

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mohamed El Bachir El Ibrahimi - Bordj Bou Arreridj –
Faculté des Sciences et de la Technologie
- Département d'Électronique -



Université Mohamed El Bachir El Ibrahimi - B.B.A -

Mémoire

Présenté pour obtenir

LE DIPLOME DE LICENCE

Filière : Electronique

Spécialité : Electronique Industrielle

Par :

- **BENREKIA HOCINE**
- **BOUAMAMA ABDARAOUF**
- **TOUTI ABDERRAHMANE**



Université Mohamed El Bachir El Ibrahimi - B.B.A -

Intitulé

Automatisation d'une Porte Coulissante Programmation avec STEP 7

Évalué le :

Par la commission d'évaluation composée de :

<i>Nom & Prénom</i>	<i>Grade</i>	<i>Qualité</i>	<i>Etablissement</i>
<i>M. A. Laouamri</i>	...	<i>Président</i>	<i>Univ-BBA</i>
<i>M. S. Bendib</i>	...	<i>Encadreur</i>	<i>Univ-BBA</i>
<i>M. Y. Belhadad</i>	...	<i>Examineur</i>	<i>Univ-BBA</i>

Remerciements

Avant tous nous remercions « Allah » de nous avoir donné la force et le courage pour réaliser ce Projet.

Nous tenons à exprimer nos vifs remerciements à notre encadreur Dr. S.Bendib de nous avoir encadré notre projet de fin d'études.

Nous tenons aussi à remercier chaleureusement les membres de jury pour l'honneur qu'ils nous ont fait en acceptant d'évaluer notre projet.

Nous exprimons également notre gratitude à tous les Enseignants de spécialité électronique industrielle MCIL.

Nous présentons enfin, notre profonde gratitude à nos parents et toute la famille, ainsi qu'à tous ceux qui nous ont aidés de près ou de loin à réaliser ce travail.

Résumé

Ce projet a pour objectif de présenter une idée générale de l'**automatisation** à l'aide d'un **automate programmable**.

Premièrement, l'**automatisation d'une porte coulissante** dans notre projet est un processus conçu pour ouvrir et fermer automatiquement cette porte via un **automate programmable SIMATIC S7-300**.

Deuxièmement, un **programme en langage à contacts** doit être créé à l'aide du logiciel **STEP 7**, puis on charge ce programme dans l'**automate** pour envoyer des commandes à l'**actionneur** qui déplace la porte (**moteur à courant continu**) en fonction des **données d'entrées** des **capteurs** et du **bouton poussoir**.

Finalement, Pour valider ce système on utilise **S7-PLCSIM**, qui est un simulateur d'**API** inclus dans logiciel **STEP 7**.

Abstract

The purpose of this project is to present a general idea of **automation** using **Programmable logic controller**.

Firstly, **automation of sliding door** in our project is a process designed to open and close this door automatically using the **SIMATIC S7-300 PLC**.

Secondly, a **ladder diagram program** must be created using **STEP 7** software, and then download it to the **PLC** to send commands to the **actuator (DC motor)** which moves the door according to the **digital input data** of the **sensors** and **push-button**.

Lastly, to validate this system we use **S7-PLCSIM**, which is a **PLC** simulator included in the **STEP 7** software.

المخلص

يهدف هذا المشروع الى تقديم فكرة عامة عن الاتمته باستعمال المبرج الالي. في البداية, ان اتمته الباب المنزلق في مشروعنا هو عبارة عن عملية مصممة لفتح و غلق هذا الباب اوتوماتيكيا باستعمال المبرج الالي سيمنس س7-300.

يجب انشاء كود بلغة السلم باستعمال البرنامج ستاب 7, ثم نقله نحو المبرمج الالي لارسال اوامر الى المشغل الذي يحرك الباب (المحرك ذو التيار المستمر) وفق البيانات الرقمية لاجهزة الاستشعار و الزر الضاغط. في الاخير, لتشغيل و اختبار هذا النظام نستخدم محاكي المبرمج الالي الموجود في برنامج ستاب 7.

