

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Ecole Nationale Polytechnique



Département Automatique

Mémoire de projet de fin d'études  
pour l'obtention du diplôme d'ingénieur d'état en  
automatique

# Etude et commande d'un ascenseur avec API

**IOUSSAIDENE Sarah**

Sous la direction du Pr **EL Madjid BERKOUK**

Présenté et soutenu publiquement le (06/07/2020)

## Composition du Jury :

Président	M. Djamel BOUKHETALA	Pr.	ENP
Promoteur	M. ELmadjid BERKOUK	Pr.	ENP
Examineur	M. Rachid ILLOUL	Mc.	ENP

0

ENP 2020



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Ecole Nationale Polytechnique



Département Automatique

Mémoire de projet de fin d'études  
pour l'obtention du diplôme d'ingénieur d'état en  
automatique

# Etude et commande d'un ascenseur avec API

**IOUSSAIDENE Sarah**

Sous la direction du Pr **EL Madjid BERKOUK**

Présenté et soutenu publiquement le (06/07/2020)

**Composition du Jury :**

Président	M. Djamel BOUKHETALA	Pr.	ENP
Promoteur	M. ELmadjid BERKOUK	Pr.	ENP
Examineur	M. Rachid ILLOUL	Mc.	ENP

ENP 2020



## ملخص:

يتعلق العمل المقدم في هذه المذكرة بتطبيق مفاهيم الحوسبة الصناعية للتحكم في المصعد. وذلك باستخدام المتحكم المنطقي المبرمج S7-1200 لشركة SIEMENS. كما استعملت لمراقبة المصعد واجهة الآلة البشرية SIMATIC TP2200 COMFORT المتوفرة في بيئة TIA Portal والتي نقوم بتوصيلها بوحدة المعالجة المركزية (CPU) عبر شبكة PROFINET.

## الكلمات المفتاحية:

مصعد ، PLC ، وحدة المعالجة المركزية ، واجهة الآلة البشرية ، TIA Portal V15 ، WinCC ، PROFINET ، النظام الآلي

## Abstract:

The work presented in this dissertation relates to the application of the concepts of industrial data processing in the automation and the supervision of an elevator. We use the automates of the series S7 of SIEMENS; which we program thanks to the software TIA Portal V15.

For the supervision of the elevator, we use the SIMATIC TP 2200 Comfort Human-Machine interface available on the TIA Portal environment; which we connect to our CPU via a PROFINET network.

## Keywords :

Elevator, API, CPU, Human-Machine interface, TIA Portal V15, WinCC, PROFINET, Automated system.

## **Résumé :**

Le travail présenté dans ce mémoire porte sur l'application des notions de l'informatique industrielle dans l'automatisation et la supervision d'un ascenseur. On utilise pour cela les automates de la série S7 de SIEMENS qu'on programme grâce au logiciel TIA Portal V15.

Pour la supervision de l'ascenseur, on utilise l'interface Homme-Machine SIMATIC TP2200 Comfort disponible sur l'environnement TIA Portal ; qu'on connecte à notre CPU via un réseau PROFINET.

## **Mots-clés :**

Ascenseur, API, CPU, interface Homme-Machine, TIA Portal V15, WinCC, PROFINET, Système automatisé.

# **Remerciements**

*Au terme de ce travail je tiens à exprimer ma profonde gratitude avant tout à **ALLAH** le tout puissant qui m'a procuré la volonté de mener à bien ce modeste travail.*

*Je tiens à exprimer toute ma reconnaissance à Monsieur **ELmadjid BERKOUK** pour son encadrement ; je le remercie de m'avoir orientée, aidée et conseillée afin de mener à bien mon travail.*

*Je remercie vivement les membres du jury d'avoir accepté et consacré du temps à la lecture et à l'examen de ce travail.*

*Un grand respect et un grand merci à mes chers parents pour leur soutien et leurs encouragements.*

*Je souhaiterais également remercier l'ensemble des enseignants du département d'automatique de l'ENP pour cette formation qui a été riche en connaissances.*

*À tous ces intervenants, mais également à tous ceux qui m'ont aidée, de près ou de loin, tout au long de mes études ; que ce mémoire soit une preuve de ma plus profonde et sincère reconnaissance.*

# ***Dédicace***

*Je dédie ce travail*

*À mes chers parents Mohamed et Karima.*

*À ma grand-mère Bahia.*

*À mes frères et sœurs.*

*À la mémoire de mon grand-père.*

*À mes chers amis : Bilel (S), Yasmine (K), Narimane (H).*

# Table des matières

<b>Liste des tableaux</b>	<b>11</b>
<b>Liste des figures</b>	<b>12</b>
<b>Liste des abréviations</b>	<b>15</b>
<b>Introduction générale</b>	<b>18</b>
<b>Chapitres 1 : Automates Programmables Industriels (API)</b>	<b>19</b>
I.1 Introduction .....	20
I.2 Généralités sur les automates programmables industrielles (API).....	20
I.2.1 Définition.....	20
I.2.3 Domaines d'applications des API.....	20
I.2.4 Avantages et inconvénients des API .....	21
I.2.5 Aspect externe des API.....	22
I.2.5.1 Type compacte ou centralisé.....	22
I.2.5.2 Type modulaire.....	23
I.2.6 Structure interne des API .....	23
I.2.6.1 L'unité centrale .....	23
I.2.6.1.1 Unité de traitement (processeur / CPU) .....	24
I.2.6.1.2 La mémoire .....	24
I.2.6.1.2.1 Zone mémoire programme.....	24
I.2.6.1.2.2 Zone mémoire système.....	24
I.2.6.1.2.3 Zone mémoire de données .....	24

I.2.6.2	Les interfaces et les cartes d'entrées/sorties .....	25
I.2.6.2.1	Interfaces et cartes d'entrées .....	25
I.2.6.2.2	Interfaces et cartes de sorties .....	25
I.2.6.3	Le bloc d'alimentation.....	26
I.2.6.4	Le bus .....	26
I.2.7	Critères de choix d'un API [3] .....	27
I.3	Les langages de programmation des API .....	28
I.3.1	Lecture du programme [11].....	28
I.3.2	Le langage Ladder .....	29
I.3.3	Le langage List.....	30
I.3.4	Langage FBD [2].....	31
I.3.5	Le langage ST .....	31
I.3.6	Le langage SFC.....	32
I.4	Conclusion.....	33
 <b>Chapitre 2 : Le logiciel TIA Portal</b>		<b>34</b>
II.1	Introduction.....	35
II.2	Présentation du logiciel .....	35
II.3	Partie programmation .....	36
II.3.1	Création d'un projet sous TIA Portal .....	36
II.3.2	La configuration du matériel .....	36
II.3.3	La vue du projet .....	39
II.3.4	La disposition des éléments sur le châssis.....	40
II.3.5	L'adressage des entrées/sorties .....	41
II.3.6	Déclaration des variables .....	42
II.3.7	Les blocs de programmation.....	43
II.3.7.1	Le bloc d'organisation OB1 .....	44
II.3.7.2	Les blocs supplémentaires.....	45

II.3.7.2.1	Les blocs fonctionnels (FB).....	46
II.3.7.2.2	Les fonctions (FC).....	46
II.3.7.2.3	Les blocs de données (DB) .....	47
II. 3.8	Les instructions de base .....	48
II.3.8.1	Operations logiques sur des bits .....	48
II.3.8.2	Operateurs de comparaison.....	48
II.3.8.3	Les compteurs .....	49
II.3.8.4	Les éléments de temporisation.....	49
II.4	La partie simulation (PLCSIM) .....	50
II.4.1	L'adresse IP de la CPU .....	50
II.4.2	Simulation .....	51
II.5	Partie supervision sous WINCC.....	57
II.5.1	la vue de l'interface Homme-Machine.....	58
II.5.2	Configuration des variables HMI .....	59
II.5.3	Configuration d'un voyant.....	60
II.5.4	Configuration d'un bouton.....	61
II.5.5	Démarrage de la simulation de l'interface .....	63
II.6	Conclusion .....	64
 <b>Chapitre 3 : Généralités sur les ascenseurs</b>		<b>65</b>
III.1	Introduction .....	66
III.2	Historique [6] .....	66
III.3	Définition .....	67
III.4	Les différents Types d'ascenseurs .....	68
III.4.1	Monte-charge .....	68
III.4.2	Monte-charge industriel .....	68
III.4.3	Ascenseur pour personne .....	68
III.5	Catégories d'ascenseurs .....	69

III.5.1	Les ascenseurs hydrauliques.....	69
III.5.1.1	Description.....	69
III.5.1.2	Les avantages de l'ascenseur hydraulique .....	70
III.5.1.3	Les inconvénients de l'ascenseur hydraulique.....	71
III.5.2	Les ascenseurs à traction à câbles .....	71
III.5.2.1	Description .....	71
III.5.2.2	Avantages des ascenseurs à traction à câbles.....	73
III.5.2.3	Inconvénients des ascenseurs à traction à câbles [7] .....	73
III.5.2.4	Motorisation des ascenseurs à traction a câbles .....	74
III.5.2.4.1	Moteur-treuil à vis sans fin.....	74
III.5.2.4.1.1	Description.....	74
III.5.2.4.1.2	Avantages des moteurs-treuil à vis sans fin .....	74
III.5.2.4.1.2	Inconvénients des moteurs-treuil à vis sans fin .....	74
III.5.2.4.2	Moteur-treuil planétaire.....	75
III.5.2.4.2.1	Description .....	75
III.5.2.4.2.2	Avantages des moteurs-treuil planétaires .....	76
III.5.2.4.3	Les moteurs à attaque directe.....	76
III.5.2.4.3.1	Description .....	76
III.5.2.4.3.2	Avantages des moteurs gearless .....	77
III.5.2.4.3.3	Inconvénients des moteurs gearless [8] [10] .....	77
III.5.2.5	Critères de choix des moteurs à traction .....	78
III.5.2.6	Les constituants d'un ascenseur à traction a câbles [7] [9] .....	78
III.6	Critères de choix des ascenseurs [7] [8] .....	85
III.7	La sécurité dans les ascenseurs .....	85
III.8	Conclusion .....	86
<b>Chapitre 4 : Automatisation et supervision des ascenseurs par API</b>		<b>87</b>
IV.1	Introduction.....	88

IV.2	Commande d'un ascenseur avec API .....	88
IV.2.1	Fonctionnement de l'ascenseur.....	88
IV.2.2	Cahier des charges .....	90
IV.2.2.1	Descendre vers le premier étage .....	92
IV.2.2.2	Vers le deuxième étage .....	92
IV.2.2.3	Vers le troisième étage.....	93
IV.2.2.6	Procédure d'urgence.....	93
IV.2.3	Les entrées et sorties de l'ascenseur.....	94
IV.2.3.1	Les entrées de l'API.....	95
IV.2.3.2	Les sorties de l'API.....	96
IV.2.4	La configuration matérielle.....	96
IV.2.4.1	La CPU 1214C DC/DC/DC.....	96
IV.2.4.1.1	Description [11].....	96
IV.2.4.2	Module d'entrées/sorties SM 1223 DI16/DQ16 x DC24V .....	97
IV.2.4.2.1	Description [12].....	97
IV.2.5	Supervision de l'ascenseur sous WinCC.....	99
IV.2.5.1	La vue principale .....	100
IV.2.5.2	La vue ascenseur .....	100
IV.2.5.3	Vue du grafctet des étages « vers étage i » .....	103
IV.2.5.4	La vue procédure d'urgence .....	105
IV.2.5.5	La vue alarmes.....	106
IV.3	Commande de deux ascenseurs avec le même API.....	107
IV.3.1	Fonctionnement des deux ascenseurs .....	107
IV.3.2	Cahier des charges .....	107
IV.3.3	La configuration matérielle.....	108
IV.3.4	Supervision des deux ascenseurs sous WinCC .....	108
IV.3.4.1	La vue principale .....	108

IV.3.4.2 La vue des ascenseurs.....	109
IV.3.4.3 La vue des procédures d'urgences .....	109
IV.4 Commande de deux ascenseurs avec deux API communicants (PROFINET) .....	111
IV.4.1 Communication entre les deux CPU .....	112
IV.4.2 Supervision des deux ascenseurs sous WinCC.....	112
IV.5 Conclusion .....	113
<b>Conclusion générale</b>	<b>114</b>
<b>Bibliographie</b>	<b>115</b>
<b>Annexe A</b>	<b>117</b>
<b>Annexe B</b>	<b>127</b>
<b>Annexe C</b>	<b>139</b>
<b>Annexe D</b>	<b>147</b>

# Liste des tableaux

Tableau I. 1 : Instructions de base en langage List. [13] .....	30
Tableau I. 2 : Les opérateurs en langage ST. [14] .....	32
Tableau II. 1 : Eléments de la configuration du matériel. ....	37
Tableau II. 2 : tableau explicatif des différents éléments présent sur la vue projet. ...	39
Tableau II. 3 : Tableau des éléments accessible dans les blocs de programme.....	44
Tableau II. 4 : Eléments disponibles sur la fenêtre « ajouter un nouveau bloc ». ....	45
Tableau II. 5: Tableau explicatif des éléments de la figure III.36. ....	58
Tableau III. 1 : Tableau comparatif entre les différentes motorisations. ....	77
Tableau IV. 1 : tableau explicatif des abréviations du grafcet de l’ascenseur. ....	91
Tableau IV. 2 : entrées de l’API. ....	95
Tableau IV. 3 : sorties de l’API. ....	96
Tableau IV. 4 : tableau explicatif des différents éléments de la vue « ascenseur »...	102
Tableau IV. 5 : tableau explicatif des éléments du grafcet. ....	104
Tableau IV. 6 : tableau explicatif des éléments de la vue « procédure d’urgence ». .	105

# Liste des figures

Figure I. 1 : automate compact.....	23
Figure I. 2: automate modulaire.....	23
Figure I. 3: Carte d'entrées typique d'un API. [5] .....	25
Figure I. 4: Carte De sortie typique d'un API. [5] .....	25
Figure I. 5 : schéma de principe d'une alimentation. [5].....	26
Figure I. 6: structure interne des API. ....	27
Figure I. 7 : Exécution d'un programme API.....	29
Figure I. 8: Composants graphiques élémentaires d'un diagramme LD. ....	29
Figure I. 9: schéma de la structure des différents blocs du langage FBD. ....	31
Figure I. 10: Grafcet.....	33
Figure II. 1 : TIA Portal. ....	35
Figure II. 2 : création d'un nouveau projet.....	36
Figure II. 3 : configuration matérielle.....	36
Figure II. 4 : ajout d'un appareil.....	37
Figure II. 5 : choix de la CPU. ....	38
Figure II. 6 : vue projet.....	39
Figure II. 7 : disposition de la CPU sur le châssis. ....	40
Figure II. 8 : ajout des modules d'entrées et de sorties. ....	41
Figure II. 9: Adressage des entrées et des sorties sur TIA Portal. ....	41
Figure II. 10: disposition et adressage des entrées/sorties. ....	42
Figure II. 11 : déclaration des variables sur TIA Portal. ....	43
Figure II. 12 : Le bloc main (OB1). ....	44
Figure II. 13: ajout d'un nouveau bloc. ....	45
Figure II. 14 : bloc fonctionnel.....	46
Figure II. 15 : Les fonctions.....	47

Figure II. 16 : ajout d'un bloc de données global. ....	47
Figure II. 17 : Les instructions de base.....	48
Figure II. 18 : Les opérateurs de comparaison. ....	48
Figure II. 19 : Les compteurs. ....	49
Figure II. 20 : Les temporisateurs.....	49
Figure II. 21 : Adresse IP de la CPU. ....	50
Figure II. 22 : Compilation du programme. ....	51
Figure II. 23 : recherche d'un appareil cible.....	51
Figure II. 24 : configuration de la connexion entre la CPU et l'appareil cible. ....	52
Figure II. 25 : chargement du programme dans l'API de simulation.....	52
Figure II. 26 : démarrage de la simulation. ....	53
Figure II. 27 : PLCSIM. ....	53
Figure II. 28 : Création d'un projet sur PLCSIM. ....	54
Figure II. 29 : table des variables. ....	54
Figure II. 30 : Variables du projet. ....	55
Figure II. 31 : Liaison en ligne. ....	55
Figure II. 32 : mise à 1 d'un bit sur PLCSIM. ....	56
Figure II. 33 : visualisation des états des variables du projet. ....	56
Figure II. 34 : choix de l'interface Homme-Machine.....	57
Figure II. 35: configuration de la connexion entre l'API et l'interface. ....	57
Figure II. 36: vue racine de l'interface Homme-Machine. ....	58
Figure II. 37: Table des variables HMI. ....	59
Figure II. 38: cycle d'acquisition des variables HMI. ....	60
Figure II. 39 : ajout de l'objet « cercle » dans la vue HMI. ....	60
Figure II. 40: Configuration d'une animation «représentation» pour l'objet cercle. ..	61
Figure II. 41: ajout du memento « A_Man ».....	61
Figure II. 42: ajout de l'objet « bouton » dans la vue HMI. ....	62
Figure II. 43: Configuration d'un évènement pour l'objet «bouton ».....	62
Figure II. 44: Compilation de l'interface Homme-machine.....	63
Figure II. 45: Lancement de WinCC.....	63
Figure II. 46: Supervision sous WinCC.....	64
Figure III. 1 : ascenseur hydraulique.....	70
Figure III. 2 : Les différentes motorisations des moteurs à traction à câbles.....	72

Figure III. 3 : Moteur-treuil à vis sans fin.....	74
Figure III. 4 : Moteur-treuil planétaire. ....	75
Figure III. 5 : Réducteur planétaire. ....	75
Figure III. 6 : Moteur à attaque directe Z4-Z20.....	76
Figure III. 7 : Suspension de la cabine. ....	79
Figure III. 8 : cuvette d'ascenseur. ....	80
Figure III. 9 : Fin de course pour ascenseur.....	80
Figure III. 10 : frein mécanique de l'ascenseur. ....	82
Figure III. 11 : armoire de commande pour ascenseur (drive). ....	82
Figure III. 12: Moto-génératrice.....	83
Figure III. 13 : réducteur de vitesse.....	84
Figure IV. 1 : graficet de l'ascenseur « vers étage 1 » . ....	92
Figure IV. 2 : graficet de l'ascenseur « vers étage 2 » . ....	92
Figure IV. 3 : graficet de l'ascenseur « vers étage 3 » . ....	93
Figure IV. 4 : graficet ascenseur « procédure d'urgence » . ....	94
Figure IV. 5 : CPU 1214C DC/DC/DC. ....	97
Figure IV. 6 : module d'entrées/sorties SM 1223 DI16/DQ16 x DC24.....	98
Figure IV. 7 : disposition de la CPU et des modules E/S sur le châssis.....	98
Figure IV. 8 : communication PROFINET PLC-HMI.....	99
Figure IV. 9 : la vue principale.....	100
Figure IV. 10: la vue ascenseur. ....	101
Figure IV. 11 : cabine d'ascenseur en mouvement (monte/descend). ....	101
Figure IV. 12 : vue « vers l'étage 5 » . ....	103
Figure IV. 13 : vue « vers l'étage 4 » . ....	104
Figure IV. 14 : vue procédure d'urgence. ....	105
Figure IV. 15: vue « alarmes » . ....	106
Figure IV. 16 : la vue principale.....	108
Figure IV. 17 : la vue des ascenseurs. ....	109
Figure IV. 18 : la vue des procédures d'urgences. ....	109
Figure IV. 19 : états des appels extérieurs avant la procédure d'urgence. ....	110
Figure IV. 20 : basculement des appels extérieurs lors de la procédure d'urgence. ...	111
Figure IV. 21 : communication PROFINET PN/IE PLC-PLC. ....	112
Figure IV. 22 : communication PROFINET PN/IE PLC_HMI_PLC. ....	113

# Liste des abréviations

CPU : Central Processing Unit.

OB : Bloc d'Organisation.

FB : Bloc Fonctionnel.

FC : Fonctions.

DB : Bloc de Données.

CTU : Counter UP.

TON : Timer ON.

API : Automates Programmables Industrielles.

ROM : Read-Only Memory.

RAM : Random Access Memory.

PROM : Programmable Read-Only Memory.

EEPROM : Electrically-Erasable Programmable Read-Only Memory.

LD : Ladder Diagram.

IL : Instruction List.

SFC : Sequential Function Chart.

FBD : Function Bloc Diagram.

ST : Structured Text.

E/S : Entrées/Sorties.

TIA Portal : Totally Integrated Automation Portal.

HMI : Interface Homme-Machine.

SM : Signal Module.

SIMATIC : Siemens Automatic.

PO : Partie Opérative.

PC : Partie Commande.

# Introduction générale

Dans le cadre de l'obtention du diplôme d'ingénieur automaticien, le département d'Automatique de l'ENP a proposé le projet de l'étude et de la commande d'un ascenseur par automates programmables. Ce projet permet à l'étudiant de synthétiser l'ensemble des connaissances obtenues lors de sa formation mais également d'enrichir ces acquis notamment en ce qui concerne l'informatique industrielle.

Ce travail a pour objectif l'élaboration d'une solution à base d'automates SIMATIC S7-1200 de SIEMENS pour l'automatisation ; et la supervision d'un ascenseur a cinq étages. Cette solution sera ensuite adaptée au cas de deux ascenseurs communicants. La réalisation de ce projet, nécessite une étude préalable des ascenseurs ainsi qu'une connaissance du logiciel de programmation des automates TIA Portal.

La supervision de l'ascenseur se fera sur le logiciel de conception des interfaces Homme-Machine, WinCC présent dans l'environnement de TIA Portal. La communication entre cette interface et l'automate est réalisée grâce à un réseau PROFINET.

Chapitre 1 :  
Les Automates  
Programmables  
Industrielles (API).

## **I.1 Introduction**

De nos jours, les automates programmables industrielles A.P.I envahissent le monde de l'automatisation. Cette utilisation en masse est due à leurs avantages multiples notamment le fait de bénéficier d'une grande flexibilité, et d'être adapter aux environnements hostiles.

Ce chapitre sera divisé en deux parties. Dans la première, nous verrons des généralités sur les automates programmables ; leur structure interne et externe, leurs avantages et inconvénients, ainsi que les fonctions principales réalisés par les API. Dans la seconde partie, nous aborderons les différents langages de programmation des API.

## **I.2 Généralités sur les automates programmables industrielles (API)**

### **I.2.1 Définition**

Un automate programmable industriel (API) est un dispositif électronique programmable, qui travaille en interaction avec les éléments du système automatisé dans le but d'assurer l'automatisation et la surveillance de processus industriels en temps réel.

Comme défini dans la norme NFC 63-850 ; l'API comporte une mémoire programmable par un utilisateur automaticien à l'aide d'un langage adapté, pour le stockage interne des instructions composant les fonctions d'automatisme.

### **I.2.3 Domaines d'applications des API**

À ce jour, les automates programmables se trouvent dans tous les champs d'applications, de la station de pompage au remplissage des bouteilles, en passant par le contrôle de robot , les automates programmables (API) sont partout et dans toutes les industries, ils s'adaptent aisément aux besoins des clients et répondent à toutes sorte de problèmes du plus simple au plus complexe.

Pour les applications simples on retrouve les API de type monobloc, alors que les API de type modulaires se trouvent dans les applications complexes d'envergure et on cite :

- l'industrie automobile
- l'industrie agroalimentaire
- les sociétés de transports
- l'industrie manufacturière
- régulation de processus chimique, métallurgique...
- industrie du bâtiment (contrôle de l'éclairage, chauffage...)

### **I.2.4 Avantages et inconvénients des API**

Les automates programmables industriels, proposent des solutions programmées, ce qui présente des avantages considérables par rapport à la technologie de logique câblée, et on cite :

- simplification du câblage et donc la réduction des coûts de câblage.
- exigent une maintenance moindre et donc la réduction des couts de maintenance.
- plus simple : un API traite une grande variété d'applications.
- plus flexible : les modifications sont mises en œuvre de façon logicielle (modification du programme en mémoire).
- Plus compacts en effet ils permettent la réduction de l'espace requis pour l'installation.
- Gain en robustesse et en fiabilité en raison de la réduction du nombre de composants mécaniques.
- Possibilité de fonctionnement dans des environnements particulièrement hostiles dû au fait qu'il soit composé d'éléments robustes.
- Plus rapide : Ils permettent d'assurer un temps d'exécution minimal, respectant un déterminisme temporel et logique, garantissant un temps réel
- Extension facile (ajout de module supplémentaire pour les besoin de l'application)

Les API sont des types particuliers d'ordinateurs adaptés pour la commande des processus en milieu industriel. Voici quelques différences et avantages des API sur les ordinateurs : [3]

- Ils sont solides et conçus pour supporter les vibrations, les températures basses ou élevées, l'humidité et le bruit, à la différence des ordinateurs personnels qui ne sont pas faits pour opérer dans des milieux hostiles.
- Les interfaces des entrées et des sorties sont intégrées à l'automate. Les API au format modulaire peuvent être facilement étendues pour recevoir un plus grand nombre d'entrées-sorties.
- Ils sont moins adaptés au stockage à long terme et à l'analyse des données.

Ils sont d'une fiabilité supérieure et moins sujets aux dysfonctionnements que les ordinateurs personnels. En contrepartie, ils présentent les inconvénients suivants : [3]

- Le prix est cher : le prix est notamment dépendant du nombre d'entrées/sorties nécessaires, de la mémoire dont on veut disposer pour réaliser le programme, de la présence ou non de modules métier.
- La connaissance des langages de programmation.

## **I.2.5 Aspect externe des API**

Les automates programmables industriels présentent deux types de structure extérieure :

### **I.2.5.1 Type compacte ou centralisé**

Souvent destinés à la commande des petits automatismes, les API de type compacte intègrent sur le même rack le processeur, les entrées/sorties et l'alimentation. Néanmoins ce type d'automates peut recevoir des extensions en nombre limité.

La programmation de ce types d'automates ce fait par ZELIO pour les API Schneider, LOGO pour les API de Siemens et MILLENIUM pour les API de Crouzet, pour ne citer que cela

### I.2.5.2 Type modulaire

A la différence des automates de type compacte, les automates programmables de type modulaire sont destinés aux automatismes plus complexes. Pour des besoins de souplesse de configuration et de flexibilité, le processeur, l'alimentation et les interfaces d'entrées/sorties résident dans des modules séparés et sont fixées sur un ou plusieurs Racks. Ces racks peuvent accueillir plusieurs modules supplémentaires allant des simples modules d'entrées/sortie TOR aux plus complexes. Elle permet ainsi des configurations évolutives et personnalisées ; Ces différents modules s'articulant autour d'un canal de communication appelé « bus interne ».

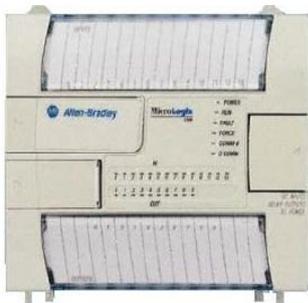


Figure I. 1 : automate compact.



Figure I. 2: automate modulaire.

## I.2.6 Structure interne des API

La structure interne d'un API est très similaire à celle d'un micro-ordinateur, elle comporte quatre principales parties :

- Une unité centrale ;
- Des modules et des interfaces d'entrées-sorties ;
- Une alimentation ;
- Le bus.

### I.2.6.1 L'unité centrale

Elle comporte

- l'unité de traitement (CPU)
- la mémoire

### **I.2.6.1.1 Unité de traitement (processeur / CPU)**

Le processeur a pour rôle est le traitement des instructions du programme, de manière cyclique. il réalise toute les fonctions de l'application (les fonctions logiques, les fonctions de temporisation, de comptage, de calcul de PID...) il est également en charge de l'organisation des relations entre la mémoire et les interfaces d'entrées/sorties, De la surveillance, du diagnostic de l'automate et du dialogue avec le terminal de programmation. [3][4]

### **I.2.6.1.2 La mémoire**

La mémoire centrale se décompose en plusieurs zones mémoire

#### **I.2.6.1.2.1 Zone mémoire programme**

Elle est destinée au stockage des instructions qui constituent le programme de fonctionnement de l'automatisme, en général pour cette zone mémoire sont utilisés des EEPROM.

#### **I.2.6.1.2.2 Zone mémoire système**

Destinée pour le stockage du système d'exploitation. En général, pour cette zone mémoire sont utilisés des mémoires PROM et parfois même des mémoires ROM. La mémoire système est programmable à la fabrication et ne peut pas être reprogrammé.

#### **I.2.6.1.2.3 Zone mémoire de données**

Destinée pour le stockage des états des entrées/sorties mises à jour par le processeur à chaque cycle, ainsi que pour le stockage des variables internes tels que : les valeurs de comptage, de temporisation ... cette zone mémoire utilise des mémoires RAM, elle peut donc être modifié autant de fois que nécessaire mais perdent l'information en cas de coupure de l'alimentation.

## I.2.6.2 Les interfaces et les cartes d'entrées/sorties

### I.2.6.2.1 Interfaces et cartes d'entrées

Elles reçoivent les informations des capteurs et du pupitre utilisateur, chaque capteur et chaque entrée du pupitre utilisateur possède une adresse spécifique sur l'interface d'entrée.

Chaque information reçu représente un signal, les cartes d'entrées adaptent ce signal en le mettant en forme par un filtrage c'est-à-dire l'élimination des parasite et en isolant électriquement (opto couplage) l'unité de commande de la partie opérative.

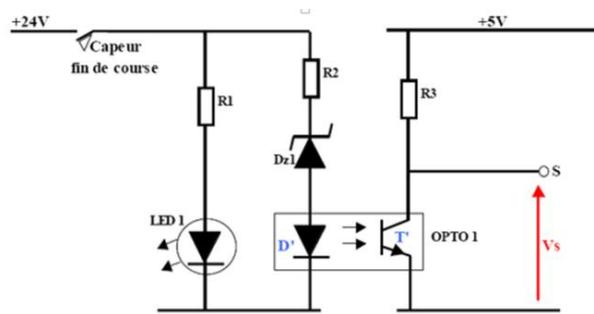


Figure I. 3: Carte d'entrées typique d'un API. [5]

### I.2.6.2.2 Interfaces et cartes de sorties

Les interfaces de sortie permettent de commander les pré-actionneurs et éléments de signalisation (voyants..), chaque pré-actionneur est relié à une adresse spécifique de l'interface de sortie.

Les cartes de sorties adaptent les niveaux de tensions de l'unité de commande à celle de la partie opérative du système en garantissant une isolation galvanique entre ces dernières.[5]

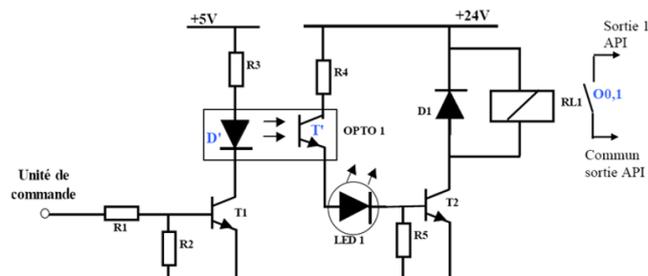


Figure I. 4: Carte De sortie typique d'un API. [5]

Le nombre de ces entrées/sorties varie suivant le type d'automate. Les cartes d'E/S ont une modularité de 8, 16 ou 32 voies. Les tensions disponibles sont normalisées (24, 48, 110 ou 230V continu ou alternatif ...). [3][5]

### I.2.6.3 Le bloc d'alimentation

Il assure la distribution de l'énergie nécessaire à l'automate. Il élabore à partir d'un réseau 220v alternatif, les tensions internes distribuées aux différents modules : +5V, 12V et 24V continu.

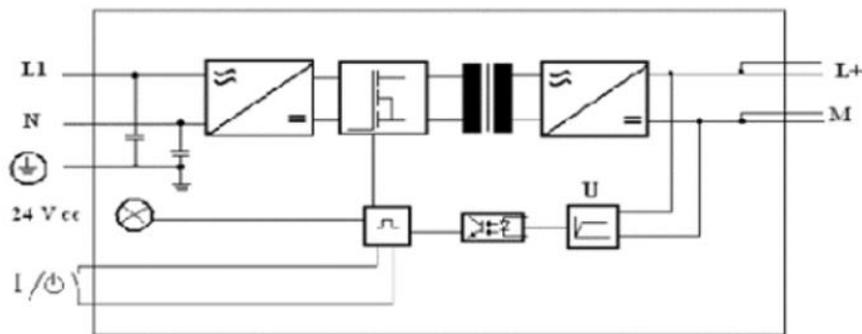


Figure I. 5 : schéma de principe d'une alimentation. [5]

### I.2.6.4 Le bus

Le bus interne assure la communication entre l'automate et ses modules, suivant l'information qu'il véhicule le bus interne est décomposé en plusieurs sous-ensembles :

- Bus de données.
- Bus d'adresses.
- Bus de contrôle
- Bus de distribution des tensions issues du bloc d'alimentation.

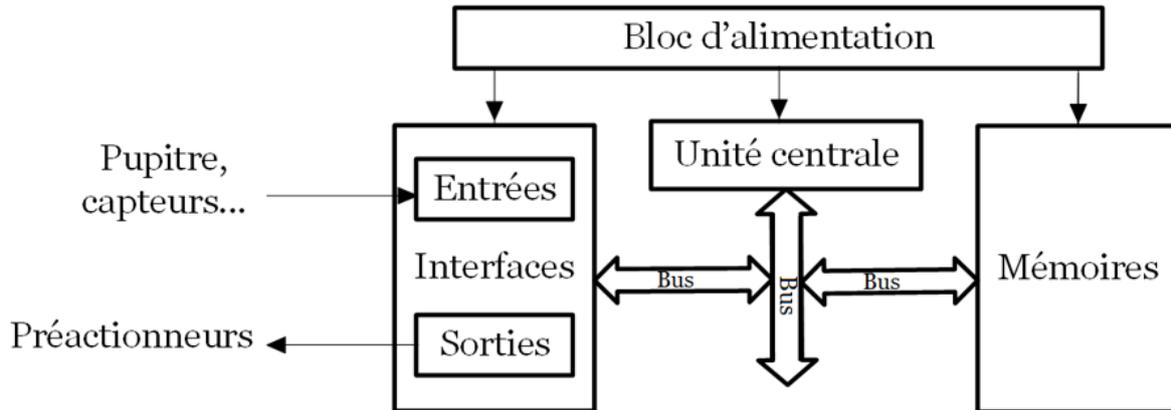


Figure I. 6: structure interne des API.

### I.2.7 Critères de choix d'un API [3]

- Le nombre et la nature d'entrées/sorties intégrés.
- Temps de traitement (vitesse).
- Capacité de la mémoire.
- Fonctions ou modules spéciaux (Nombre de compteurs, Nombre de temporisateurs...)
- La communication avec les autres systèmes
- Les moyens de dialogue et le langage de programmation
- La documentation, le service après-vente, durée de la garantie, la formation.
- Les moyens de sauvegarde du programme
- La fiabilité, robustesse, immunité aux parasites

## I.3 Les langages de programmation des API

Les API possèdent l'avantage d'être programmable et donc un même automate peut être utilisé pour différentes applications, il suffit de changer le programme contenu dans l'API.

Un programme est une suite d'instruction, qui lorsqu'elles sont bien formulées, permettent l'exécution des opérations nécessaires au bon fonctionnement du système.

La norme IEC 1131-3 définit cinq langages de programmation pour les API, ces cinq langages sont :

- Le langage ladder **LD** (« Ladder Diagram », ou schéma à relais).
- Le langage list **IL** (« Instruction List », ou liste d'instructions).
- Le langage **FBD** (« Function Block Diagram », ou schéma par blocs).
- Le langage **SFC** (« Sequential Function Char »).
- Le langage **ST** (« Structured Text » ou texte structuré).

L'API traduit le langage de programmation en langage compréhensible directement par le microprocesseur. Ce langage est propre à chaque constructeur, il est lié au matériel mis en œuvre. [11]

### I.3.1 Lecture du programme [11]

En fonctionnement, le microprocesseur de l'API lit en permanence le programme et exécute les instructions les unes à la suite des autres. L'ensemble de ces tâches réalisées par le microprocesseur s'appelle une scrutation.

Une sortie spécifique de l'automate qui change d'état lorsqu'une défaillance de l'automate survient permet de couper le circuit d'arrêt d'urgence du système, cette sortie de sécurité est appelée chien de garde.

La lecture du programme est conditionné par la fonction RUN / STOP de l'API. (Marche/Arrêt).



Figure I. 7 : Exécution d'un programme API.

### I.3.2 Le langage Ladder

Le langage Ladder, aussi appelé schéma Ladder est de loin le langage de programmation le plus utilisé pour la programmation d'automates, il est facile à comprendre et simple à manipuler. Le langage Ladder est une représentation graphique d'équations booléennes, chaque équation représente une instruction, l'instruction est composée d'une séquence de contacts (interrupteurs à l'état fermé ou ouvert) disposés en série et/ou en parallèle, représentant les entrées ; et de bobines représentant les sorties, ces dernières traduisent l'état logique du système. Chaque instruction Ladder est représentée par un réseau.

—   — Contact normalement ouvert	— >= — Contact supérieur ou égal ???
— / — Contact normalement fermé	— <= — Contact inférieur ou égal ???
—(S)— Relais à action SET « mise à un »	— > — Contact supérieur strictement ???
—(R)— Relais à action RESET « remise à zéro »	— < — Contact inférieur strictement ???
— P — Contact à détection de front montant	— == — Contact égalité ???
— N — Contact à détection de front descendant	
—( )— Relais associé à une variable de sortie ou interne	

a) Opérateurs logiques

b) Opérateurs de comparaison

Figure I. 8: Composants graphiques élémentaires d'un diagramme LD.

### I.3.3 Le langage List

Le langage List (instruction List) est un langage textuel de bas niveau constitué d'une suite d'instructions qui opèrent sur un résultat courant (ou registre IL) ; Chaque instruction doit contenir un opérateur pour indiquer l'opération qui doit être effectuée entre le résultat courant et l'opérande. Le résultat obtenu est stocké dans le résultat courant. [2]

Le langage liste d'instruction permet de transcrire sous forme de liste [13] :

- Un schéma à contact
- Un logigramme, équations booléennes
- Un grafcet

Il réalise aussi des fonctions d'automatisme telles que temporisation, comptage, pas à pas...

Instructions de test	
Désignation	Fonctions
LD	Le résultat est égal à l'opérande (load : lire la valeur).
LDN	Le résultat est égal à l'inverse de l'opérande (contact ouverture).
AND	ET logique entre le résultat et précédent et l'état de l'opérande.
ANDN	ET logique entre le résultat et précédent et l'état inverse de l'opérande.
OR	OU logique entre le résultat et précédent et l'état de l'opérande.
ORN	OU logique entre le résultat et précédent et l'état inverse de l'opérande.
XOR, XORN	OU exclusif.
Instructions d'action	
ST	L'opérande associé prend la valeur de la zone de test.
STN	L'opérande associé prend la valeur inverse de la zone de test.
S	L'opérande associé est mis à 1 lorsque le résultat de la zone de test est à 1.
R	L'opérande associé est mis à 1 lorsque le résultat de la zone de test est à 1.

Tableau I. 1: Instructions de base en langage List. [13]

### I.3.4 Langage FBD [2]

Le langage FBD (Functional Block Diagram) est un langage graphique. Il permet la construction de procédures complexes à partir de fonctions existantes de la librairie standard ou de la section des fonctions ou des blocs fonctionnels.

Le diagramme FBD décrit une fonction entre des variables d'entrée et des variables de sortie. Une fonction est décrite comme un réseau de blocs élémentaires. Les variables d'entrée et de sortie sont connectées aux blocs par des arcs de lien. Une sortie d'un bloc peut aussi être connectée sur une entrée d'un autre bloc.

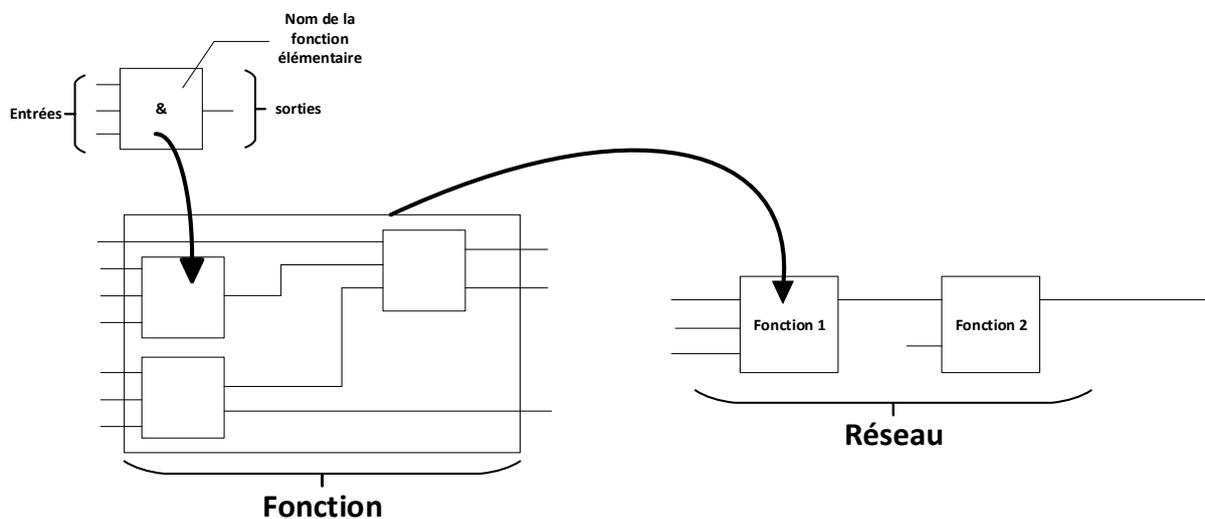


Figure I. 9: schéma de la structure des différents blocs du langage FBD.

