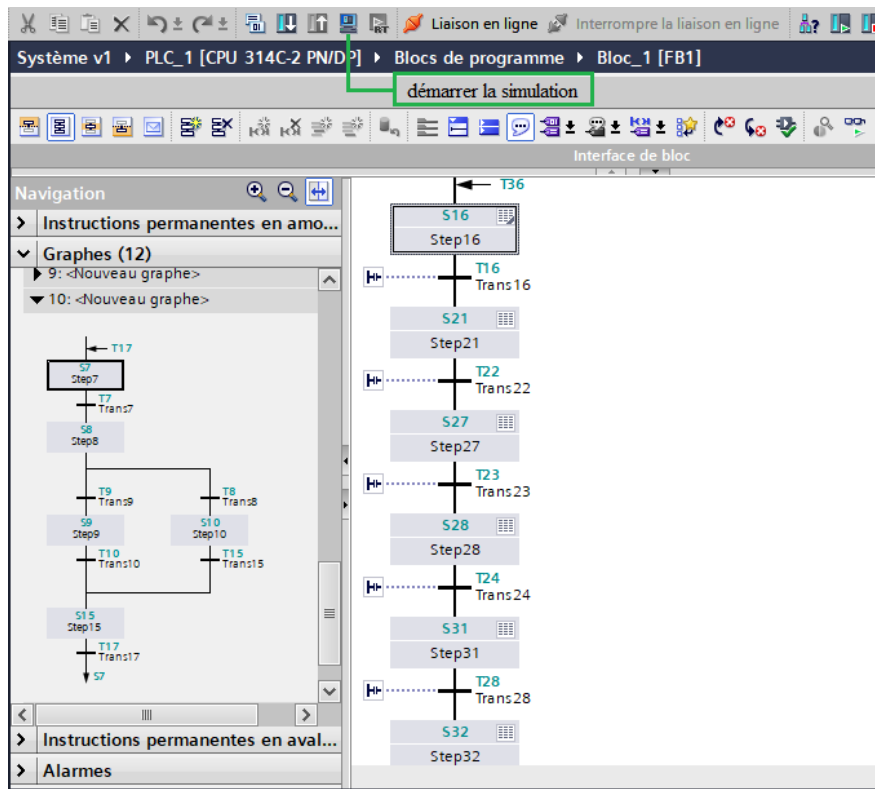


FIG. 3.12 : Démarrage de la simulation avec S7-PLCSIM

Choisissez le type d'interface PG/PC, puis cliquez sur "lancer le recherche" ensuite cliquez sur "charger".

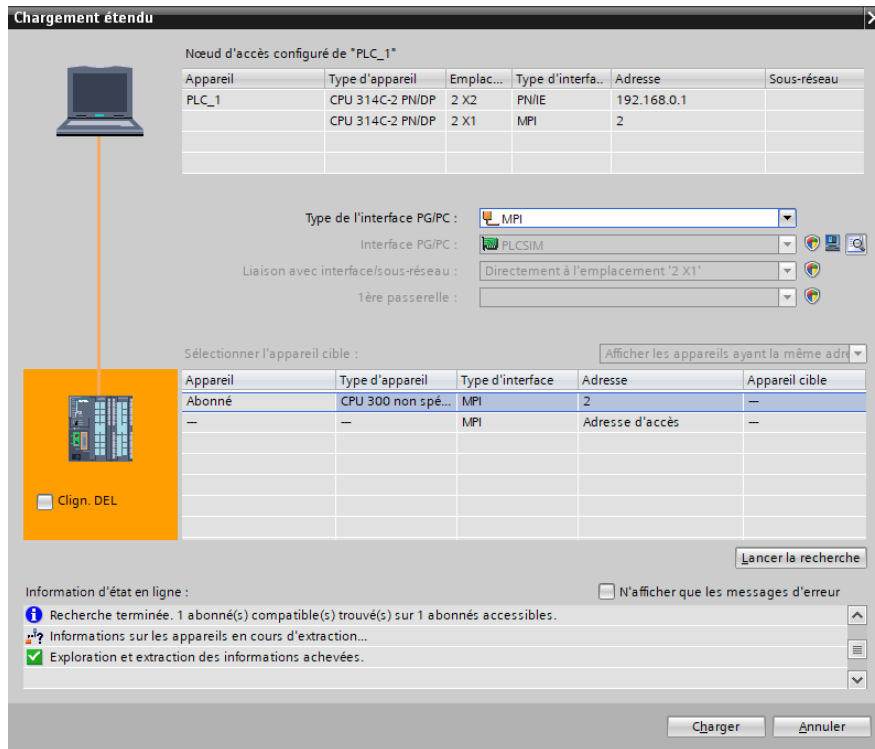


FIG. 3.13 : Choix du type d'interface PG/PC et chargement vers PLCSIM

Cliquez sur "charger dans l'appareil". ensuite cliquez sur "charger "

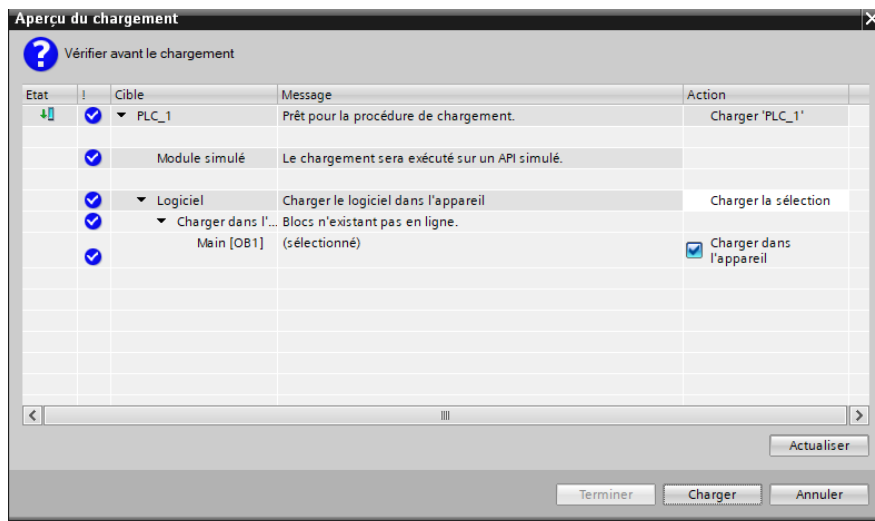


FIG. 3.14 : Chargement du programme dans l'automate

Dans FACTORY I/O, un driver I/O est une fonctionnalité intégrée chargée d'assurer la communiquer avec un contrôleur externe. FACTORY I/O inclut de nombreux driver I/O, chacun étant conçu pour une technologie spécifique.

Pour afficher le menu des drivers, cliquez sur Fichier > drivers (F4). Vous pouvez également effectuer un clic gauche sur le nom du driver affiché dans la barre d'état.