Sommaire

	Page
Introduction générale	1
I-Partie Théorique :	
Chapitre I : l'automatisation de la porte coulissante	
I-1 Introduction.....	2
I-2 la porte coulissante automatique.....	2
I-3 Structure d'un système automatisé.....	3
I-4 les équipements électriques et électroniques.....	4
I-4-1 les boutons.....	4
I-4-2 les voyants lumineux.....	5
I-4-3 les capteurs.....	5
I-4-3-1 les interrupteurs de position.....	6
I-4-3-2 les détecteurs de proximité photoélectrique.....	6
I-4-4 Relais électromécanique.....	8
I-4-5 Moteur à courant continu.....	8
I-4-6 Automates programmables industriels.....	9
I-4-6-1 Définition de l'API.....	9
I-4-6-2 Structure des APIs.....	10
I-4-6-3 Programmation des APIs.....	11
I-4-6-4 Fonctionnement des APIs.....	12
I-4-6-5 Choix d'un API.....	12
I-4 Conclusion.....	12
Chapitre II : l'automate SIMATIC S7-300 et logicielle STEP 7	
II-1 Introduction.....	13
II-2 l'automate SIMATIC S7-300.....	13
II-2-1 Caractéristiques de l'automate S7-300.....	13
II-2-2 Modularité du S7-300.....	14
II-3 Programmation STEP 7.....	17
II-3-1 Définitions du logiciel STEP7.....	17
II-3-2 L'adressage absolu des modules de signaux.....	18
II-3-3 Création d'un projet STEP7.....	19
II-4 Simulation S7-PLCSIM.....	23
II-4-1 Présentation du S7-PLCSIM.....	23
II-4-2 Mise en route du logiciel S7-PLCSIM.....	24
II-5 Conclusion.....	26
II-Partie Pratique :	
I-1 Présentation du cahier de charge.....	27
I-1-1 Liste d'équipements.....	27
I-1-2 Description de cycle du fonctionnement.....	28
I-2 Réalisation hardware et software.....	29
I-3 Simulation.....	35
Conclusion générale	38

Liste des Figures :

	Page
I-Partie Théorique :	
Figure I-1 : Porte coulissante automatique à deux vantaux.....	2
Figure I-2 : Système de motorisation.....	2
Figure I-3 : Structure d'un système automatisé.....	3
Figure I-4 : Grandeur physique et signal d'un capteur.....	5
Figure I-5 : Principe d'un détecteur de proximité photoélectrique.....	6
Figure I-6 : Système détection par barrage.....	7
Figure I-7 : Système détection par réflex.....	7
Figure I-8 : Système détection par proximité.....	7
Figure I-9 : Schéma principe d'un relais électromécanique.....	8
Figure I-10 : Circuit H-Bridge pour un seul moteur DC.....	8
Figure I-11 : Circuit d'un générateur de signal PWM.....	9
Figure I-12 : Compact API (LOGO SIEMENS).....	10
Figure I-13 : Modulaire API (SIMATIC S7-300).....	10
Figure I-14 : Structure générale d'un automate programmable.....	11
Figure I-15 : Principe du langage à contacts.....	11
Figure I-16 : Principe de fonctionnement d'un automate programmable.....	12
Figure II-1: Automate SIMATIC S7-300.....	13
Figure II-2 : Disposition des modules de l'API S7-300.....	14
Figure II-3 : Différentes module d'alimentation SIMATIC S7-300.....	14
Figure II-4 : CPU SIMATIC S-300.....	15
Figure II-5 : Câble de programmation pour automate S7-300.....	15
Figure II-6 : Micro carte mémoire SIMATIC.....	16
Figure II-7 : Vue d'ensemble de l'automatisme.....	17
Figure II-8 : Exemple d'adressage absolu d'un module TOR.....	18
Figure II-9 : SIMATIC Manager qui lance l'assistant de STEP7.....	19
Figure II-10 : Assistant de STEP 7 : nouveau projet.....	19
Figure II-11 : Fenêtre de choix de la CPU.....	20
Figure II-12 : Choix des Blocs à utilisés et de langage.....	20
Figure II-13 : Nomination du projet.....	21
Figure II-14 : Répertoire de la station SIMATIC et de la CPU.....	21
Figure II-15 : Configuration du matériel.....	22
Figure II-16 : La table de mnémoniques (pour activer et désactiver d'un voyant).....	22
Figure II-17 : Création du programme (langage à contacts, pour activer et désactiver d'un voyant.).....	23
Figure II-18 : Fenêtre du S7-PLCSIM.....	24
Figure II-19 : Simulateur S7-PLCSIM.....	25
Figure II-20 : Visualisation de l'état du programme (activer et désactiver un moteur DC).....	26