### I.3.5 Le langage ST

Le langage ST (Structured Text) est un langage textuel de haut niveau dédié aux applications d'automatisation. Ce langage est principalement utilisé pour implémenter des procédures complexes, difficilement modélisables avec les langages graphiques. Le langage ST peut être utilisé pour la programmation des actions dans les étapes et les conditions associées aux transitions du SFC ou pour décrire les actions et les tests du langage FC. [2]

Opérateurs	Description	Opérateur	Description
Fonctions logiques			
NOT	NON	OR	OU
AND	ET	XOR	OU Exclusif
Opérateurs arithmétiques			
+	Addition	-	Soustraction
mod	Reste de la division entière	/	Division
		*	Multiplication
Opérateurs de comparaison			
=	Égal à	<>	Différent de
>	Supérieur à	>=	Supérieur ou égal à
<	Inférieur à	<=	Inférieur ou égal à

Tableau I. 2 : Les opérateurs en langage ST. [14]

### I.3.6 Le langage SFC

Le langage **Sequential Function Chart (SFC)** est un langage graphique utilisé pour décrire des opérations séquentielles. Le procédé est représenté comme une suite connue d'étapes, reliées entre elles par des transitions. Une condition booléenne est attachée à chaque transition. Un ensemble d'actions est attaché à chaque étape. Les conditions et actions sont décrites avec d'autres langages (ST, IL ou LD). [2]

Le langage SFC est régi par des règles graphiques :

- Deux étapes ne peuvent se suivre.
- Deux transitions ne peuvent se suivre.
- Dans un programme SFC ; il doit au moins exister une étape initiale.

Les composants de base du langage sont :

- étapes et étapes initiales ;
- transitions ;
- liaisons orientées ;
- renvoi à une étape.

Comme pour les grafcet, le langage SFC donne la possibilité de créer des liens de connexions multiples qui représentent des convergences ou des divergences ; il permet également de créer des macroétapes, qui sont des parties graphiques isolées (grafcets secondaires) représenté dans le graphique principal par un seul symbole.

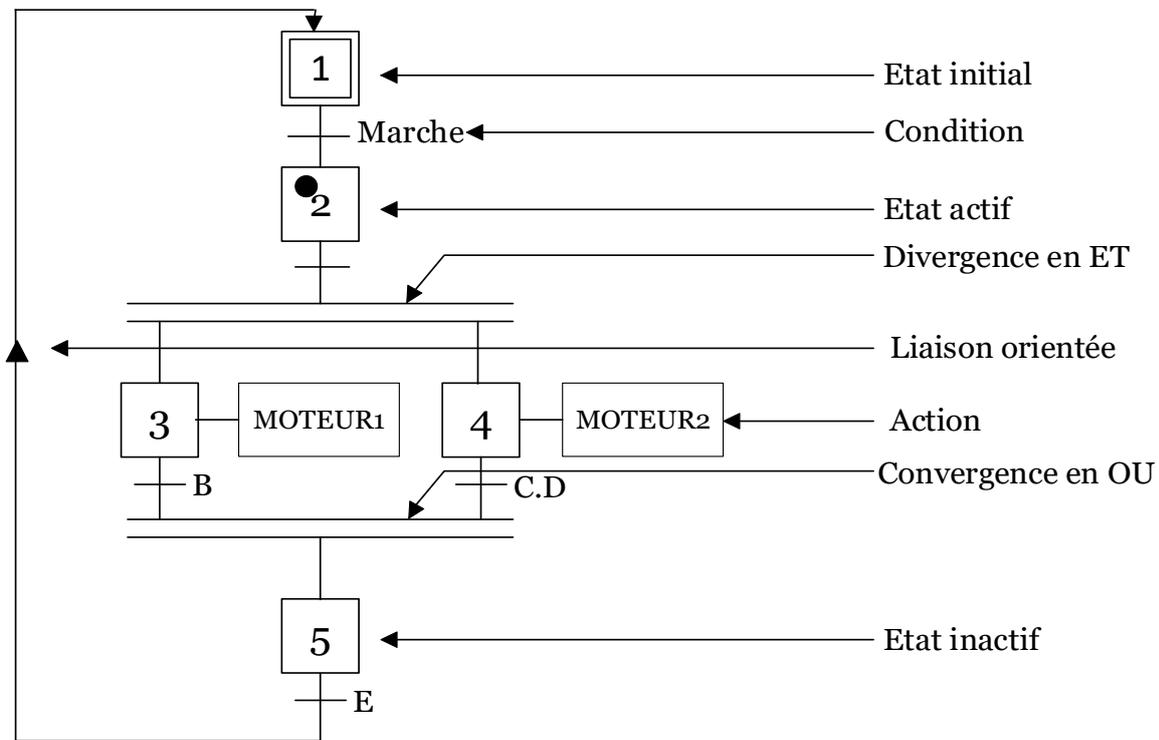


Figure I. 10: Grafcet.

## I.4 Conclusion

Grâce à leur architecture flexible qui s'adapte selon les besoins de l'application ; les automates programmables industriels sont de nos jours présents dans tous les secteurs de l'industrie. À travers ce chapitre on a abordé de plus près des notions de base de ces unités de commande ; notamment les différents langages de programmation.

# Chapitre 2 :

Le logiciel TIA Portal

## II.1 Introduction

Dans ce chapitre, nous allons nous intéresser au logiciel de programmation des automates de Siemens, à savoir le TIA Portal (Totally Integrated Automation Portal) qui permet la programmation des automates de la famille S7 (S7-300, S7-400, S7-1200 et les S7-1500). Nous verrons en détails les étapes à suivre pour la création d'un projet d'automatisation et sa supervision.

## II.2 Présentation du logiciel

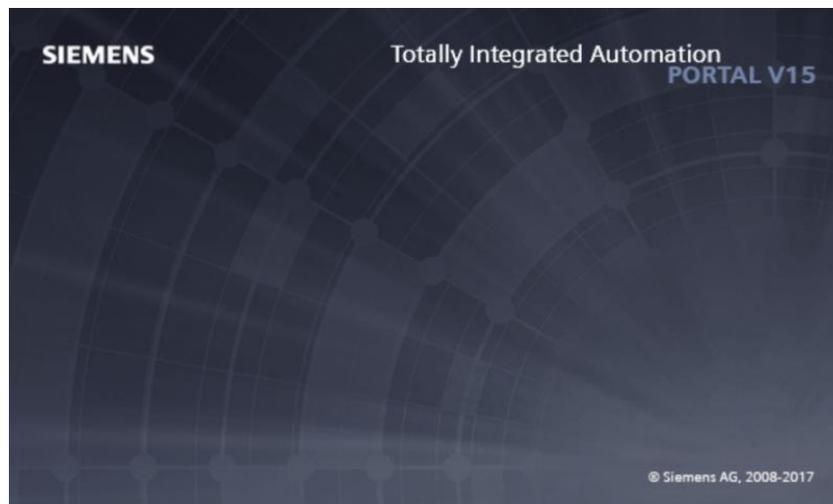


Figure II. 1 : TIA Portal.

« Totally Integrated Automation Portal » plus connu sous l'abréviation de TIA Portal est un logiciel de programmation des Automates Siemens de la famille S7. Il permet la gestion intégrale d'un projet d'automatisation d'un processus de la programmation à la mise en réseau de la solution. TIA portal est un logiciel assez complet. Il intègre la simulation sous PLCSim, et la visualisation (supervision) sous WinCC.

## II.3 Partie programmation

### II.3.1 Création d'un projet sous TIA Portal

À l'ouverture de TIA Portal, allez dans « créer un projet » et définissez le nom, et le chemin du projet en cours de création, puis cliquez sur « créer ».

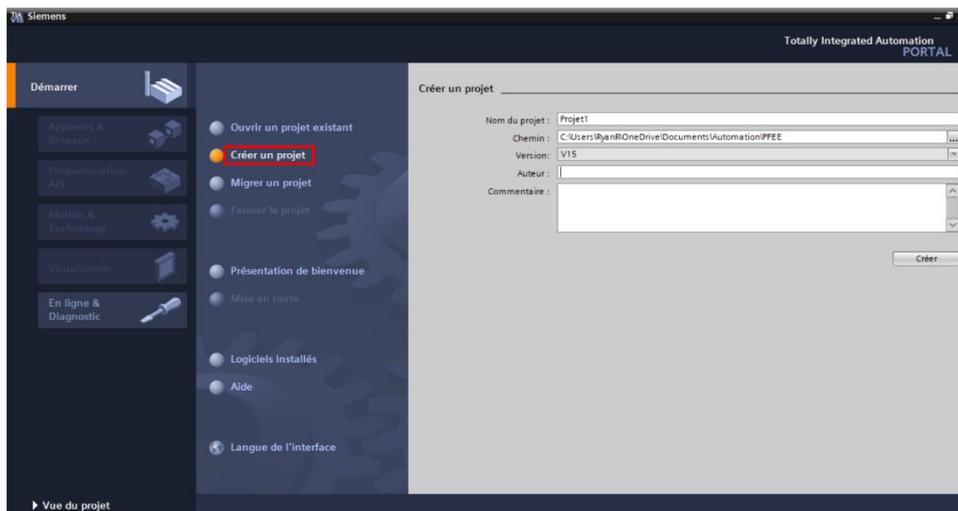


Figure II. 2 : création d'un nouveau projet.

### II.3.2 La configuration du matériel

Une fois le projet créé avec succès, le logiciel TIA Portal vous redirigera automatiquement vers « mise en route ». Sélectionnez « configurer un appareil ».

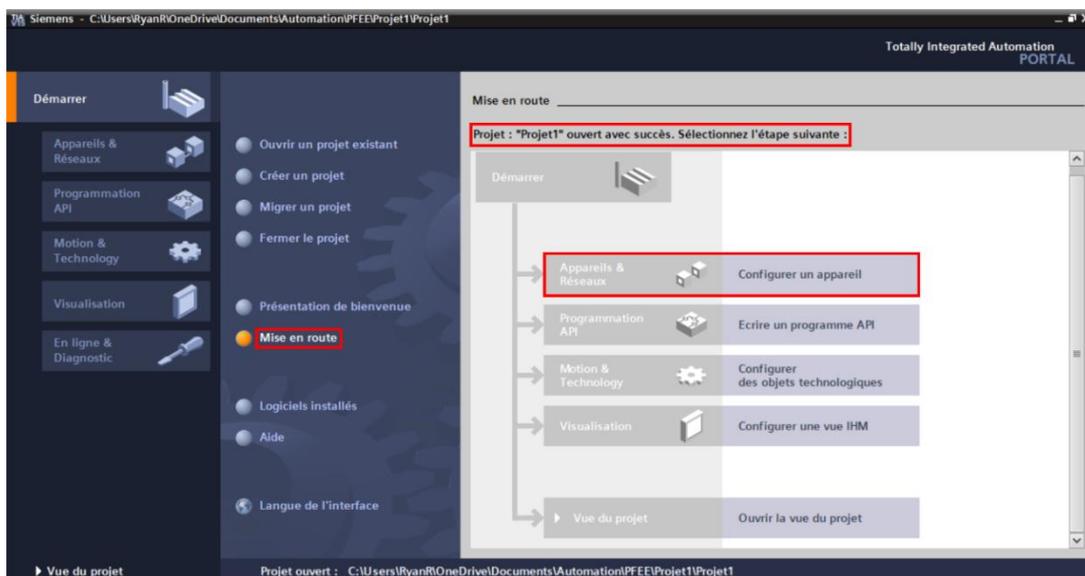


Figure II. 3 : configuration matérielle.

Une fois dans appareils et réseaux, cliquez sur « ajouter un appareil » ; puis sur « contrôleurs ».

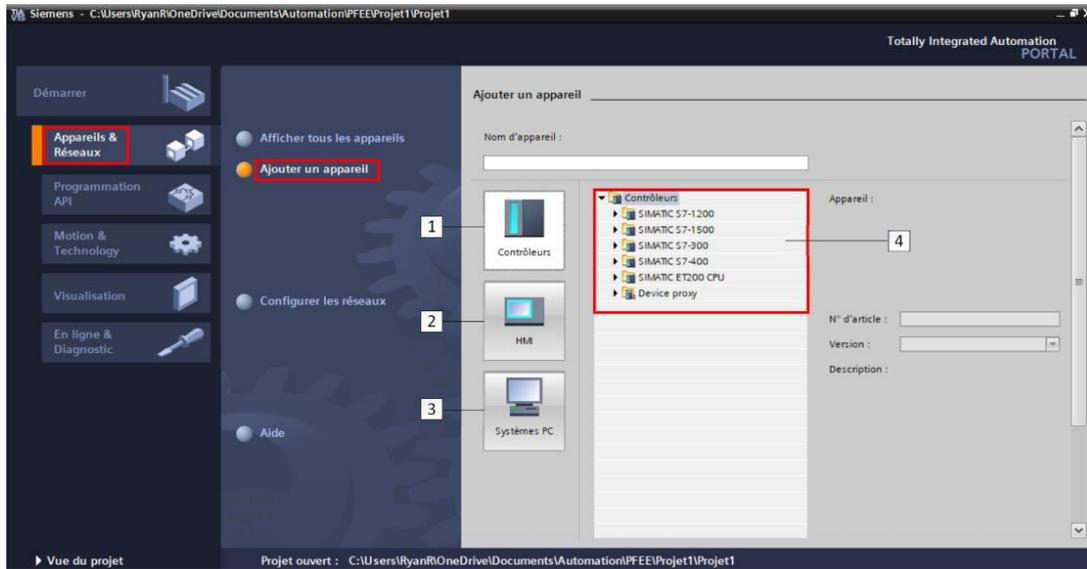


Figure II. 4 : ajout d'un appareil.

N°	Désignation
1	Un clique sur "contrôleurs" permet de choisir un automate parmi l'ensemble des automates de la série S7 disponible sur TIA Portal.
2	En cliquant sur "HMI" (interface homme machine) on choisi une des HMI parmi la liste des HMI disponible sur TIA Portal.
3	Un clique sur "Systèmes PC" nous donne accès aux micro-ordinateurs qu'on peut utilisé a l'instar des HMI.
4	la liste des automates programmables disponible sur TIA Portal et on retrouve : les S7-300, S7-400, S7-1200, S7-1500

Tableau II. 1 : Eléments de la configuration du matériel.

Dans contrôleurs, sélectionnez la famille d'automates qui vous intéresse parmi celle disponible, puis choisissez l'automate qui répond le plus au besoin de votre application.

A titre d'exemple, nous choisirons un automate de la famille « SIMATIC S7-1200 », en prenant comme CPU, la 1212C DC/DC/DC de référence : 6ES7 212-1AE40-0XB0.

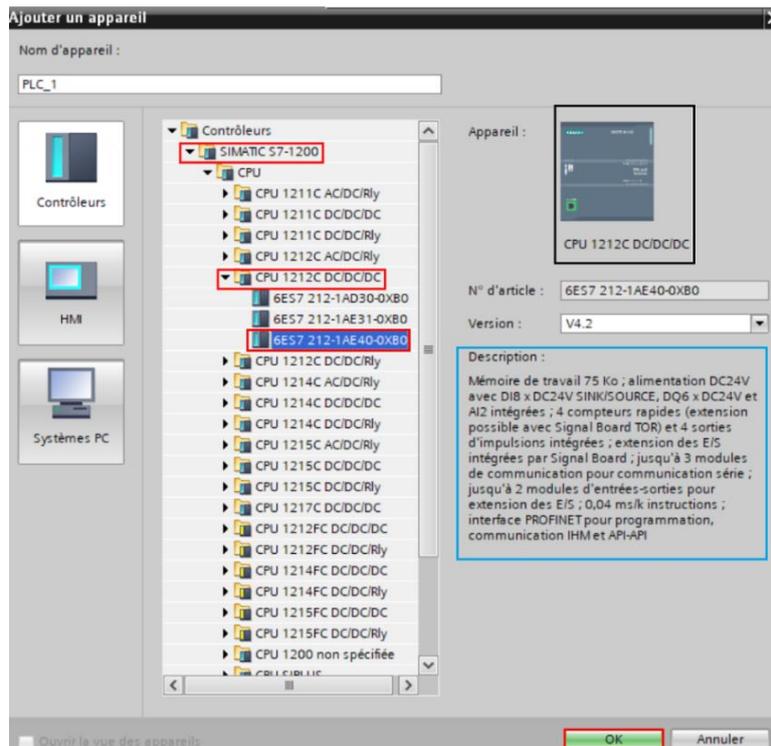


Figure II. 5 : choix de la CPU.

Le rectangle noir, nous donne une image réelle de l'automate et le rectangle en bleu est un bref descriptif des spécifications de la CPU sélectionnée (mémoire de travail, alimentation, mode de communication, les modules intégrés...).

### II.3.3 La vue du projet

A la différence de la vue du portail qui est une vue générale du projet, la vue du projet est une vue structurée de l'ensemble des éléments qui constitue le projet.

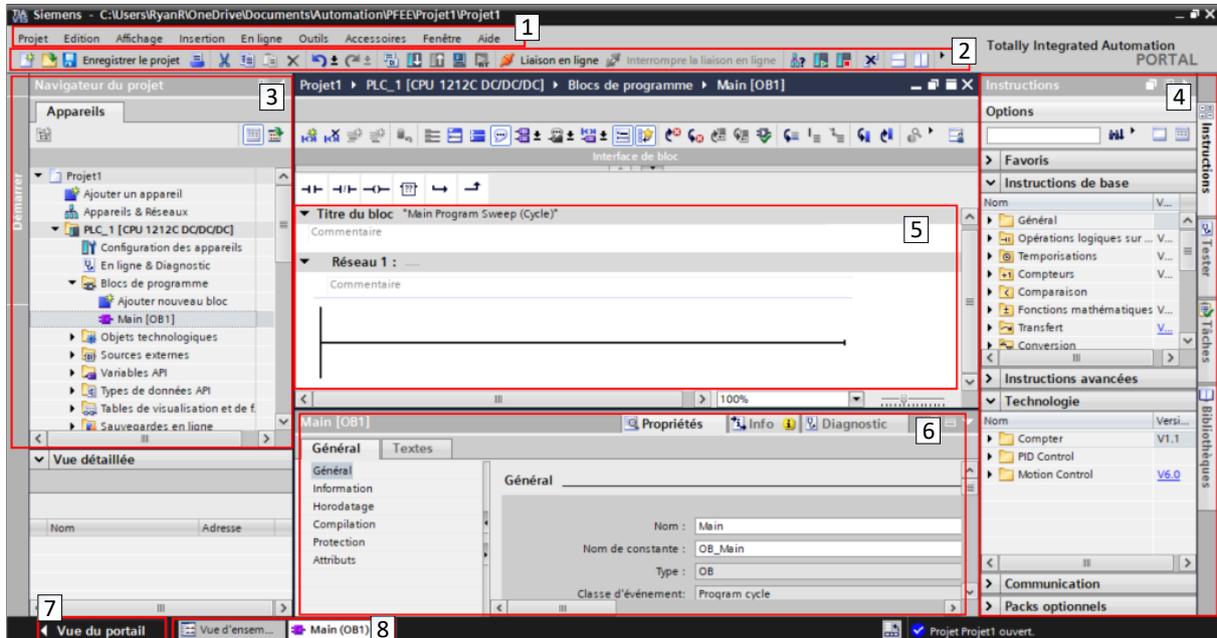


Figure II. 6 : vue projet.

N°	Désignation
1	Barre de menu
2	Barre des fonctions, et on y retrouve :  : compiler.  : chargement dans l'API.  : démarrer la simulation/démarrage de PLCSIM.  : RUN.  : STOP.
3	Navigateur de projet, on y retrouve l'ensemble des éléments du projet la CPU, les blocs de programmation, les variables de l'API, la HMI ...
4	Intrusions de base, instructions avancés, librairies...
5	Le bloc central, permet l'affichage et la manipulation de l'élément sélectionné dans le navigateur de projet, ici le bloc d'organisation OB1
6	Propriétés et informations du bloc central ou d'un élément sélectionné de ce bloc.
7	Accès a la vue du portail.
8	L'ensemble des fenêtres du projet qui sont ouvertes (éléments du navigateur de projet).

Tableau II. 2 : tableau explicatif des différents éléments présent sur la vue projet.

## II.3.4 La disposition des éléments sur le châssis

Le châssis est l'élément sur lequel est disposé le processeur ainsi que l'ensemble des modules qui lui sont joints, il permet donc une configuration modulaire de l'API.

Afin de réaliser la disposition des différents modules, allez sur la partie « navigateur de projet » puis allez dans votre PLC et enfin cliquez sur « Configuration des appareils ».

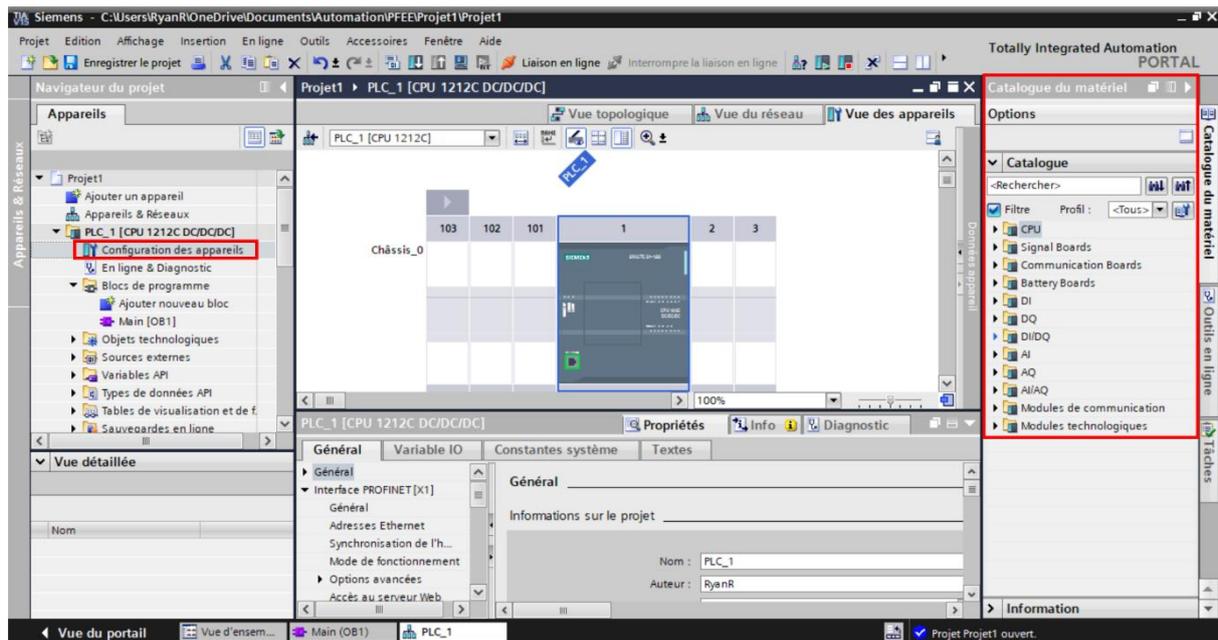


Figure II. 7 : disposition de la CPU sur le châssis.

Le rectangle en rouge à droite, désigne le catalogue du matériel qu'on peut disposer sur le châssis. On y trouve l'ensemble des CPU disponible, les modules entrées/sortie analogique et numériques, les modules de communication...

On continue sur l'exemple précédent de la CPU 1212C DC/DC/DC, à laquelle on décide d'ajouter un module de 8 entrées/8 sorties digitales.

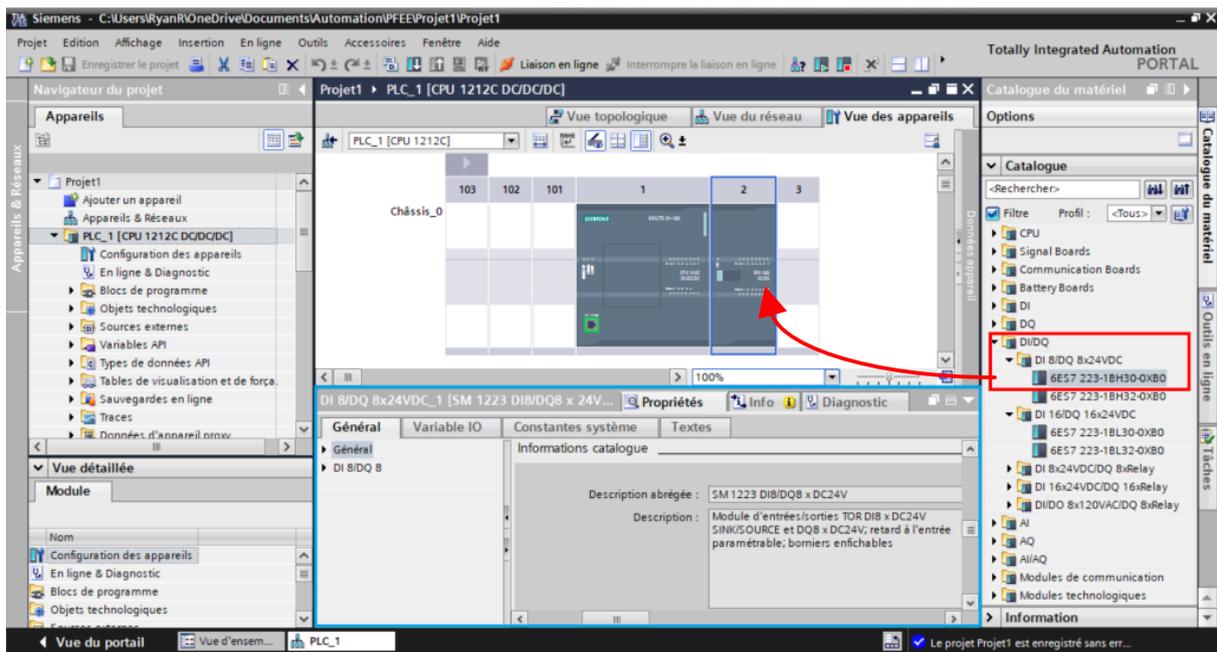


Figure II. 8 : ajout des modules d'entrées et de sorties.

Sur le châssis, la CPU est toujours à l'emplacement « 1 ». Le module entrées/sorties de références 6ES7 223-1BH30-0XB0 prendra automatiquement l'emplacement 2.

La partie en bas de la fenêtre centrale (encadré en bleu), nous donne les propriétés de l'élément sélectionné, dans le cas de la figure précédente l'élément sélectionné est le module d'entrées/sorties numériques.

### II.3.5 L'adressage des entrées/sorties

Avant d'entamer la programmation, il faut d'abord définir les entrées et les sorties du processus qu'on souhaite automatiser. Pour ce faire il est important de connaître la forme de l'adressage des E/S sur TIA Portal.



Figure II. 9: Adressage des entrées et des sorties sur TIA Portal.

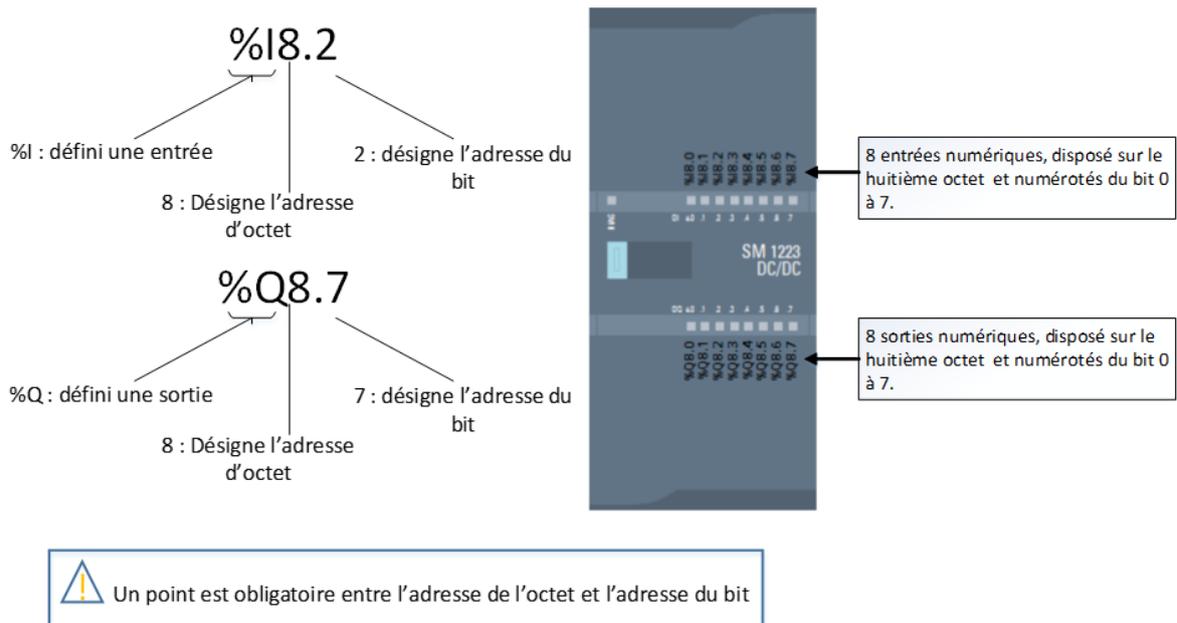


Figure II. 10: disposition et adressage des entrées/sorties.

### II.3.6 Déclaration des variables

Pour faire la déclaration des variables d'un programme sur TIA Portal, dans la partie « navigateur de projet » aller sur votre PLC puis dans « variables API » et enfin cliquez sur « Tables de variables standards ».

La table des variables standards s'ouvre dans la partie centrale de TIA Portal, et il vous sera alors possible de commencer la déclaration des variables de votre programme.

Lors de la déclaration de variables, vous devrez donner un nom distinct a chaque variable, définir son types de données (booléens, integer, word, double word... ) et lui attribuer une adresse propre parmi celles disponible.

Les adresses disponibles sont celles qui ont une existence physique et donc ce sont les adresses des modules d'entrées sorties disponible sur le chassis. Il est cependant possible d'avoir des entrées/sorties intégré dans la CPU comme c'est le cas pour la CPU 1212C DC/DC/DC, ces dernieres peuvent elle aussi etre utilisé dans la déclaration de variables.

Vous pouvez ajouter autant de variables que d'adresses disponibles ; si les adresses disponibles sont insuffisantes pour accueillir l'ensemble des variables de votre projet, vous serez dans l'obligation d'ajouter des modules entrées/sorties supplémentaires.

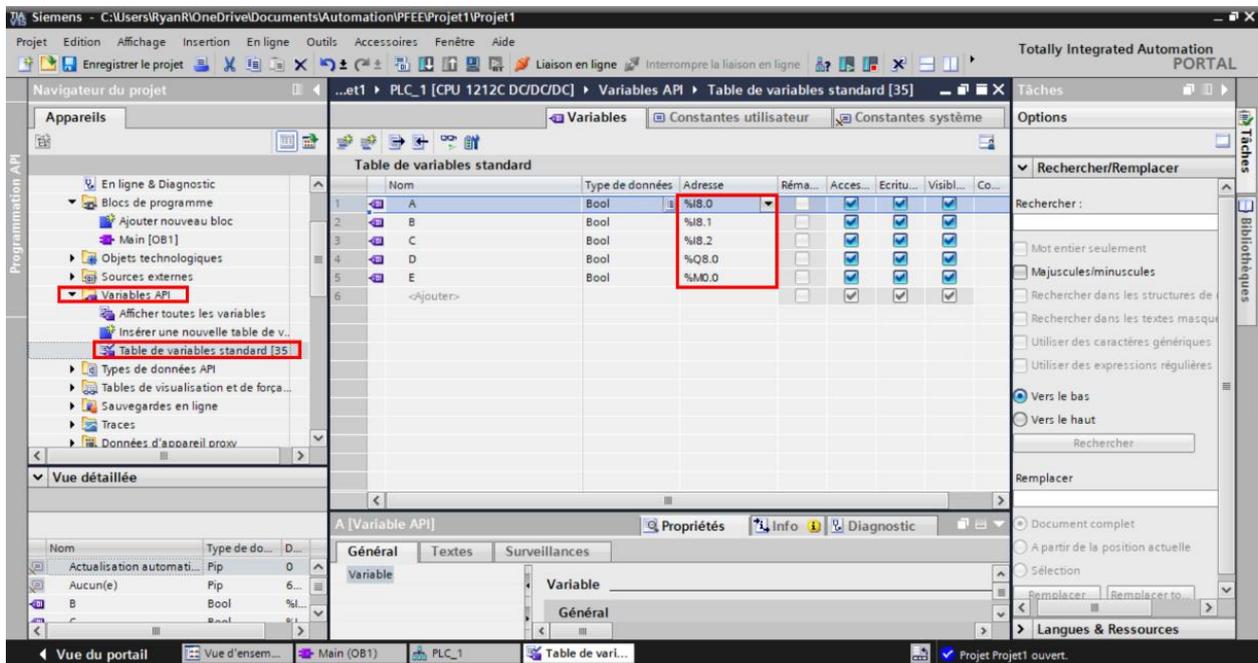


Figure II. 11 : déclaration des variables sur TIA Portal.

Les variables ainsi définies sont des variables globales et donc accessibles partout dans le programme.

### II.3.7 Les blocs de programmation

Sur TIA Portal, les séquences du programme sont écrites dans ce qu'on appelle des blocs. Un bloc d'organisation OB1 est automatiquement créé lors de l'ajout d'une CPU, aussi connu sous le nom main OB1. Ce bloc permet d'accueillir le programme principal.

Le bloc OB1 est divisé en plusieurs réseaux. Sur chaque réseau est représentée une équation du programme principal écrite dans un des langages de programmation des automates (ladder, list ...). Le bloc OB1 est traité de manière cyclique.

Afin de mieux structurer un programme, et d'éviter la réécriture des tâches répétitives, des blocs supplémentaires peuvent être créés, et appelés dans le bloc main.

### II.3.7.1 Le bloc d'organisation OB1

Afin d'accéder au bloc OB1, dans le « Navigateur de projet » allez dans votre PLC puis dans « blocs de programmation » et cliquez sur « main [OB1] ».

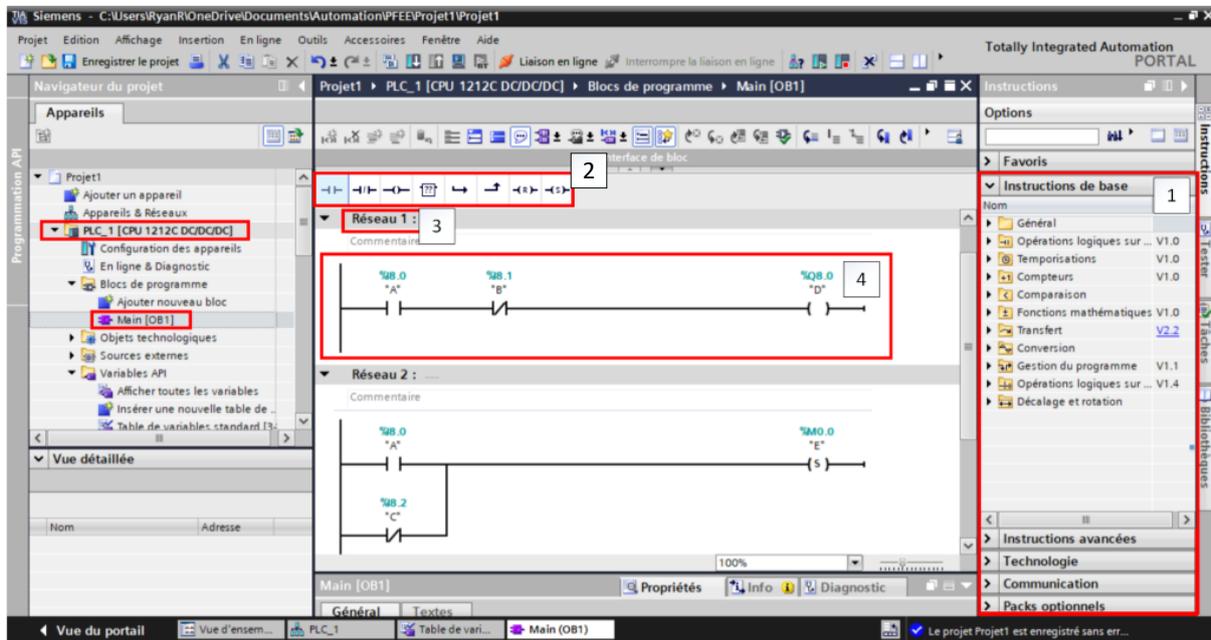


Figure II. 12 : Le bloc main (OB1).

N°	Désignation
1	L'ensemble des instructions disponible sur TIA Portal.
2	Barre des instructions de bases, on y retrouve:  : Contact a ouverture.  : Contact a fermeture .  : Affectation ou Bobine de sortie.  : Bloc vide  : Ouvrir branche pour une opération en "OU"  : fermée branche  : Set / Mise a un de la variable.  : Reset / Mise a zero de la variable.
3	Le premier réseau.
4	Equation Ladder du premier réseau.

Tableau II. 3 : Tableau des éléments accessible dans les blocs de programme.

### II.3.7.2 Les blocs supplémentaires

Afin d'ouvrir un bloc supplémentaire, allez dans « navigateur du projet » et sélectionnez votre PLC, ensuite allez dans « blocs de programme », puis cliquez sur « ajouter nouveau bloc ». Une fenêtre « ajouter nouveau bloc » s'ouvrira et vous donnera accès aux blocs complémentaires disponibles sur TIA Portal. Vous devrez en sélectionner un des quatre disponibles.

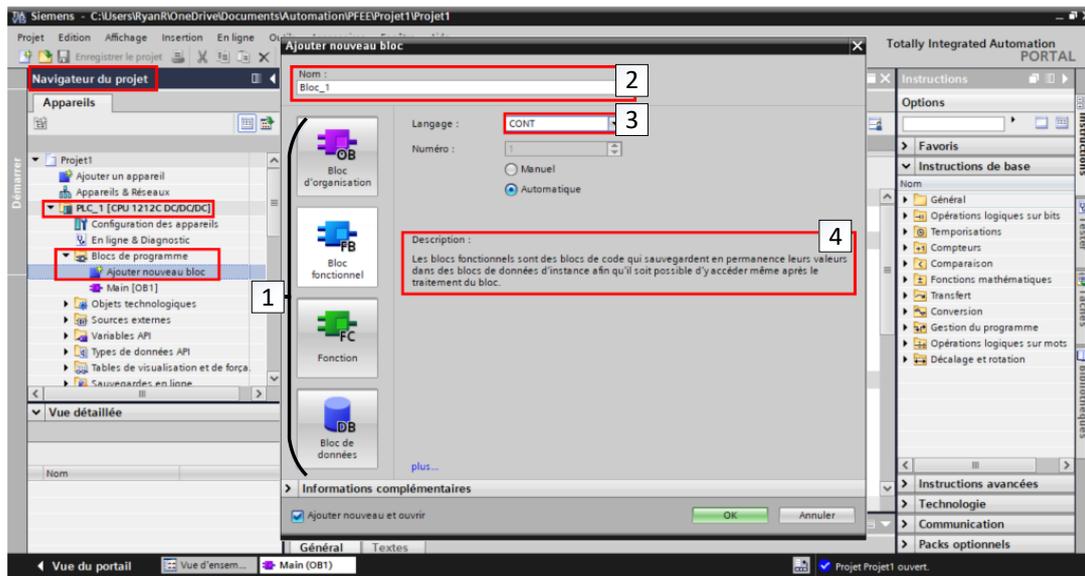


Figure II. 13: ajout d'un nouveau bloc.

N°	Désignation
1	Les blocs complémentaires disponibles sur TIA Portal: blocs d'organisations (OB). blocs fonctionnels (FB). Les fonctions (FC). Blocs de données (DB). dans le cas de l'exemple le bloc sélectionné est le bloc fonctionnel (FB)
2	Nom du bloc.
3	Choix du langage de programmation parmi les trois disponibles : LOG/CONT/SCL
4	Description du bloc sélectionné.

Tableau II. 4 : Eléments disponibles sur la fenêtre « ajouter un nouveau bloc ».

### II.3.7.2.1 Les blocs fonctionnels (FB)

Les blocs fonctionnels sont des blocs de code qui mémorisent durablement leurs paramètres d'entrée, de sortie et d'entrée/sortie dans des blocs de données d'instance afin qu'il soit possible d'y accéder même après le traitement de blocs. C'est pourquoi ils sont également appelés "Blocs avec mémoire".

Les blocs fonctionnels peuvent aussi travailler avec des variables temporaires. Cependant, les variables temporaires ne sont pas enregistrées dans la DB d'instance mais disponibles uniquement tout le temps d'un cycle.