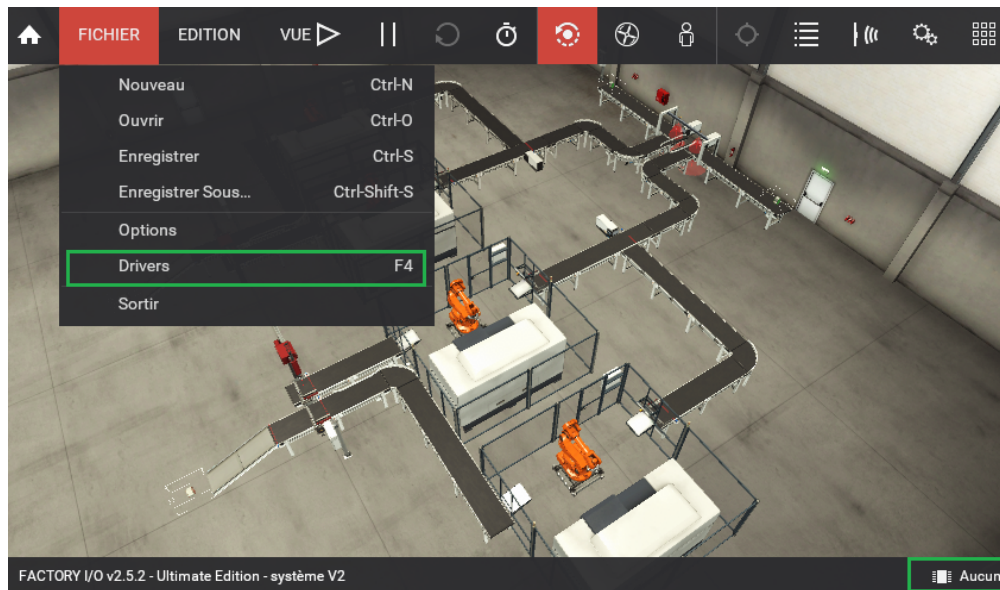


FIG. 3.15 : Ouverture du menu des drivers pour la configuration de la connexion

Dans le menu pilote (drivers) ,selectionnez Simens S7-PLCSIM dans la liste déroulante .

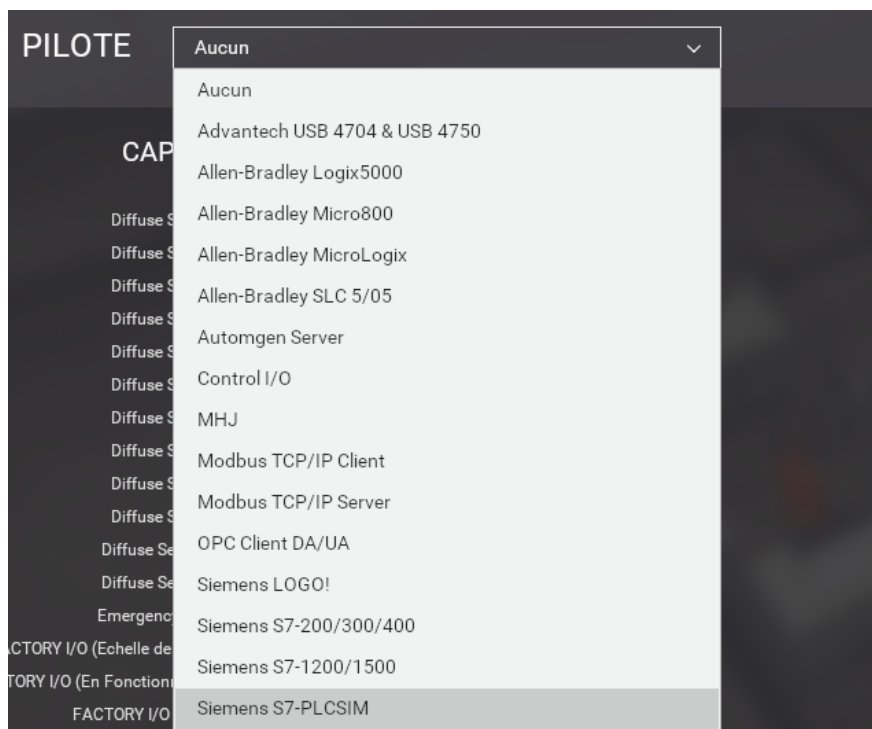


FIG. 3.16 : Sélection du driver Siemens S7-PLCSIM dans Factory I/O

Ouvrez le panneau de configuration du driver en cliquant sur « CONFIGURATION ».

Assurez-vous que S7-300 est sélectionné dans la liste déroulante modèle, puis modifiez les points E/S selon les

besoins de votre système.

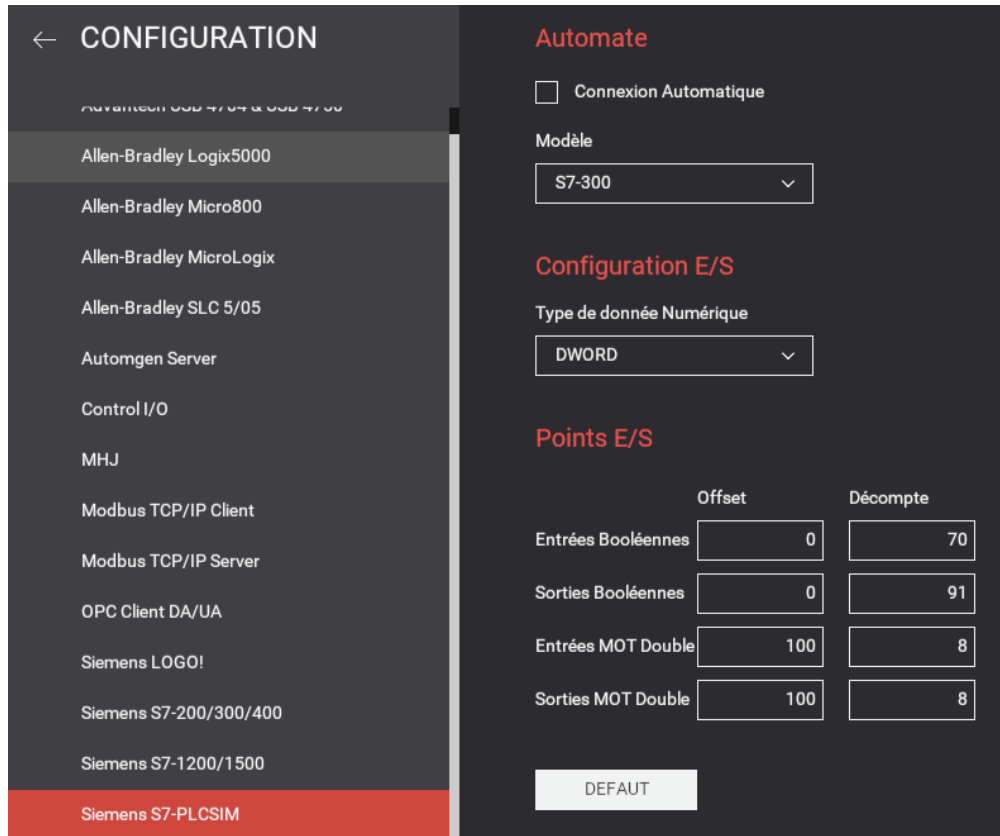


FIG. 3.17 : Vérification de la configuration du driver S7-PLCSIM

Appuyez sur Échap pour revenir à la fenêtre principale du pilote.

Ensuite, associez-les entrées et sorties de votre système aux adresses correspondantes déjà définies dans la table des variables de TIA Portal.

Par exemple, si vous avez défini dans la table des variables l'entrée "Bouton Démarrer 0" avec l'adresse I0.1, vous devez connecter, dans le pilote de Factory I/O, le capteur "Start Button 0" à cette même adresse (I0.1) dans le module S7-PLCSIM.

Il est essentiel que les adresses soient strictement identiques pour assurer une communication correcte entre Factory I/O et TIA Portal.

Une fois toutes les entrées et sorties correctement configurées, cliquez sur «CONNECTER» pour établir la liaison avec le simulateur.

Si la connexion est réussie, une icône verte s'affichera à côté du pilote sélectionné et son nom apparaîtra dans la barre d'état.



FIG. 3.18 : Association des E/S aux adresses du programme et test de la connexion

3.7 La programmation de l'API

3.7.1 Programmation et Architecture des Blocs Fonctionnels du Système Automatisé avec GRAFCET

A_ Programme principal MAIN [OB1]

Après avoir créé le tableau des variables, qui contient 172 variables (entrées, sorties et quelques variables internes), nous allons commencer la programmation. Le bloc MAIN OB1 contient le programme principal, tandis que les autres blocs (FB1, FB2) représentent les sous-programmes correspondant aux différentes lignes de production.