II-Partie Pratique :

Figure I-1 : Porte coulissante automatique à deux vantaux.....	28
Figure I-2 : Schéma électrique circuit commande.....	29
Figure I-3 : Schéma électrique circuit puissance.....	29
Figure I-4 : Configuration matérielle.....	31
Figure I-5 : Table des mnémoniques.....	31
Figure I-6 : Réseau 1, l'ouverture de la porte.....	32
Figure I-7 : Réseau 2, la temporisation 5secondes.....	33
Figure I-8 : Réseau 3, fermeture de la porte.....	34
Figure I-9 : Réseau 4, cas d'urgence.....	34
Figure I-10 : Simulation d'ouverture de porte.....	35
Figure I-11 : Simulation de temporisation 5s.....	35
Figure I-12 : Simulation de fermeture de porte.....	36
Figure I-13 : Simulation de la porte qui est complètement fermée.....	37
Figure I-14 : Simulation du cas d'urgence.....	37

Liste des Tableaux :

Page

I-Partie Théorique :

Tableau I-1 : Les types de contact selon CEI.....4

Tableau I-2 : Les symboles de boutons selon CEI.....4

Tableau I-3 : Code de couleur normalisé pour les voyants lumineux selon CEI.....5

Tableau I-4 : Les symboles d'interrupteur de position selon la norme CEI.....6

II-Partie Pratique :

Tableau I-1 : La liste des équipements.....27

Tableau I-2 : Symbole des équipements.....30

Liste des abréviations :

API : Automate Programmable Industriel.
PLC : Programmable logic controller.
TOR : Tout Ou Rien.
NO : Normalement ouvert.
NF : Normalement fermé.
CEI : Commission Electrotechnique Internationale.
PWM : Pulse Width Modulation.
DC : Direct Current.
AC : Alternative current.
PS : Power Supply.
CPU : Central Processing Unit.
MPI : Multi Point Interface.
PG : Programming Console.
SM : Signal Modules.
LIST : Langage liste.
LOG : langage logigramme.
CONT : Langage contact.
OB : Bloc d'Organisation.
FB : Bloc fonctionnel.
DB : Bloc de données.
FC : Bloc Fonctions.
GRAFCET : Graphe de Commande Etape Transition.
PLCSIM : Programmable logic controller Simulator.
SPDT : Single Pole Double Throw.

Introduction générale :

De nos jours, les portes coulissantes sont partout (dans les hôpitaux, les centres commerciaux, les aéroports...). L'usage manuel de ces portes est devenu une perte de temps et d'efforts, pour éviter ce problème il est nécessaire d'automatiser ces portes.

Notre projet consiste à mettre en œuvre une porte coulissante automatique en utilisant un automate programmable SIMATIC S7-300, bouton poussoir, des détecteurs de position et de proximité photoélectrique et un actionneur qui est le moteur à courant continu. Effectivement l'automate doit être programmé à l'aide d'un logiciel STEP 7 en langage à contacts.