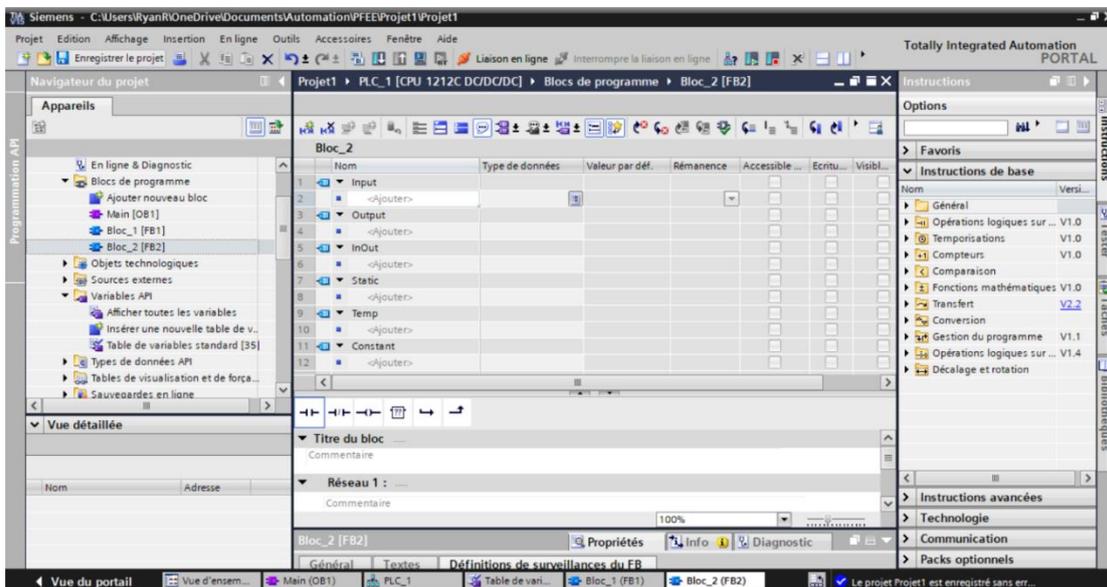


Figure II. 14 : bloc fonctionnel.

### II.3.7.2.2 Les fonctions (FC)

Les fonctions sont des blocs de code sans mémoire. Elles n'ont pas de mémoire de données dans laquelle il est possible d'enregistrer les valeurs de paramètres de bloc. C'est pourquoi des paramètres effectifs doivent être fournis à tous les paramètres formels lors de l'appel d'une fonction. Pour enregistrer les données durablement, les fonctions disposent de blocs de données globaux.

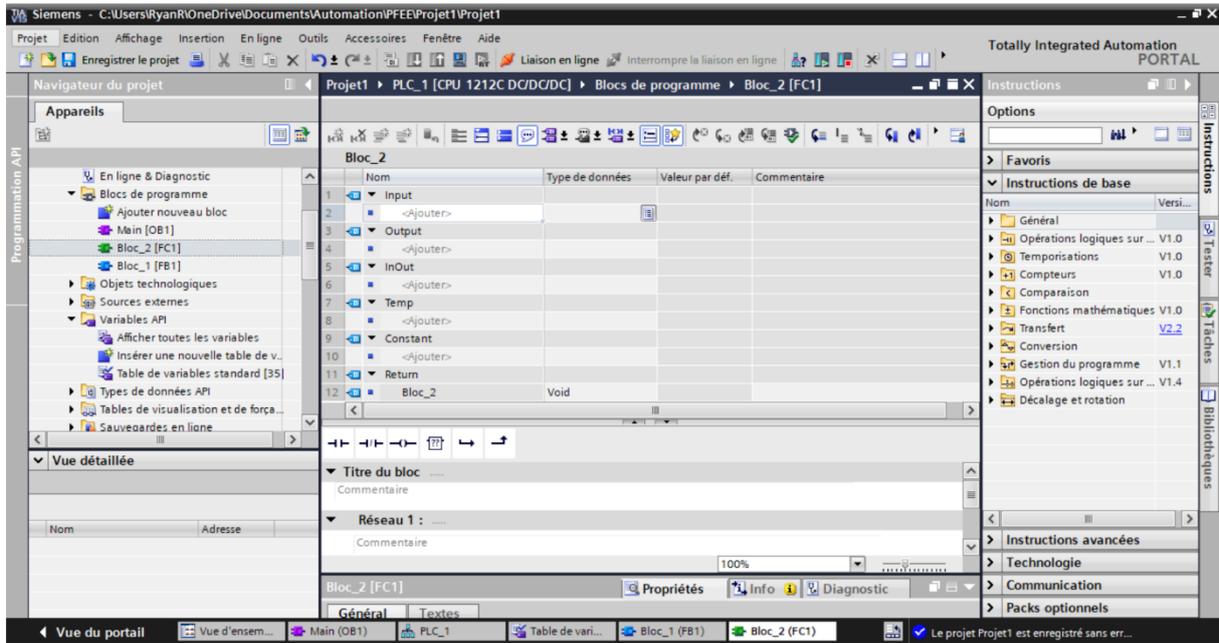


Figure II. 15 : Les fonctions.

### II.3.7.2.3 Les blocs de données (DB)

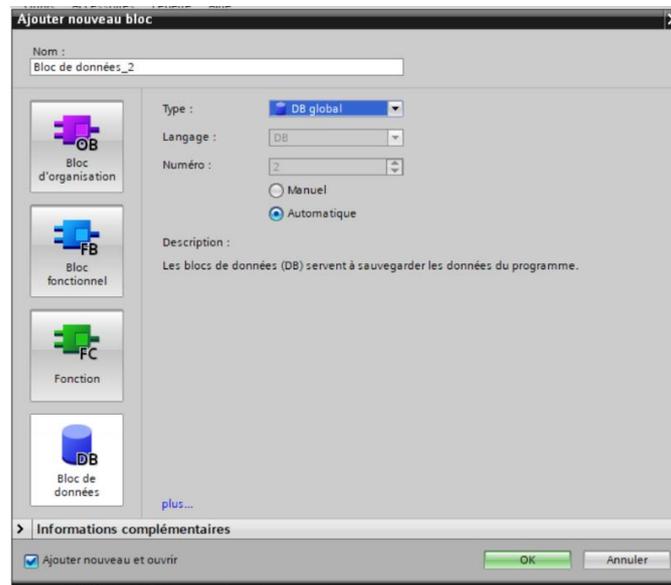


Figure II. 16 : ajout d'un bloc de données global.

## II. 3.8 Les instructions de base

Dans cette partie on énumèrera quelques instructions de base disponible sur TIA Portal.

### II.3.8.1 Operations logiques sur des bits

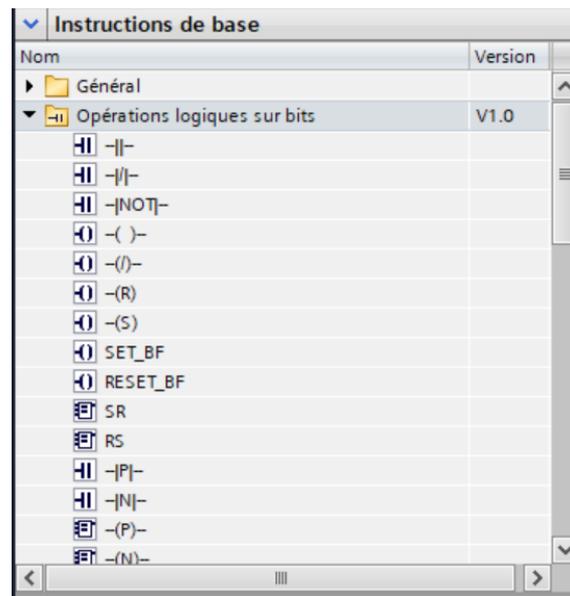


Figure II. 17 : Les instructions de base.

### II.3.8.2 Opérateurs de comparaison

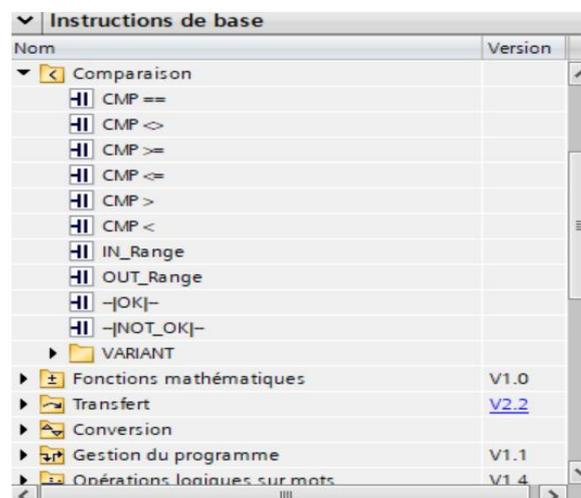


Figure II. 18 : Les opérateurs de comparaison.

### II.3.8.3 Les compteurs

▼ +1 Compteurs	V1.0
■ CTU	V1.0
■ CTD	V1.0
■ CTUD	V1.0

Figure II. 19 : Les compteurs.

### II.3.8.4 Les éléments de temporisation

Instructions de base	
Nom	Version
▶ Général	
▶ Opérations logiques sur bits	V1.0
▼ Ⓢ Temporisations	V1.0
■ TP	V1.0
■ TON	V1.0
■ TOF	V1.0
■ TONR	V1.0
Ⓜ -(TP)-	
Ⓜ -(TON)-	
Ⓜ -(TOF)-	
Ⓜ -(TONR)-	
Ⓜ -(RT)-	
Ⓜ -(PT)-	
▶ +1 Compteurs	V1.0
▶ < Comparaison	
▶ ± Fonctions mathématiques	V1.0
▶ Transfert	V2.2

Figure II. 20 : Les temporisateurs.

**Remarque :**

- Les compteurs, les temporisateurs et les blocs fonctionnels nécessitent l'utilisation de blocs de données (DB)
- Pour plus d'informations sur les fonctionnalités d'un élément, sélectionnez ce dernier puis cliquez sur la touche F1 de votre clavier. (Ouverture du HELP de TIA Portal)

## II.4 La partie simulation (PLCSIM)

### II.4.1 L'adresse IP de la CPU

Dans un même programme, chaque CPU possède une adresse IP qui lui est propre. La connaissance de cette adresse est obligatoire lors du chargement et de la connexion à l'automate réel.

Pour connaître l'adresse IP d'une CPU et pouvoir la modifier, allez dans « navigateur du projet » ; puis dans votre PLC et cliquez sur « Configuration des appareils » puis sur le châssis sélectionnez la CPU ; en bas de la fenêtre centrale vous trouverez les propriétés de votre CPU allez dans « Général » puis dans « interface PROFINET », enfin cliquez sur « adresse Ethernet », l'adresse IP ainsi que le masque de sous réseau apparaitront, vous devez changer l'adresse IP par celle de l'automate réel en votre possession.

Les étapes citées précédemment sont résumées dans la figure suivante :

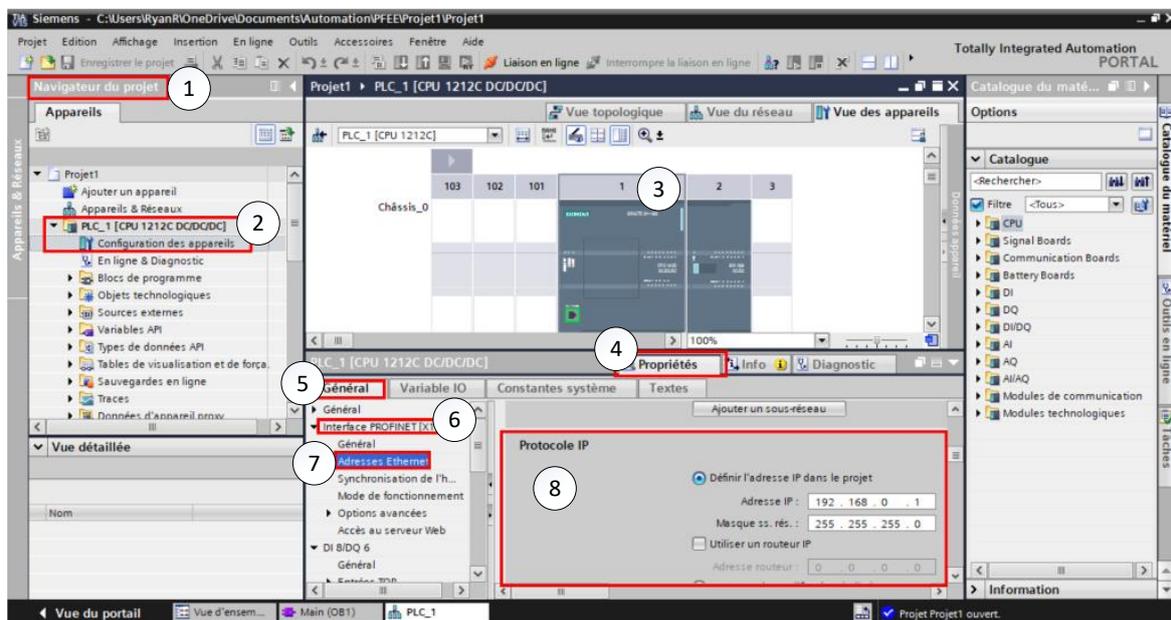


Figure II. 21 : Adresse IP de la CPU.



Une fois l'appareil cible détecté, sélectionnez-le, puis cliquez sur « charger ».

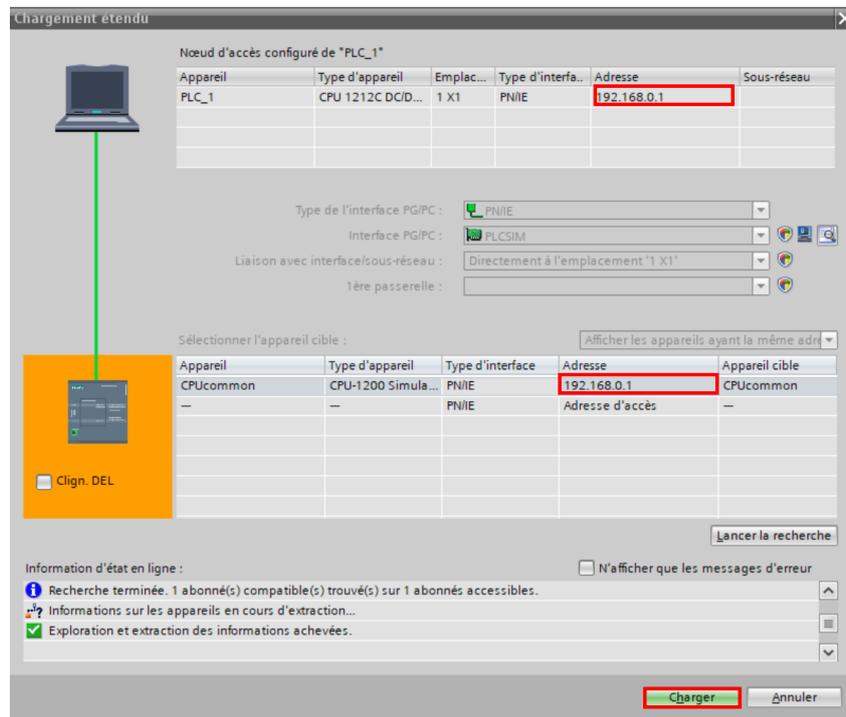


Figure II. 24 : configuration de la connexion entre la CPU et l'appareil cible.

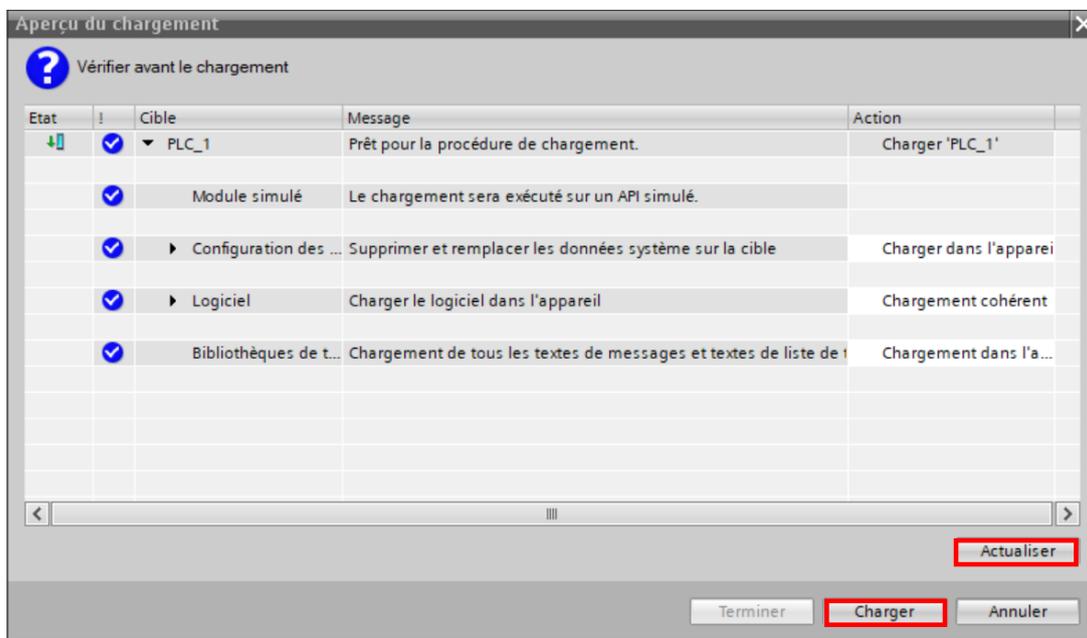


Figure II. 25 : chargement du programme dans l'API de simulation.

Enfin cliquez sur « terminer ».

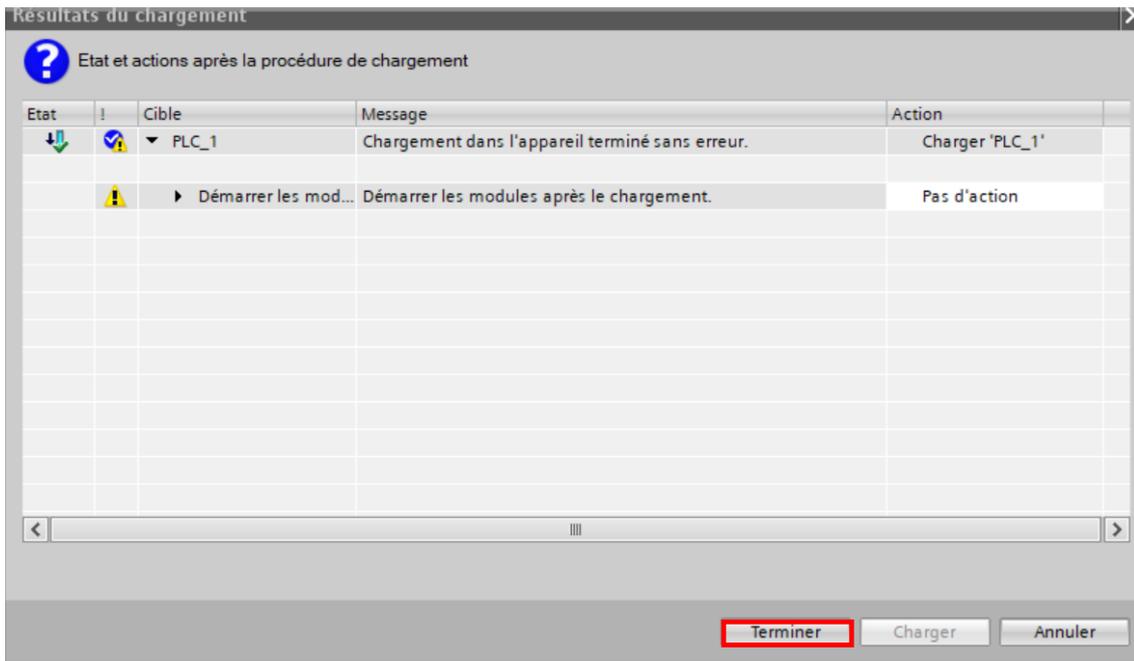


Figure II. 26 : démarrage de la simulation.

**Remarque :**

- Cliquez sur  afin d'avoir accès à l'intégralité des fonctionnalités de PLCSIM.

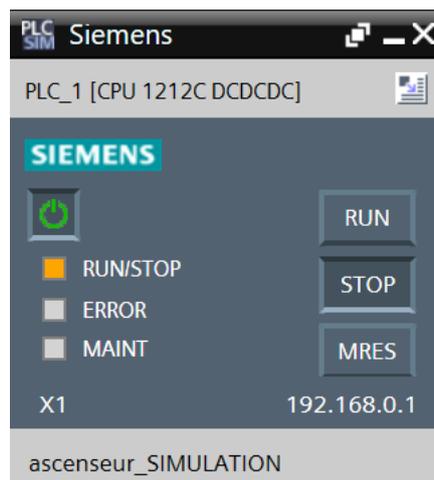


Figure II. 27 : PLCSIM.

Sur PLCSIM, cliquez sur « nouveau projet » une fenêtre créer un projet s’ouvrira, vous devrez alors donner un nom à votre projet et lui définir un chemin, puis cliquez sur « créer ».

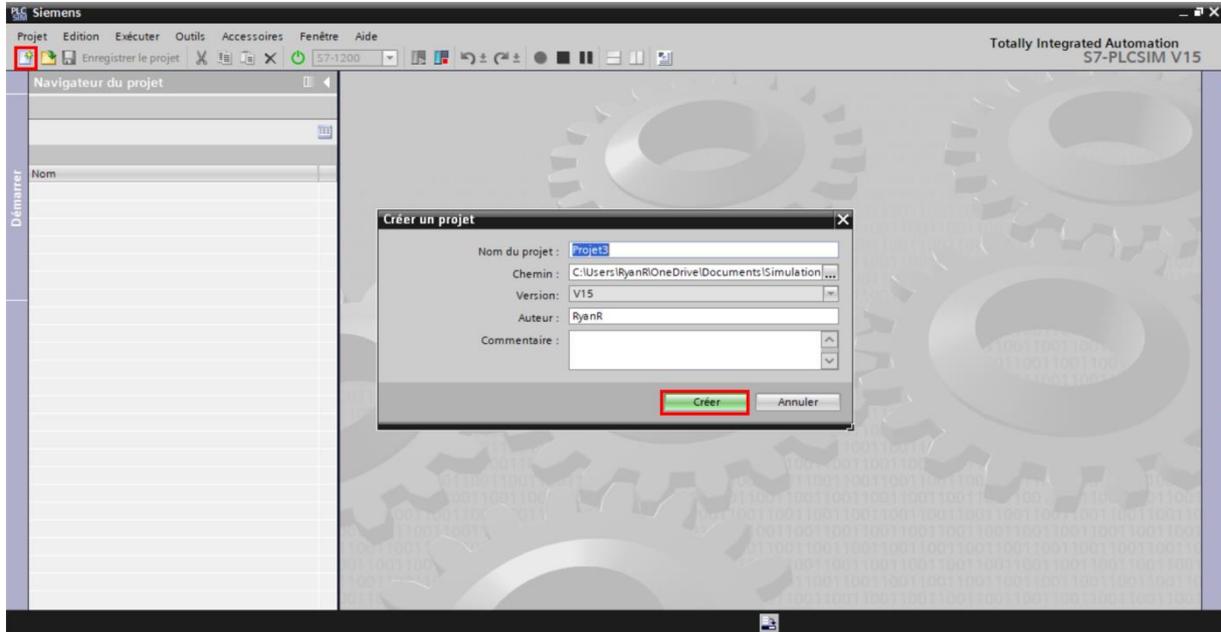


Figure II. 28 : Création d’un projet sur PLCSIM.

Une fois le projet ouvert, allez dans « navigateur du projet » puis dans votre PLC et cliquez sur « tables SIM ». Une fois à ce niveau, vous aurez le choix entre créer une nouvelle table en cliquant sur « ajouter une nouvelle table SIM » ou bien d’utiliser la table SIM par défaut en cliquant sur « table SIM\_1 ».



Figure II. 29 : table des variables.

La table s'ouvrira dans la section centrale de PLCSIM et vous pourrez alors commencer à ajouter les variables du projet (les mêmes variables que vous aviez défini sur TIA Portal).

**Remarque :**

- Vous pouvez cliquer sur  pour que l'ensemble des variables de votre projet soient ajouté de façon automatique.

	Nom	Adresse	Format d'affichage	Valeur visualisée/de forçage	Bits	Forçage par lot		Co...
	*A*:P	%I8.0:P	Bool	FALSE		<input type="checkbox"/> FALSE	<input type="checkbox"/>	
	*B*:P	%I8.1:P	Bool	FALSE		<input type="checkbox"/> FALSE	<input type="checkbox"/>	
	*C*:P	%I8.2:P	Bool	FALSE		<input type="checkbox"/> FALSE	<input type="checkbox"/>	
	*D*	%Q8.0	Bool	FALSE		<input type="checkbox"/> FALSE	<input type="checkbox"/>	
	*E*	%M0.0	Bool	TRUE		<input checked="" type="checkbox"/> FALSE	<input type="checkbox"/>	

Figure II. 30 : Variables du projet.

Retournez sur TIA Portal, et cliquez sur  puis sur  pour démarrer la simulation.

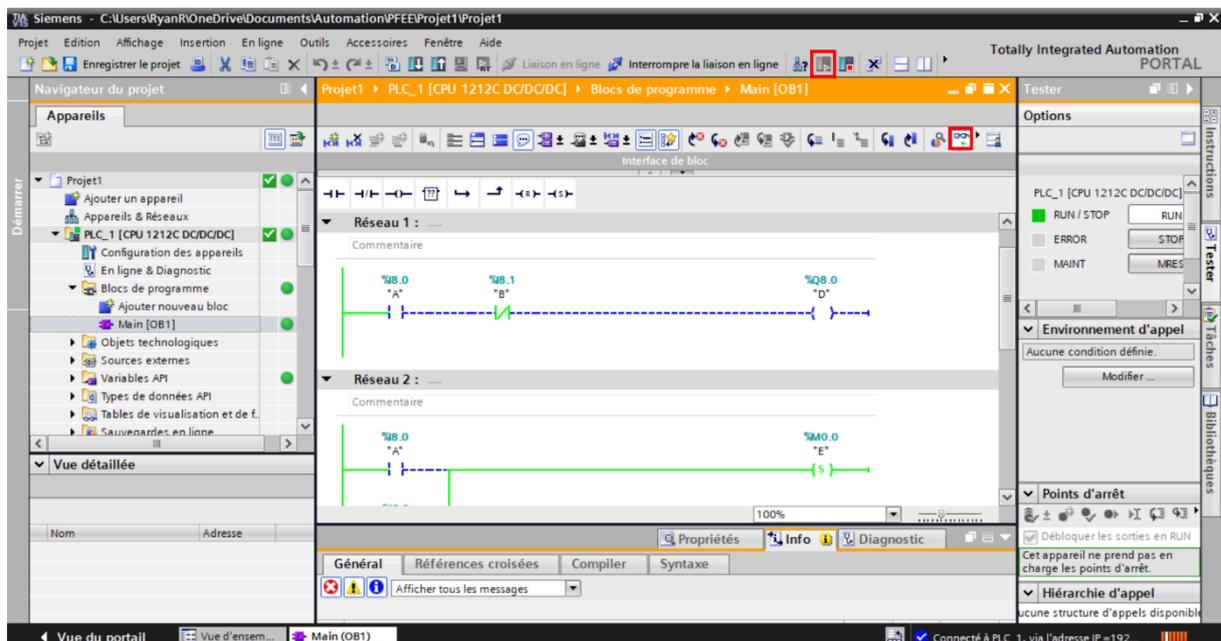


Figure II. 31 : Liaison en ligne.

Pour changer l'état des variables d'entrée, retournez sur PLCSIM, puis dans votre table SIM allez dans « bits » et cocher le bit que vous voulez mettre à un.

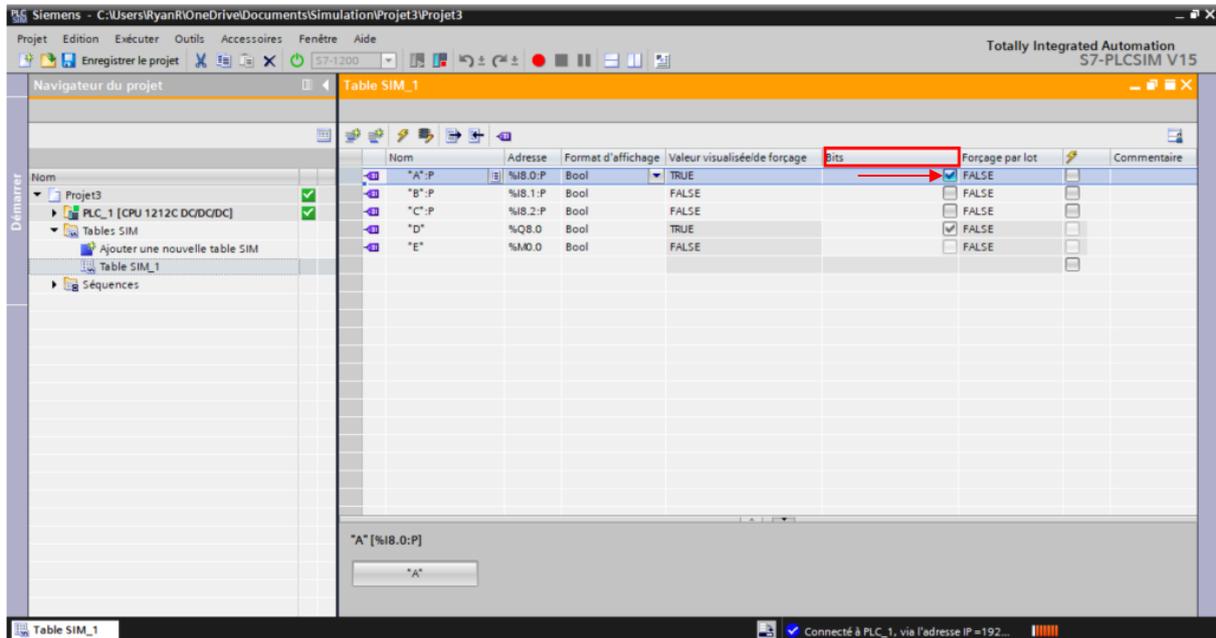


Figure II. 32 : mise à 1 d'un bit sur PLCSIM.

Afin de mieux visualiser l'état des sorties retournez dans TIA Portal.

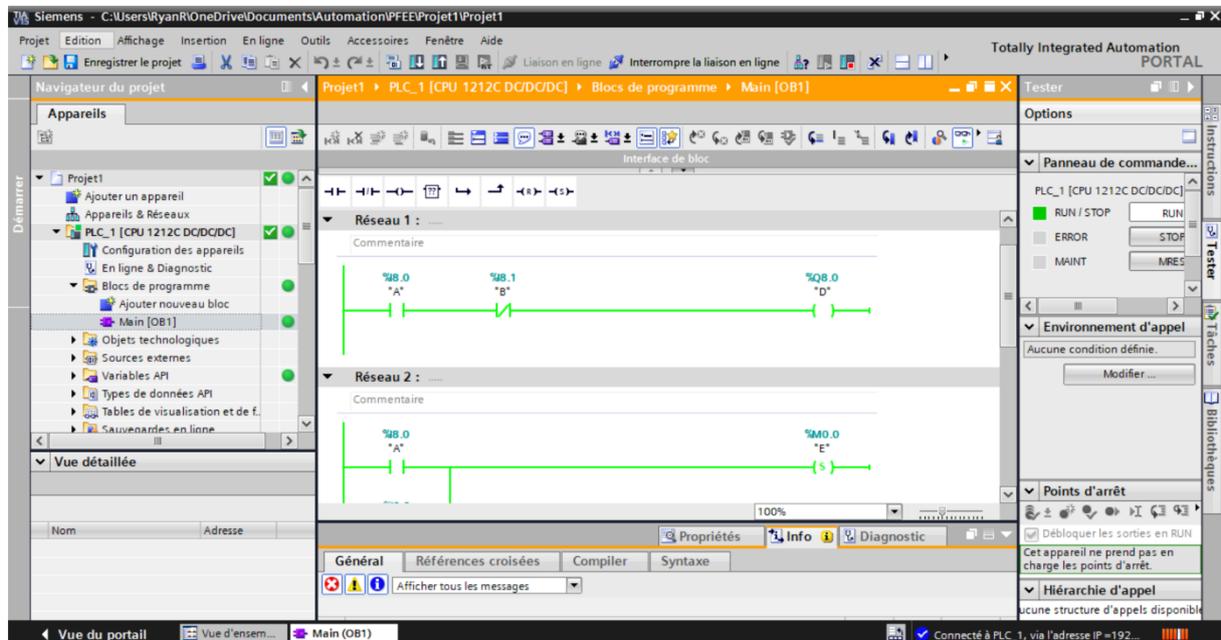


Figure II. 33 : visualisation des états des variables du projet.

## II.5 Partie supervision sous WINCC

Afin de réaliser la supervision d'un projet d'automatisme sur TIA Portal, il faut créer une interface Homme Machine « HMI ». Pour ce faire, allez dans « Navigateur du projet » puis cliquez sur « ajouter un appareil » ; enfin sélectionnez « HMI » et choisissez l'interface que vous souhaitez intégrer à votre projet.

A titre d'exemple, on choisit l'interface « TP2200 COMFORT PRO ».

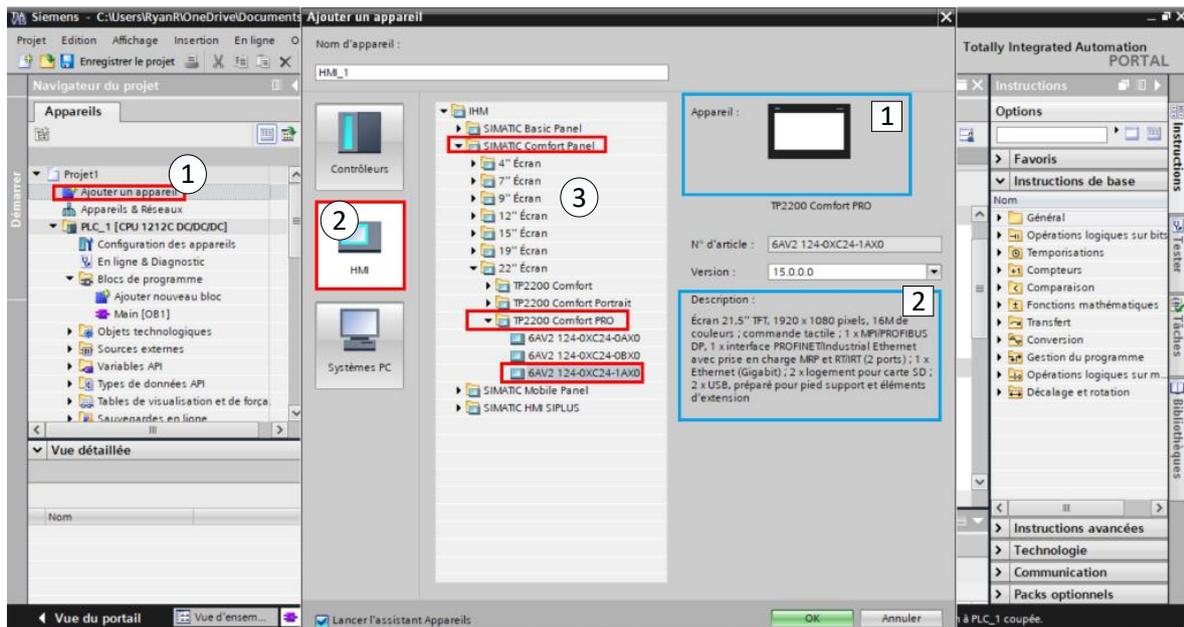


Figure II. 34 : choix de l'interface Homme-Machine.

Une fois l'interface choisie, vous devez établir la liaison entre votre IHM et la CPU, pour cela allez dans « parcourir » et sélectionnez votre CPU.

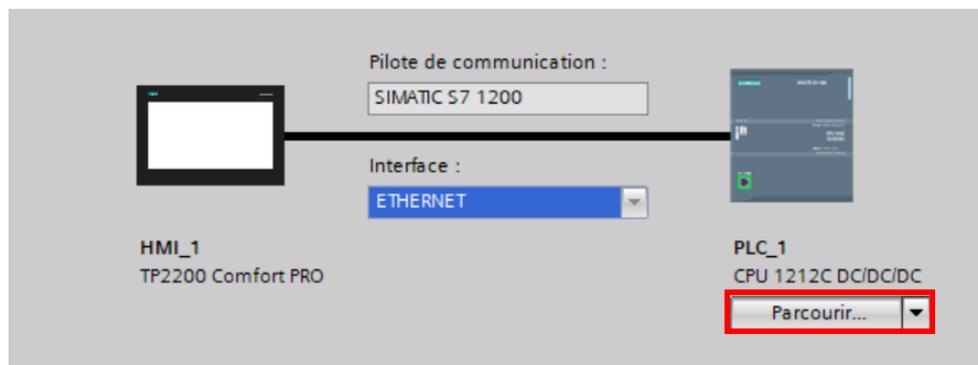


Figure II. 35: configuration de la connexion entre l'API et l'interface.

## II.5.1 la vue de l'interface Homme-Machine

Une fois la configuration de l'HMI terminée, la vue racine de votre interface s'ouvrira de manière automatique sur la partie centrale de TIA Portal comme illustré dans la figure ci-dessous.

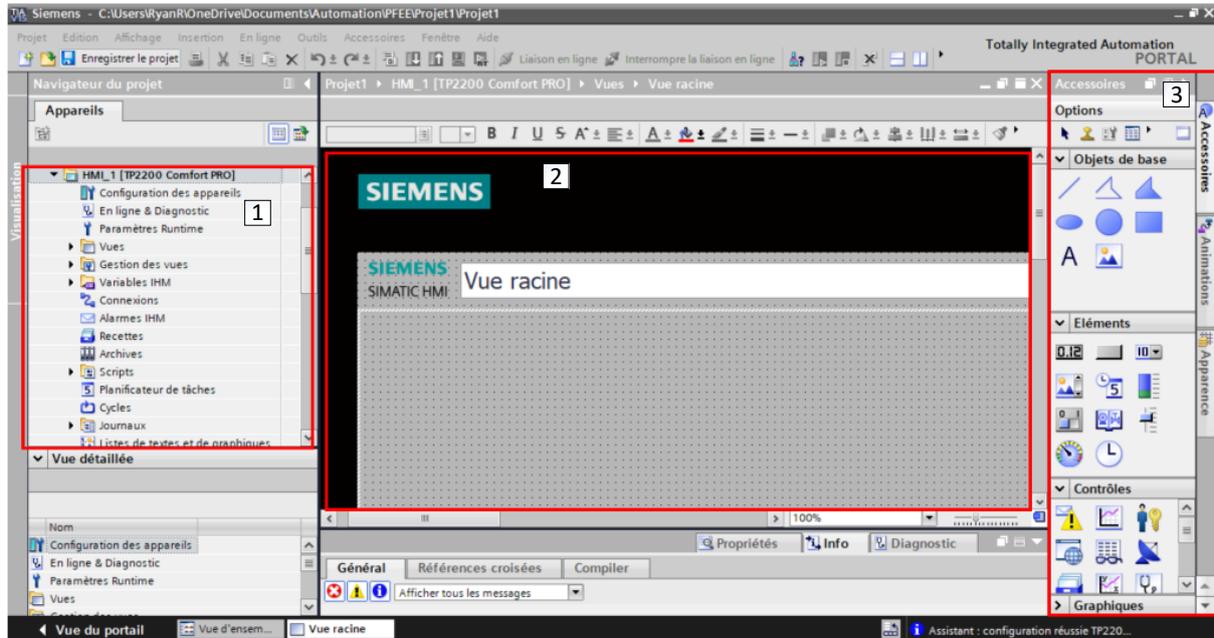


Figure II. 36: vue racine de l'interface Homme-Machine.

N°	Désignation
1	La vue globale de l'ensemble des éléments de l'HMI on y retrouve: Les vues; La gestion des vues; Les variables HMI ...
2	La vue racine ou vue principale de l'interface homme machine.
3	L'ensemble des accessoires utilisé pour créer des vues de supervision. on y retrouve des objets de base tel que des rectangles, des cercles qui seront utilisés pour la visualisation des états des sorties, mais aussi des éléments de contrôle pour le suivi du processus et sa surveillance (alarmes...)

Tableau II. 5: Tableau explicatif des éléments de la figure III.36.

## II.5.2 Configuration des variables HMI

Pour accéder à la table des mnémoniques, allez dans « variables HMI » et sélectionnez « table des variables standard ».

Une fois dans la table des variables, vous devez importer les variables de votre projet ; pour ce faire cliquez sur « ajouter » et donnez un nom à votre variable HMI, puis importer la variable API que vous souhaitez lui associer en allant dans « variable API ».

### Remarque :

- la variable HMI et la variable API lui correspondant peuvent avoir le même nom.