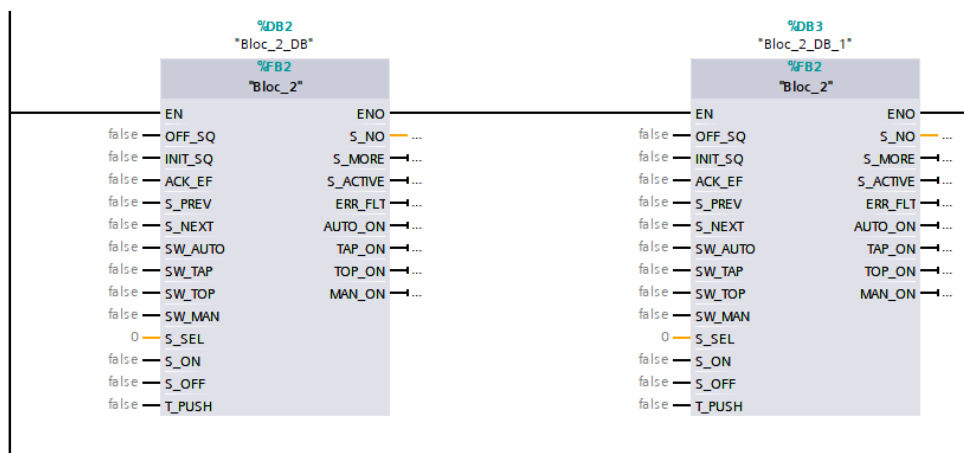


FIG. 3.19 : MAIN BLOC programme principale

B_ Les blocs fonctionnels

Chaque bloc fonctionnel (FB1 et FB2) est responsable d'une ligne de production. Il contient 12 unités GRAFCET, chacune chargée de programmer une partie spécifique de cette ligne. J'ai choisi de diviser le programme en 12 GRAFCETs afin de garantir le fonctionnement indépendant de chaque section de la ligne de production.

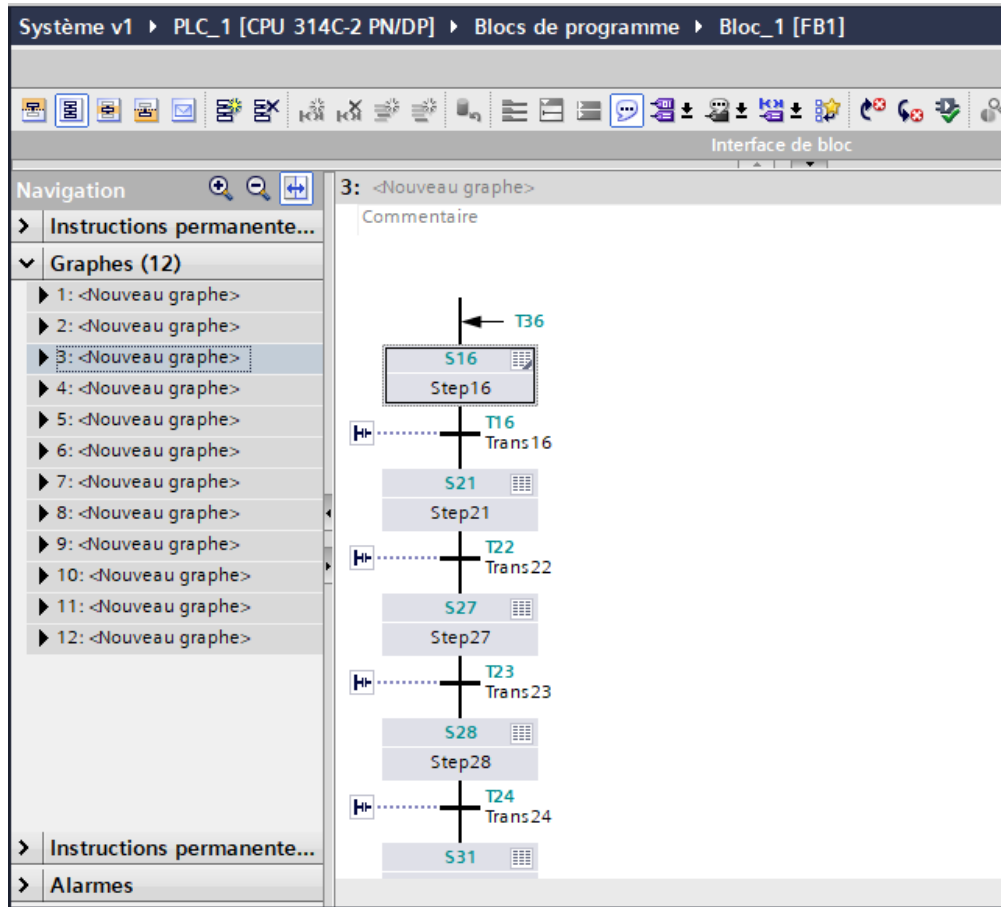


FIG. 3.20 : bloc fonctionnel FB1

□ GRAFCET de Supervision et contrôle de l'armoire de commande

Le premier GRAFCET occupe une position centrale dans l'architecture du système automatisé. Il est chargé de la gestion de l'armoire de commande et constitue l'unité de supervision globale. Ce GRAFCET agit sur les 11 autres GRAFCETs à travers des variables booléennes et se compose de trois étapes principales :

- **Étape 1 : Arrêt de la ligne de production**

Cette étape permet de désactiver toutes les étapes des 11 GRAFCETs secondaires, stoppant ainsi la ligne de production dans son ensemble. Elle peut être déclenchée par un bouton STOP.

- **Étape 2 : Réinitialisation**

Accessible uniquement après un arrêt complet (étape 1), cette étape est activée par l'appui sur le bouton RESET. Elle remet toutes les variables internes à zéro et prépare le système pour un redémarrage propre. Cette

réinitialisation ne peut être effectuée que dans un état sécurisé, entre les étapes 1 et 3.

- **Étape 3 : Démarrage (START)**

Cette étape est activée par le bouton START. Elle déclenche simultanément les étapes initiales des 11 autres GRAFCETs, permettant ainsi la reprise coordonnée et contrôlée du processus de production.

Le GRAFCET 1 gère également l'arrêt d'urgence, via un bouton dédié. En cas d'urgence, toutes les étapes sont immédiatement désactivées, quel que soit leur état, garantissant une sécurité maximale.

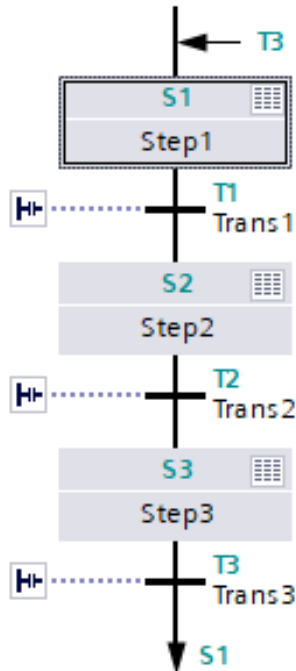


FIG. 3.21 : GRAFCET de gestion de l'armoire de commande

□ GRAFCET d'acheminement des pièces vers les centres d'usinage

Ce GRAFCET assure l'orientation des pièces détectées par un capteur photoélectrique vers l'un des deux centres d'usinage. Il repose sur un compteur logiciel (variable interne) qui s'incrémente à chaque détection. En fonction de la parité (paire ou impaire) de ce compteur, la pièce est dirigée vers le centre de traitement approprié.

- Principe de fonctionnement :

Lorsqu'une pièce est détectée, la variable compteur est incrémenté.

Un test de parité est ensuite effectué (division modulo 2) :

- Si le compteur est impair, le pusher est activé : la pièce est déviée vers la voie 1, menant au centre d'usinage des bases.
- Si le compteur est pair, le pusher reste inactif : la pièce suit naturellement la voie 2, menant au centre d'usinage des couvercles.