Ce projet est organisé autour de deux parties suivantes :

Partie théorique : consiste en une description de la porte coulissante automatique, puis une étude sur le système automatisé qui utilise des dispositifs électriques et électroniques, et enfin une connaissance du fonctionnement des équipements de ce projet.

Partie pratique : consiste à présenter le cahier de charge, la réalisation hardware et software et la simulation.

En terminant notre projet par une conclusion générale.

Partie Théorique

Chapitre I :

L'automatisation de la

porte coulissante

I-1 Introduction :

L'automatisme est un système qui, par le moyen de dispositifs mécaniques, pneumatiques, hydrauliques ou électriques, est capable de remplacer l'être humain pour certaines tâches simples ou complexes, les systèmes automatisés font partie de l'environnement de l'être humain.

L'automatisation de la porte coulissante de notre projet nécessite des dispositifs électriques et électroniques, pour cela en va étudier les différentes parties qui assurent ce type d'automatisation.

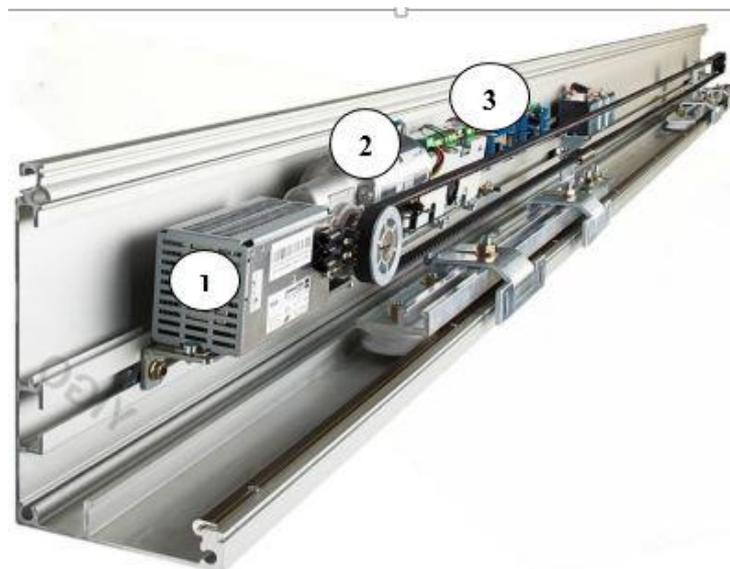
I-2 La porte coulissante automatique :

La porte coulissante automatique est un système automatisé facilite l'accès aux établissements accueillant le public tout en apportant une sécurité.

La porte coulissante automatique se compose d'un ou plusieurs vantaux, un système de motorisation permettant d'assurer l'ouverture et la fermeture du dispositif et un système de contrôle (les détecteurs, les boutons...)



Figure I-1 : Porte coulissante automatique à deux vantaux.



- 1 - Alimentation électrique.
- 2 - Moteur avec une courroie crantée.
- 3 - Unité de traitement.

Figure I-2 : Système de motorisation. [1]

I-3 Structure d'un système automatisé : [2]

Les systèmes automatisés possèdent une structure de base identique Figure II-1.