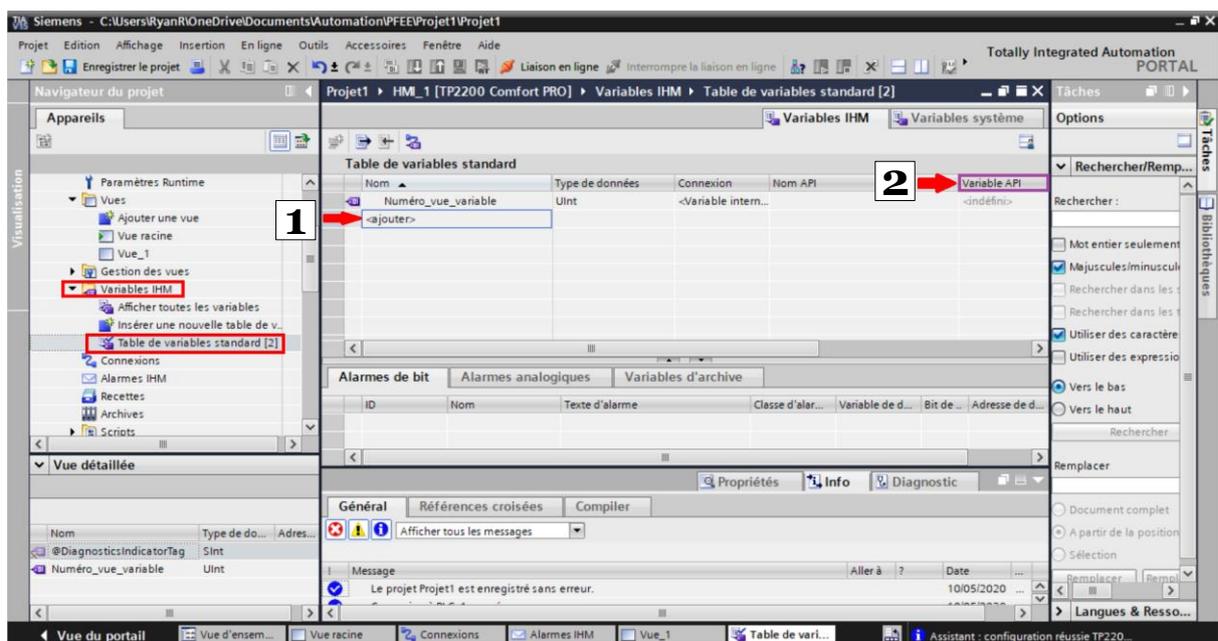


Figure II. 37: Table des variables HMI.

### Remarque :

- Il faut faire attention au cycle d'acquisition. Un cycle d'acquisition trop grand engendre un fonctionnement non désiré.

Table de variables standard									
Nom	Type de données	Connexion	Nom API	Variable API	Adresse	Mode d'accès	Cycle d'acquisit...	Archivé	Comm...
Numéro_vue_variable	UInt	<Variable intern...		<indéfini>			1 s	<input type="checkbox"/>	
A	Bool	HMI_Liaison_1	PLC_1	A		<accès symbolique>	100 ms	<input type="checkbox"/>	
B	Bool	HMI_Liaison_1	PLC_1	B		<accès symbolique>	100 ms	<input type="checkbox"/>	
C	Bool	HMI_Liaison_1	PLC_1	C		<accès symbolique>	100 ms	<input type="checkbox"/>	
D	Bool	HMI_Liaison_1	PLC_1	D		<accès symbolique>	100 ms	<input type="checkbox"/>	
E	Bool	HMI_Liaison_1	PLC_1	E		<accès symbolique>	100 ms	<input type="checkbox"/>	

Figure II. 38: cycle d'acquisition des variables HMI.

### II.5.3 Configuration d'un voyant

Allez dans une vue de votre choix, puis dans « accessoires » choisissez le cercle que vous trouverez dans « objets de base » et faites le glisser dans votre vue.

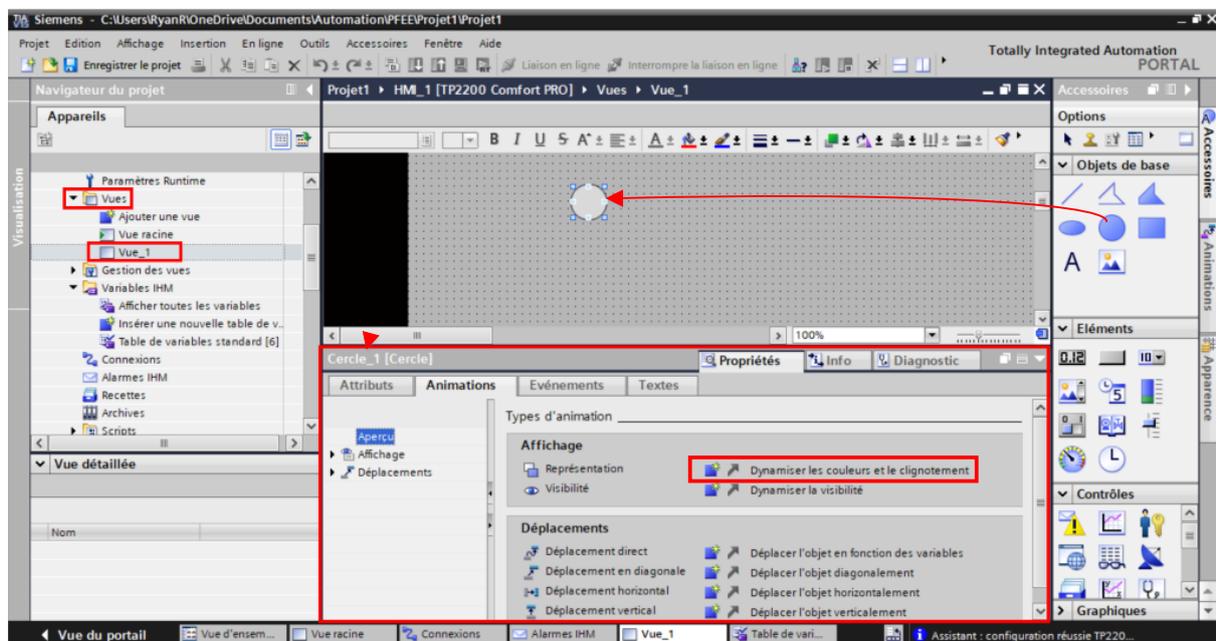


Figure II. 39 : ajout de l'objet « cercle » dans la vue HMI.

En bas de votre vue, vous trouverez les propriétés de votre cercle ; allez dans « animations » puis sélectionnez « dynamiser les couleurs et clignotement ».

Enfin dans représentation ; choisissez la variable que vous souhaitez dynamiser.

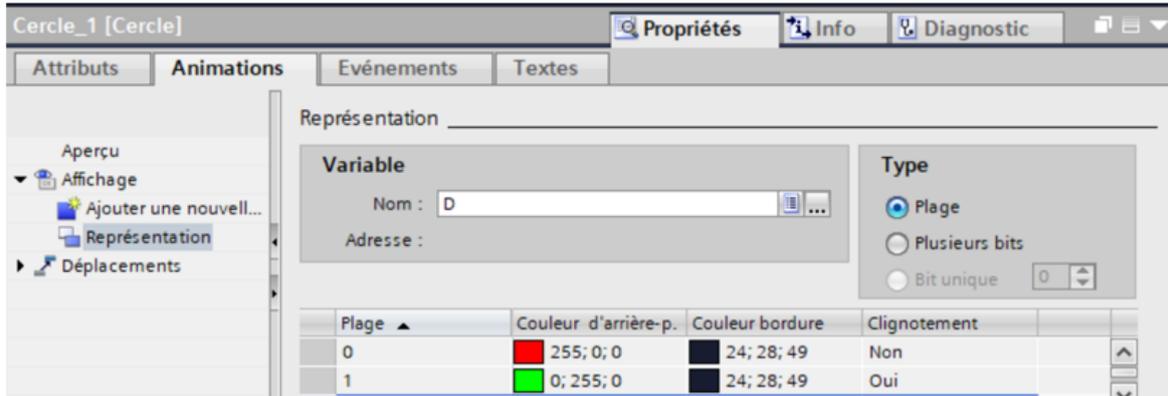


Figure II. 40: Configuration d'une animation « représentation » pour l'objet cercle.

## II.5.4 Configuration d'un bouton

Avant d'entamer la configuration d'un bouton, nous allons créer une variable de type entrée/sortie (%M) « A\_man ». Cette dernière va nous servir à l'activation manuelle de la sortie D. C'est-à-dire l'activation à partir de la vue de supervision.

Afin que cette variable puisse permettre l'activation de la sortie D ; il faut qu'elle soit ajoutée à notre programme principale dans le réseau consacré à la sortie D.

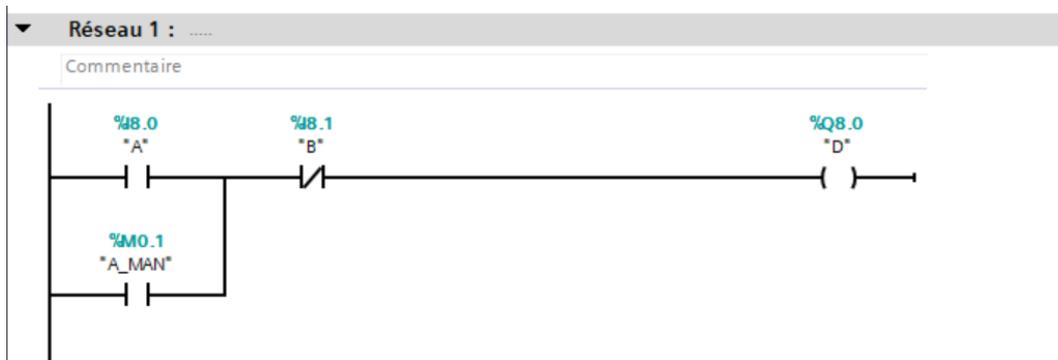


Figure II. 41: ajout du memento « A\_Man ».

Allez dans «accessoires», puis dans « éléments » et faites glisser le bouton dans votre vue. Votre bouton est sélectionné, vous verrez donc apparaître en bas de votre vue, les propriétés de votre bouton, allez dans «Evènements» puis sélectionnez «Clic».

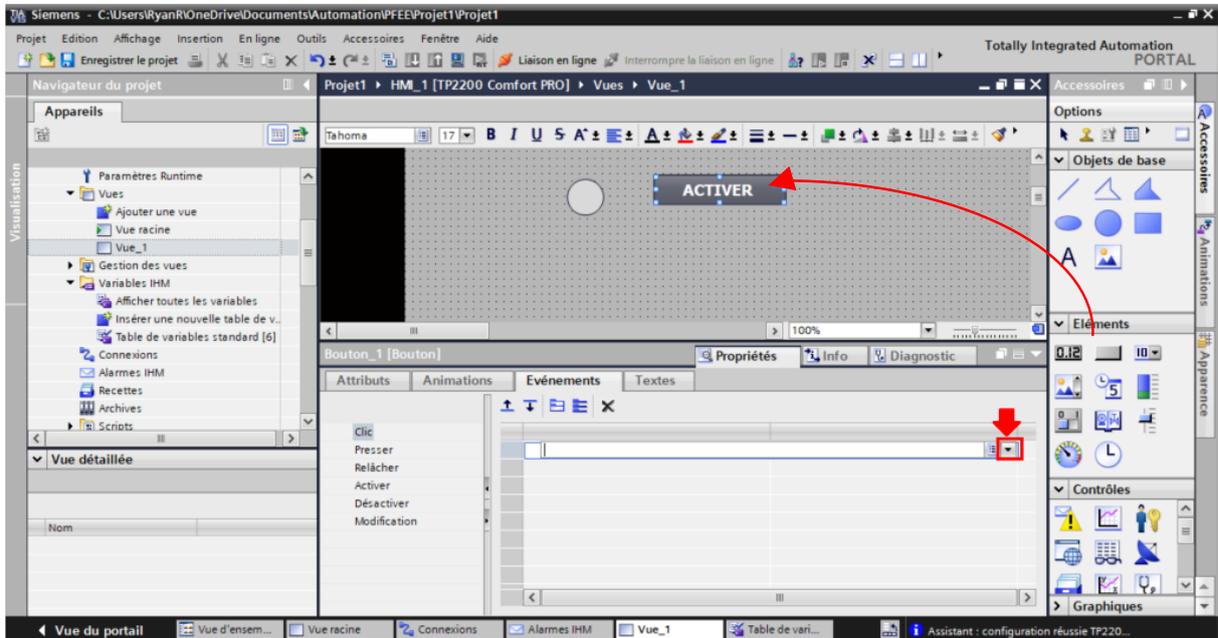


Figure II. 42: ajout de l'objet « bouton » dans la vue HMI.

TIA Portal propose un large choix de fonctions préprogrammées que vous pouvez utiliser pour configurer votre bouton ; sélectionnez celle qui vous intéresse.

Pour notre exemple, on choisit la fonction « inverser bit ».

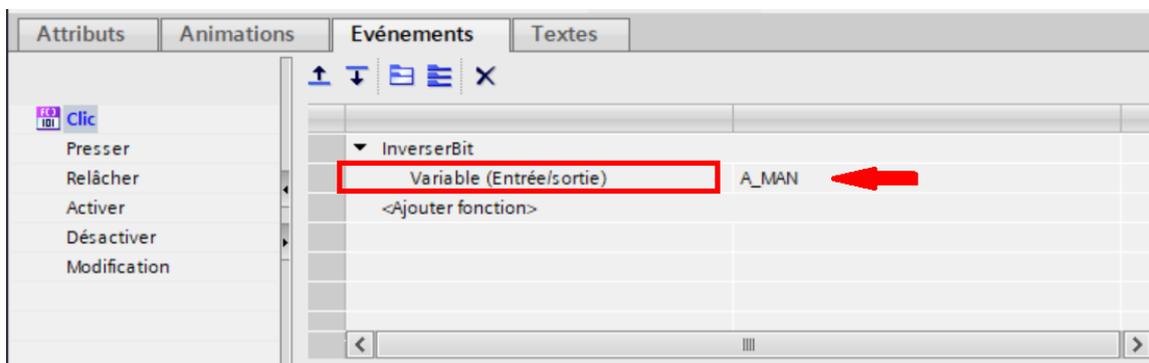


Figure II. 43: Configuration d'un évènement pour l'objet «bouton ».

## II.5.5 Démarrage de la simulation de l'interface

Avant de démarrer la simulation de votre l'interface, vous devez compiler votre programme main (OB1) et lancer PLCSIM. Retournez ensuite à votre interface et cliquez sur compiler puis sur démarrer la simulation.

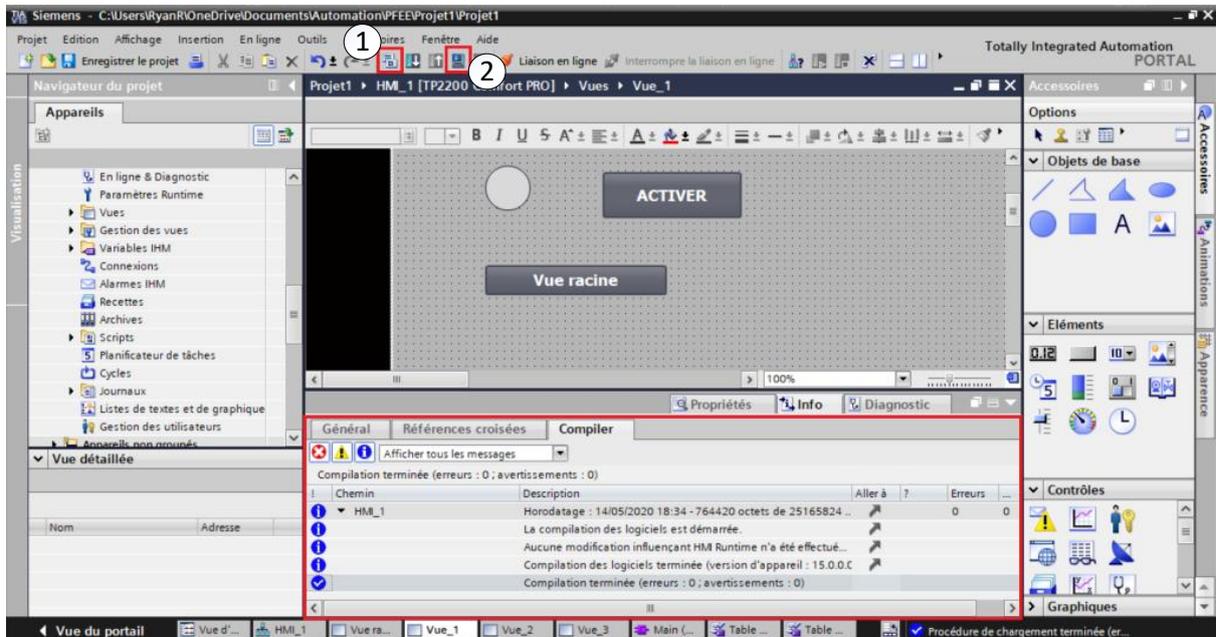


Figure II. 44: Compilation de l'interface Homme-machine.

Une fois SIMATIC WinCC lancé ; cliquez sur « RUN » dans PLCSIM pour démarrer votre simulation.



Figure II. 45: Lancement de WINCC.

Dans l'exemple traité, un clic sur le bouton « ACTIVER » met à 1 la sortie D et donc le voyant clignote en vert.

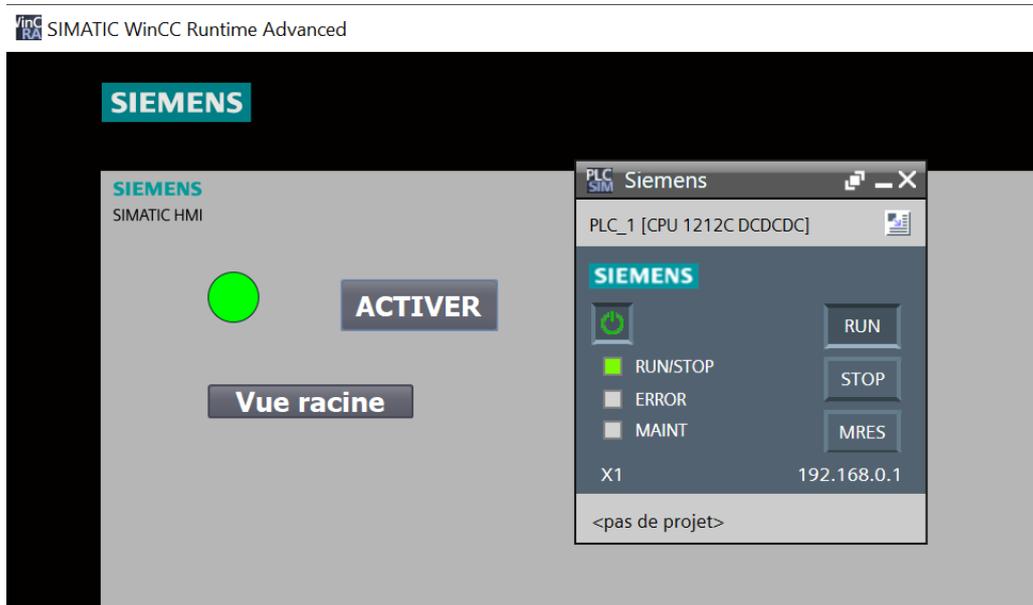


Figure II. 46: Supervision sous WINCC.

## II.6 Conclusion

Ce chapitre regroupe les étapes essentielles à la création d'un projet d'automatisation sur TIA Portal. On y retrouve la configuration matérielle des automates et des interfaces E/S, la programmation, la simulation sur PLCSIM ainsi que la création et configuration d'interfaces Homme-Machine pour la supervision du système.

# Chapitre 3 :

## Généralités sur les ascenseurs.

## III.1 Introduction

Quand l'homme a eu besoin de se déplacer en hauteur, il a commencé à réfléchir à un moyen de déplacement rapide minimisant ses efforts et surtout lui assurant une sécurité maximale. L'homme s'est alors inspiré des ascenseurs rudimentaires déjà présent au moyen âge et utilisés pour remonter des charges lourdes ; pour développer des ascenseurs plus performant en apportant des améliorations sur aussi bien sur le plan technique mais aussi sur le plan esthétique.

De nos jours, les ascenseurs garantissent une accessibilité optimale et une facilité d'utilisation qui font de l'ascenseur un appareil incontournable du quotidien. En effet, il permet un accès sécurisé et simplifié aux différents étages d'un immeuble. Il facilite ainsi les déplacements des individus et tout particulièrement le transport des personnes handicapés.

À travers ce chapitre, nous allons nous familiariser avec cet appareil de déplacement révolutionnaire en étudiant de plus près ses composants, les différents types ainsi que les catégories d'ascenseurs existants de nos jours et en tirer les avantages et les inconvénients de chaque catégorie. Enfin, nous verrons les critères de choix d'un ascenseur.

## III.2 Historique [6]

En 236 avant J.C., Archimède avait mis au point un treuil comportant des cordes et des poulies, la corde s'enroulait sur un tambour actionné par l'homme.

Au Moyen-âge des treuils servaient à monter des personnes et des marchandises dans des endroits isolés, tels que les monastères.

En 1835, un monte-charge à vapeur, appelé le « teagle », servait à monter les matériaux dans une usine anglaise.

En 1845, Sir William Thompson mit au point le premier élévateur hydraulique, dont on ne verra la commercialisation qu'en 1850. Il faut attendre 1852, pour voir le premier « ascenseur » sécurisé avec l'invention d'Elisha Graves Otis, qui réalisa un « parachute ».

Le 23 mars 1857, Otis inaugure le **premier ascenseur** du monde, dans un magasin de porcelaine. Le bâtiment comportait cinq étages et l'ascenseur fonctionnait grâce à une série d'arbres et de courroies entraînés par une centrale à vapeur. Sa capacité était de 450 kilogrammes à une vitesse de 0,2 mètre par seconde.

En 1868, Otis avait mis au point un ascenseur à vapeur et des dispositifs de **sécurité** si élaborés que les étages supérieurs prirent plus de valeur. La hauteur des bâtiments augmenta.

En 1878, Otis disposait de deux nouveaux produits : un ascenseur hydraulique très rapide (de 3 à 4 mètres par seconde) et un dispositif de parachute actionné par un limiteur de vitesse. L'arrêt en cas d'urgence se faisait alors progressivement.

En 1880, c'est en Allemagne cependant qu'on pensera à un **ascenseur électrique**, les moteurs électriques et transformateurs se développant pour l'industrie et la traction ferroviaires qui ont utilisé le câble et le treuil fixe pour tracter les trains à locomotive pour monter les fortes côtes.

En 1924, un ascenseur sans liftier (machiniste) fera son apparition, exigeant la mise au point d'**automatismes et de dispositifs de sécurité**. Les commandes deviennent électriques puis électroniques et se dotent de mémoire. Les grilles fixes ou articulées des portes disparaissent, les portes pleines métalliques se verrouillent automatiquement.

Dans les années 1880 à 1890 les architectes ne pouvaient pas dépasser 10 à 12 étages, limiter par la charge que les murs en briques pouvaient supporter. Mais en 1885, avec la naissance de la technique des « gratte-ciel » avec la charpente métallique, la course était lancée pour les ascenseurs Otis.

### III.3 Définition

L'ascenseur est un appareil élévateur, comportant une cabine servant au transport de marchandises ou de personne sur plusieurs niveaux. Les ascenseurs se déplacent le long de guides verticaux ou dont l'inclinaison est inférieure à 15°. [7] [8]

## **III.4 Les différents Types d'ascenseurs**

Selon leurs utilisations, les ascenseurs peuvent être de 03 types :

### **III.4.1 Monte-charge**

C'est un type particulier d'ascenseur dont les dimensions et la constitution de la cabine sont suffisamment grande pour permettre le transport de marchandises lourdes et de personnes.

### **III.4.2 Monte-charge industriel**

Les ascenseurs « monte charges industriels » sont des appareils de levage comportant une cabine ou un plateau suffisamment grand pour permettre le transport exclusive de marchandise sur plusieurs niveaux, et dont la commande ne peut se faire que de l'extérieur , le transport des personnes est donc interdit néanmoins ce type d'ascenseur est accessible aux personnes pour le chargement ou déchargement.

### **III.4.3 Ascenseur pour personne**

Destiné exclusivement au transport de personnes, ce type d'ascenseurs présente une plus grande sécurité et un meilleur confort en comparaison des types cités précédemment. Ils sont présent aussi bien dans les bâtiments à intérêt public, que dans le milieu privé où ils desservent par exemple les occupants d'une maison individuelle

On retrouve dans ce type d'ascenseurs, des ascenseurs spécialement dédiés au transport de personnes handicapées, debout ou en fauteuil roulant, avec ou sans accompagnateur

## III.5 Catégories d'ascenseurs

Selon le mode de mise en mouvement, on distingue deux catégories d'ascenseurs [7] :

- Les ascenseurs hydrauliques.
- Les ascenseurs à traction électrique aussi appelé à traction à câbles.

### III.5.1 Les ascenseurs hydrauliques

#### III.5.1.1 Description

Le déplacement de la cabine d'un ascenseur hydraulique se fait grâce à un système de vérin. La cabine est propulsée par le piston d'un vérin alimenté par de l'huile sous pression provenant d'une centrale hydraulique (Pompe).

L'huile sous pression pousse le piston hors du cylindre ce qui engendre le déplacement de la cabine vers le haut. Pour redescendre la vanne (by-pass) de la pompe s'enclenche permettant ainsi l'évacuation de l'huile du cylindre vers un réservoir fonctionnant en cycle fermé. Cette action fait diminuer la pression et donc permet la descente de la cabine.

L'inconvénient majeur des ascenseurs hydrauliques est que l'absence de contrepoids pour équilibrer la cabine engendre une importante consommation énergétique alors que les déplacements sont plus lents et plus courts en comparaison à des ascenseurs à traction. Leur déplacement maximum est de l'ordre de 18m.

L'ascenseur hydraulique peut être installé à l'intérieur d'une gaine maçonnée, ou à l'intérieur d'un **pylône métallique**, dans ce cas il s'agit d'un ascenseur hydraulique **autoportant**.

Plusieurs modèles d'ascenseurs hydrauliques sont proposés :

- à cylindre enterré ;
- à cylindre de surface ;
- télescopique.

L'ascenseur hydraulique est composé notamment : [7] [8]

- d'une cabine ;
- D'un ensemble pistons-cylindres hydrauliques placé sous la cabine de l'ascenseur.
- d'un réservoir d'huile ;
- d'un moteur électrique ;
- d'une pompe hydraulique.
- De guides.
- D'un contrôleur.

Les différents modèles permettent de tenir compte de certains critères tels que: [8]

- La place.
- La hauteur d'immeuble à desservir vu que la hauteur est limitée.
- La stabilité du sol et du sous-sol.
- Le risque de pollution par rapport au sol et plus spécifiquement aux nappes phréatiques.
- L'esthétique.

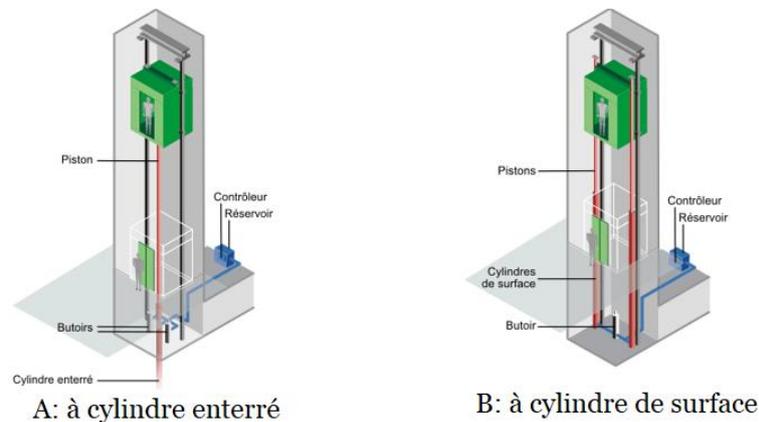


Figure III. 1 : ascenseur hydraulique.

### III.5.1.2 Les avantages de l'ascenseur hydraulique

- Implémentation facile notamment dans un immeuble existant.
- Les déplacements de la cabine sont précis et s'effectuent en douceur.

- Il est silencieux.
- Entretien relativement simple.
- Capacité de charge importante.
- Le réglage de la vitesse de déplacement est facile.
- Pas de local de machinerie.

### **III.5.1.3 Les inconvénients de l'ascenseur hydraulique**

- Consommation importante en énergie
- Consommation importante d'huile ce qui complique la sécurité incendie
- Vitesse de déplacement relativement lente ;
- Le nécessaire renforcement de la dalle de sol lors de son installation ;
- Risque de pollution des sous-sols.
- Possède une élévation limitée.

## **III.5.2 Les ascenseurs à traction à câbles**

### **III.5.2.1 Description**

Les ascenseurs à traction à câbles est le type d'ascenseur le plus répandu. De nombreux utilisateurs les empruntent quotidiennement, notamment au sein de bâtiments tertiaires. Selon la motorisation, on distingue plusieurs types d'ascenseur à traction à câbles :

- A moteur-treuil planétaire
- A moteur sans treuil ou « Gearless »
- A moteur-treuil avec une vis sans fin

Suspendus dans la gaine, la cabine et son contrepoids sont reliés par des câbles de tractions. Le déplacement de l'ensemble cabine-contrepoids se fait grâce à un moteur électrique qui actionne une poulie. Cette dernière entraîne les câbles de tractions ce qui provoque le déplacement dans des sens contraire de la cabine et du contrepoids.

Selon le mode de traction, deux variantes existent :

- **Direct** : le treuil à tambour tracte directement la cabine. Ce mode de traction garanti un encombrement réduit mais est limité aux applications avec de petites capacités de charges.
- **Mouflet** : ce mode de traction est utilisé dans le cas de grandes charges, nécessitant des vitesses de traction importante

De manière générale, les ascenseurs à traction par câbles sont tous constitués :

- Une cabine
- Un contrepoids.
- Des câbles reliant la cabine au contrepoids.
- Des guides.
- Un système de traction
- D'un système antichute

En termes d'énergie, les ascenseurs à traction à câbles sont nettement moins énergétiques que les ascenseurs hydrauliques du fait de la présence du contrepoids qui équilibre le poids de la cabine et donc réduit grandement la consommation en énergie et en puissance et ceci indépendamment du type de motorisation.

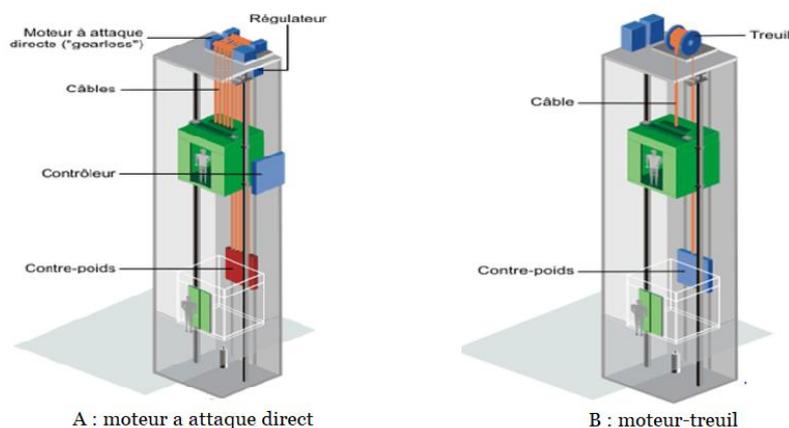


Figure III. 2 : Les différentes motorisations des moteurs à traction à câbles.

### **III.5.2.2 Avantages des ascenseurs à traction à câbles**

Comparé à des ascenseurs hydrauliques, les ascenseurs à tractions à câbles présentent de nombreux avantages, voici quelques-uns :

- Effectue des déplacements rapides et précis.
- respecte l'environnement.
- efficacité énergétique importante.
- simplifier le déplacement des utilisateurs.
- large plage de variation des vitesses.
- La limitation des consommations et des appels de puissance. [7]

### **III.5.2.3 Inconvénients des ascenseurs à traction à câbles [7]**

- un entretien régulier doit être respecté pour assurer le bon fonctionnement de l'appareil, mais surtout pour la sécurité des utilisateurs.
- la version standard des ascenseurs nécessite l'installation d'un cabanon technique en toiture.
- Nécessite de tenir compte du poids de la cabine, des câbles, du contre poids, de la structure de la salle des machines, et des équipements de la salle des machines. Le poids total repose sur la structure du bâtiment (colonne ou mur de gaine porteur renforcé) et se reporte au niveau des fondations.
- Peut imposer un volume construit inesthétique visible sur le toit.
- Problème d'accessibilité.
- Compacité de la gaine réduite par la présence de la cabine et du contrepoids, et par conséquent, réduction de la surface utile dans les étages du bâtiment.

Il est à noter que chaque modèle d'ascenseur électrique offre ses propres avantages. Le choix dépend des besoins de l'application.

### III.5.2.4 Motorisation des ascenseurs à traction a câbles

#### III.5.2.4.1 Moteur-treuil à vis sans fin

##### III.5.2.4.1.1 Description

D'abord couplé avec des moteurs à courant continu à excitation indépendante ou shunt. Ce type de motorisation avait l'avantage de pouvoir faire varier très facilement la vitesse de rotation. De nos jours les treuils à vis sans fin sont couplés à des moteurs à courant alternatifs et c'est en principe des moteurs à deux vitesses. Une première vitesse de démarrage lente, puis un léger choc d'accélération annonce le passage à la seconde vitesse qui est la vitesse de déplacement optimale.

Bien qu'avec les avancées technologiques et le perfectionnement des outils, les rendements de ce type de motorisation ont atteint 60% à 65%. Cependant les moteurs-treuil à vis sans fin sont abandonnés au profit des moteurs à attaque direct (sans réducteur « gearless »).

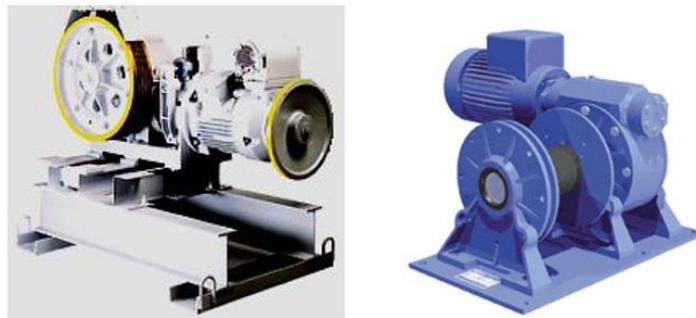


Figure III. 3 : Moteur-treuil à vis sans fin.

##### III.5.2.4.1.2 Avantages des moteurs-treuil à vis sans fin

- Précision dans les déplacements et sur la régulation de vitesse
- Grande plage de variation de vitesse.
- Couple élevé.

##### III.5.2.4.1.2 Inconvénients des moteurs-treuil à vis sans fin

- Entretien important.
- Efficacité énergétique faible.
- Consommation électrique non négligeable.

### III.5.2.4.2 Moteur-treuil planétaire

#### III.5.2.4.2.1 Description

Les appareils à treuil planétaire utilisent le système de réduction de vitesse par engrenages planétaires. Accouplés à un moteur électrique, ils permettent d'avoir un rapport de réduction appréciable pour obtenir une plage de vitesse compatible avec le confort et l'efficacité de déplacement souhaitée. [6]

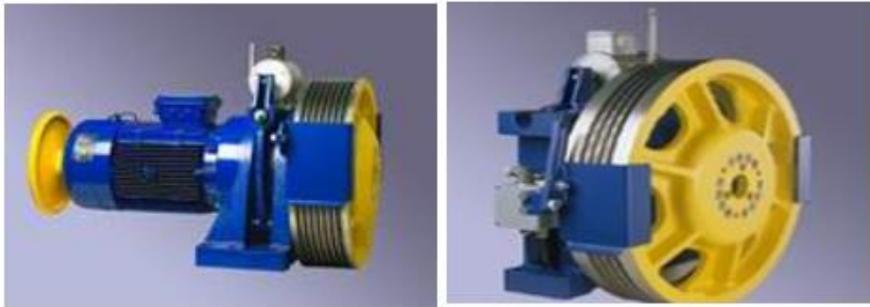


Figure III. 4 : Moteur-treuil planétaire.

Les moteurs électriques utilisés pour ce type de motorisation sont:

- à courant continu (pour une grande plage de variation de vitesse).
- à courant alternatif asynchrone à deux vitesses.
- à courant alternatif asynchrone commandé par un variateur de fréquence.



Figure III. 5 : Réducteur planétaire.

### III.5.2.4.2.2 Avantages des moteurs-treuil planétaires

- en fonction du type de moteur accouplé :
  - grande plage de variation de vitesse (courant continu ou alternatif à variation de fréquence).
  - rendement énergétique important entraînant une diminution des coûts à la conception (puissance installée plus faible) et des consommations moindres à l'exploitation.
- précision dans les déplacements et sur la régulation de vitesse.
- Couple important.

### III.5.2.4.3 Les moteurs à attaque directe

#### III.5.2.4.3.1 Description



Figure III. 6 : Moteur à attaque directe Z4-Z20.

Les moteurs à attaque directe sans réducteur ont fait leur apparition avec la venue des variateurs de fréquence. Les installations deviennent tellement compactes qu'il est possible à l'heure actuelle de se passer de local des machines sur le toit des immeubles. [6]

Ce système est énergétiquement performant principalement de par la présence d'un variateur de fréquence qui optimise la consommation énergétique. [6]

### III.5.2.4.3.2 Avantages des moteurs gearless

- vitesse optimisée par le variateur de fréquence.
- compacité du système.
- pas de lubrifiant.
- faible niveau de bruit.
- pertes mécaniques réduites.
- efficacité énergétique intéressante.
- poids réduit.
- pas de cabanon technique nécessaire pour les ascenseurs.
- précision dans les déplacements et sur la régulation de vitesse.

### III.5.2.4.3.3 Inconvénients des moteurs gearless [8] [10]

- la compacité peut entraîner des difficultés de maintenance.
- difficulté d'intervention dans la cage d'ascenseur.

Type de motorisation		Rendement	Courant nominal	Courant de démarrage
Moteur hydraulique		de l'ordre de 20 %	3In	12 à 18 In
Moteur-treuil a vis sans fin	Ancienne génération	de l'ordre de 45 %	In	2 à 3,5 In
	Nouvelle génération	60 à 65 %		
Moteur sans treuil (« gearless »)		de l'ordre de 78 %	**	**
Moteur-treuil planétaire		de l'ordre de 97 %	**	**

Tableau III. 1 : Tableau comparatif entre les différentes motorisations.

### III.5.2.5 Critères de choix des moteurs à traction

On différencie les motorisations à traction en fonction des critères principaux suivants :

- la performance énergétique.
- l'encombrement des équipements.

Les critères secondaires, mais néanmoins importants, prennent en compte le poids, le niveau acoustique, ...

### III.5.2.6 Les constituants d'un ascenseur à traction a câbles [7] [9]

Les ascenseurs à traction à câbles sont en général constitués de :

- **Cabine d'ascenseur** : Élément composé d'un plancher, de parois et d'un toit destiné à accueillir les personnes et les marchandises (la partie visible de l'ascenseur). Cet élément est inséré et fixé dans un cadre appelé suspension cabine.
- **Porte de cabine** : Porte à fermeture généralement automatique destinée à confiner l'utilisateur dans la cabine pendant le déplacement de celle-ci, lui interdisant tous contact avec les parties extérieures à la cabine.
- **Porte palière** : C'est la porte externe de l'ascenseur. Chaque ascenseur est équipé d'autant de porte palière que de nombre d'étage. Elles peuvent être battantes et commandées manuellement, ou automatiques et coulissantes (à ouverture centrale ou latérale). Elles doivent être équipées d'un dispositif empêchant leurs ouvertures si la cabine n'est pas sur le niveau et bloquant le départ pendant leur ouverture.
- **Boutons d'appels** : boutons installés aux paliers.
- **Boutons d'envois** : boutons sont installés dans la cabine.
- **Contrepoids** : Élément destiné à contre balancer le poids de la suspension cabine, augmenté de la moitié de la charge utile. Celui-ci est constitué d'une suspension métallique contenant des gueuzes en fonte destinées à l'alourdir. Lorsque la cabine d'ascenseur monte, le contrepoids descend.

- **Suspension cabine** : Cadre rigide, aussi appelé étrier, composé de poutrelles en acier et destiné à accueillir la cabine. Ce cadre est suspendu dans la gaine par les câbles de traction. Des coulisseaux épousant la forme des guides de cabine et fixés à la suspension, assure le déplacement parfait de la cabine.
- **Gaine d'ascenseur** : Gaine verticale dans laquelle se déplacent l'ascenseur et son contrepoids.
- **Guides** : Profilés en acier, généralement en forme de T, destinés à guider la cabine et le contrepoids dans la gaine (figure I.7).
- **Ancrage de guide** : Pièce métallique servant à fixer les guides aux murs de la gaine.
- **Coulisseaux** : éléments fixés à la suspension, garnis d'une fourrure, épousant la forme des guides et destinés à orienter celle-ci dans la gaine.



Figure III. 7 : Suspension de la cabine.

- **Cuvette** : Partie la plus basse de la gaine de l'ascenseur contenant les poulies de renvoi et les amortisseurs.
- **Amortisseurs** : Ressorts puissants placés en cuvette et destinés à ralentir la suspension cabine ou le contrepoids en cas de dépassement des "fin de course" de sécurité. Dans le cas d'un ascenseur à grande vitesse, on utilise des amortisseurs à huile.