- Remarques :

- Il s’agit d’un OU exclusif : seule l’une des deux actions est exécutée selon le résultat du test de parité.
- Le compteur logiciel est remis à zéro lors de l’activation de l’étape de réinitialisation du GRAFCET de l’armoire de commande (voir Figure 18).
- La ligne de convoyage étant unique, il n’existe aucun risque de collision ou d’arrivée simultanée de plusieurs pièces.

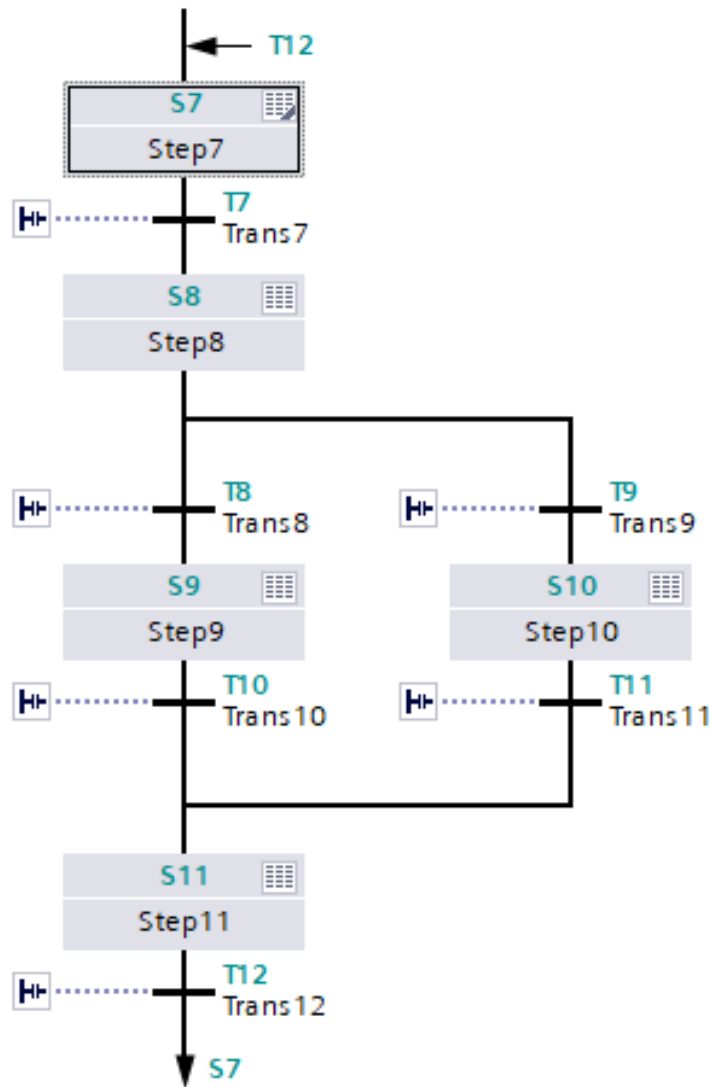


FIG. 3.22 : GRAFCET d’orientation des pièces selon la parité vers les centres d’usinage

□ GRAFCET de serrage des bases dans la station d’assemblage

Ce GRAFCET contrôle le positionneur de bases dans la station d’assemblage. Il s’agit d’un GRAFCET séquentiel, conçu pour maintenir la pièce en position tant que les conditions et de synchronisation avec le robot ne sont pas remplies.

• Principe de fonctionnement :

- **Détection d’une base :** Le GRAFCET est initialisé dès qu’un capteur détecte la présence d’une base. Cette

détection déclenche la fermeture du positionneur afin de maintenir la pièce en place.

- **Maintien de la position pendant l'assemblage** : Tant que le robot n'a pas terminé son cycle, le positionneur reste fermé pour assurer la stabilité de la pièce. Aucun changement d'étape n'a lieu dans le GRAFCET tant que le robot n'est pas revenu à sa position initiale.
- **Déplacement vertical en fin de cycle** : Une fois le cycle du robot terminé, un signal est reçu. Le GRAFCET du positionneur passe alors à l'étape suivante, déclenchant un mouvement vertical pour libérer la pièce assemblée et l'acheminer vers l'étape suivante du processus.
- **Synchronisation avec le robot** : Ce GRAFCET fonctionne en parfaite coordination avec celui du robot pick-and-place. La synchronisation est assurée via des variables booléennes d'échange.

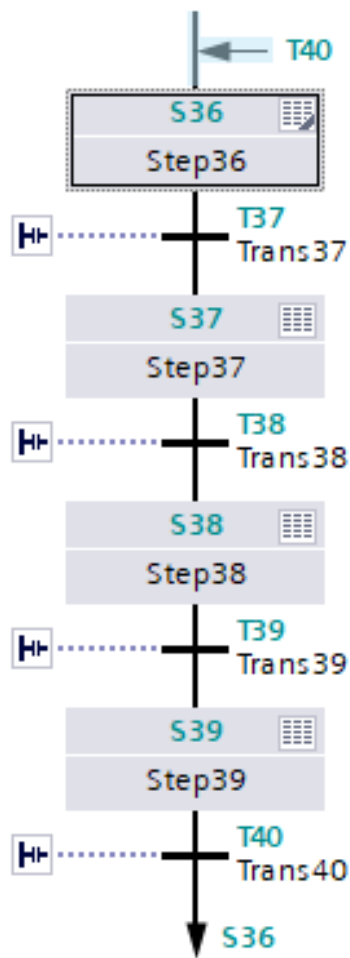


FIG. 3.23 : GRAFCET de contrôle du positionneur de bases dans la station d'assemblage

3.7.2 Test, Simulation et Supervision avec S7 PLCSIM et Factory I/O

Dans cette section, nous procédons à la validation, la simulation et la supervision du comportement de notre système automatisé à l'aide de deux outils complémentaires : S7 PLCSIM et Factory I/O.

D'une part, le simulateur S7 PLCSIM, développé par Siemens, permet d'exécuter le programme automate (API) dans un environnement virtuel sans utiliser de matériel physique. Grâce à son interface simple et intuitive, il est possible de visualiser en temps réel l'évolution des signaux de sortie en fonction des variations des entrées et du déroulement du temps. Une fonction essentielle de S7 PLCSIM est la possibilité de forcer l'état des bits d'entrée, ce qui permet de tester différents scénarios rapidement et efficacement. La figure 24 illustre ces variations de signaux, mettant en évidence la réponse du système à diverses conditions.

D'autre part, la connexion entre S7 PLCSIM et Factory I/O enrichit considérablement l'environnement de simulation. Factory I/O offre une simulation visuelle en 3D du système automatisé, en reproduisant fidèlement un environnement industriel réaliste (convoyeurs, capteurs, actionneurs, etc.). Cette visualisation dynamique permet de superviser en temps réel l'ensemble du processus : chaque action effectuée dans le programme automate est immédiatement reflétée dans la scène virtuelle, et inversement. Cela permet une observation intuitive et détaillée du fonctionnement du système.

Factory I/O joue ainsi un rôle fondamental de supervision, car il permet non seulement d'observer le comportement du système, mais aussi de détecter rapidement les dysfonctionnements, de valider la logique de commande, et d'assurer la cohérence des interactions entre les différentes unités automatisées. Il devient également un outil de formation et de perfectionnement, en offrant un espace sûr pour expérimenter et comprendre les séquences de fonctionnement sans risque matériel.

En résumé, l'association de S7 PLCSIM pour la simulation logique et de Factory I/O pour la supervision visuelle constitue un environnement puissant, interactif et réaliste pour le test, le débogage et la validation des systèmes automatisés avant leur mise en service réelle.