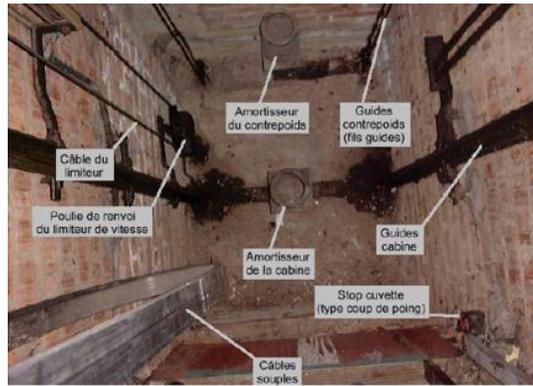


Figure III. 8 : cuvette d'ascenseur.

- **Fin de course:** Contact de sécurité placé généralement en gaine et destiné à stopper l'ascenseur en cas de dépassement de sa course normale. La fin de course peut aussi se trouver en machinerie. Dans ce cas, il est actionné soit par le tambour de traction soit par le câble du limiteur.



Figure III. 9 : Fin de course pour ascenseur.

- **Câble de sélecteur d'étage :** Généralement, les sélecteurs d'étage mécaniques sont entraînés par le treuil ou le limiteur de vitesse. Cependant, certains sélecteurs d'étage ont leur propre câble d'entraînement. Celui-ci, relié entre la cabine d'ascenseur et le contrepoids entraîne un petit tambour qui actionne le sélecteur d'étage.
- **Machinerie :** est un local placé généralement au-dessus de la gaine, destiné à contenir l'appareillage et le système de traction. Il est aussi appelé "salle des machines".

- **Appareillage:** constitue une armoire placée en machinerie, contenant les relais, et autres équipements destinés à commander l'ascenseur.
- **Treuil:** il s'agit d'une machine composée d'un dispositif de freinage et d'un moteur, destinée à actionner les câbles de traction de l'ascenseur. On distingue trois types de treuil : Les anciennes machines à tambour de traction, les machines, équipées d'un réducteur, appelées Geared (avec boîte de vitesse), les machines à traction directe (sans boîte de vitesse), sont appelées Gearless.
- **Volant de dépannage :** Dans le but d'assurer le déplacement manuel de l'ascenseur, un volant de dépannage est généralement fixé soit sur l'arbre du moteur de traction, soit sur l'axe du treuil.
- **Marquage des câbles :** Tous les ascenseurs équipés de treuil, devraient être équipés d'un marquage sur les câbles de traction, matérialisant la mise à niveau de la cabine. Celui-ci est à niveau lorsque la marque des câbles est en face d'une marque de même couleur peinte sur le châssis du treuil. Certains ascenseurs possèdent quant à eux un voyant installé sur l'appareillage, qui lorsqu'il est allumé, informe de la mise à niveau de la cabine.
- **Câbles de compensation :** Dans les immeubles de grande hauteur, le poids des câbles de traction devient trop important par rapport à la charge utile de la cabine d'ascenseur. Il existe donc une forte différence selon que la cabine se trouve en haut ou en bas de la gaine. On résout ce problème par l'adjonction de câbles, ou de chaîne de compensation. Ceux-ci, d'un poids identique aux câbles de traction sont suspendus entre la cabine et le contrepoids et permettent de maintenir l'ensemble en équilibre, quelle que soit la position de la cabine.
- **Chaîne :** Sur quelques ascenseurs spéciaux, certains câbles sont remplacés par des chaînes remplissant les mêmes fonctions. (Câbles de compensation, câble de limiteur, câble de sélecteur d'étage, câbles de traction).
- **Servofrein :** Mécanisme composé de pignons et d'un secteur de roue dentée. Celui-ci, commandé par un moteur est destiné à actionner le mécanisme de défreinage des mâchoires de frein. Le servofrein a depuis, été remplacé par l'électrofrein.

- **Electrofrein** : Electroaimant puissant destiné à assurer le défreinage des mâchoires de frein. Dans le but de permettre un défreinage manuel, l'électrofrein est généralement équipé d'un levier fixé à demeure. Si celui-ci est amovible, il doit être laissé à disposition dans la machinerie.
- **Tambour de frein** : Pièce cylindrique fixée solidement sur l'axe de la vis du treuil. Lors de l'arrêt de l'ascenseur, les mâchoires de frein sont appliquées fermement sur celui-ci pour immobiliser l'ascenseur.
- **Garnitures de frein** : Patins de friction en matière spéciale collés ou rivés sur les mâchoires de frein et destinés à bloquer efficacement le tambour de frein.



Figure III. 10 : frein mécanique de l'ascenseur.

- **Drive** : Le drive est la partie de l'appareillage qui commande le système de traction de l'ascenseur.



Figure III. 11 : armoire de commande pour ascenseur (drive).

• **Moto génératrice** : Machine destinée à créer une alimentation en courant continu de forte puissance. Cette machine est composée, d'une part, d'un moteur à induction alternatif triphasé et, d'autre part, d'une génératrice à courant continu. Ces deux machines sont solidaires l'une de l'autre.

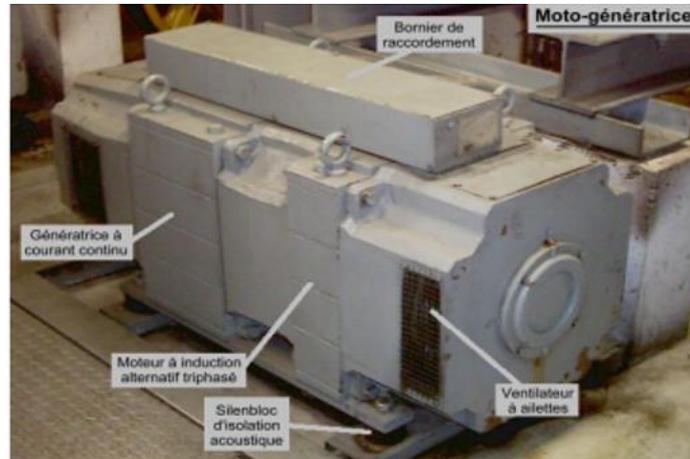


Figure III. 12: Moto-génératrice.

• **Poulie de renvoi** : Poulie tournant librement et destinée à guider les câbles entre la cabine et le contrepoids (figure I-11).

• **Poulie de mouflage** : Certains ascenseurs à grande capacité sont mouflés. C'est-à-dire qu'une démultiplication est installée à l'aide de poulies de mouflage. Lorsque la cabine parcourt un mètre, les câbles au niveau du treuil en parcourent deux ou trois. Cette méthode permet d'installer des treuils moins puissants mais augmente la longueur des câbles et le coût de leur remplacement.

• **Limiteur de vitesse** : Organe mécanique équipé de masselottes et placé généralement en machinerie. Le limiteur (solidaire à l'ascenseur par un câble limiteur) tourne au déplacement de l'ascenseur. Si la vitesse dépasse anormalement la vitesse maximale autorisée, les masselottes se lèvent et coupent un contact de sécurité.

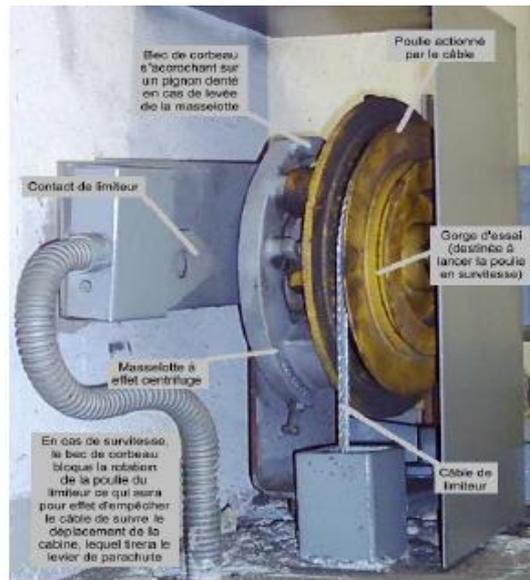


Figure III. 13 : réducteur de vitesse.

- **Parachute** : Organe mécanique placé sur la suspension de cabine et commandé par un câble de limiteur. En cas de rupture des câbles de traction ou de survitesse exagérée en descente, le mécanisme du parachute assure un blocage mécanique de la suspension dans les guides, évitant la chute libre de la cabine. Ce dispositif peut, dans certains cas, équiper le contrepoids.

- **Commande de rappel** : Dans le cas d'une grosse installation, il existe en machinerie un boîtier de manœuvre de rappel, destiné à ramener l'ascenseur au niveau d'un palier, pour dégager une personne bloquée dans la cabine.

Si ce boîtier existe, il faut tenter de déplacer l'ascenseur à l'aide de celui-ci. Si cela ne donne pas de résultat, il faudra couper le courant avant d'effectuer la manœuvre manuellement.

- **Commande de révision** : La commande de révision est composée d'un boîtier placé sur le toit de la cabine de l'ascenseur. Ce boîtier, équipé de bouton de marche montée et descente, ainsi que d'un bouton d'arrêt d'urgence, permet au préposé à l'entretien de manœuvrer, en toute sécurité et à faible allure, l'ascenseur pour inspecter et graisser les organes placés en gaine.

- **Inverseur d'étage** : Sur les anciens ascenseurs, contact inverseur qui était placé en gaine à chaque étage et servait de sélecteur d'étage.

• **Manœuvre** : Les ascenseurs se définissent en nombre d'appareil dans une batterie d'ascenseurs et par le genre de boutons d'appel utilisé pour les faire venir à l'étage. Ces paramètres déterminent le type de manœuvre. Il existe trois grandes familles de manœuvres : La manœuvre à blocage, la manœuvre collective et la manœuvre sélective qui n'est plus fabriquée actuellement.

### III.6 Critères de choix des ascenseurs [7] [8]

- Constructifs, tels que la hauteur de bâtiment, l'espace disponible au niveau des étages, la possibilité de placer une salle des machines au sommet de la gaine, stabilité du terrain, ...
- Organisationnels, comme le type de fonction du bâtiment, son occupation et son type de fonctionnement en garantissant une performance de confort (le temps d'attente moyen maximum et la vitesse de déplacement de la cabine) et de trafic (rapport charge/vitesse).
- énergétique, La consommation et l'appel de puissance (courant de démarrage) influencent directement le choix d'une technologie.
- de sécurité, est la préoccupation première des constructeurs d'ascenseur et le critère prépondérant pour le choix de la technologie.

### III.7 La sécurité dans les ascenseurs

La sécurité constitue la première qualité exigée pour tout ascenseur. En effet l'ascenseur fait l'objet de démarches et d'améliorations successives. Les innovations apportées ont en fait un appareil de référence pour le quotidien.

Les éléments assurant la sécurité dans les ascenseurs actuels sont :

- Systèmes pour ventilation et aération permanentes de la cabine.
- Portes coulissantes automatiques des cabines (butée en ouverture et fermeture, etc.).

- Dispositif d'alerte en cas de blocage de la cabine entre deux étages (téléphone, bouton d'alarme, caméras, etc.).
- Détecteurs d'obstacle (cellule empêchant la fermeture de la porte en présence d'objet ou lors du passage d'une personne).
- Dispositif conçu pour chaque type d'ascenseur, le permettant de ne pas démarrer si la charge maximale (poids de la cabine et ceux des personnes) est dépassée.
- Frein parachute, limiteur de vitesse, dispositif antichute et de secours (descente manuelle en cas de panne de courant).
- Cabines opaques (protection contre le vertige).

### **III.8 Conclusion**

De nos jours, les ascenseurs sont de plus en plus performants et sur ; ce qui fait d'eux des appareils indispensables pour les déplacements verticaux de l'homme au quotidien. Dans ce chapitre, on a vu qu'il existe deux catégories d'ascenseurs, et aux vues des avantages et des inconvénients de chaque catégorie ; l'ascenseur à traction à câbles s'est avéré être la catégorie la plus utilisée dans le monde. En effet ces ascenseurs se sont distingués par rapport aux ascenseurs hydrauliques, par leurs performances, leur coût réduit, et la possibilité d'un déplacement plus grand par rapport à un ascenseur hydraulique qui lui est limitée à un déplacement maximum de 18m

Chapitre 4 :  
Automatisation et  
supervision des ascenseurs  
par API

## IV.1 Introduction

Dans ce chapitre, nous allons appliquer l'ensemble des notions abordés dans les sections précédentes pour faire l'automatisation d'un ascenseur sur TIA PORTAL .Ce chapitre sera divisé en trois parties :

Dans un premier temps, nous élaborerons un descriptif du fonctionnement de l'ascenseur duquel on tirera un cahier des charges fonctionnel. Grâce à ce dernier, on pourra déterminer les éléments matériels nécessaires à notre projet. On clôturera cette partie en présentant les vues de supervision obtenues sur WinCC.

Dans la deuxième partie, on traitera le cas de deux ascenseurs qui travaillent en parallèle. On gardera le cahier des charges de la partie précédente auquel on apportera quelques modifications afin de prendre en compte la communication entre les deux ascenseurs. Dans cette partie, la programmation des deux ascenseurs se fera sur la même CPU, pour finir on présentera les vues de supervision sur WinCC.

La troisième partie suit le même concept que celle qui la précède à la différence que cette fois la programmation se fera sur deux CPU différentes, dont il faudra faire la communication afin d'échanger des informations entre les deux ascenseurs.

## IV.2 Commande d'un ascenseur avec API

### IV.2.1 Fonctionnement de l'ascenseur

L'ascenseur dont nous allons étudier le fonctionnement est un ascenseur desservant 5 étages. Il dispose d'un bouton « RUN » pour sa mise en fonction ; mais ce dernier ne peut démarrer que sous les conditions initiales suivantes :

- Non arrêt d'urgence.
- Poids non dépassé.

Une fois ces conditions satisfaites, le passager provoque la mise en route de l'ascenseur soit par appuie sur un bouton cabine d'appel d'étage (BI1, BI2, BI3, BI4, BI5) ou par appuie sur un bouton palière d'appel cabine (B1, BD2, BH2, BD3, BH3, BD4, BH4, B5), et selon sa position, la cabine monte ou descend. Arrivée à l'étage demandé, le détecteur de position de la cabine à l'étage (D1, D2, D3, D4, D5 )

s'enclenche et la cabine s'arrête , ce qui permet l'ouverture des portes de l'étage en question ( PO1, PO2, PO3, PO4, PO5 ), au même moment une temporisation de 10 Secondes est enclenchée, ce qui permet au passager de circuler (entrée dans la cabine ou en sortir).Une fois la temporisation écoulée, et si le bouton manuel d'ouverture des portes (ouv\_man\_ext1, ouv\_man\_ext2, ouv\_man\_ext3, ouv\_man\_ext4, ouv\_man\_ext5, ouv\_man\_int) et le détecteur de présence de personnes au niveau des portes palières (P1, P2, P3, P4, P5) sont désactivés, les portes se referment et l'ascenseur se dirige vers une nouvelle destination selon l'ordre reçu par appuie sur les bouton d'appels. Si l'ascenseur est en train de monter vers un étage défini, et qu'il reçoit un appel extérieur pour monter depuis un étage se trouvant sur son chemin, ce dernier est pris en compte et l'ascenseur s'arrête à cet étage. Néanmoins pour éviter les arrêts brusques de l'ascenseur, l'appel est pris en compte si la cabine est au moins à un étage en dessous de l'étage d'appel. La même procédure est effectuée lors de la descente et prend en compte les appels extérieurs pour descendre depuis un étage.

La prise en compte des **appels d'étages** (boutons intérieurs) s'effectue seulement pendant la temporisation. Cependant les appels effectués après la fin de la temporisation sont mémorisés pour une prise en compte ultérieure. Une fois les portes fermées, l'ascenseur se dirige vers un des étages demandé. L'ordre des priorités est défini dans le programme présent dans L'API. Le choix est fait pour garantir un déplacement minimal de la cabine.

Tant qu'un étage demandé n'est pas atteint, les appels le concernant sont mémorisés, et cette mémorisation ne se désactive que lorsque l'ascenseur s'arrête à cet étage et les portes s'ouvrent.

Si arrivée à un étage, aucune demande n'est effectuée les portes se referment et l'ascenseur s'arrête à cet étage. Deux cas peuvent se présenter :

**Cas 1** : un usager appelle l'ascenseur depuis un autre étage. Une temporisation de 5 secondes est lancée pour prendre en compte les éventuels appels qui pourraient survenir. Une fois la temporisation écoulée, l'ascenseur se dirige vers l'étage demandé

**Cas 2** : un ou plusieurs usagers sont au même étage que l'ascenseur et veulent descendre ou monter, ils actionnent l'ouverture des portes de l'ascenseur par appuie sur le bouton extérieur d'ouverture des portes. Une fois les portes ouvertes, une

temporisation s'enclenche pour prendre en compte l'ensemble des demandes puis un choix de priorité est effectué.

Des dispositifs de sécurité et des alarmes sont présents dans l'ascenseur pour garantir le transport en toute sécurité des usagers.

En cas de problème ; la procédure d'urgence est enclenchée et l'ascenseur s'arrête. L'équipe de maintenance dispose d'une manette pour faire monter ou descendre l'ascenseur vers l'étage le plus proche, l'ouverture des portes de la cabine se fait alors de manière manuelle grâce à une clé.

## IV.2.2 Cahier des charges

En se basant sur le mode de fonctionnement de l'ascenseur à cinq étages décrit dans le paragraphe précédent, on a défini le cahier des charges de l'ascenseur et on l'a mis sous forme d'un grafcet a 25 états en plus de l'état initial Xo.

L'appuie sur le bouton « RUN » active l'état Xo, et si et seulement si les conditions initiales CI sont vérifiées que l'ascenseur peut démarrer (état X1 activé).Un choix d'étage est effectué en prenant en compte la position actuelle de la cabine et les demandes des usagers de manière à garantir un déplacement minimal ainsi qu'un temps d'attente moindre.

Une fois le choix d'étage effectué ; une des conditions ET1, ET2, ET3, ET4, ET5 est mise à un et active de ce fait un des cinq états (X2, X5, X9, X13, ou X17), ce qui provoque la fermeture des portes et le lancement d'une temporisation (T2) de 5 secondes. Cette temporisation permet de prendre en compte les appels extérieurs de la cabine à un étage se trouvant sur le chemin de l'ascenseur ; prenons à titre d'exemple que l'ascenseur est au troisième étage et va descendre vers le premier étage ; si un usager se trouve au deuxième étage et demande à descendre ; sa requête sera traitée en priorité c'est-à-dire que l'ascenseur s'arrêtera à cet étage avant de continuer son chemin pour atteindre le premier étage. Une fois la temporisation écoulée l'ascenseur démarre.

Arrivée à l'étage choisi (X4, X8, X12, X16, ou X19), les portes au niveau de cet étage s'ouvrent et une temporisation de 10secondes est enclenchée. Une fois cette dernière écoulee, l'état X1 est réactivé et un nouveau cycle commence.

Elements du CDC	désignation
ET1, ET2, ET3, ET4, ET5	Les conditions sur les boutons d'appel et la position de la cabine, qui doivent être vérifié pour que l'ascenseur se dirige vers l'étage 1, 2, 3, 4, 5 respectivement.
POi i=1, 2, 3, 4, 5.	Porte de l'étage i ouverte.
Di i=1, 2, 3, 4, 5.	Position de l'ascenseur à l'étage i.
BHi i= 2, 3, 4.	Appel extérieur pour monter depuis l'étage i.
BDi i= 2, 3, 4.	Appel extérieur pour descendre depuis l'étage i.
B1	Appel extérieur pour monter depuis l'étage 1.
B5	Appel extérieur pour descendre depuis l'étage 5.
Bli i= 1, 2, 3, 4, 5.	Appel intérieur pour monter/descendre vers l'étage i.
Ti i=1,2	Temporisateur de type TON , c'est-a-dire a retard au démarrage.
Ouv_man_exti i= 1, 2, 3, 4, 5.	Bouton manuel extérieur d'ouverture des portes.
Ouv_man_int	Bouton manuel intérieur d'ouverture des portes

Tableau IV. 1 : tableau explicatif des abréviations du grafctet de l'ascenseur.

### IV.2.2.1 Descendre vers le premier étage

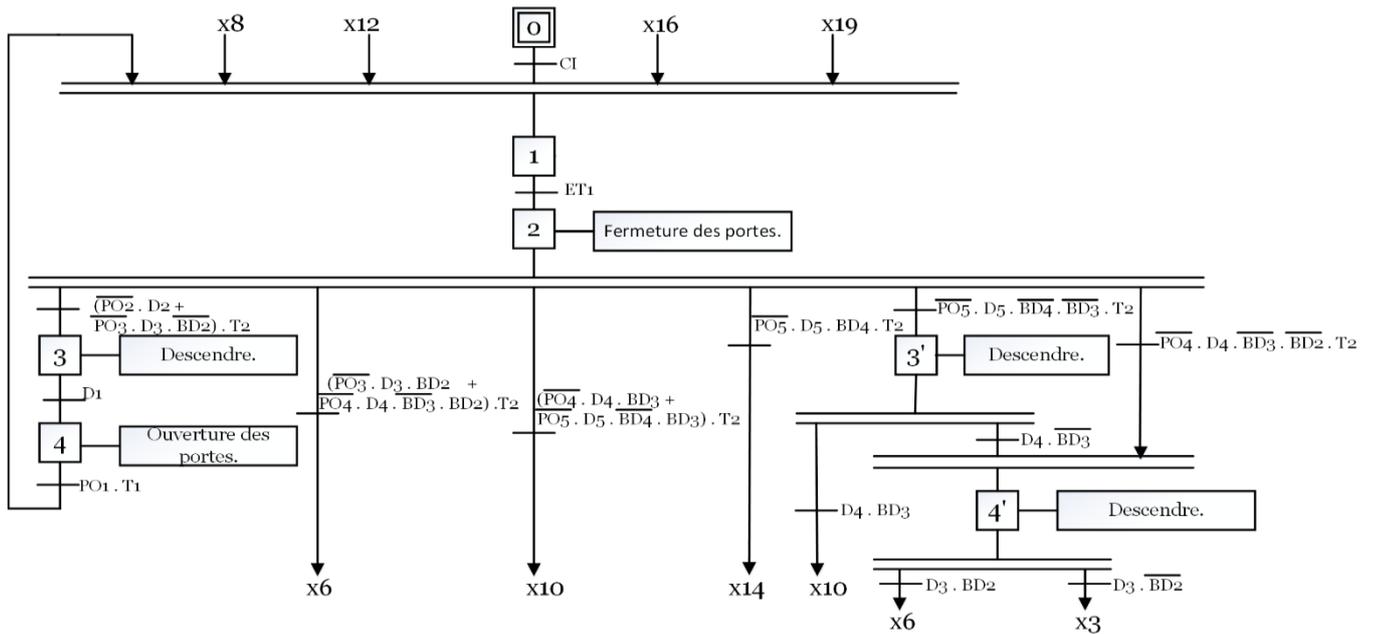


Figure IV. 1 : grafcet de l'ascenseur « vers étage 1 ».

### IV.2.2.2 Vers le deuxième étage

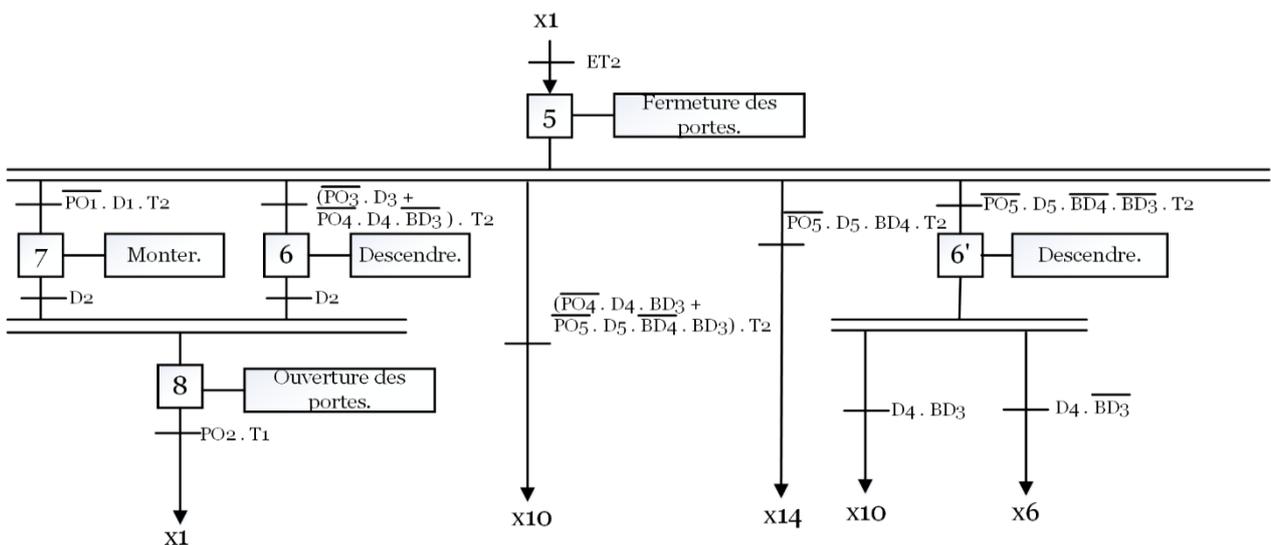


Figure IV. 2 : grafcet de l'ascenseur « vers étage 2 ».

### IV.2.2.3 Vers le troisième étage

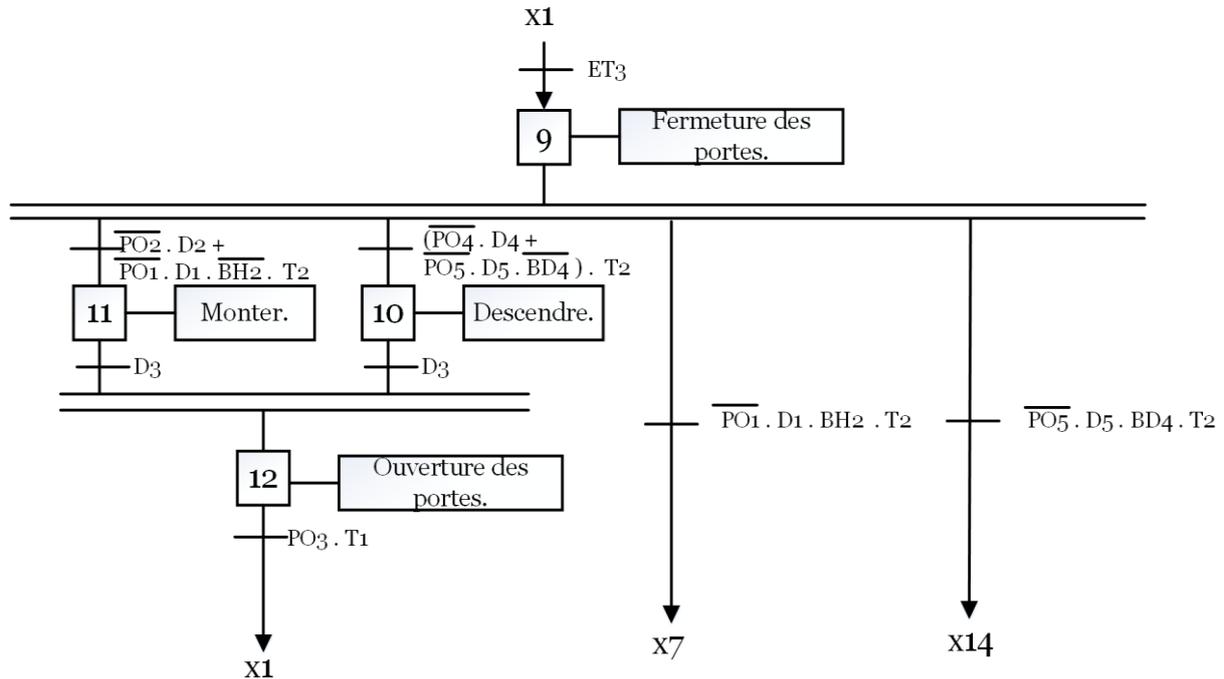


Figure IV. 3 : grafcet de l'ascenseur « vers étage 3 ».

Les grafkets des étages 5 et 4 sont analogues à ceux des étages 1 et 2 respectivement ; c'est pour cette raison qu'on ne présentera pas leurs grafket dans cette partie.

### IV.2.2.6 Procédure d'urgence

Si l'une des conditions de marche CM (vitesse non dépassée, poids non dépassé, non arrêt d'urgence...) n'est plus vérifiée, une alarme se déclenche provoquant l'arrêt de l'ascenseur, et la mise en route de la procédure d'urgence. Deux cas peuvent se présenter :

- Dans le premier cas, au moment de l'arrêt de l'ascenseur est au niveau d'un des cinq étages (le détecteur  $D1, D2, D3, D4$ , ou  $D5$  est à 1), une ouverture manuelle des portes se fait grâce à une clé.

- Dans le second cas, l'ascenseur s'est arrêté entre deux étages. Le déplacement de l'ascenseur vers l'étage le plus près est réalisé manuellement grâce à une manette se trouvant dans la pièce de machinerie (salle machine). Une fois arrivé au niveau de l'étage, les portes sont ouvertes manuellement par une clé.

Une fois la procédure d'urgence terminée, l'état X5 est activé occasionnant de ce fait l'activation de l'état Xo du grafcet principal. Une fois le problème solutionné et les conditions de marche rétablies (CM=1) l'état X5 est désactivé et l'ascenseur peut redémarrer (état X1 du grafcet principal).

Le grafcet décrivant la procédure d'urgence est illustré dans la figure ci-dessous :

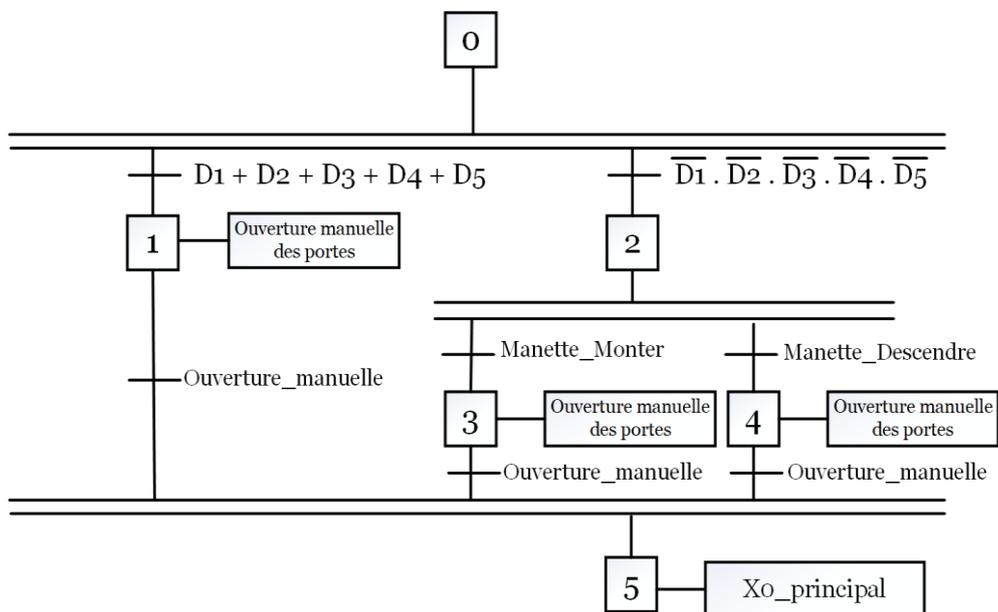


Figure IV. 4 : grafcet ascenseur « procédure d'urgence ».

### IV.2.3 Les entrées et sorties de l'ascenseur

Afin de pouvoir réaliser la configuration matérielle sur TIA Portal, il faut être en mesure de déterminer le nombre d'entrées et de sorties nécessaires pour l'automatisation de l'ascenseur à cinq étages. En effet la connaissance du nombre d'entrées et de sorties ainsi que leurs type permet de bien choisir les modules d'entrées/sorties à associer à l'automate.

A partir du grafcet fonctionnel décrit précédemment ; on peut conclure que les entrées et les sorties pour le cas de l'automatisation d'un ascenseur sont toutes digitales.

### IV.2.3.1 Les entrées de l'API

Les entrées de l'API sont représentées dans le tableau ci-dessous :

Entrées de l'organe de commande (API)	Nombre
Boutons palières d'appel cabine	8
Boutons cabine d'appel étage	5
Boutons manuel extérieur d'ouverture des portes	5
Bouton manuel intérieur d'ouverture des portes	1
Capteurs de détection de portes palières ouvertes	5
Capteurs de détection de porte cabine ouverte	1
Capteur de détection de présence de personnes au niveau des portes Palières.	5
Capteurs de détection de la cabine (position)	5
Capteurs de fin de course de la cabine (haut et bas)	2
Capteur de détection de dépassement de la vitesse	1
Capteur de détection de dépassement du poids limite	1
Mise en marche (RUN)	1
Arrêt	1
Arrêt d'urgence	1

Tableau IV. 2 : entrées de l'API.

### IV.2.3.2 Les sorties de l'API

Les sorties de l'API sont représentées dans le tableau ci-dessous :

Entrées de l'organe de commande (API)	Nombre
Voyants de prise en compte d'appels cabine	8
Voyants de prise en compte appels d'étage	5
Commande de montée de la cabine	1
Commande de descente de la cabine	1
Commande d'ouverture des portes	5
Commande de fermeture des portes	5

Tableau IV. 3 : sorties de l'API.

## IV.2.4 La configuration matérielle

Afin de réaliser l'automatisation de l'ascenseur sur TIA Portal, il faudra dans un premier temps définir le matériel nécessaire pour le bon fonctionnement de l'ascenseur.

### IV.2.4.1 La CPU 1214C DC/DC/DC

#### IV.2.4.1.1 Description [11]

Mémoire de travail 100 Ko ; alimentation DC24V avec DI14 x DC24V SINK/SOURCE, DQ10 x DC24V et AI2 intégrées ; 6 compteurs rapides et 4 sorties d'impulsions intégrées ; extension des E/S intégrées par Signal Board ; jusqu'à 3 modules de communication pour communication série ; jusqu'à 8 modules d'entrées-sorties pour extension des E/S ; 0,04 ms/k instructions ; interface PROFINET pour programmation, communication IHM et API-API.

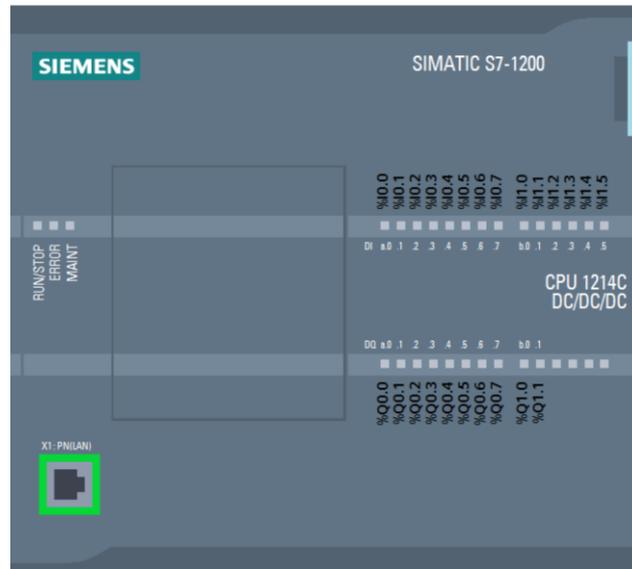


Figure IV. 5 : CPU 1214C DC/DC/DC.

Au vue de la description fournie par TIA Portal, notre choix s'est porté sur la CPU 1214C DC/DC/DC.

#### IV.2.4.2 Module d'entrées/sorties SM 1223 DI16/DQ16 x DC24V

Le logiciel TIA Portal propose des modules intégrant à la fois les entrées et les sorties. Ces modules regroupent les entrées/sorties par groupe de 8 ou de 16. Le choix le plus judicieux est d'utiliser des modules de 16 entrées/16 sorties digitales.

Notre choix s'est porté sur le module : SM 1223 DI16/DQ16 x DC24V

##### IV.2.4.2.1 Description [12]

Module d'entrées/sorties TOR DI16 x DC24V SINK/SOURCE et DQ16 x DC24V; retard à l'entrée paramétrable; borniers enfichables.

Etant donnée qu'on aura besoin de 42 entrées et 25 sorties digitales, on utilisera trois modules d'entrées/sortie SM 1223 DI16/DQ16 x DC24V.

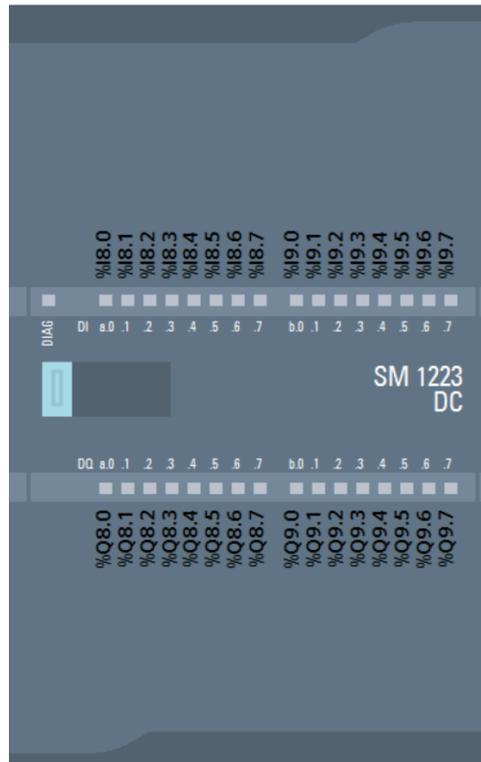


Figure IV. 6 : module d'entrées/sorties SM 1223 DI16/DQ16 x DC24.

La disposition des éléments de l'API sur TIA portal se fait sur un châssis et est donnée dans l'ordre suivant :

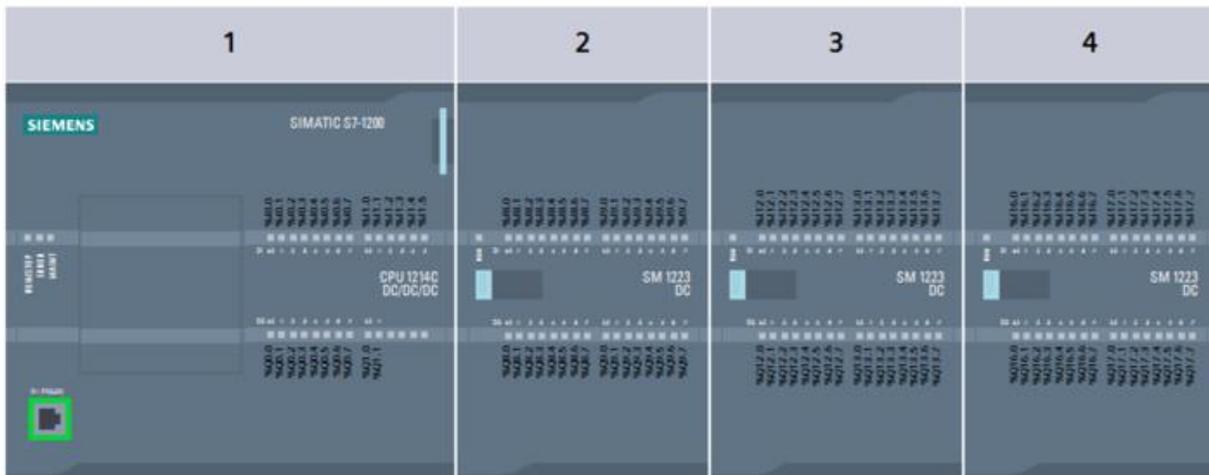


Figure IV. 7 : disposition de la CPU et des modules E/S sur le châssis.

L'adressage du premier module SM 1223 DI16/DQ16 x DC24V commence à l'octet 8 (%I8.0 pour les entrées et %Q8.0 pour les sorties) et s'étant sur 16bits soit 2 octets atteignant ainsi le bit %I9.7 pour les entrées et %Q9.7 pour les sorties. L'adressage du 2<sup>ème</sup> et du 3<sup>ème</sup> module E commence à l'octet 12 et 16 respectivement.

La CPU choisie dispose de 2 entrées analogiques (%IW64, %IW66) qu'on utilisera par la suite pour définir une variable de type « INT » qui sera consacrée à la programmation des alarmes de l'ascenseur.

## IV.2.5 Supervision de l'ascenseur sous WinCC

Pour mieux visualiser le fonctionnement de l'ascenseur ; nous avons associé à notre projet l'interface Homme-Machine « TP2200 Comfort Portrait ». Cette dernière est connectée à notre CPU par une communication PROFINET (PN/IE).

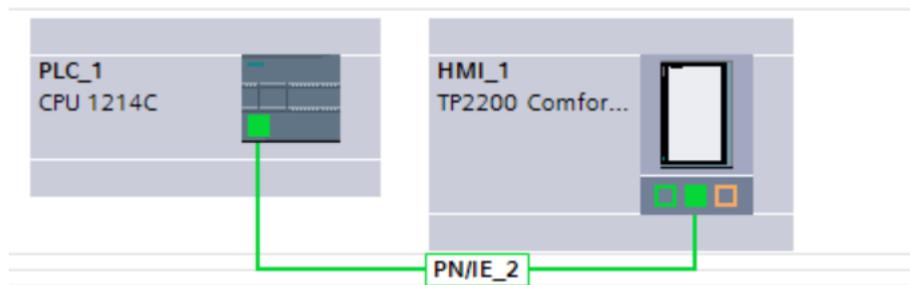


Figure IV. 8 : communication PROFINET PLC-HMI.