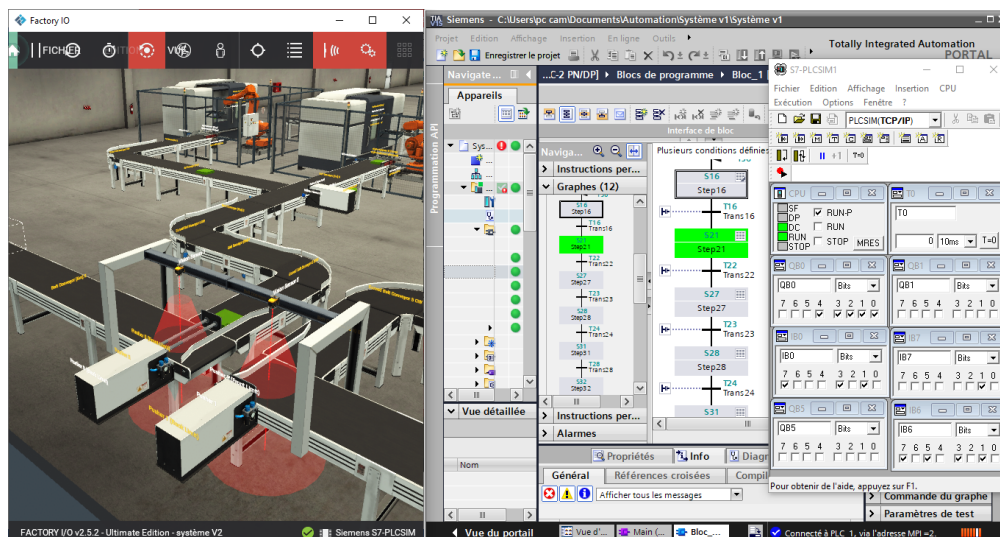


FIG. 3.24 : Simulation du système automatisé

Au vu des résultats obtenus, nous pouvons conclure que le système développé fonctionne avec succès.

3.8 Conclusion

Ce chapitre a permis de démontrer de manière concrète l'apport du TIA Portal dans la conception et la simulation d'un système automatisé complet. Grâce à son intégration fluide avec des outils tels que S7-PLCSIM et Factory I/O, il devient possible de valider la logique de fonctionnement d'un automate sans recourir immédiatement à un environnement matériel réel.

L'utilisation du langage GRAFCET a facilité la structuration du programme en unités fonctionnelles claires et indépendantes, renforçant ainsi la modularité, la lisibilité et la maintenabilité du système. L'approche adoptée a non seulement permis de minimiser les risques liés à la mise en service, mais a également favorisé une meilleure compréhension et une maîtrise plus approfondie du comportement du processus automatisé.

En somme, cette méthodologie représente une solution moderne, pédagogique et sécurisée pour aborder efficacement les problématiques liées à l'automatisation industrielle.[32], [33]

CONCLUSION GÉNÉRALE

Dans un contexte industriel en pleine mutation, où les technologies numériques redéfinissent les standards de la production, ce projet de fin d'études s'est inscrit dans une dynamique ambitieuse et moderne qui est celle de l'Industrie 4.0. En articulant modélisation 3D, programmation logique structurée, configuration virtuelle et simulation intégrée, nous avons pu concevoir, développer et valider un système automatisé et robotisé, fidèle aux réalités d'un environnement industriel connecté et interopérable.

Au cœur de cette réalisation, trois outils ont joué un rôle central, chacun apportant une valeur ajoutée déterminante dans la réussite du projet. Le premier outil est Factory I/O qui nous a permis d'implémenter une maquette numérique réaliste d'un atelier de production comportant deux lignes de production automatisées et robotisées. Grâce à cet environnement immersif en trois dimensions, nous avons pu observer le comportement dynamique des machines et des capteurs en interaction, simuler différents scénarios de fonctionnement, et surtout valider visuellement l'efficacité des programmes de commande. Cette phase de modélisation nous a sensibilisés à l'importance croissante des jumeaux numériques dans les industries modernes, où la simulation précède souvent l'investissement matériel.

Le deuxième principal outil utilisé dans notre travail est le langage de modélisation standardisé GRAFCET, qui nous a permis de structurer la logique de fonctionnement du système de façon hiérarchisée, claire et modulaire. Chaque étape, chaque transition, chaque condition d'activation a été pensée pour répondre à une logique industrielle cohérente, reflétant la séquence réelle d'un processus de production. Le recours à cette méthode a été crucial pour garantir la lisibilité du programme, faciliter la détection d'erreurs, et envisager aisément les extensions futures. En tant qu'outil pédagogique, il nous a permis de faire le lien entre les exigences fonctionnelles du système et leur traduction en code de commande.

Enfin TIA Portal, l'environnement de développement intégré de Siemens, a constitué la plateforme principale pour la programmation, la configuration matérielle et la simulation du système automatisé. Grâce à l'outil S7-PLCSIM, nous avons pu tester les logiques de commande dans un environnement virtuel complet, sans recourir à du matériel physique. Cette démarche a permis une précision remarquable dans la validation des scénarios d'automatisation. L'intégration des éléments matériels et logiciels dans une seule interface nous a permis de reproduire fidèlement les conditions de fonctionnement d'un véritable système industriel moderne piloté par un API Siemens.

Ce projet a été l'occasion d'acquérir et de consolider des compétences transversales en automatisation, informatique industrielle, communication entre systèmes, gestion de processus complexes et programmation des API. Il nous a également permis de comprendre les enjeux liés à la conception et la simulation des systèmes industriels complexes

virtuels en utilisant des outils de conception 3D et son rôle primordial avant la conception des systèmes industriels physiques. Cette expérience nous a permis aussi de développer une vision complète du cycle de vie d'un système industriel automatisé, depuis sa conception théorique jusqu'à sa simulation finale. Elle nous a également sensibilisés à l'importance de la modélisation fonctionnelle, de la gestion modulaire des systèmes complexes, et de l'approche projet collaborative. L'utilisation d'outils industriels reconnus dans le monde professionnel renforce notre employabilité et nous prépare efficacement à des environnements de travail où la transformation numérique est déjà bien amorcée. En somme, ce projet se veut à la fois une démonstration pratique de l'intégration des technologies de l'Industrie 4.0 et une expérience d'apprentissage approfondie. Il constitue une réponse concrète aux besoins des entreprises en quête d'efficacité, de flexibilité et de réduction des coûts, tout en assurant des niveaux élevés de qualité et de sécurité. Il nous a permis de mesurer l'apport stratégique des jumeaux numériques dans la phase de conception et d'exploitation des systèmes industriels modernes.

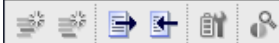
Pour conclure, ce mémoire illustre la capacité à mobiliser les outils technologiques de pointe dans une démarche structurée et cohérente, au service de la modernisation des systèmes industriels. Il constitue une base solide pour aborder des projets professionnels dans les domaines de l'automatisation avancée, de la robotique, de la supervision à distance, ou encore du développement de solutions numériques pour l'industrie intelligente. Il ouvre également la voie à des travaux de recherche plus approfondis sur l'intégration de l'intelligence artificielle, de l'analyse prédictive, ou de la maintenance intelligente dans les processus industriels, en phase avec les exigences de compétitivité et d'innovation de l'industrie du futur.