En plus de pouvoir visualiser le fonctionnement de l'ascenseur, cette interface va nous permettre de représenter la procédure d'urgence en cas de défaut, les alarmes, ainsi que les différents états du processus ; et pour ce faire nous allons créer 9 vues de supervisions.

### IV.2.5.1 La vue principale

La vue principale représente une vue de l'ensemble du projet. Cette vue permet de basculer vers les différentes vues de l'interface.

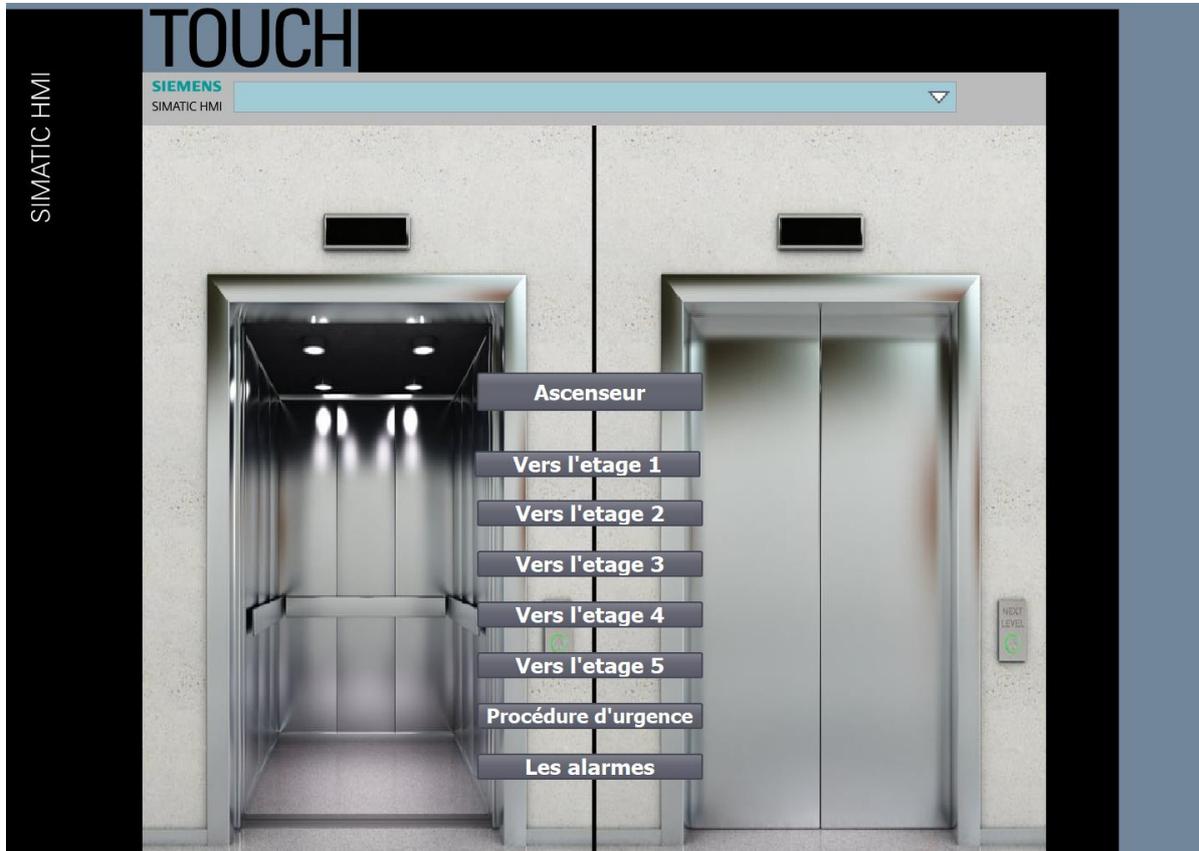


Figure IV. 9 : la vue principale.

### IV.2.5.2 La vue ascenseur

A travers cette vue, nous pouvons visualiser les déplacements de l'ascenseur à 5 niveaux grâce à une représentation réaliste de l'ascenseur. Des animations sont réalisées de façon à dynamiser l'ensemble et donner un effet réel à notre vue.

L'ensemble des capteurs (présence, position, portes, fin de course ....) ; ainsi que l'ensemble des boutons d'appels y sont représentés.

L'appuie sur le bouton « RUN » met en marche l'ascenseur et donc les différentes animations de notre vue.

L'appuie sur « vue principale » permet de revenir à la vue portail de notre projet.

La vue « ascenseur » est donnée sur la figure ci-dessous :

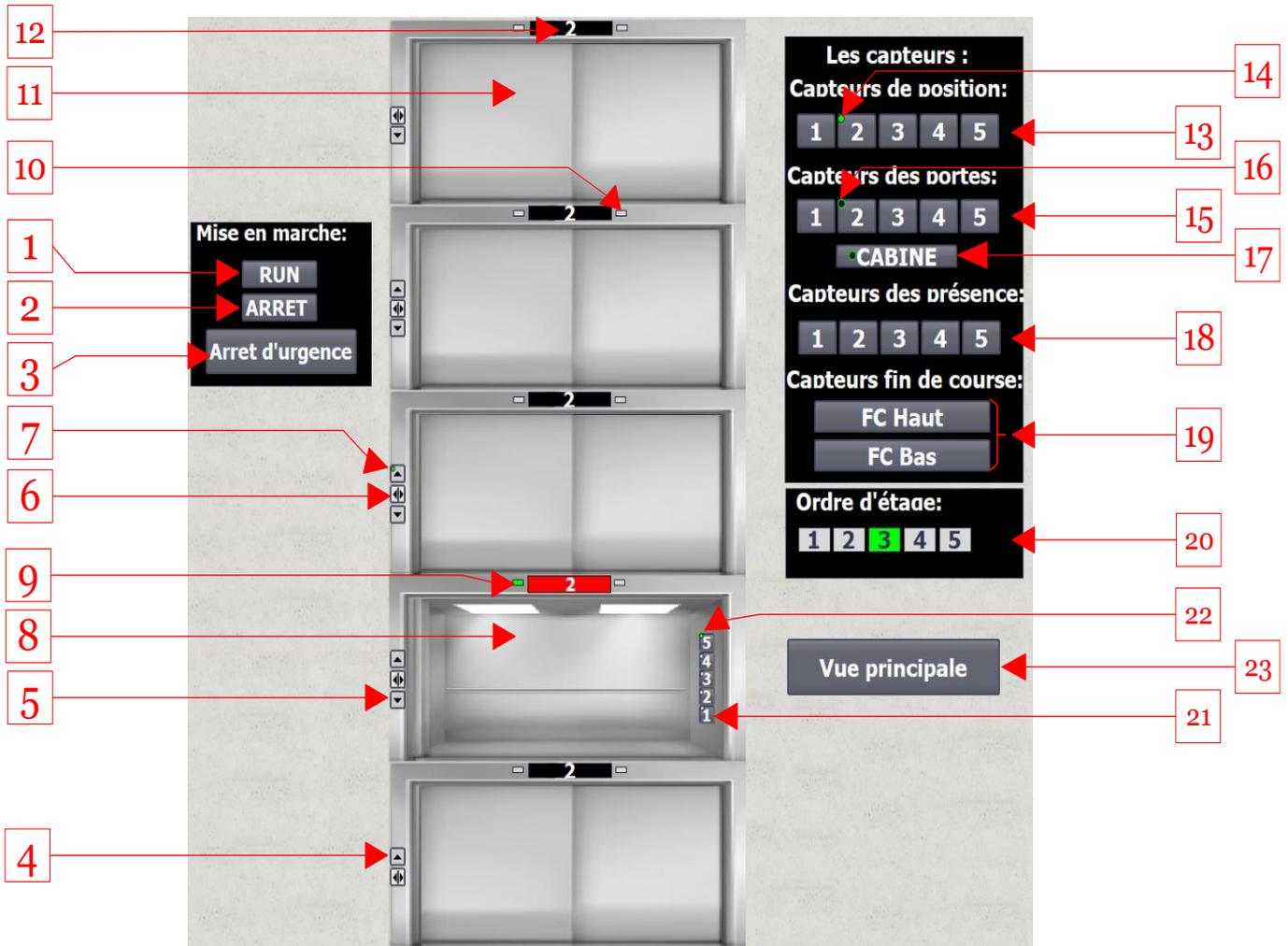


Figure IV. 10: la vue ascenseur.



Figure IV. 11 : cabine d'ascenseur en mouvement (monte/descend).

N°	Désignation
1	Bouton de mise en marche « RUN ».
2	Bouton arrêt de l'ascenseur.
3	Bouton d'arrêt d'urgence de l'ascenseur.
4	Bouton extérieur d'appel cabine a l'étage pour monter
5	Bouton extérieur d'appel cabine a l'étage pour descendre.
6	Ouverture manuelle des portes depuis l'extérieur.
7	Voyant de prise en compte de l'appel de la cabine.
8	Vue de l'intérieur de la cabine (portes ouvertes).
9	Voyant visualisant l'ordre de fermeture/ouverture des portes palières.
10	Voyant visualisant l'ordre de fermeture/ouverture des portes cabine.
11	Portes fermées.
12	Afficheur de la position de la cabine.
13	Capteurs détecteurs de la position de la cabine a un étage.
14	Voyant de visualisant la présence de la cabine a l'étage.
15	Capteurs détecteurs de portes palières ouvertes.
16	Voyant visualisant l'ouverture des portes palières.
17	Capteur détecteur de portes cabine ouvertes.
18	Capteurs détecteurs de la présence de personnes au niveau des portes.
19	Capteurs détecteurs de fin de course de la cabine (haut et bas).
20	Ordres d'étages.
21	Boutons intérieur d'appels étage.
22	Voyants de prise en compte des appels intérieur d'étage.
23	Aller vers la vue principale.
24	Ascenseur en marche, la cabine monte.
25	Ascenseur en marche, la cabine descend.

Tableau IV. 4 : tableau explicatif des différents éléments de la vue « ascenseur ».

### IV.2.5.3 Vue du grafcet des étages « vers étage i »

Les vues « vers l'étage i »  $i=1, 2, 3, 4, 5$  ; sont la représentation du grafcet de l'ascenseur ; chaque vue décrit le déplacement vers un des cinq étages, cela permet une visualisation rapide des états actifs du processus.

Supposons que la cabine de l'ascenseur est arrêtée au troisième étage. Une demande intérieure pour monter vers le cinquième étage est effectuée, les portes étant toujours ouvertes ; l'unique état actif du grafcet est l'état X17 qui se trouve dans la vue « vers étage 5 ».

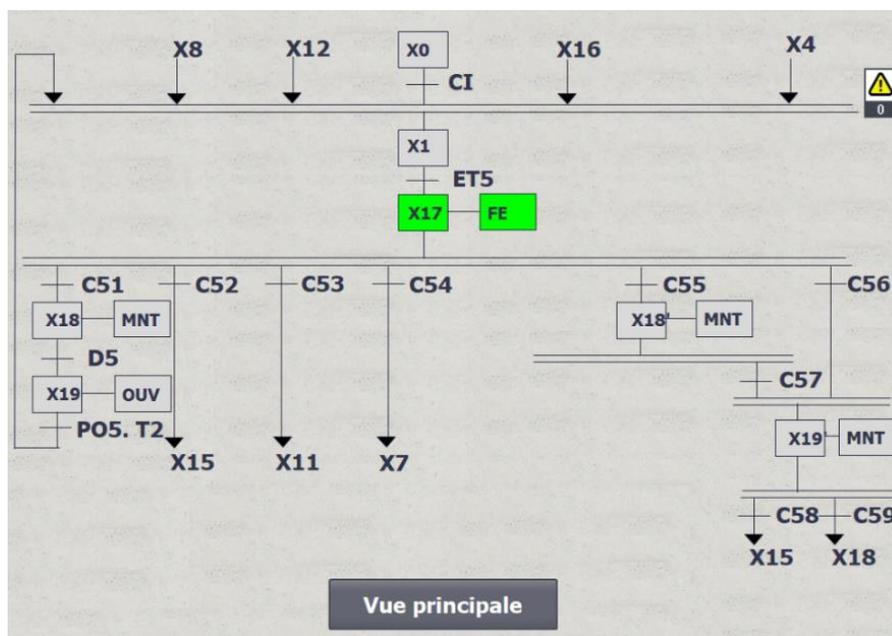


Figure IV. 12 : vue « vers l'étage 5 ».

Supposons maintenant que les portes se ferment, et qu'un usager effectue une demande de la cabine à l'étage 4 pour monter ; la cabine ayant pour consigne de monter et l'étage 4 se trouvant sur sa route, la demande de l'utilisateur est prise en priorité, par conséquent l'état X17 de la vue « vers étage 5 » se désactive ; l'état X15 de la vue « vers étage 4 » s'active et l'ascenseur commence à monter.

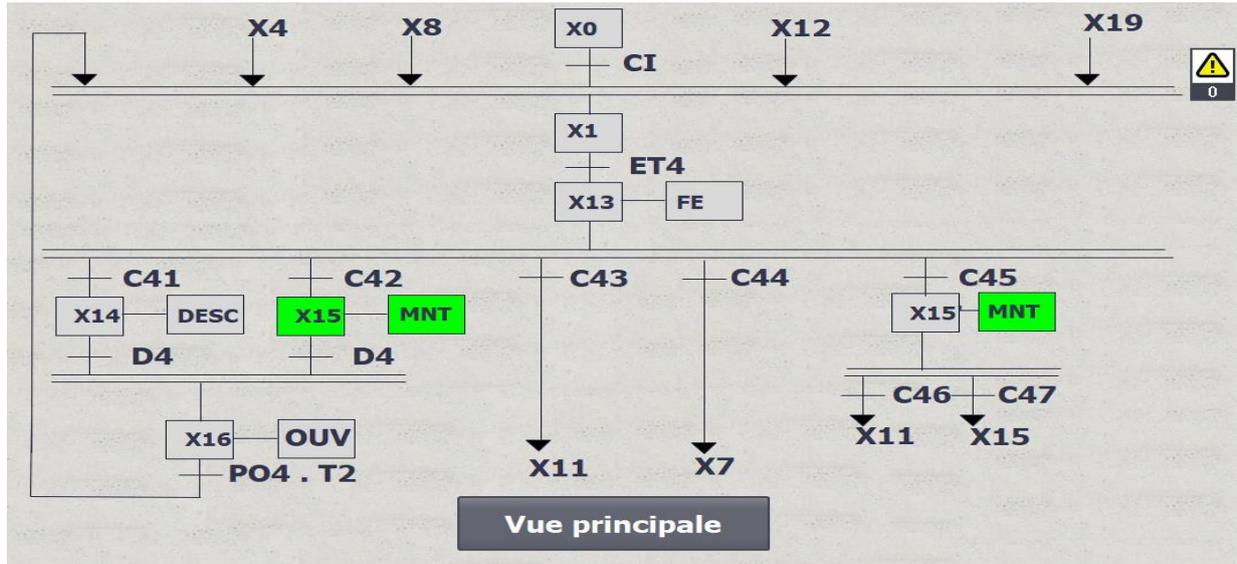


Figure IV. 13 : vue « vers l'étage 4 ».

	Désignation
FER	Ordre de fermeture de la porte de cabine.
OUV	Ordre d'ouverture de la porte de cabine.
MNT	Commande de la montée de la cabine.
DESC	Commande de la descente de la cabine.
Cij	Conditions de transitions entre les états (voir Grafcet ).

Tableau IV. 5 : tableau explicatif des éléments du grafcet.

Une fois la demande BH4 traitée (l'ascenseur s'est arrêté à l'étage 4) ; la demande de l'ascenseur au cinquième étage sera prise en compte et l'ascenseur monte en direction de ce dernier.

#### IV.2.5.4 La vue procédure d'urgence

La procédure d'urgence est décrite par son grafcet. Ce mode de fonctionnement représente le fonctionnement manuel en cas de problèmes.

Le déplacement de la cabine (montée/descente) est réalisé de façon manuelle grâce à une manette.

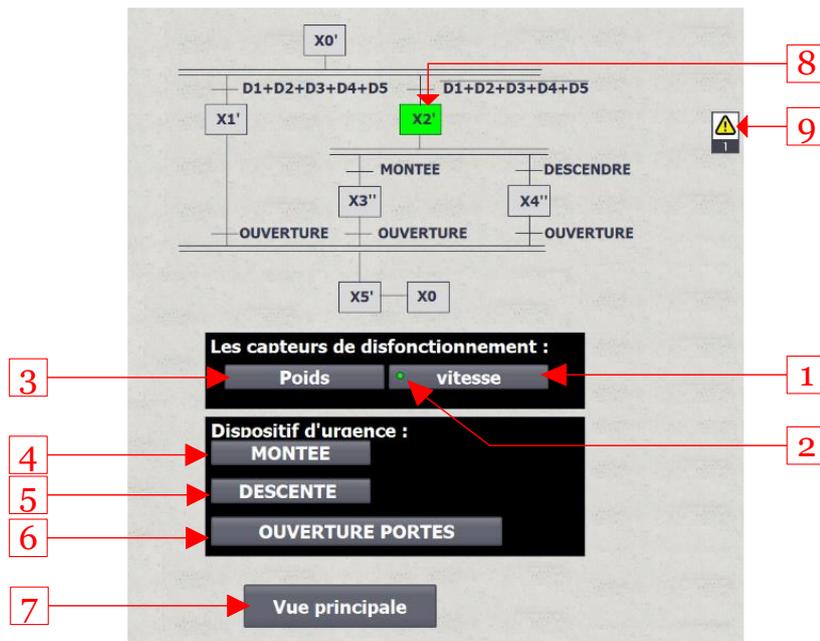


Figure IV. 14 : vue procédure d'urgence.

N°	Désignation
1	Capteur de dépassement de la vitesse.
2	Voyant indiquant le dépassement de la vitesse.
3	Capteur de dépassement du poids limite.
4	Montée manuelle de la cabine
5	Descente manuelle de la cabine
6	Ouverture manuelle des portes.
7	Aller a la « vue principale ».
8	Etat X2 de la procédure d'urgence activé.
9	Signalisation de présence d'alarme active.

Tableau IV. 6 : tableau explicatif des éléments de la vue « procédure d'urgence ».

### IV.2.5.5 La vue alarmes

La sécurité des usagers étant la première consigne à respecter pour les ascenseurs ; on définit dans notre projet 12 alarmes qui pourraient constituer si elles sont activées, un risque pour l'utilisateur et donc l'arrêt de l'ascenseur est primordial.

- cinq alarmes pour signaler l'ouverture intempestive des portes palières à chaque étage ; une alarme se déclenche si les portes palières d'un étage sont ouvertes alors que la cabine de l'ascenseur n'est pas présente à ce niveau.
- Alarme de porte cabine ouverte alors que l'ascenseur est en marche.
- Alarme de dépassement de la vitesse maximum.
- Alarme de dépassement du poids limite.
- Deux alarmes de fin de course (haut / bas).
- Alarme d'arrêt d'urgence.
- Alarme de présence de personnes au niveau des portes.

Sur l'interface HMI, on définit une vue alarme sur laquelle on dispose un tableau de signalisation ; afin de visualiser les alarmes, et des informations utiles les concernant ; tels que leur date d'occurrence ; l'heure ; l'état ainsi que des messages textes spécifiques à chaque alarme.

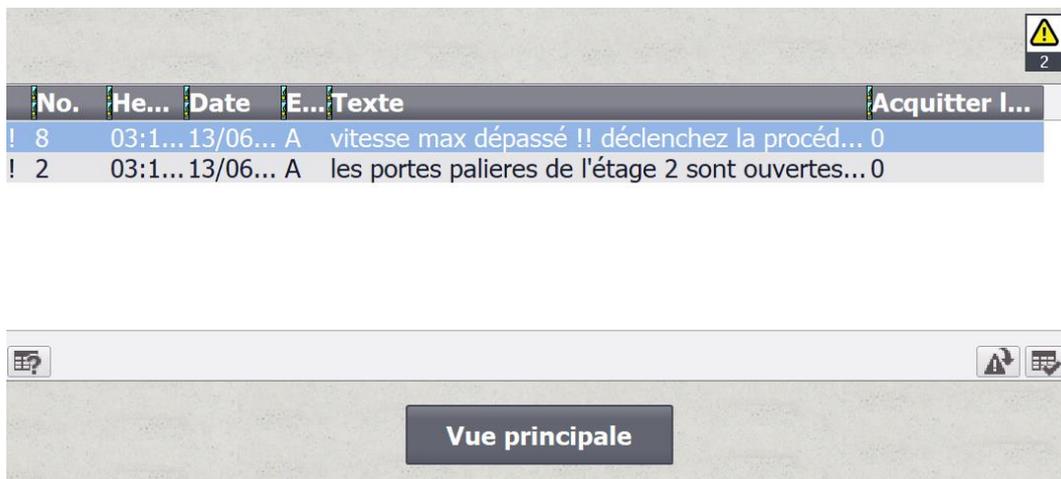


Figure IV. 15: vue « alarmes ».

## **IV.3 Commande de deux ascenseurs avec le même API**

### **IV.3.1 Fonctionnement des deux ascenseurs**

Le fonctionnement des deux ascenseurs est le même que celui de l'ascenseur vu dans la partie précédente, à la différence que dans le cas de deux ascenseurs, il doit y avoir une communication entre les deux ; c'est-à-dire que les deux ascenseurs doivent être en mesure de s'échanger des informations sur leurs positions respectives et leurs déplacements afin d'optimiser leurs fonctionnement et de garantir aux usagers un temps d'attente minimal.

### **IV.3.2 Cahier des charges**

Le même cahier des charges que celui de la partie précédente est adopté dans le sens où les états, et les conditions de transition restent inchangés. Cependant il existe une différence conséquente en ce qui concerne la mémorisation des appels extérieurs de la cabine à l'étage, en effet si l'un des deux ascenseurs ; appelons-le l'ascenseur A ; est arrêté à un étage, ou bien s'il se dirige vers cet étage et qu'un usager demande le second ascenseur à ce même étage, cette demande ne sera pas mémorisée et l'ascenseur B prendra en charge d'autres demandes, tandis que la demande de l'utilisateur sera prise en charge par l'ascenseur A .Ce mode de fonctionnement évite les déplacements inutiles et optimise le fonctionnement des deux ascenseurs.

En cas de procédure d'urgence au niveau d'un des deux ascenseurs; le second prendra en charge les demandes extérieurs de l'ascenseur en panne ; en d'autres termes si l'ascenseur A est en panne, et qu'il avait des demandent extérieurs en attentes, ces dernières basculerons vers l'ascenseur B qui lui est toujours en fonction.

Il est logique que les appels intérieurs de chaque ascenseur lui soient propres ; et donc il n'existe pas d'échanges de ces informations.

Les boutons « RUN » et « Arrêt » sont communs aux deux ascenseurs.

### IV.3.3 La configuration matérielle

Pour réaliser la configuration matérielle de 2 ascenseurs commandés par la même CPU on a choisi

- CPU : 1214C DC/DC/DC
- Module E/S digitales : SM 1223 DI16/DQ16 x DC24V.

### IV.3.4 Supervision des deux ascenseurs sous WinCC

Afin de mieux visualiser le fonctionnement et la communication entre les deux ascenseurs ; nous avons associé à notre projet l'interface Homme-Machine « TP2200 Comfort Pro ».

Dans le cas de la supervision de deux ascenseurs, on aura besoin de 13 vues ; en plus de la vue principale qui constitue un portail d'accès aux autres vues.

#### IV.3.4.1 La vue principale



Figure IV. 16 : la vue principale.

### IV.3.4.2 La vue des ascenseurs

De manière identique à la première partie ; nous avons créé une vue de supervision des deux ascenseurs.

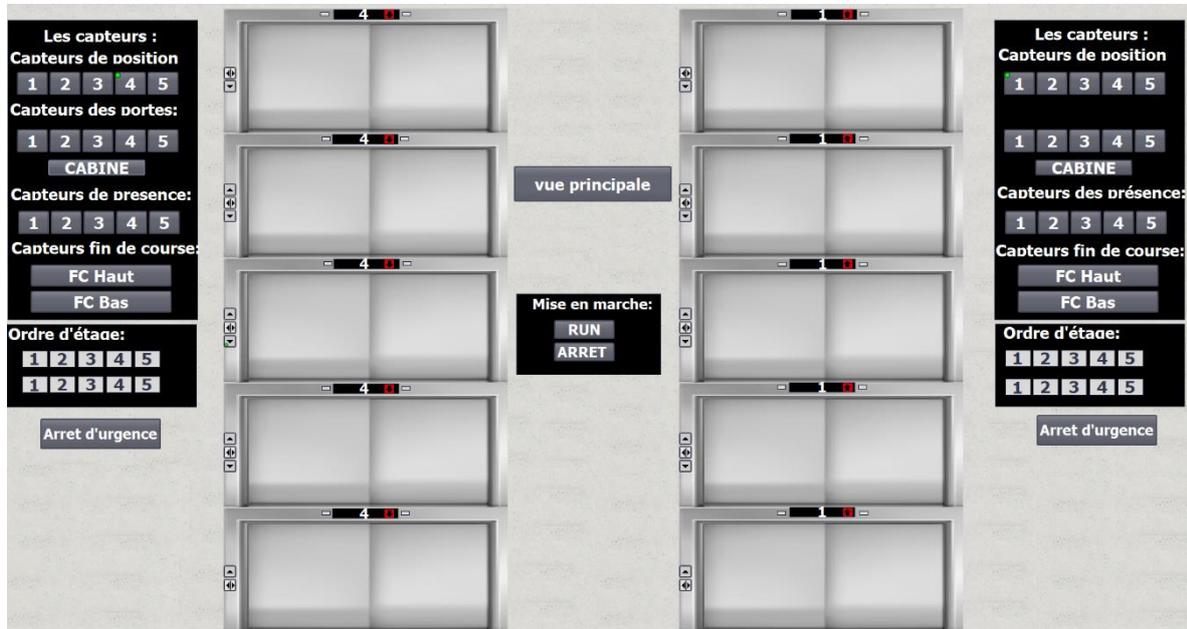


Figure IV. 17 : la vue des ascenseurs.

### IV.3.4.3 La vue des procédures d'urgences

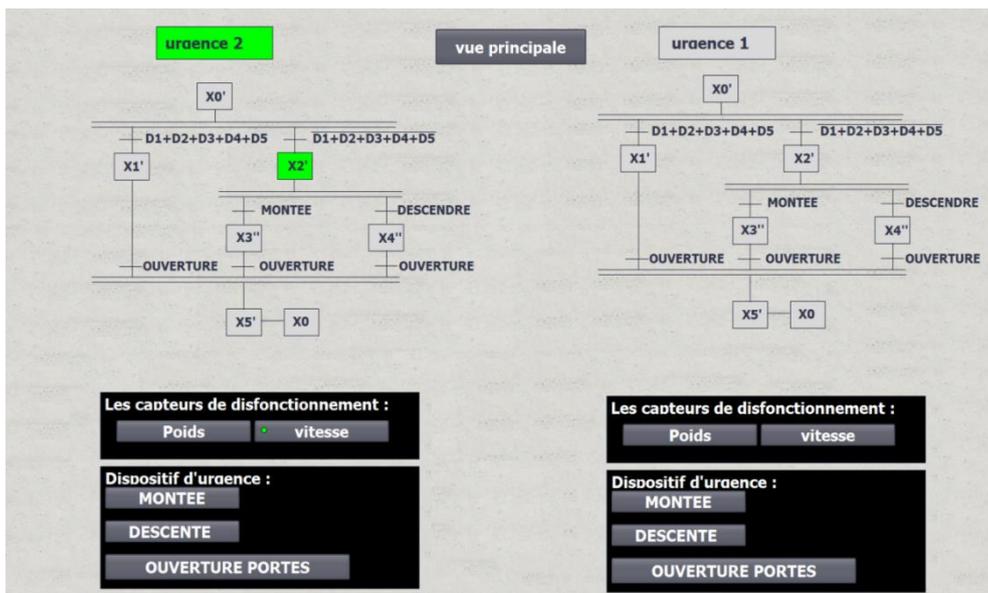


Figure IV. 18 : la vue des procédures d'urgences.

Comme décrit dans le cahier des charges ; lorsqu'un des ascenseurs rencontre un problème, les demandes extérieurs de ce dernier seront prises en charge par le second ascenseur. Les figures II.23 ET II.24 illustrent parfaitement ce cas.

Dans la figure ci-dessous ; l'ascenseur B est en train de descendre depuis le 5ème étage ; un appel extérieur pour descendre depuis le 4ème étage (BD4) et un autre pour monter depuis le 3ème étage (BH3) sont effectués ; tandis que l'ascenseur A est monte depuis le premier étage.

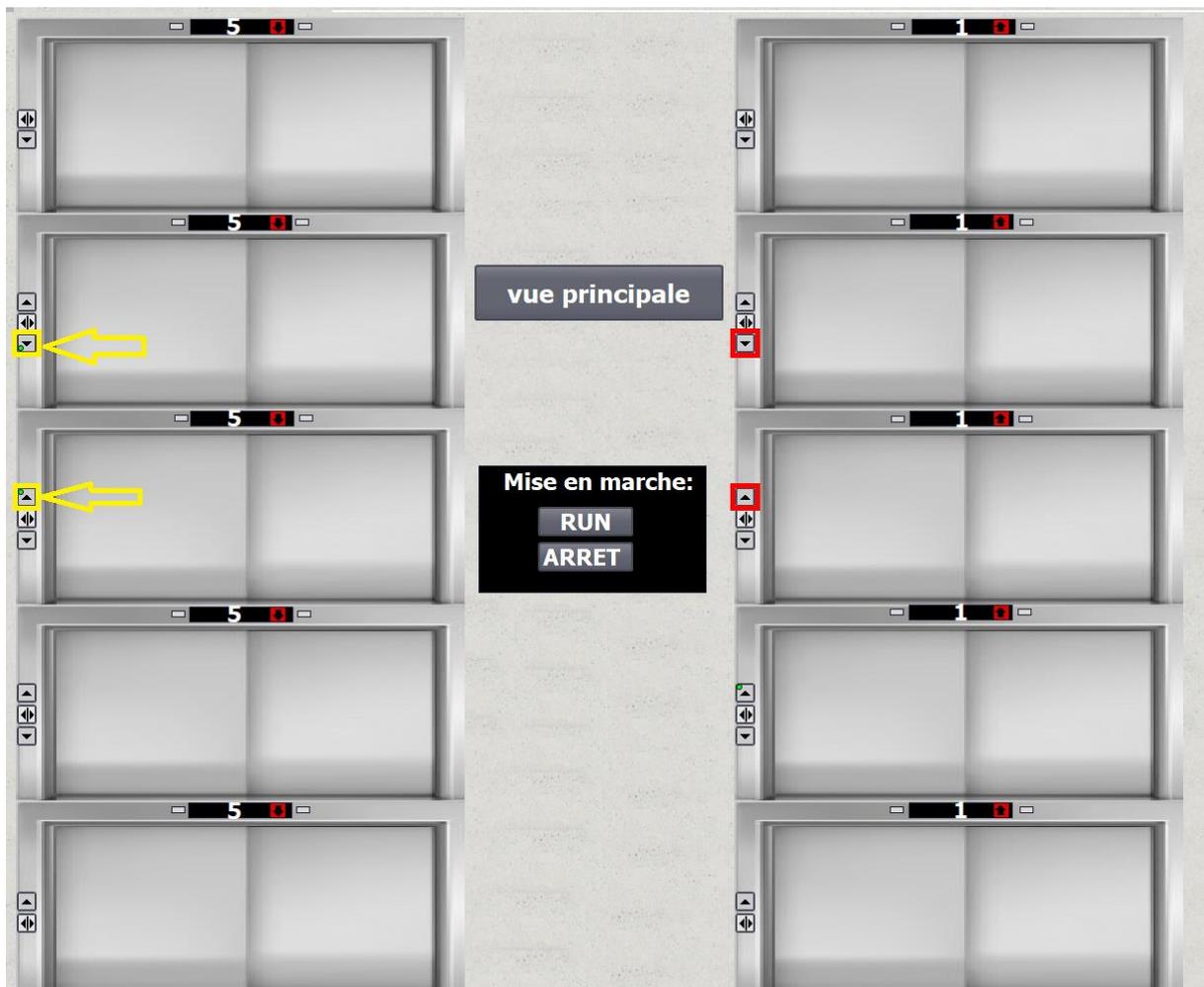


Figure IV. 19 : états des appels extérieurs avant la procédure d'urgence.

Dans la figure II.20 ; un problème de dépassement de la vitesse seuil survient pour l'ascenseur B ; ce qui conduit à l'arrêt de ce dernier et l'enclenchement d'une procédure d'urgence ; néanmoins les appels extérieurs (BD4, BH3) de cet ascenseur ne sont pas perdus ; ils basculent en effet vers l'ascenseur A.

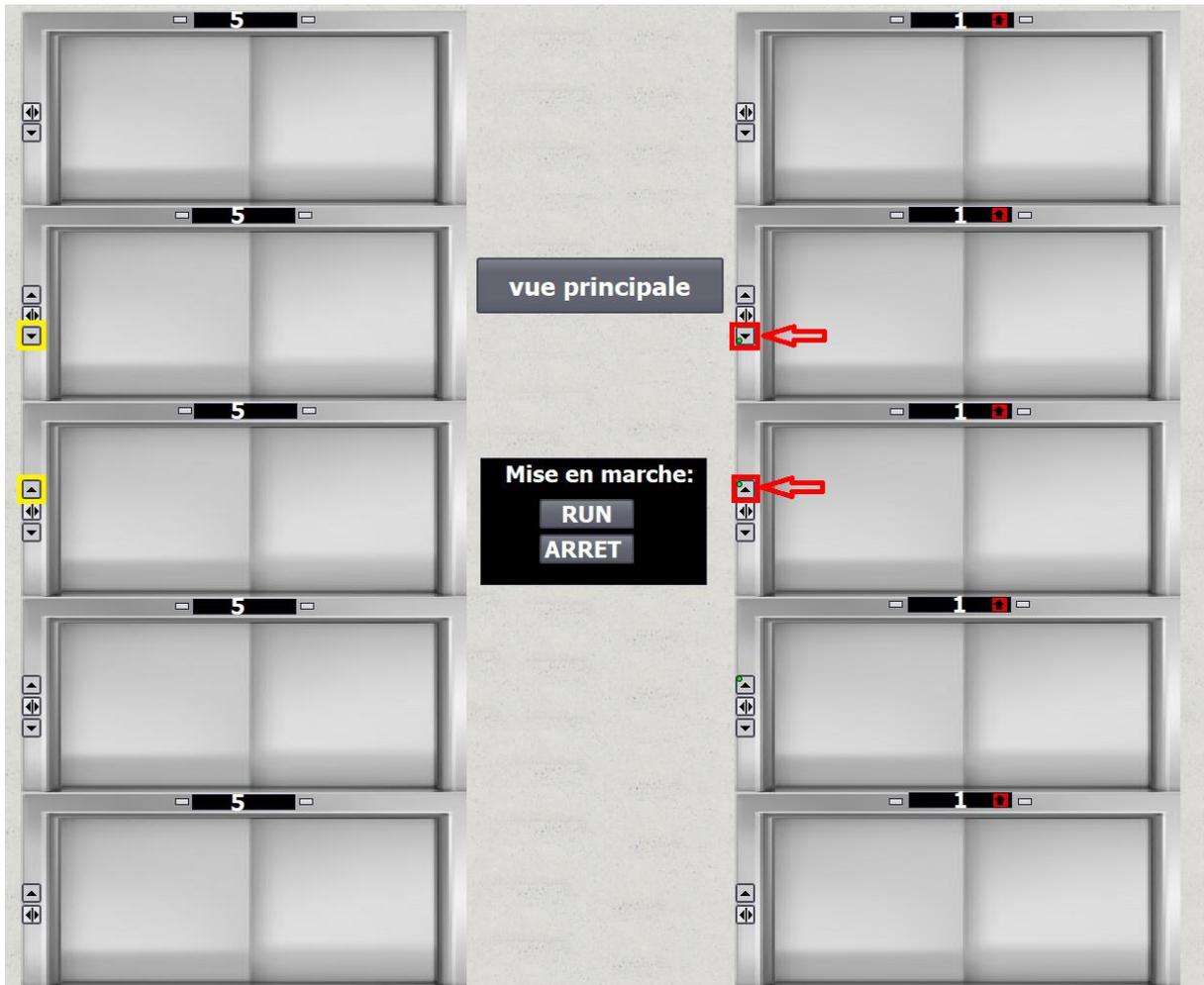


Figure IV. 20 : basculement des appels extérieurs lors de la procédure d'urgence.

## IV.4 Commande de deux ascenseurs avec deux API communicants (PROFINET)

Le mode de fonctionnement ; le cahier des charges ; ainsi que les vues de supervision de l'interface Homme-machine sont identiques à ceux de la partie précédente ; l'unique différence réside dans l'utilisation de deux CPU différentes c'est-à-dire que chaque ascenseur aura sa propre CPU ; il faudra alors communiquer les CPU entre elles par une liaison PROFINET (PN/IE) afin de pouvoir échanger les informations sur la position et les déplacements de chaque ascenseur et donc de garantir un fonctionnement optimal.

## IV.4.1 Communication entre les deux CPU

Pour cette partie de notre projet, on a choisi d'utiliser deux CPU Identiques : CPU 1214C.

La CPU 1214C contient des interfaces PROFINET. La liaison PN/IE ne peut se faire que si l'interface PROFINET de la première CPU et celle de la seconde appartiennent au même sous réseau.

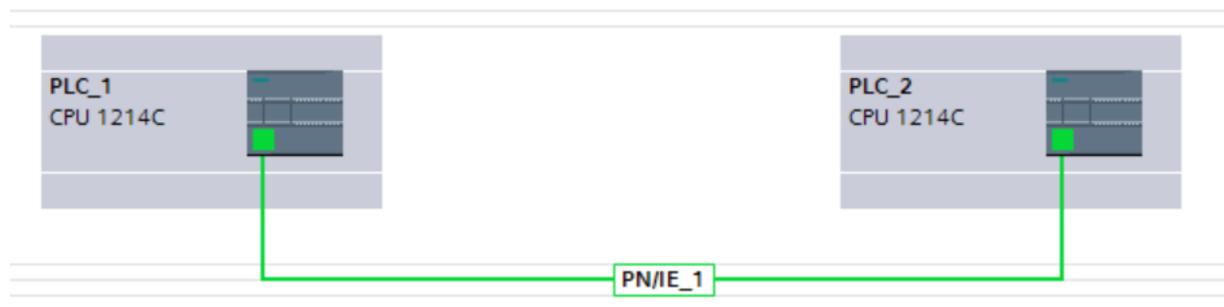


Figure IV. 21 : communication PROFINET PN/IE PLC-PLC.

Une fois que la liaison PN/IE est bien établie, les deux CPU peuvent communiquer et l'échange de données entre les deux ascenseurs devient possible. A ce moment, on pourra réaliser le même cahier des charges décrit dans la deuxième partie.

## IV.4.2 Supervision des deux ascenseurs sous WinCC

Afin de pouvoir réaliser la supervision des deux ascenseurs sous WinCC, on a choisi l'interface Homme-Machine « TP2200 Comfort Pro », qu'on a ensuite connectée aux deux CPU contenant toutes deux le programme d'un des ascenseurs.

La communication entre les deux CPU et l'HMI est une communication PROFINET et c'est la même que celle utilisée pour la communication entre les deux CPU, on parle donc de la liaison PN/IE\_1.

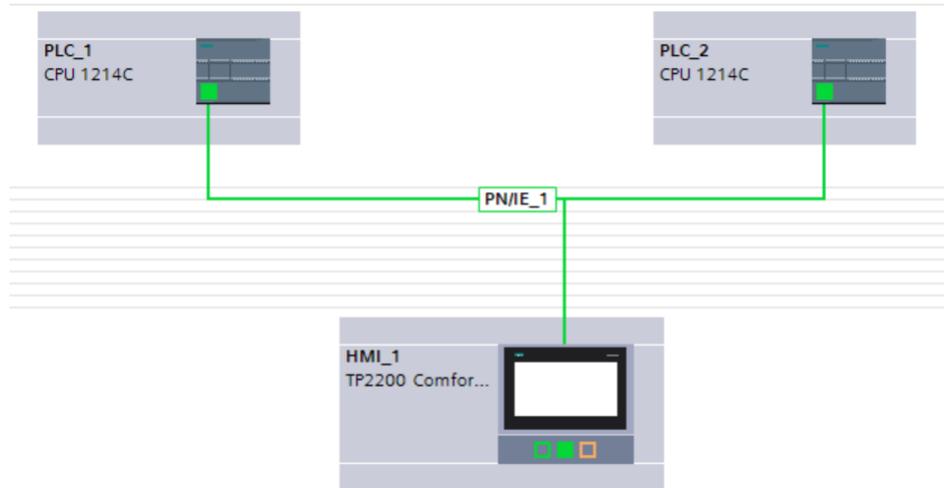


Figure IV. 22 : communication PROFINET PN/IE PLC\_HMI\_PLC.

## IV.5 Conclusion

A travers ce chapitre nous avons expliqué le fonctionnement d'un ascenseur, on a essayé de traiter tous les cas qui pourraient se présenter et nous en avons fait un cahier des charges qu'on a ensuite programmé en langage Ladder sur TIA Portal.

Le cahier des charges a ensuite été étendu pour prendre en charge le fonctionnement deux ascenseurs travaillant en parallèle, d'abord sur la même CPU puis sur deux CPU différentes où on a dû réaliser une communication par des interfaces PROFINET.

Enfin ; pour mieux visualiser le fonctionnement des ascenseurs, on a créé des vues de supervision. Cela a été possible grâce aux HMI disponible sur TIA Portal.

# Conclusion générale

Dans ce projet de fin d'études, notre contribution s'est portée sur l'élaboration d'un cahier des charges assurant le fonctionnement d'un ascenseur à cinq étages ; de sa programmation ainsi que sa supervision.

Afin de mener à bien ce travail, nous avons réalisé une solution à base d'automates programmables SIMATIC S7-1200 de SIEMENS ; ces automates assurent la sécurité, la rapidité ainsi que la fiabilité du processus. La supervision de l'ascenseur a été réalisée grâce à l'interface Homme-Machine TP2200 COMFORT.

La programmation ainsi que la supervision de ces automates ont été réalisés sur le logiciel TIA Portal ; ce dernier garanti une exploitation optimale des fonctionnalités de notre automate ainsi qu'une programmation rapide et facile. La présentation graphique offerte par WinCC ; permet de visualiser notre processus, et de faciliter ainsi sa compréhension et sa maintenance en cas de panne.

Enfin, ce travail nous a permis de développer les connaissances acquises tout au long de notre formation en automatique mais aussi d'acquérir de nouvelles compétences notamment en ce qui concerne la programmation et la supervision des API.

# Bibliographie

[1] BEN SAOUD, Slim. Chapitre 1, Les automatismes. In : Automatismes.

[2] Techniques ingénieur. <https://www.techniques-ingenieur.fr/base-documentaire/automatique-robotique-th16/supervision-des-systemes-industriels-42396210/langages-de-programmation-pour-systemes-automatisees-norme-cei-61131-3-s8030/>

[3] DOUDOU, Sofiane. Automates programmables industrielles, chapitre 4 architecture des API, pp. 39-43.

[4] *BEN SAOUD, Slim. Chapitre 2, Les automates programmables industrielles. In : Automatismes.*

[5] Technologuepro. <https://www.technologuepro.com/cours-automate-programmable-industriel/Les-automates-programmables-industriels-API.htm>

[6] Syderal. <https://www.syderal.fr/histoire-ascenseur/>

[7] M. Lbouhmadi, J. Laayoun « Etude d'ascenseur commandé par automate programmable » projet de fin d'étude, Université sidi Mohammed ben Abdallah, Fès, Maroc 2007.

[8] HASSAN, Ali « Automatisation d'un ascenseur par un API » projet de fin d'étude, Ali HASSANI, Université Mouloud Mammeri, Tizi-Ouzou, Algérie, 2018.

[9] ZIDANE ; Taous « Développement d'un outil numérique destiné à l'étude des performances dynamiques d'un entraînement électrique : application aux ascenseurs » mémoire de fin d'étude, Université Mouloud Mammeri, Tizi-Ouzou, Algérie, 2010.

[10] Energieplus. « Systèmes de motorisation » <https://energieplus-lesite.be/techniques/ascenseurs7/systemes-de-motorisation/> , septembre 2007.

[11] Siemens. Manuel CPU 1214C DC/DC/DC  
<https://support.industry.siemens.com/cs/pd/255094?pdti=td&dl=fr&lc=fr-FR>

[12] Siemens. Module d'entrées et de sorties TOR SM1223  
<https://support.industry.siemens.com/cs/pd/380435?pdti=td&dl=fr&lc=fr-FR>

[13] Manuel de Siemens « SIMATIC PROFINET Description du système », référence : 6ES7398-8FA10-8CA

# Annexe A

Les systèmes automatisés

## I Introduction

Quand les tâches à réaliser sont devenues pénibles, répétitives voire dangereuses pour l'homme ; l'automatisation des tâches est apparue, et **les systèmes automatisés** ont commencé à se répandre.

De nos jours les systèmes automatisés sont présents partout, du plus simple au plus complexe. Ces systèmes envahissent le quotidien de l'homme pour son plus grand plaisir.

Dans ce chapitre, nous allons nous pencher un peu plus en détails sur ces systèmes, étudier de plus près leur structure et comprendre leur fonctionnement.

## II Définition

Un système automatisé est un ensemble d'éléments en interaction les uns avec les autres, dont la fonction est d'agir sur une matière d'œuvre afin de réaliser une valeur ajoutée et ceci en limitant l'intervention humaine au simple fait de donner des consignes.

On appelle systèmes automatisés tout système de production qui agit de manière autonome en effectuant des actions cycliques préétabli.

La figure ci-dessous illustre un exemple de système automatisé : la commande de vols primaires de l'Airbus A380

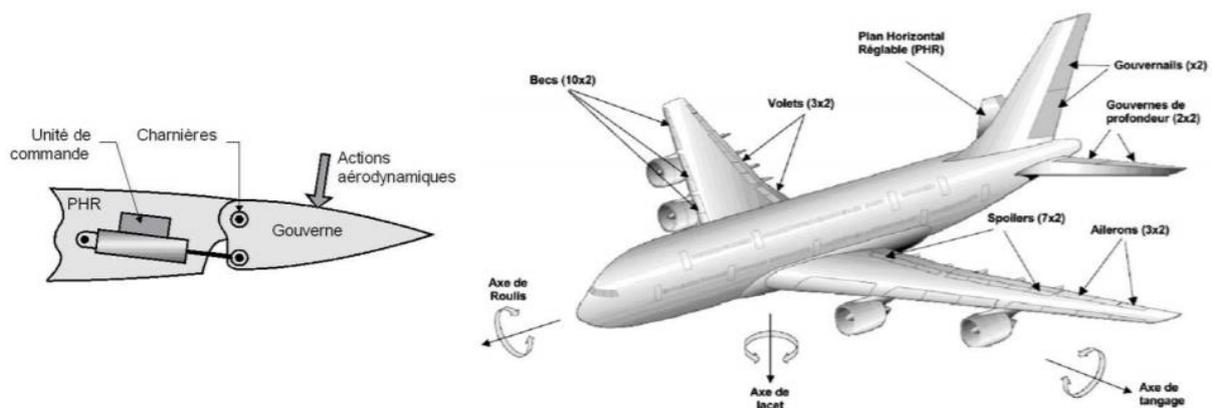


Figure a. 1 : Système automatisé.

### III Structure d'un système automatisé

Un Système Automatisé se décompose en trois parties :

- La Partie Commande (P.C.).
- La partie opérative (P.O).
- Le Poste de contrôle (le Pupitre ou Unité de Dialogue (U.D)).

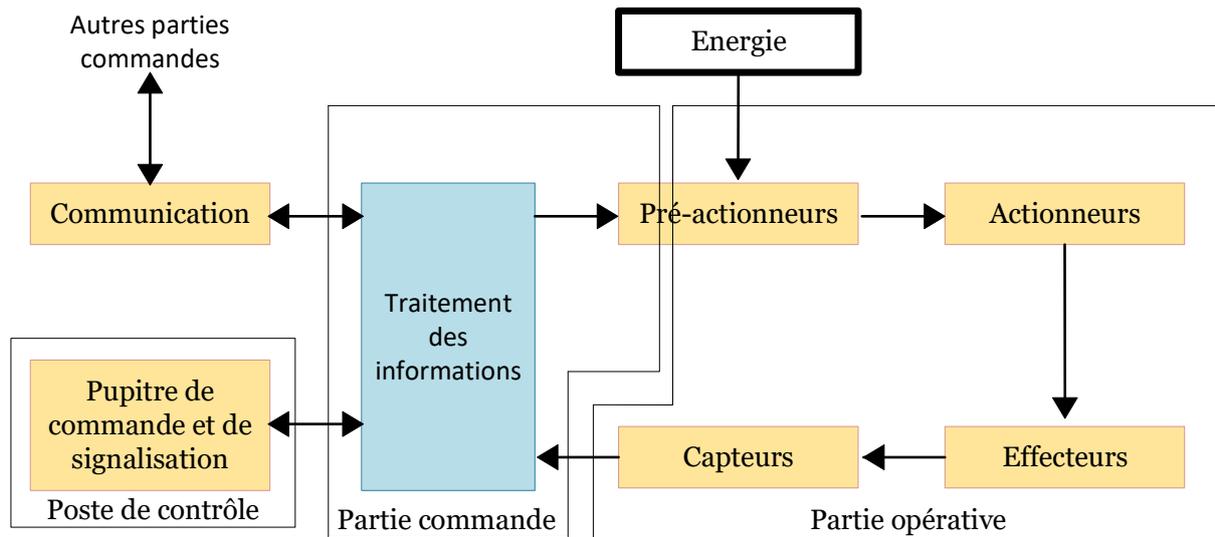


Figure a. 2 : structure d'un système automatisé.

#### III.1 La partie commande

c'est la partie chargé de la prise de décision du système automatisé, elle reçoit les informations de la partie opérative par l'intermédiaire des capteurs, traite ces informations en fonction des consignes du poste de contrôle, puis à travers les pré-actionneurs, renvoie les ordres qui doivent être exécuté par la partie opérative , elle permet également l'envoi de message de communication avec l'opérateur et les autres systèmes.

#### III.2 La partie opérative

C'est la partie qui est chargé de la conversion de puissance, c'est elle aussi qui effectue les ordres de la partie commande en opérant sur la matière d'œuvre.

La partie opérative est composée de :

### III.2.1 Les pré-actionneurs

Sur ordre de la partie de commande, les pré-actionneurs assurent la distribution de l'énergie aux actionneurs, ce sont donc les interfaces de puissance entre la partie commande et la partie opérative.

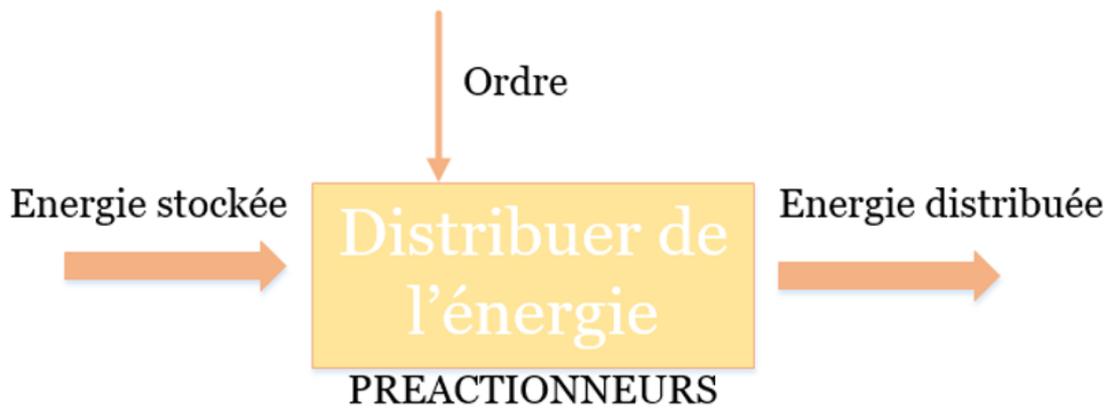


Figure a. 3 : rôle des pré-actionneurs.

Les pré-actionneurs constituent un élément de dialogue entre PC-PO ; et les plus utilisés sont :

- Les contacteurs pour les moteurs électriques.
- Les distributeurs pour les vérins pneumatiques ou hydrauliques.



Figure a. 4 : les pré-actionneurs.

### III.2.2 Les actionneurs

Convertissent l'énergie qu'ils reçoivent des pré-actionneurs, énergie qui est souvent pneumatique, hydraulique, ou encore électrique en une énergie « mécanique » utilisée par les effecteurs, créant ainsi un mouvement de translation ou de rotation.

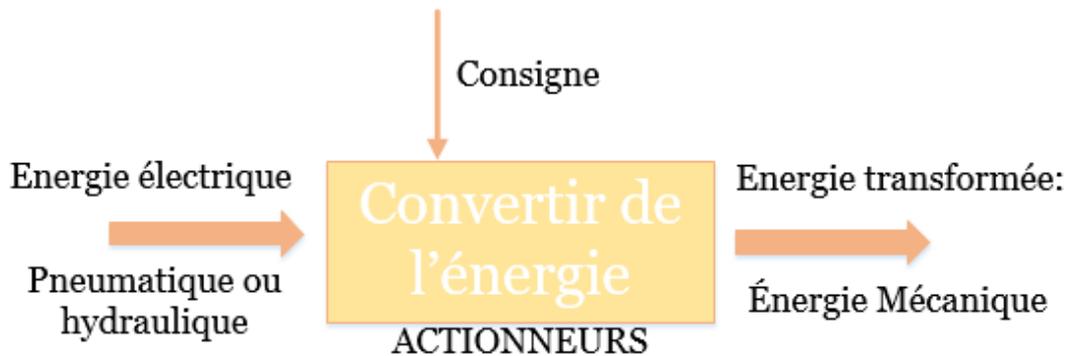


Figure a. 5 : rôle des actionneurs.

Les actionneurs les plus utilisés sont :

- Les moteurs pour les systèmes électriques.
- Les vérins pour les systèmes pneumatiques et hydrauliques



Moteur



Vérin

Figure a. 6 : les actionneurs.

### III.2.3 Les transmetteurs

Le transmetteur a pour rôle d'adapter l'énergie mécanique délivrée par l'actionneur pour la rendre utilisable par l'effecteur et la transmettre à ce dernier.

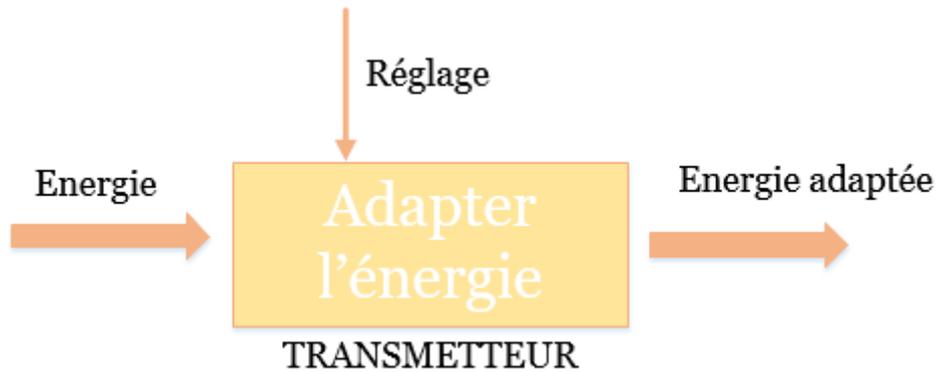


Figure a. 7 : rôle des transmetteurs.

Il existe deux catégories de transmetteurs :

- **Transmetteurs sans transformation de mouvements** : ils adaptent et transmettent l'énergie à l'effecteur sans transformation du mouvement. Exemple : roue et vis sans fin, pignon – chaîne, engrenage, poulie-courroie



Figure a. 8 : Transmetteurs sans transformation de mouvements.

- **Transmetteurs avec transformation de mouvement** :



Figure a. 9 : Transmetteurs avec transformation de mouvements.

### III.2.4 Les effecteurs

Tout organe en contact avec la matière d'œuvre et dont le rôle est d'agir directement sur celle-ci afin de réaliser l'effet souhaité (valeur ajoutée) est dit effecteur.

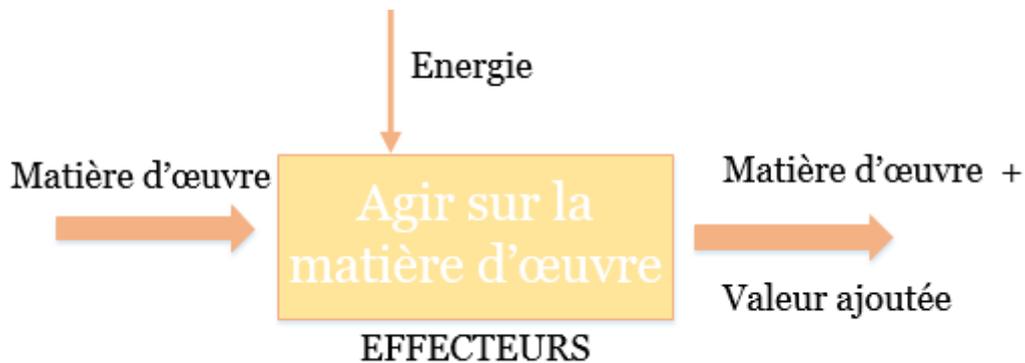


Figure a. 10 : rôle des effecteurs.



Figure a. 11 : effecteurs de robot.

L'ensemble pré-actionneurs, actionneurs, transmetteurs et effecteurs constituent ce qu'on appelle **chaîne d'énergie** ou **chaîne d'action** du système automatisé. Moyennant l'énergie disponible, cet ensemble de composant a pour rôle d'exécuter un ordre émis par la partie commande. Ce dernier se traduit par un effet sur la matière d'œuvre.

### III.2.5 Les capteurs

Ce sont des outils de prélèvement qui transforment une grandeur physique (force, accélération, vitesse, position pression, débit, intensité, température etc...) en un signal logique ou analogique porteur d'une information exploitable par la partie commande (PC).

Les capteurs sont donc en charge de la communication PO-PC, en effet la grandeur physique est une information sur l'état de la partie opérative.

Les capteurs sont souvent classés en fonction de leurs informations de sortie : [1]

- Logique : l'information de sortie est binaire, elle ne prend donc que la valeur 0 ou 1.
- Analogique : l'information est obtenue sous forme d'une tension ou un courant proportionnels à la grandeur mesurée.
- Numérique : La sortie est une séquence d'états logiques qui, en se suivant, forment un nombre.

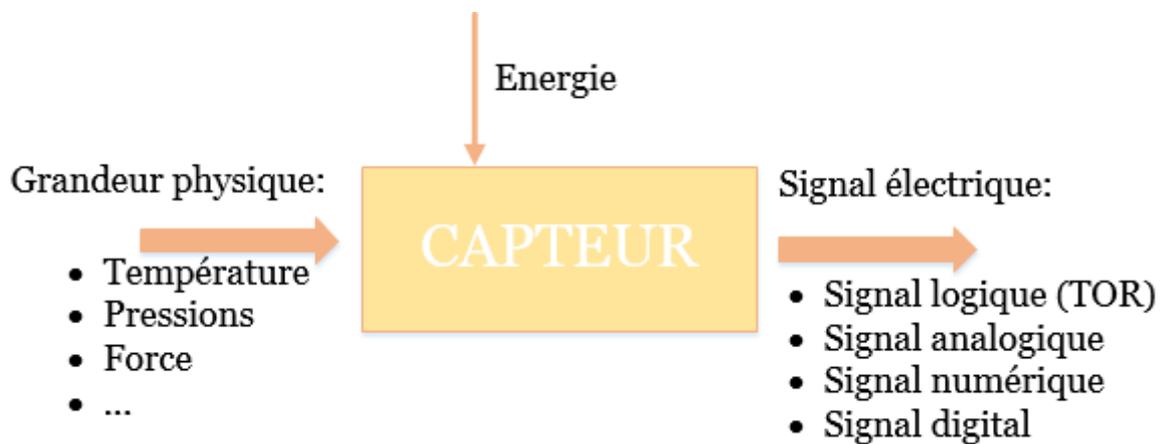


Figure a. 12 : rôle des capteurs.

Les capteurs de position sont les plus utilisés sur les systèmes automatisés. Ils prennent des formes variées : interrupteurs de position avec contact ou détecteurs de proximité sans contact (signal tout ou rien) et capteurs de déplacement (signal analogique ou numérique). [1]

Aux côtés de la partie commande, des codeurs, de l'interface homme-machine et du périphérique réseau, les capteurs forment ce qu'on appelle **la chaîne d'acquisition** ou **chaîne d'information**. Cette dernière a pour rôle l'acquisition, le codage éventuel, le traitement et la communication des informations.

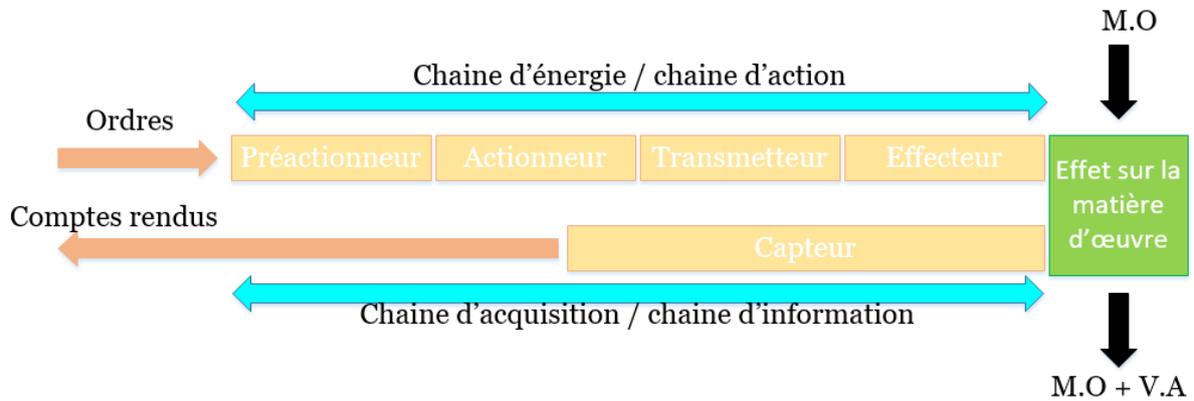


Figure a. 13 : structure de la partie opérative d'un système automatisé.

### III.3 Poste de contrôle

C'est l'organe servant d'interface Homme Machine (HMI). Il est composé du pupitre de commande par le biais duquel l'opérateur donne des ordres (consignes) à la partie commande (PC) ; et d'un pupitre de signalisation qui permet de visualiser les informations du système (par voyants, alarmes ...).

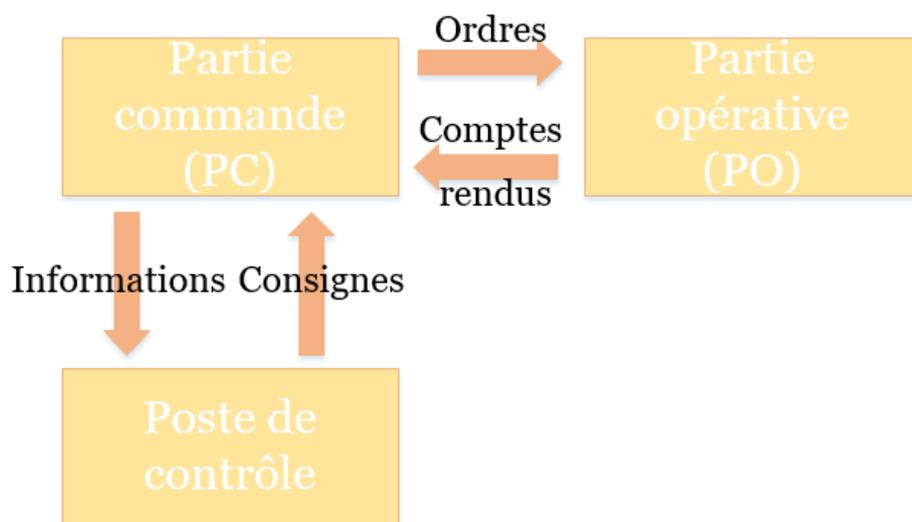


Figure I. 14 : schéma représentatif des interactions entre les différentes parties d'un système automatisé.

## **IV Conclusion**

A travers ce chapitre nous avons eu un aperçu global des systèmes automatisés et de leurs structures. On a ensuite mis en avant la fonction de chaque composant du système et l'importance du travail en chaîne dans l'automatisation du processus.

# Annexe B

Communication PROFINET  
entre deux CPU

# **I Présentation de la communication par PROFINET [13]**

## **I.1 Qu'est-ce que PROFINET?**

Dans le contexte de la Totally Integrated Automation (TIA), PROFINET est le prolongement cohérent de :

- PROFIBUS DP, du bus de terrain établi
- Industrial Ethernet, du bus de communication au niveau cellule.

Les acquis des deux systèmes ont été et continueront à être intégrés dans PROFINET.

En tant que standard d'automatisation conçu sur la base d'Ethernet par PROFIBUS International, PROFINET est un modèle non propriétaire de communication et d'ingénierie.

## **I.2 Objectifs de PROFINET**

La norme PROFINET a les objectifs suivants :

- Standard Ethernet ouvert, conçu pour l'automatisation sur la base de Industrial Ethernet : Les composants Industrial Ethernet et Ethernet standard sont compatibles entre eux, toutefois les appareils Industrial Ethernet sont plus robustes et donc mieux adaptés à un environnement industriel (température, immunité aux perturbations, etc.).
- Utilisation de TCP/IP et de standards des NTIC
- Automatisation avec Ethernet temps réel
- Intégration sans faille de systèmes de bus de terrain

## **I.3 Avantages de PROFINET**

PROFINET précise les fonctions servant à réaliser une solution d'automatisation globale, de l'installation du réseau au diagnostic sur le Web. Grâce à sa structure modulaire, il sera très facile de lui ajouter des fonctions à l'avenir. Il en résulte les avantages suivants :

- Flexibilité grâce à l'exploitation d'Ethernet et des standards éprouvés des NTIC
- Economies au niveau de l'ingénierie et de la mise en service grâce à la modularisation
- Rentabilisation des investissements en matériel et applications PROFIBUS
- Plus rapide que les bus spécifiques actuels dans le domaine du Motion Control
- Vaste gamme de produits disponibles sur le marché.

## **I.4 Architecture PROFINET**

Architecture PROFINET PROFIBUS International a tenu compte des aspects suivants dans l'architecture PROFINET :

- Communication entre appareils de terrain tels que les périphériques et les entraînements (Possibilité d'intégration des architectures PROFIBUS existantes. La rentabilisation des investissements en matériel et applications PROFIBUS est ainsi assurée).
- Communication entre automates en tant que composants de systèmes répartis (La modularisation technique se traduit par des économies au niveau de l'ingénierie et de la maintenance).
- Connectique faisant appel à des connecteurs et composants de réseau standardisés

## II Communication entre deux CPU par interfaces PROFINET

Afin d'illustrer les étapes à suivre pour réaliser la communication PROFINET entre deux CPU, On utilisera les CPU :

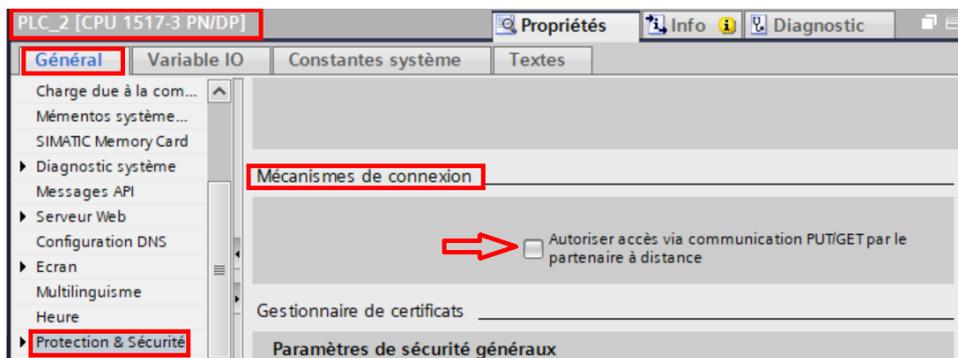
- CPU 1518\_4 PN/DP.
- CPU 1518\_3 PN/DP.

La communication entre les 2 CPU suit plusieurs étapes qu'on va énumérés ci-dessous :

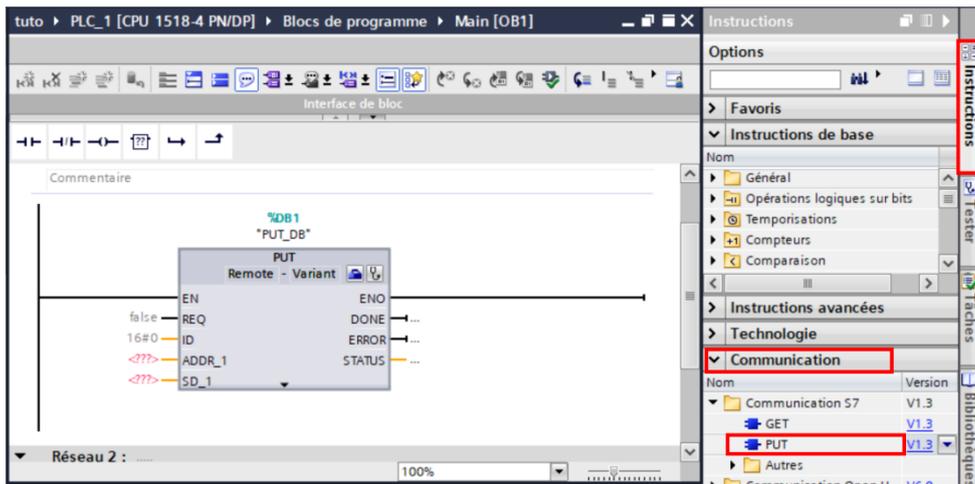
### II.1 Les étapes

#### II.1.1 Etape 1

Pour commencer ; il faut activer l'accès via communication PUT/GET pour chaque CPU et pour ce faire, allez dans les propriétés de chaque CPU, et dans général sélectionné « protection & sécurité » puis enfin autoriser l'accès.

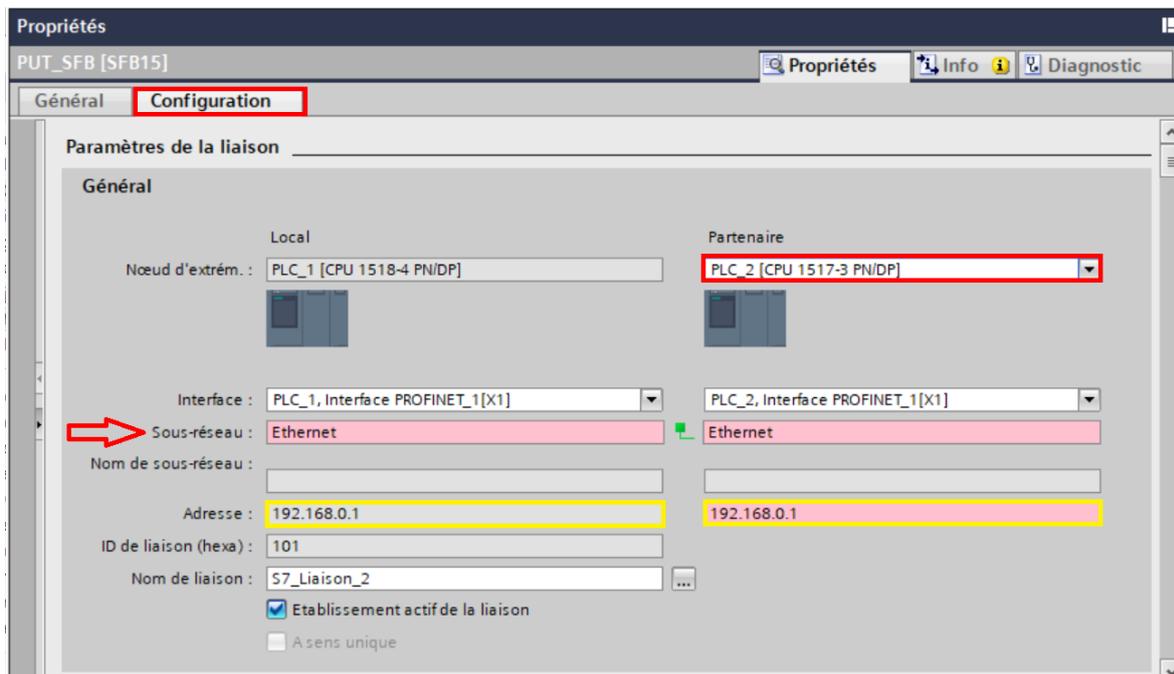


Dans le bloc main OB1 de votre première CPU; allez dans « communications » et faites glisser le bloc « PUT » dans le premier réseau.



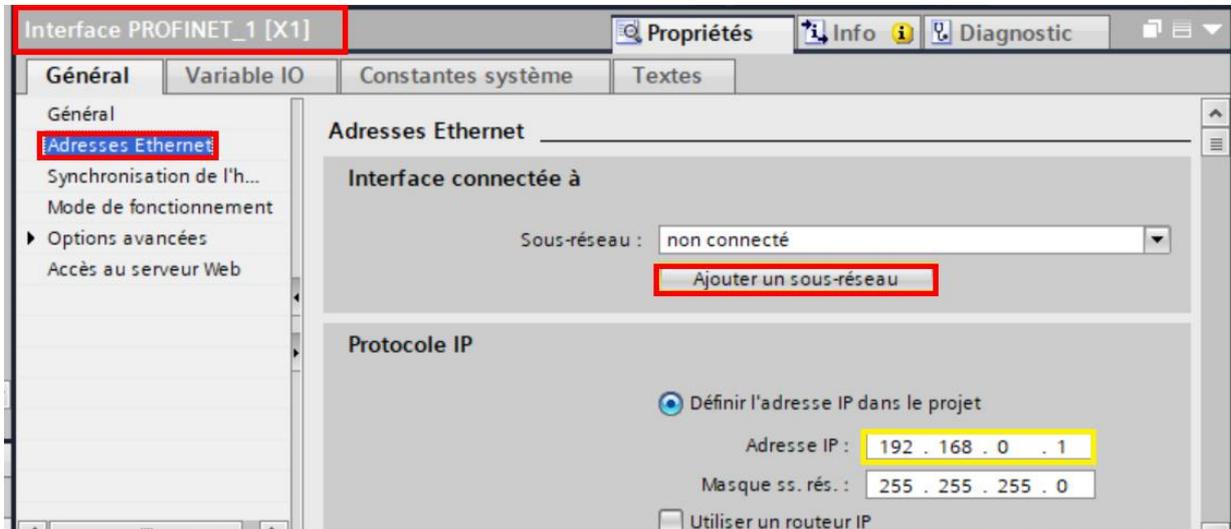
## II.1.2 Etape 2

Sélectionnez le bloc « PUT » pour accéder à ses propriétés; puis allez dans configuration. Enfin dans partenaire, sélectionnez la deuxième CPU.

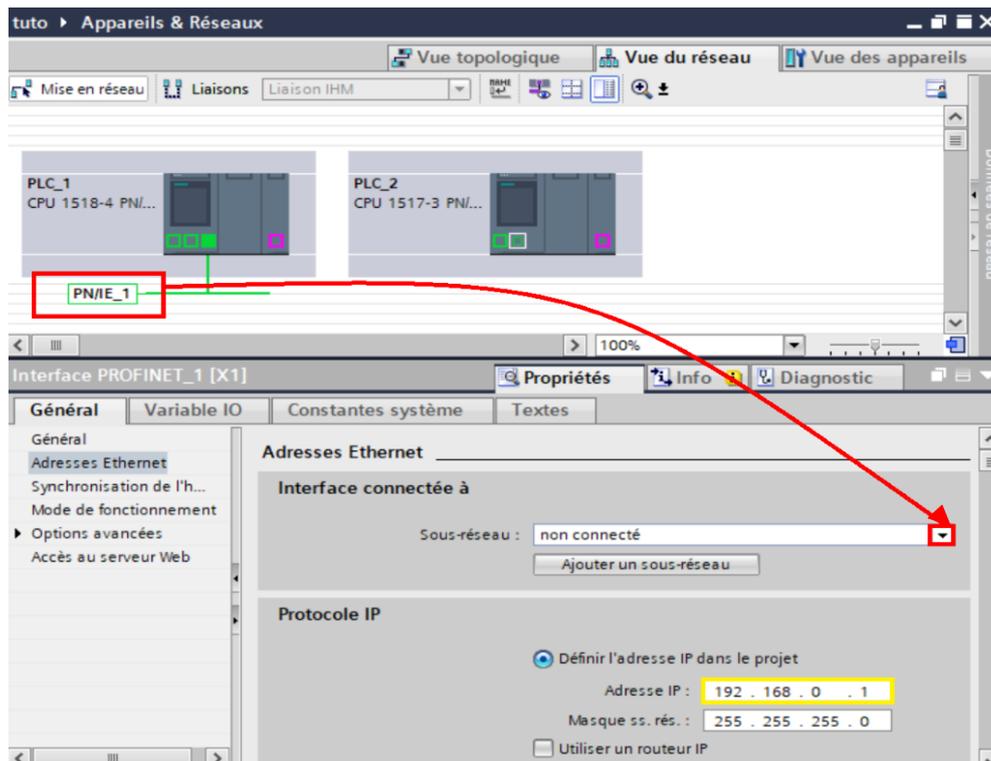


Arrivée à ce niveau il faudra définir un « sous réseau » qui doit être le même pour les deux CPU ; pour ce faire allez dans « appareils et réseaux ».

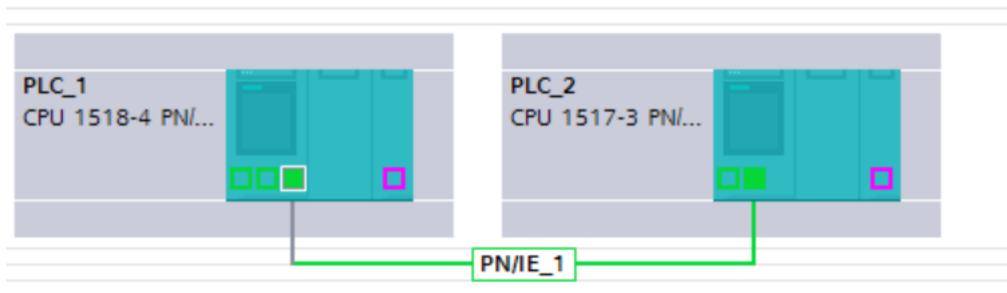
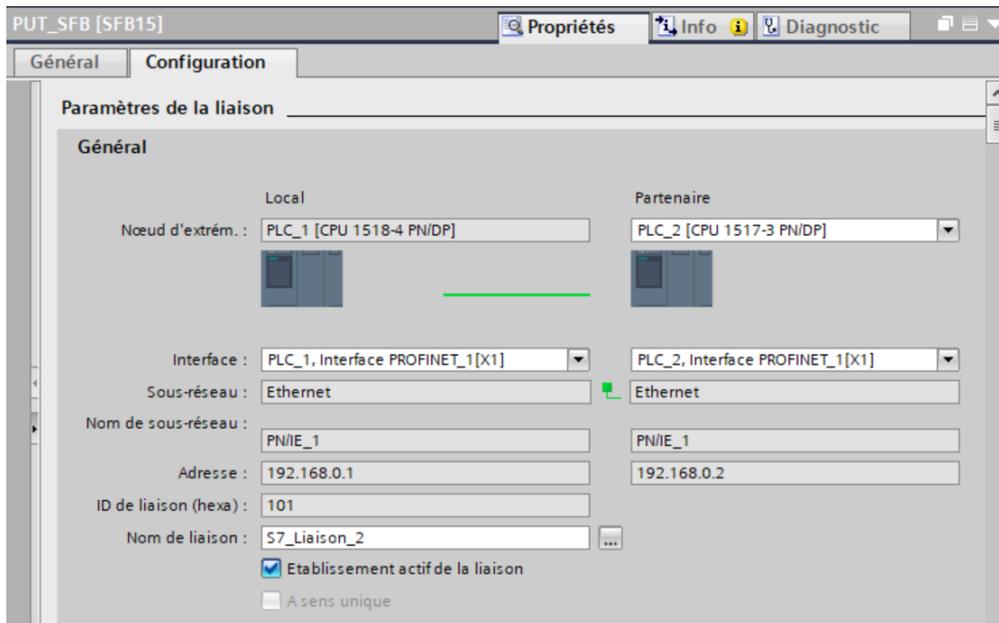
- sur la première CPU, sélectionnez l'interface PROFINET dont l'adresse est la même que celle encadré en jaune dans les paramètres de la liaison (figure précédente) ; puis allez « adresses Ethernet » et cliquez sur « ajouter un sous-réseau ».



- Sur la deuxième CPU ; sélectionnez l'interface PROFINET dont l'adresse est la même que celle défini dans les paramètres de la liaison; puis allez dans « adresses Ethernet » et sélectionnez le même sous-réseau défini pour la première CPU.



Vous pouvez désormais commencer le partage de données entre les deux CPU.