Variables API							
	Nom	Table des variables	Type de données	Adresse	Réma...	Acces...	Visibl...
1	start0	Table de variables s..	Bool	%I0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	stop0	Table de variables s..	Bool	%I0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	reset0	Table de variables s..	Bool	%I0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4	AU	Table de variables s..	Bool	%I0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	VS1	Table de variables s..	Bool	%I0.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
6	VS2	Table de variabl...	Bool	%I0.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7	BL0	Table de variables s..	Bool	%I0.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
8	BL1	Table de variables s..	Bool	%I0.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
9	BL2	Table de variables s..	Bool	%I1.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
10	BL3	Table de variables s..	Bool	%I1.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
11	FL0	Table de variables s..	Bool	%I1.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
12	FL1	Table de variables s..	Bool	%I1.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
13	FL2	Table de variables s..	Bool	%I1.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14	FL3	Table de variables s..	Bool	%I1.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
15	DS0	Table de variables s..	Bool	%I1.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
16	DS1	Table de variables s..	Bool	%I1.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
17	DS2	Table de variables s..	Bool	%I2.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
18	DS3	Table de variables s..	Bool	%I2.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
19	DS4	Table de variables s..	Bool	%I2.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
20	DS6	Table de variables s..	Bool	%I2.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
21	LTO	Table de variables s..	Bool	%I3.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
22	clamped0	Table de variables s..	Bool	%I3.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
23	LT1	Table de variables s..	Bool	%I3.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
24	clamped1	Table de variables s..	Bool	%I3.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>



Variables API

	Nom	Table des variables s..	Type de données	Adresse	Réma...	Acces...	Visibl...
25	LT2	Table de variables s..	Bool	%I3.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
26	clamped2	Table de variables s..	Bool	%I3.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
27	LT3	Table de variables s..	Bool	%I4.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
28	clamped3	Table de variables s..	Bool	%I4.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
29	LT4	Table de variables s..	Bool	%I4.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
30	clamped4	Table de variables s..	Bool	%I4.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
31	LT5	Table de variables s..	Bool	%I4.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
32	clamped5	Table de variables s..	Bool	%I4.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
33	LT6	Table de variables s..	Bool	%I4.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
34	clamped6	Table de variables s..	Bool	%I4.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
35	LT7	Table de variables s..	Bool	%I5.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
36	clamped7	Table de variables s..	Bool	%I5.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
37	error0	Table de variables s..	Bool	%I5.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
38	opened0	Table de variables s..	Bool	%I5.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
39	progress0	Table de variables s..	Bool	%I5.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
40	by0	Table de variables s..	Bool	%I5.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
41	error1	Table de variables s..	Bool	%I5.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
42	opened1	Table de variables s..	Bool	%I5.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
43	progress 1	Table de variables s..	Bool	%I6.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
44	by1	Table de variables s..	Bool	%I6.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
45	error2	Table de variables s..	Bool	%I6.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
46	opened2	Table de variables s..	Bool	%I6.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
47	progress 2	Table de variables s..	Bool	%I6.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
48	by2	Table de variables s..	Bool	%I6.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>



Variables API

	Nom	Table des variables s..	Type de données	Adresse	Réma...	Acces...	Visibl...
49	error3	Table de variables s..	Bool	%I6.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
50	opened3	Table de variables s..	Bool	%I6.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
51	progress3	Table de variables s..	Bool	%I7.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
52	by3	Table de variables s..	Bool	%I7.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
53	Mz0	Table de variables s..	Bool	%I7.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
54	Mx0	Table de variables s..	Bool	%I7.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
55	detected0	Table de variables s..	Bool	%I7.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
56	GR0	Table de variables s..	Bool	%I7.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
57	RT0	Table de variables s..	Bool	%I7.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
58	Mz1	Table de variables s..	Bool	%I7.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
59	Mx1	Table de variables s..	Bool	%I8.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
60	detected1	Table de variables s..	Bool	%I8.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
61	GR1	Table de variables s..	Bool	%I8.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
62	RT1	Table de variables s..	Bool	%I8.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
63	emitter 0	Table de variables s..	Bool	%Q0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
64	emitter 1	Table de variables s..	Bool	%Q0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
65	remover 0	Table de variables s..	Bool	%Q0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
66	remover 1	Table de variables s..	Bool	%Q0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
67	pusher 0	Table de variables s..	Bool	%Q0.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
68	pusher 1	Table de variables s..	Bool	%Q0.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
69	pusher 2	Table de variables s..	Bool	%Q0.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
70	pusher 3	Table de variables s..	Bool	%Q0.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
71	clamp 0	Table de variables s..	Bool	%Q1.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
72	raise 0	Table de variables s..	Bool	%Q1.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>



Variables API

	Nom	Table des variables s..	Type de données	Adresse	Réma...	Acces...	Visibl...
73	clamp 1	Table de variables s..	Bool	%Q1.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
74	raise 1	Table de variables s..	Bool	%Q1.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
75	clamp 2	Table de variables s..	Bool	%Q1.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
76	raise 2	Table de variables s..	Bool	%Q1.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
77	clamp 3	Table de variables s..	Bool	%Q1.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
78	DS5	Table de variables s..	Bool	%I2.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
79	DS7	Table de variables s..	Bool	%I2.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
80	DS8	Table de variables s..	Bool	%I2.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
81	DS9	Table de variables s..	Bool	%I2.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
82	DS10	Table de variables s..	Bool	%I3.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
83	DS11	Table de variables s..	Bool	%I3.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
84	raise 3	Table de variables s..	Bool	%Q1.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
85	clamp 4	Table de variables s..	Bool	%Q2.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
86	raise 4	Table de variables s..	Bool	%Q2.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
87	clamp 5	Table de variables s..	Bool	%Q2.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
88	raise 5	Table de variables s..	Bool	%Q2.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
89	clamp 6	Table de variables s..	Bool	%Q2.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
90	raise 6	Table de variables s..	Bool	%Q2.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
91	clamp 7	Table de variables s..	Bool	%Q2.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
92	raise 7	Table de variables s..	Bool	%Q2.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
93	MC start 0	Table de variables s..	Bool	%Q3.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
94	MC Reset 0	Table de variables s..	Bool	%Q3.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
95	MC Stop 0	Table de variables s..	Bool	%Q3.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
96	PLO	Table de variables s..	Bool	%Q3.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>



Variables API

	Nom	Table des variables s..	Type de données	Adresse	Réma...	Acces...	Visibl...
97	MC start 1	Table de variables s..	Bool	%Q3.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
98	MC Reset 1	Table de variables s..	Bool	%Q3.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
99	MC Stop 1	Table de variables s..	Bool	%Q3.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
100	PL1	Table de variables s..	Bool	%Q3.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
101	MC start 2	Table de variables s..	Bool	%Q4.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
102	MC Reset 2	Table de variables s..	Bool	%Q4.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
103	MC Stop 2	Table de variables s..	Bool	%Q4.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
104	PL2	Table de variables s..	Bool	%Q4.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
105	MC start 3	Table de variables s..	Bool	%Q4.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
106	MC Reset 3	Table de variables s..	Bool	%Q4.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
107	MC Stop 3	Table de variables s..	Bool	%Q4.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
108	PL3	Table de variables s..	Bool	%Q4.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
109	Move X0	Table de variables s..	Bool	%Q5.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
110	Move Z0	Table de variables s..	Bool	%Q5.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
111	Grab 0	Table de variables s..	Bool	%Q5.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
112	Rotate CW 0	Table de variables s..	Bool	%Q5.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
113	Rotate CCW 0	Table de variables s..	Bool	%Q5.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
114	Gripper CW 0	Table de variables s..	Bool	%Q5.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
115	Gripper CCW 0	Table de variables s..	Bool	%Q5.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
116	Move X1	Table de variables s..	Bool	%Q5.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
117	Move Z1	Table de variables s..	Bool	%Q6.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
118	Grab 1	Table de variables s..	Bool	%Q6.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
119	Rotate CW 1	Table de variables s..	Bool	%Q6.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
120	Rotate CCW 1	Table de variables s..	Bool	%Q6.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>