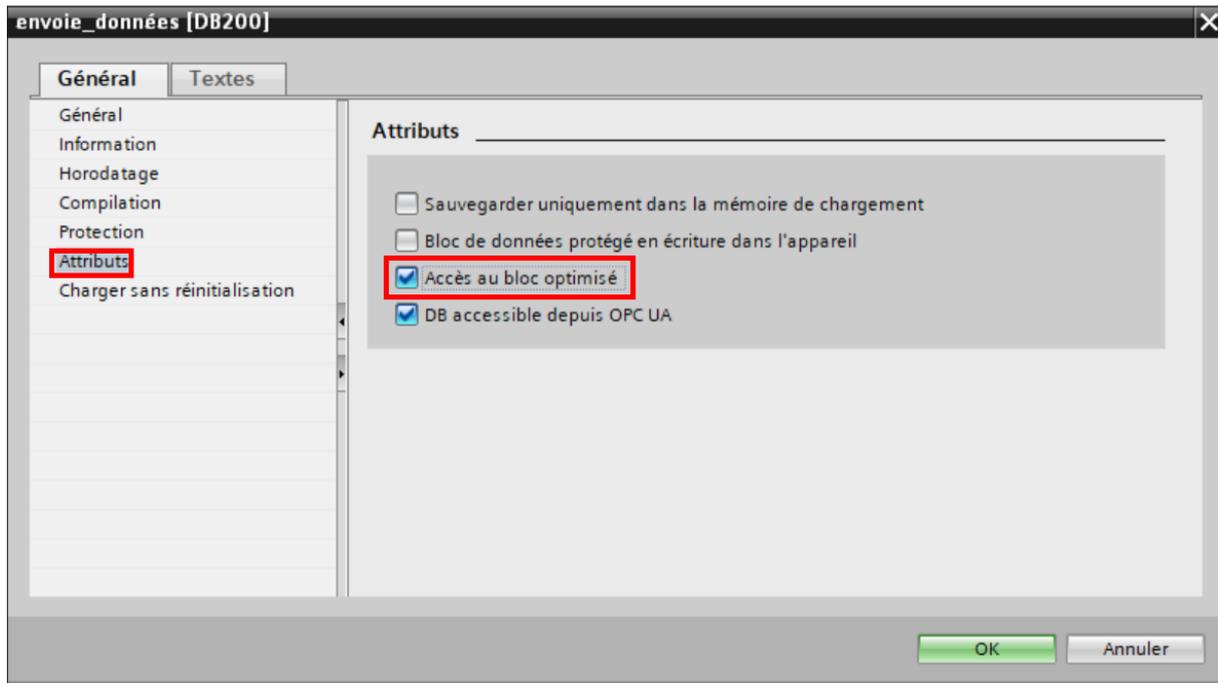
### II.1.3 Etape 3

Pour pouvoir échanger les données entre les deux CPU, il faudra définir des blocs de données globaux (DB) dans chaque CPU. Les DB doivent avoir une structure identique.

Pour la première CPU ; on a créé un DB « envoie\_données » comme suite :

envoie_données								
	Nom	Type de données	Valeur de départ	Rémanence	Accessible ...	Ecritu...	Visible da...	Val...
1	Static			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	bool 1	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	bool 2	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	entier	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

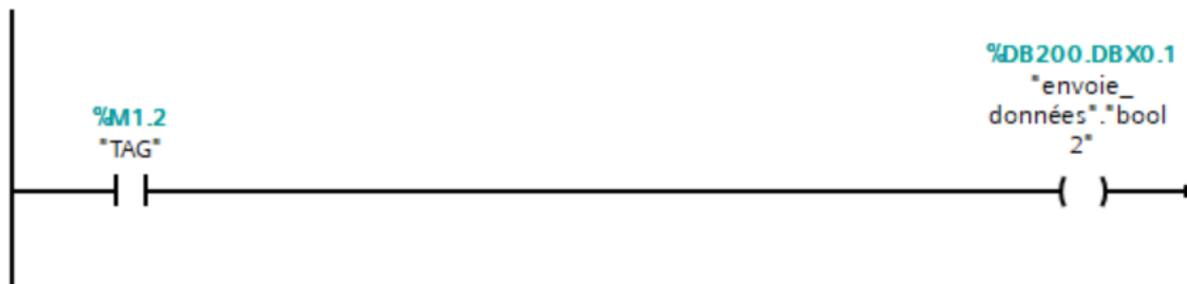
Allez dans les propriétés de chaque DB et désactivez « Accès au bloc optimisé ».



Vous devez reproduire les mêmes étapes pour la création d'un DB dans la seconde CPU. (Création du DB100 « recoie\_données »).

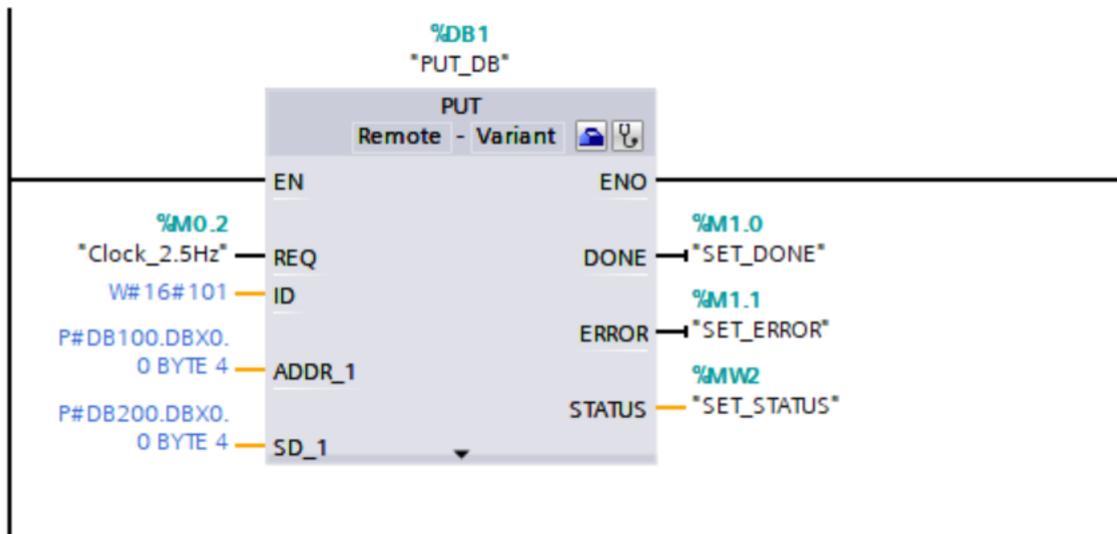
#### II.1.4 Etape 4

Enfin pour envoyer une variable spécifique de la première CPU vers la seconde ; créez un nouveau réseau simple où vous affecté la variable que vous voulez partager a une variable de votre DB, cela ce fait comme dans la figure ci-dessous.

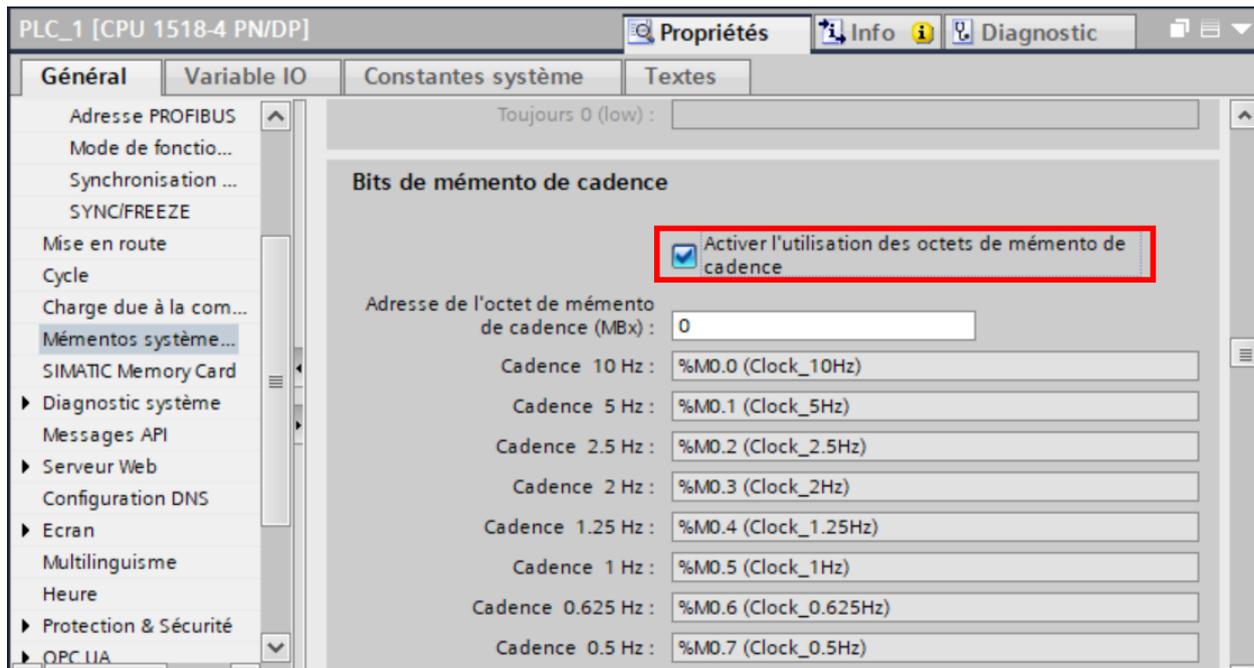


## II.1.2 Etape 5

Retournez dans le bloc main de votre première CPU, et définissez les entrées et sorties du bloc « PUT » comme suite :



- L'entrée REQ représente l'entrée de commande c'est-à-dire que l'instruction PUT est exécutée à chaque front montant de REQ.
- ID est le paramètre d'adressage de la liaison avec la CPU partenaire.
- SD\_1 désigne la zone d'émission des données, dans notre cas c'est le DB200 de la première CPU « envoie\_données ».
- ADDR\_1 désigne la zone de réception des données transmises, dans notre cas c'est le DB100 de la deuxième CPU « recoie\_données »

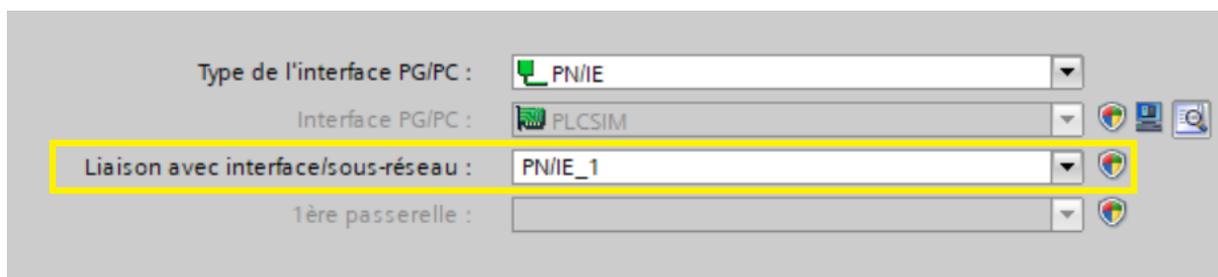


Dans notre exemple on a affecté une variable un memento de cadence pour l'activation du bloc « PUT ». Pour activer les mementos de cadence d'une CPU ; sélectionnez-la puis dans général, allez a « mementos systèmes » et activer l'utilisation des octets de memento de cadence

## II.2 La simulation

Avant de démarrer la simulation il faut compiler puis charger le programme.

Choisissez la liaison PN/IE qui fait la communication entre les deux CPU.



Une fois la simulation lancée ; allez dans le DB de la CPU qui envoie les données (dans notre cas envoie\_données de la CPU1) et modifier les valeurs

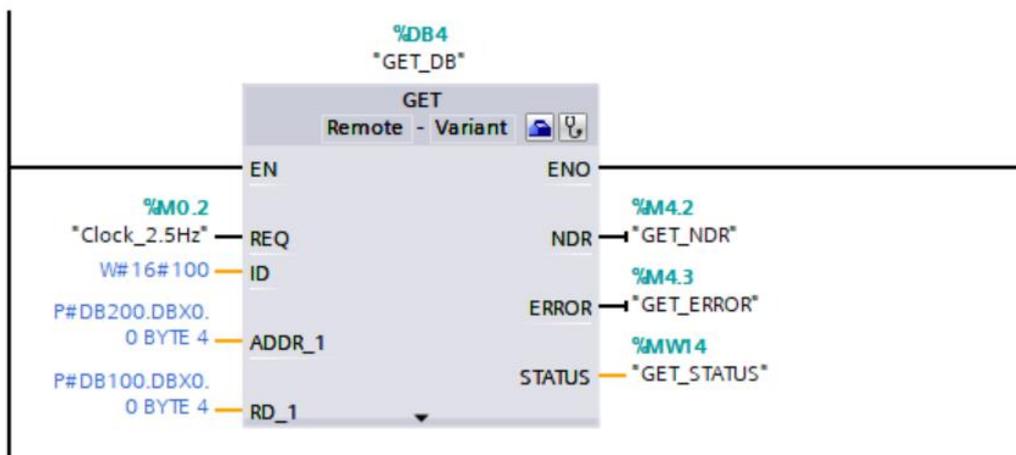
	Nom	Type de données	Décalage	Valeur de départ	Valeur de visualisati.	Rémanence	Accessi...
1	Static					<input type="checkbox"/>	<input type="checkbox"/>
2	bool 1	Bool	0.0	false	TRUE	<input type="checkbox"/>	<input checked="" type="checkbox"/>
3	bool 2	Bool	0.1	false	FALSE	<input type="checkbox"/>	<input checked="" type="checkbox"/>
4	entier	Int	2.0	0	70	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Allez dans la CPU partenaire (CPU qui reçoit les données) ; sélectionnez son DB de réception (dans notre cas « recoie\_données ») et vérifiez si la réception a eu lieu

	Nom	Type de données	Décalage	Valeur de départ	Valeur de visualisati.	Rémanence	Accessi...
1	Static					<input type="checkbox"/>	<input type="checkbox"/>
2	bool 1	Bool	0.0	false	TRUE	<input type="checkbox"/>	<input checked="" type="checkbox"/>
3	bool 2	Bool	0.1	false	FALSE	<input type="checkbox"/>	<input checked="" type="checkbox"/>
4	entier	Int	2.0	0	70	<input type="checkbox"/>	<input checked="" type="checkbox"/>

## II.3 La réception de données (instruction GET)

Pour l'envoi de données depuis la seconde CPU vers la première, il faudra utiliser le bloc « GET », et reproduire les même étapes à partir de l'étape 3.



Le bloc « GET » est disposé dans le main de la première CPU et non dans le main de la CPU partenaire

- L'entrée REQ représente l'entrée de commande c'est-à-dire que l'instruction GET est exécutée à chaque front montant de REQ.
- ID est le paramètre d'adressage de la liaison (la même liaison utilisé pour le bloc SET)
- RD\_1 : Pointeur désignant les zones de la propre CPU où stocker les données lues.
- ADDR\_1 : Pointeur désignant les zones à lire dans la CPU partenaire.

# Annexe C

## Fiches techniques

## I CPU 1214C DC/DC/DC [11]

SIMATIC S7-1200, CPU 1214C, CPU compacte CC/CC/CC, I/O intégrées: 14 entrées TOR 24V CC; 10 sorties TOR 24V CC; 2 AI 0- 10V CC, alimentation: CC 20,4-28,8V CC, mémoire de programme / de données 100Ko



Informations générales	
Désignation du type de produit	CPU 1214C CC/CC/CC
Version du firmware	V4.2
Ingénierie avec	
<ul style="list-style-type: none"> <li>• Pack de programmation</li> </ul>	à partir de STEP 7 V14
Tension d'alimentation	
Valeur nominale (CC)	
<ul style="list-style-type: none"> <li>• 24 V CC</li> </ul>	Oui
Plage admissible, limite inférieure (CC)	20,4 V
Plage admissible, limite supérieure (CC)	28,8 V
Protection contre l'inversion de polarité	Oui
Tension de charge L+	
<ul style="list-style-type: none"> <li>• Valeur nominale (CC)</li> </ul>	24 V
<ul style="list-style-type: none"> <li>• Plage admissible, limite inférieure (CC)</li> </ul>	20,4 V
<ul style="list-style-type: none"> <li>• Plage admissible, limite supérieure (CC)</li> </ul>	28,8 V
Courant d'entrée	
Consommation (valeur nominale)	500 mA; uniquement CPU
Consommation, maxi	1 500 mA; CPU avec tous les modules d'extension
Courant d'appel, maxi	12 A; sous 28,8 V
I <sup>2</sup> t	0,5 A <sup>2</sup> -s
Courant de sortie	
pour bus interne (5 V CC), max.	1 600 mA; max. 5 V CC pour SM et CM

Alimentation des capteurs	
Alimentation des capteurs 24 V	
• 24 V	L+ moins 4 V CC min.
Puissance dissipée	
Puissance dissipée, typ.	12 W
Mémoire	
Mémoire de travail	
• Intégré	100 kbyte
• extensible	Non
Mémoire de chargement	
• Intégré	4 Mbyte
• enfichable (SIMATIC Memory Card), max.	Carte mémoire SIMATIC
Sauvegarde	
• présente	Oui
• sans maintenance	Oui
• sans pile	Oui
Temps de traitement CPU	
pour opérations sur bits, typ.	0,08 µs; / instruction
pour opérations sur mots, typ.	1,7 µs; / instruction
pour opérations à virgule flottante, typ.	2,3 µs; / instruction
CPU-blocs	
Nombre de blocs (total)	DB, FC, FB, compteurs et temporisations Le nombre maximal de blocs va de 1 à 65535. Il n'y a pas de limitations ; utilisation de l'ensemble de la mémoire de travail.
OB	
• Nombre, maxi	Limité uniquement par la mémoire de travail pour le code
Plage d'adresses	
Mémoire image du processus	
Configuration matérielle	
Nombre de modules par système, maxi	3 modules de communication, 1 Signal Board, 8 modules d'entrées-sorties
Entrées TOR	
Nombre d'entrées TOR	14; intégré
• dont entrées utilisables pour les fonctions technologiques	6; HSC (compteur rapide)
Type M/P	Oui
Nombre d'entrées activables simultanément	
Toutes les positions de montage	
— jusqu'à 40 °C, maxi	14
Tension d'entrée	
• Valeur nominale (CC)	24 V
• pour état log. "0"	5 V CC à 1 mA
• pour état log. "1"	15 V CC à 2,5 mA

<b>Retard d'entrée (pour valeur nominale de la tension d'entrée)</b>	
pour entrées standard	
— paramétrable	0,2 ms, 0,4 ms, 0,8 ms, 1,6 ms, 3,2 ms, 6,4 ms et 12,8 ms, sélectionnable par groupe de 4
— pour "0" vers "1", mini	0,2 ms
— pour "0" vers "1", maxi	12,8 ms
pour entrées d'alarme	
— paramétrable	Oui
pour fonctions technologiques	
— paramétrable	monophasé : 3 @ 100 KHz & 3 @30kHz, différentiel : 3 @ 80 kHz & 3 @30kHz
<b>Longueur de câble</b>	
• blindé, maxi	500 m; 50 m pour les fonctions technologiques
• non blindé, max.	300 m; pour fonctions technologiques : Non
<b>Sorties TOR</b>	
Nombre de sorties TOR	10
• dont les sorties rapides	4; Sortie de trains d'impulsions 100 KHz
Limitation de la tension de coupure inductive à	L+ (-48 V)
<b>Pouvoir de coupure des sorties</b>	
• pour charge résistive, max.	0,5 A
• pour charge de lampes, maxi	5 W
<b>Tension de sortie</b>	
• pour état log. "0", max.	0,1 V; avec charge 10 kohm
• pour état log. "1", mini	20 V
<b>Courant de sortie</b>	
• pour état log. "1" valeur nominale	0,5 A
• pour état log. "0" courant résiduel, maxi	0,1 mA
<b>Temps de retard de sortie pour charge ohmique</b>	
• pour "0" vers "1", maxi	1 µs
• pour "1" vers "0", max.	5 µs
<b>Fréquence de commutation</b>	
• des sorties d'impulsions, pour charge résistive, maxi	100 kHz
<b>Sorties relais</b>	
• Nombre de sorties à relais	0
<b>Longueur de câble</b>	
• blindé, maxi	500 m
• non blindé, max.	150 m
<b>Entrées analogiques</b>	
Nombre d'entrées analogiques	2
<b>Etendues d'entrée</b>	
• Tension	Oui
<b>Etendues d'entrée (valeurs nominales), tensions</b>	
• 0 à +10 V	Oui
— Résistance d'entrée (0 à 10 V)	≥100 kOhm
<b>Longueur de câble</b>	
• blindé, maxi	100 m; torsadé et blindé

1. Interface	
Type d'interface	PROFINET
Physique	Ethernet
avec séparation galvanique	Oui
Détermination automatique de la vitesse de transmission	Oui
Autonégociation	Oui
Autocrossing	Oui
Réalisation physique de l'interface	
• Nombre de ports	1
• Commutateur intégré	Non
Protocoles	
• Automate PROFINET IO	Oui
• Périphérique PROFINET IO	Oui
• Communication SIMATIC	Oui
• Communication IE ouverte	Oui
• Serveur Web	Oui
• Redondance des média	Non
Automate PROFINET IO	
• Vitesse de transmission, maxi	100 Mbit/s
Services	
— Communication PG/OP	Oui
— Routage S7	Oui
— Mode synchrone	Non
— IRT	Non
— MRP	Non
— MRPD	Non
— PROFIenergy	Non
— Démarrage prioritaire	Oui
— Nombre de périphériques IO avec démarrage priorisé, max.	16
— Nombre de périphériques IO raccordables, max.	16
— Nombre de périphériques d'E/S raccordables pour RT, maxi	16
— dont en ligne, maxi	16
— Activation/Désactivation de périphériques d'E/S	Oui
— Nombre de périphériques IO activables/désactivables simultanément, maxi	8
— Temps de rafraîchissement	La valeur min. de temps d'actualisation dépend du jeu de composants de communication pour PROFINET IO, du nombre de périphériques IO et de la quantité de données utilisateur configuré.

## II SM 1223, DI 16x24 V CC, DQ 16x24 V CC [12]

SIMATIC S7-1200, E/S TOR SM 1223, 16DI/16DO, 16 entrées TOR 24V CC, Sink/Source, 16DO, transistor 0,5A



Informations générales	
Désignation du type de produit	SM 1223, DI 16x24 V CC, DQ 16x24 V CC
Tension d'alimentation	
Valeur nominale (CC)	24 V
Plage admissible, limite inférieure (CC)	20,4 V
Plage admissible, limite supérieure (CC)	28,8 V
Courant d'entrée	
sur bus interne 5 V CC, maxi	185 mA
Entrées TOR	
<ul style="list-style-type: none"> <li>sur tension de charge L+ (sans charge), maxi</li> </ul>	4 mA; par voie
Tension de sortie	
Tension d'alimentation des transmetteurs	
<ul style="list-style-type: none"> <li>présente</li> </ul>	Oui
Puissance dissipée	
Puissance dissipée, typ.	4,5 W
Entrées TOR	
Nombre d'entrées TOR	16
<ul style="list-style-type: none"> <li>par groupes de</li> </ul>	2
Caractéristique d'entrée selon CEI 61131, type 1	Oui

<b>Nombre d'entrées activables simultanément</b>	
Toutes les positions de montage	
— jusqu'à 40 °C, maxi	16
Montage horizontal	
— jusqu'à 40 °C, maxi	16
— jusqu'à 50 °C, maxi	16
Montage vertical	
— jusqu'à 40 °C, maxi	16
<b>Tension d'entrée</b>	
• Type de tension d'entrée	CC
• Valeur nominale (CC)	24 V
• pour état log. "0"	5 V CC à 1 mA
• pour état log. "1"	15 V CC à 2,5 mA
<b>Courant d'entrée</b>	
• pour état log. "0", max. (courant de repos admissible)	1 mA
• pour état log. "1", mini	2,5 mA
• pour état log. "1", typ.	4 mA
<b>Retard d'entrée (pour valeur nominale de la tension d'entrée)</b>	
pour entrées standard	
— paramétrable	Oui; 0,2 ms, 0,4 ms, 0,8 ms, 1,6 ms, 3,2 ms, 6,4 ms et 12,8 ms, sélectionnable par groupe de 4
pour entrées d'alarme	
— paramétrable	Oui
<b>Longueur de câble</b>	
• blindé, maxi	500 m
• non blindé, max.	300 m
<b>Sorties TOR</b>	
Nombre de sorties TOR	
• par groupes de	1
Protection contre les courts-circuits	
	Non; à prévoir en externe
Limitation de la tension de coupure inductive à	
	L+ (-48 V)
<b>Pouvoir de coupure des sorties</b>	
• pour charge résistive, max.	0,5 A
• pour charge de lampes, maxi	5 W
<b>Tension de sortie</b>	
• Valeur nominale (CC)	24 V
• pour état log. "0", max.	0,1 V; avec charge 10 kohm
• pour état log. "1", mini	20 V CC
<b>Courant de sortie</b>	
• pour état log. "1" plage admissible, maxi	0,5 A
• pour état log. "0" courant résiduel, maxi	10 µA
<b>Temps de retard de sortie pour charge ohmique</b>	
• pour "0" vers "1", maxi	50 µs
• pour "1" vers "0", max.	200 µs
<b>Courant total des sorties (par groupe)</b>	
Montage horizontal	
— jusqu'à 50 °C, maxi	8 A; Courant par masse

Sorties relais	
Pouvoir de coupure des contacts	
— pour charge inductive, maxi	0,5 A
— pour charge de lampes, maxi	5 W
— pour charge résistive, max.	0,5 A
Longueur de câble	
• blindé, maxi	500 m
• non blindé, max.	150 m
Alarmes/diagnostic/information d'état	
Alarmes	
• Alarme de diagnostic	Oui
Signalisation de diagnostic par LED	
• pour l'état des entrées	Oui
• pour l'état des sorties	Oui
Séparation galvanique	
Séparation galvanique entrées TOR	
• entre les voies, par groupes de	2
Séparation galvanique sorties TOR	
• entre les voies, par groupes de	1
• entre voies et bus interne	500 V CA
Degré et classe de protection	
Indice de protection IP	IP20
Normes, homologations, certificats	
Marquage CE	Oui
Homologation CSA	Oui
Homologation UL	Oui
cULus	Oui
Homologation FM	Oui
RCM (anciennement C-TICK)	Oui
Homologation KC	Oui
Agrément pour constructions navales	Oui

# Annexe D

## Graficet de l'ascenseur

## I. Grafcet global du fonctionnement de l'ascenseur

Sur la base de la description donnée dans le cahier des charges ; on tire le grafcet global du fonctionnement de l'ascenseur.

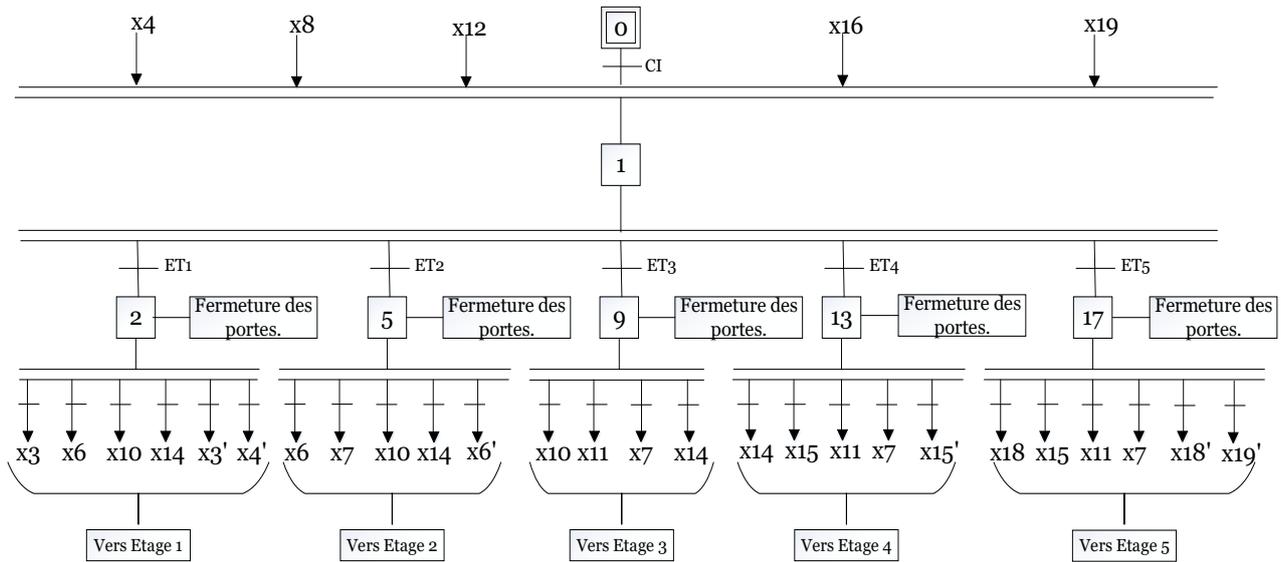


Figure D.1 le grafcet global

À l'état 1, l'ascenseur prend en compte les demandes des usagers, et en fonction de ces dernières il choisit la destination à atteindre.

Afin de mieux comprendre le fonctionnement global de l'ascenseur et les conditions d'accès à chaque étage ; le grafcet global est divisé en 5 grafcets « Vers étage  $i$  ».

## II. Descendre vers le premier étage

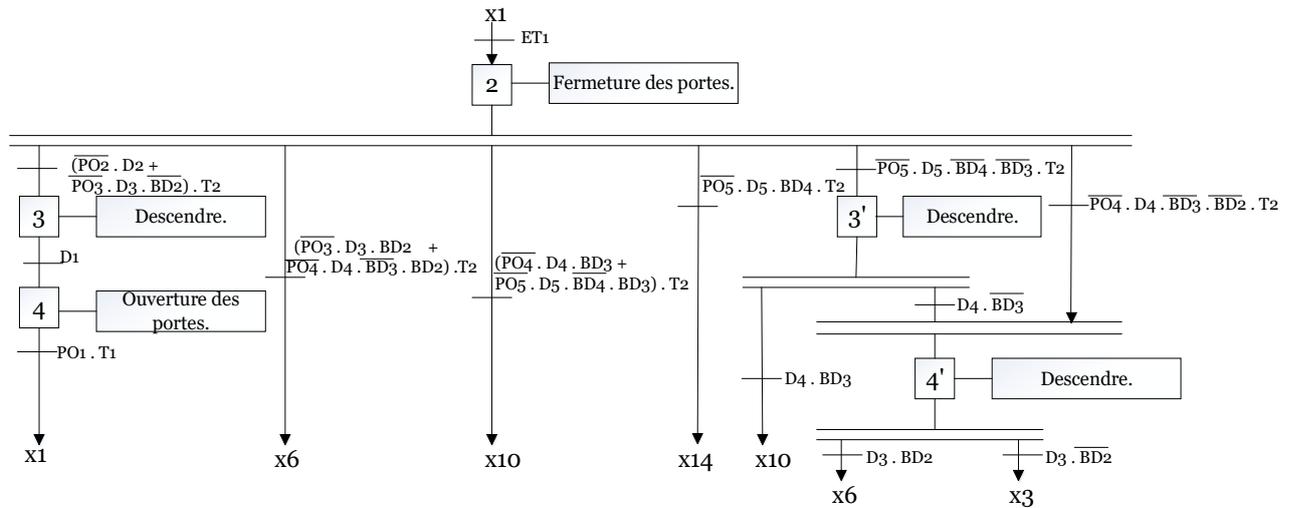


Figure D.2 : grafcet de l'ascenseur « vers étage 1 ».

## III. Vers le deuxième étage

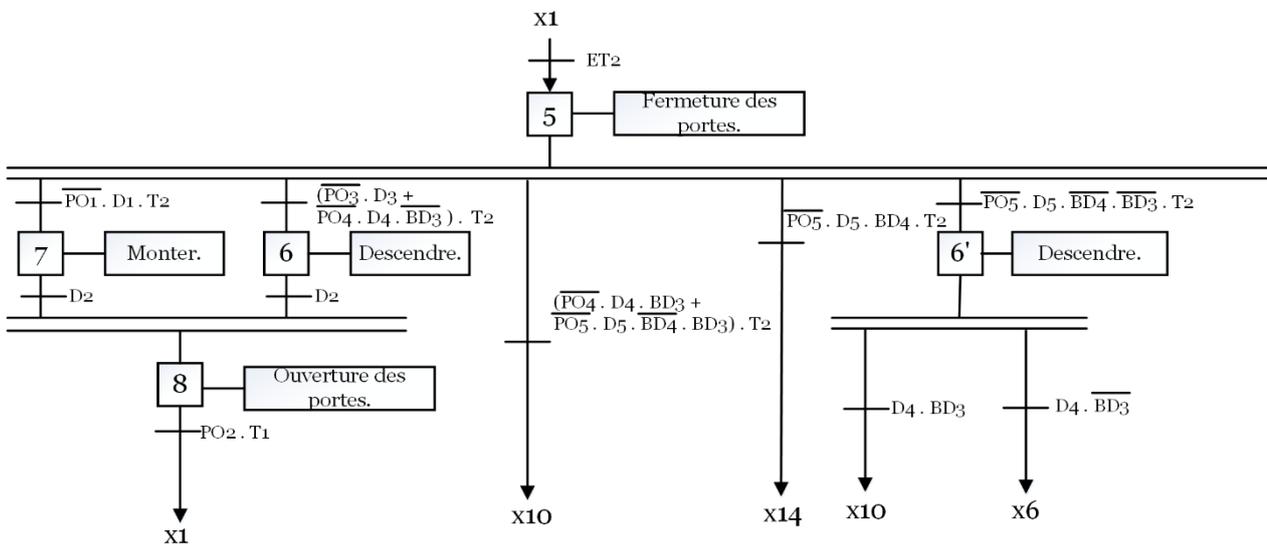


Figure D.3 : grafcet de l'ascenseur « vers étage 2 ».

#### IV. Vers le troisième étage

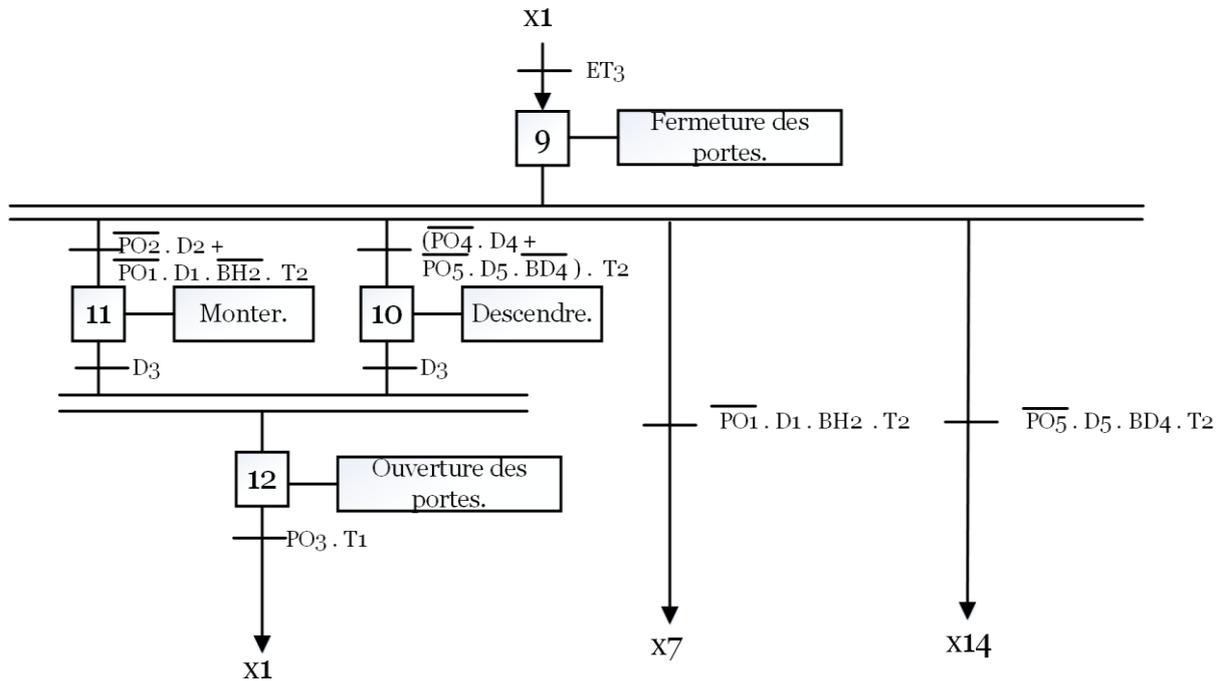


Figure D.4 : grafcet de l'ascenseur « vers étage 3 ».

#### V. Vers le quatrième étage

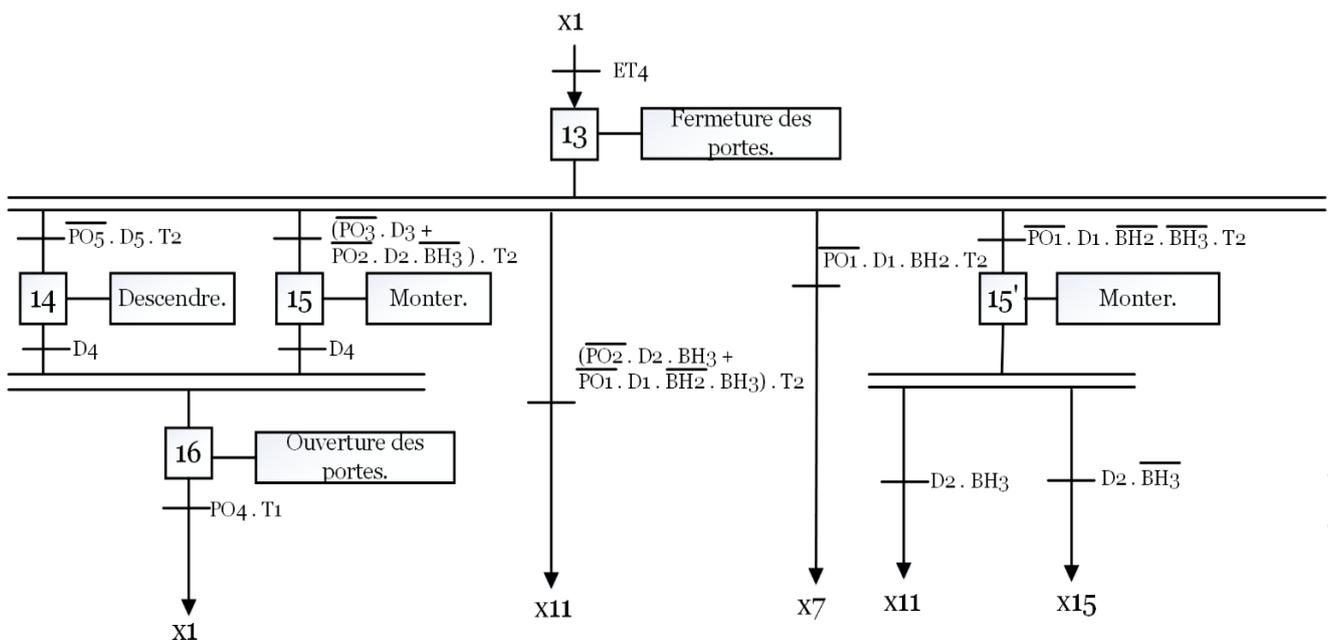


Figure D.5 : grafcet de l'ascenseur « vers étage 4 ».

## VI. Monter vers le cinquième étage

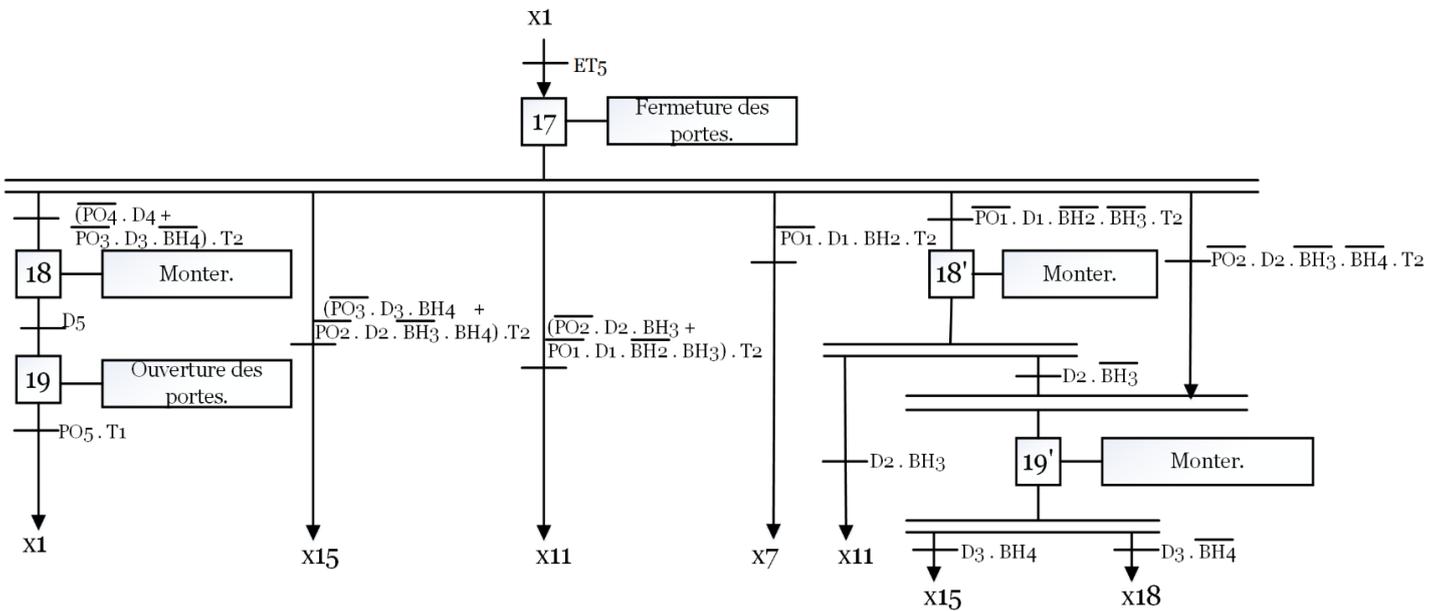


Figure D.6 : grafcet de l'ascenseur « vers étage 5 ».

Le grafcet de l'ascenseur représente des analogies entre l'étage 1 et l'étage 5, ainsi qu'entre les étages 2 et 4.