Variables API

	Nom	Table des variables s..	Type de données	Adresse	Réma...	Acces...	Visibl...
121	Gripper CW 1	Table de variables s..	Bool	%Q6.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
122	Gripper CCW 1	Table de variables s..	Bool	%Q6.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
123	Cnv 2/0	Table de variables s..	Bool	%Q6.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
124	Cnv 2/1	Table de variables s..	Bool	%Q6.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
125	Cnv 2/2	Table de variables s..	Bool	%Q7.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
126	Cnv 2/3	Table de variables s..	Bool	%Q7.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
127	Cnv 2/4	Table de variables s..	Bool	%Q7.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
128	Cnv 2/5	Table de variables s..	Bool	%Q7.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
129	Cnv 2/6	Table de variables s..	Bool	%Q7.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
130	Cnv 4/0	Table de variables s..	Bool	%Q7.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
131	Cnv 4/1	Table de variables s..	Bool	%Q7.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
132	Cnv 4/2	Table de variables s..	Bool	%Q7.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
133	Cnv 4/3	Table de variables s..	Bool	%Q8.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
134	Cnv 4/4	Table de variables s..	Bool	%Q8.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
135	Cnv 4/5	Table de variables s..	Bool	%Q8.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
136	Cnv 4/6	Table de variables s..	Bool	%Q8.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
137	Cnv 4/7	Table de variables s..	Bool	%Q8.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
138	Cnv 6/0	Table de variables s..	Bool	%Q8.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
139	Cnv 6/1	Table de variables s..	Bool	%Q8.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
140	Cnv 6/2	Table de variables s..	Bool	%Q8.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
141	Cnv 6/3	Table de variables s..	Bool	%Q9.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
142	Cnv 6/4	Table de variables s..	Bool	%Q9.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
143	Cnv 6/5	Table de variables s..	Bool	%Q9.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
144	Cnv 6/6	Table de variables s..	Bool	%Q9.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>



Variables API

	Nom	Table des variables s..	Type de données	Adresse	Réma...	Acces...	Visibl...
145	Cnv 6/7	Table de variables s..	Bool	%Q9.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
146	Cnv CW0	Table de variables s..	Bool	%Q9.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
147	Cnv CCW1	Table de variables s..	Bool	%Q9.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
148	Cnv CCW2	Table de variables s..	Bool	%Q9.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
149	Cnv CCW3	Table de variables s..	Bool	%Q10.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
150	Cnv CW4	Table de variables s..	Bool	%Q10.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
151	Cnv CW5	Table de variables s..	Bool	%Q10.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
152	Cnv CW6	Table de variables s..	Bool	%Q10.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
153	Cnv CCW7	Table de variables s..	Bool	%Q10.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
154	FACTORY I/O (reset)	Table de variables s..	Bool	%Q10.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
155	CNT0	Table de variables s..	Counter	%C1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
156	image cnt0	Table de variables s..	Dint	%MD0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
157	modulo 0	Table de variables s..	Dint	%MD2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
158	image modulo 0	Table de variables s..	Int	%MWO		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
159	MGRB0	Table de variables s..	Bool	%M0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
160	light stop 0	Table de variables s..	Bool	%Q10.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
161	light start 0	Table de variables s..	Bool	%Q10.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
162	light reset 0	Table de variables s..	Bool	%Q11.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
163	start1	Table de variables s..	Bool	%I8.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
164	stop1	Table de variables s..	Bool	%I8.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
165	light start 1	Table de variables s..	Bool	%Q11.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
166	light stop 1	Table de variables s..	Bool	%Q11.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
167	CNT1	Table de variables s..	Counter	%C2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
168	image cnt1	Table de variables s..	Dint	%MD1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Système v1 ▶ PLC_1 [CPU 314C-2 PN/DP] ▶ Variables API

Variables

Variables API

	Nom	Table des variables s..	Type de données	Adresse	Réma...	Acces...	Visibl...
168	image cnt1	Table de variables s..	Dint	%MD1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
169	modulo 1	Table de variables s..	Dint	%MD3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
170	image modulo 1	Table de variables s..	Int	%MW1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
171	MGRB1	Table de variables s..	Bool	%MO.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
172	CNT2	Table de variables s..	Counter	%C3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
173	image cnt2	Table de variables s..	Dint	%MD4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
174	digital display0	Table de variables s..	Dint	%QD100		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
175	CNT3	Table de variables s..	Counter	%C4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
176	image cnt3	Table de variables s..	Dint	%MD5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
177	digital display1	Table de variables s..	Dint	%QD104		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
178	<Ajouter>					<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

BIBLIOGRAPHIE

- [1] C. DANJOU, L. RIVEST et R. PELLERIN, “Industrie 4.0: des pistes pour aborder l’ère du numérique et de la connectivité”, *Unknown Journal*, n.d. (Cf. p. 4, 6-8, 12).
- [2] A. BITAILLOU, B. PARREIN et G. ANDRIEUX, “Synthèse sur les protocoles de communication pour l’Internet des objets de l’industrie 4.0”, *Unknown Journal*, 2019 (cf. p. 4, 5).
- [3] M. LOUART, J.-C. L. LANN, F. L. ROY, A. BOUDRAA et J.-J. SZKOLNIK, “Émulation de Systèmes Cyber-Physiques sur FPGA”, *Unknown Journal*, n.d. (Cf. p. 9).
- [4] P. B. EDDY, B. AURIAC et G. KOENIG, “Objets Connectés Industriels : Identification et Géolocalisation dans une Infrastructure Internet des Objets”, *Unknown Journal*, n.d. (Cf. p. 10).
- [5] C. B, “Impact de l’intelligence artificielle dans la recherche clinique et la collecte de données en vie réelle”, *Actual Pharm*, t. 57, p. 22-24, 2018. DOI : 10.1016/j.actpha.2018.05.012 (cf. p. 10-12).
- [6] P. LE BRUN, *Automates programmables industriels : Technologie, choix et mise en œuvre des automates programmables industriels*. Lycée Louis Armand, 1999 (cf. p. 14, 15).
- [7] A. GONZAGA, *Les automates programmables industriels*, <http://www.geea.org>, Accessed March 21, 2025, 2004 (cf. p. 15, 18-20).
- [8] MABÉO INDUSTRIES, *Unité centrale automate compacte Simatic S7-1200*, <https://www.mabeo-industries.com/A-509587-siemens-unite-centrale-automate-compacte-simatic-s7-1200>, Accessed March 21, 2025 (cf. p. 17).
- [9] A. SENSE, *Comment choisir un module d’E/S pour votre automate?*, <https://www.automation-sense.com/blog/automatisme/comment-choisir-un-module-d-e-s-pour-votre-automate.html>, Accessed : 2025-03-21 (cf. p. 17).
- [10] H. AHMED, “Chapitre 2 Les capteurs dans les systèmes automatisés de production (SAP)”, *Unknown Journal*, n.d. (Cf. p. 20, 22).
- [11] M. R. B. J. MABROUKI et M. L. MABROUKI, *Cours Capteurs et actionneurs en instrumentation*, <http://www.technologiepro.com/cours-genie-electrique/cours-27-capteurs-actionneurs-instrumentation/>, Accessed March 21, 2025 (cf. p. 21).
- [12] H. AHMED, “Chapitre 4 Les Actionneurs dans les systèmes automatisés de production (SAP)”, *Unknown Journal*, n.d. (Cf. p. 23).
- [13] S. BENKRAOUDA, *Réseaux industriels*, Unknown Source, 2022 (cf. p. 23).

- [14] U. AUTHOR, *Automates Programmables Industriels Desc*, Unknown Source, n.d. (Cf. p. 24-26).
- [15] B. KATTAN, “Synthèse structurelle d’un contrôleur basée sur le Grafcet”, *Unknown Journal*, n.d. (Cf. p. 26).
- [16] U. AUTHOR, *Système de production 2*, <https://jackadit.com/index.php?p=sysprod2>, Accessed March 21, 2025 (cf. p. 26).
- [17] S. MASSART et D. GRALAND, *Les outils de simulation de systèmes automatisés*, Unknown Source, n.d. (Cf. p. 27).
- [18] H. MEZDOUR et U. AUTHOR, *Etude et réalisation d’un système de supervision sous YOKOGAWA CS3000*, Unknown Source, 2019 (cf. p. 27).
- [19] U. AUTHOR, *Editions - Factory I/O*, <https://factoryio.com/editions/>, Accessed February 1, 2025, 2025 (cf. p. 30).
- [20] U. AUTHOR, *Manual - FACTORY I/O*, <https://docs.factoryio.com/manual/>, Accessed February 1, 2025 (cf. p. 30, 31, 33, 38, 39).
- [21] U. AUTHOR, *CONTROL I/O*, <https://docs.factoryio.com/controlio/>, Accessed February 15, 2025 (cf. p. 48, 50-52).
- [22] FACTORY I/O DOCUMENTATION, *Interface graphique de Control I/O montrant les blocs fonctionnels et la zone de simulation*, <https://docs.factoryio.com/controlio/img/getting-started/ui.jpg>, Image, accessed February 1, 2025, 2025 (cf. p. 49).
- [23] FACTORY I/O DOCUMENTATION, *Représentation des blocs fonctionnels dans Control I/O et leurs connexions logiques*, <https://docs.factoryio.com/controlio/img/getting-started/blocks.jpg>, Image, accessed February 1, 2025, 2025 (cf. p. 50).
- [24] U. AUTHOR, *TIA Portal - Usine Nouvelle*, <https://www.usinenouvelle.com/expo/tia-portal-p225270.html>, n.d. (Cf. p. 61, 68).
- [25] M. KARA, “Conception et réalisation d’une maquette de tri automatique des pièces métalliques par automate programmable industriel”, Mémoire de fin d’études, Université Saad Dahlab de Blida 1, 2022. adresse : <https://di.univ-blida.dz/xmlui/bitstream/handle/123456789/19915/kara%20finale.pdf?sequence=1&isAllowed=y> (cf. p. 62-66, 68).
- [26] UNIVERSIDAD DE PIURA, *Laboratoire d’Automatisation et de Contrôle – Logiciel TIA Portal*, <https://www.udpep.edu.pe/labsac/wp-content/uploads/sites/85/2023/03/soft12.png>, 2023 (cf. p. 62).
- [27] F. MOSBAH et T. A. SEBGAG, “Gestion Et Supervision D’une Station De Pompage à Base D’automate SIEMENS”, Thèse de doctorat, Université de Ghardaïa, 2021 (cf. p. 65).
- [28] M. SIACI, “API et GRAFCET”, Thèse de doctorat, Unknown Institution, 2022 (cf. p. 66).
- [29] B. KATTAN, “Synthèse structurelle d’un contrôleur basée sur le Grafcet”, Thèse de doctorat, Université Joseph-Fourier-Grenoble I, 2004 (cf. p. 66, 67).
- [30] P. SCIARRETTA, *Technologie collègue - Le GRAFCET expliqué*, <https://technologie-sciarretta.ovh/?p=633>, 2017 (cf. p. 67).
- [31] A. RIADH, “Automatisation du système d’air frais et de gaz de fumée à l’aide d’un API SIEMENS S7-300”, Thèse de doctorat, Université Mouloud Mammeri, 2015 (cf. p. 68, 71).
- [32] SIEMENS AG, *SIMATIC S7-300 – CPU 314C-2 PN/DP*, <https://mall.industry.siemens.com/mall/en/WW/Catalog/Product/6ES7314-6EH04-0AB0>, 2020 (cf. p. 69, 91).
- [33] SIEMENS AG, *SIMATIC S7-300 CPU 314C-2 PN/DP – Technical Data Sheet*, <https://docs.rs-online.com/93ac/0900766b8131a200.pdf>, PDF document, accessed May 1, 2025, 2020 (cf. p. 72, 91).

Résumé

Dans le cadre de ce projet de fin d'études, nous avons développé un système de production automatisé et robotisé virtuel qui comporte deux lignes de production, en combinant modélisation 3D, programmation d'automate et simulation. L'environnement Factory I/O a été utilisé pour reproduire de manière réaliste ce système de production composé de plusieurs stations : distribution, usinage et assemblage. La commande du système a été assurée par un automate Siemens S7, programmé dans TIA Portal, avec exécution et test via le simulateur S7-PLCSIM. La logique de commande a été structurée en GRAFCET, traduite ensuite sous forme de blocs fonctionnels modulaires, ce qui a facilité la lisibilité du programme, sa maintenance et sa possible évolution. Ce projet s'inscrit dans la dynamique de l'Industrie 4.0, en exploitant des outils numériques avancés pour simuler et valider les processus industriels de manière virtuelle. Il illustre l'importance croissante des jumeaux numériques, de la connectivité et de la flexibilité logicielle dans les systèmes automatisés modernes. Ce travail nous a permis de renforcer nos compétences dans la conception des systèmes industriels ainsi que la modélisation, la programmation et la simulation d'automates, dans une approche concrète et conforme aux exigences de l'industrie du futur.

Mots clés :

API, GRAFCET, Programmation, Supervision, TIA Portal, Conception, Simulation 3D, FACTORY IO.

Abstaract :

As part of this final-year project, we developed a virtual automated and robotized production system comprising two production lines, using a combination of 3D modeling, PLC programming and simulation. The Factory I/O environment was used to realistically reproduce this production system comprising several stations: distribution, machining and assembly. The system was controlled by a Siemens S7 PLC, programmed in TIA Portal, with execution and testing via the S7-PLCSIM simulator. The control logic was structured in GRAFCET, then translated into modular function blocks, making the program easy to read, maintain and upgrade. This project is part of the industry 4.0 dynamic, exploiting advanced digital tools to simulate and validate industrial processes virtually. It illustrates the growing importance of digital twins, connectivity and software flexibility in modern automated systems. This work has enabled us to strengthen our skills in industrial system design, as well as PLC modeling, programming and simulation, in a concrete approach that meets the requirements of the industry of the future.

Key words :

PLC, GRAFCET, Programming, Supervision, TIA Portal, Design, 3D Simulation, FACTORY IO.

ملخص :

كجزء من مشروع السنة النهائية هذا، قمنا بتطوير نظام إنتاج افتراضي مؤتمت وآلي يتألف من خطي إنتاج، باستخدام مزيج من النمذجة ثلاثية الأبعاد وبرمجة PLC والمحاكاة. استُخدمت بيئة FACTORY I/O لإعادة إنتاج نظام الإنتاج هذا بشكل واقعي والذي يتألف من عدة محطات: التوزيع، والتشغيل الآلي، والتجميع. تم التحكم في النظام من خلال نظام التحكم المنطقي القابل للبرمجة S7 Siemens وتمت برمجته في TIA Portal مع التنفيذ والاختبار عبر محاكي S7-PLCSIM. تمت هيكلة منطق التحكم في تم GRAFCET تُرجمت إلى كتل وظيفية معيارية، مما جعل البرنامج أسهل في القراءة والصيانة والترقية. يعد هذا المشروع جزءاً من ديناميكية الصناعة 4.0، باستخدام أدوات رقمية متقدمة لمحاكاة العمليات الصناعية والتحقق من صحتها افتراضياً.

وهو يوضح الأهمية المتزايدة للتوائم الرقمية والاتصال ومرونة البرمجيات في الأنظمة المؤتمتة الحديثة. لقد مكنتنا هذا العمل من تعزيز مهارتنا في تصميم الأنظمة الصناعية ونمذجة وبرمجة ومحاكاة وحدات التحكم المنطقي القابلة للبرمجة في نهج عملي يلبي متطلبات الصناعة في المستقبل.

الكلمات المفتاحية :

FACTORY. I/O, المحاكاة ثلاثية الأبعاد, التصميم, Portal, TIA, بوابة الإشراف, البرمجة, الإشراف, PLC, GRAFCET, البرمجة, الإشراف, بوابة TIA, التصميم, المحاكاة ثلاثية الأبعاد, I/O. FACTORY.