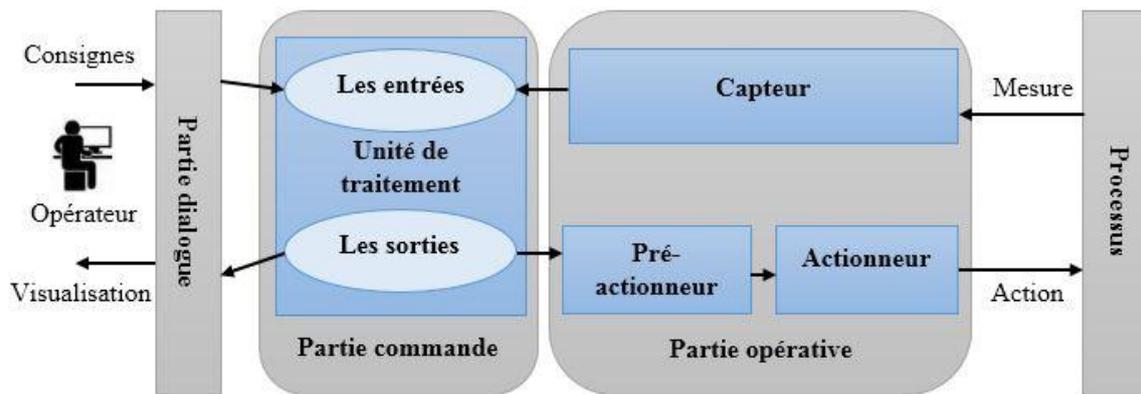


Figure I-3 : Structure d'un système automatisé.

Ils sont donc constitués de plusieurs parties plus ou moins complexes reliées entre elles :

a/ La partie opérative :

C'est la partie visible du système. Elle comporte les éléments du procédé, c'est-à-dire :

- **Des pré-actionneurs** qui reçoivent des ordres de la partie commande pour distribuer de l'énergie à un actionneur, ce dernier convertissant l'énergie reçue en énergie utile (généralement mécanique).
- **Des actionneurs** qui apportent au procédé l'énergie nécessaire à son fonctionnement à partir d'une source d'alimentation externe (cas d'un moteur par exemple, il apporte au procédé l'énergie mécanique à partir d'une source électrique).
- **Des capteurs** qui créent, à partir des grandeurs physiques de natures divers (déplacement, température, pression...) des informations utilisables par la partie commande.

b/ La partie commande :

Généralement, cette partie est l'unité de traitement (API, microcontrôleur, microprocesseur...). Elle gère le déroulement ordonné des opérations à effectuer (donne des ordres pour actionner et visualiser des opérations) selon : le programme stocké à l'unité, des informations reçues de la partie opérative et des consignes de la partie dialogue.

c/ La partie dialogue :

Cette partie est de plus en plus intégrée dans la partie commande.

Elle permet à l'opérateur de : visualiser et surveiller les opérations selon les ordres de la partie commande en utilisant (écran de surveillance, des voyants lumineux, des alarmes sonores...) et créer des consignes de commandes en utilisant (des boutons, écran tactile...).

I-4 les équipements électriques et électroniques :

I-4-1 les boutons :

Les boutons sont les éléments de dialogue entre l'opérateur et la partie commande. On trouve plusieurs types de boutons (les boutons poussoirs, les boutons tournants...). Chaque type de bouton consiste un contact peuvent être :

- Normalement Ouvert (NO) : le contact électrique est établi par une action sur le bouton.
- Normalement fermé (NF) : le contact électrique est établi sans action. Lorsqu'on fait une action sur le bouton, le contact électrique devient ouvert.

Type de contact	Représentation Verticale	Représentation Horizontale
Contact normalement ouvert NO = Normaly Open <i>(Contact à fermeture)</i>		
Contact normalement fermé NC = Normaly Closed <i>(Contact à ouverture)</i>		

Tableau I-1 : Les types de contact selon CEI.

On peut symboliser les types de boutons le plus couramment utilisés dans le tableau suivant :

Symbole	Désignation
	Bouton poussoir à contact NO Pour (mise en marche)
	Bouton poussoir à contact NF Pour (mise à l'arrêt)
	Bouton tournant avec un contact NF Pour (mise à l'arrêt)
	Contacts 2 directions et 3 positions fixes à bouton tournant
	Contact coup de poing à accrochage contact NF Généralement, pour (mise à l'arrêt d'urgence)

Tableau I-2 : Les symboles de boutons selon CEI.

I-4-2 les voyants lumineux :

Les voyants lumineux sont omniprésents dans la partie dialogue d'un système. Ils comportent une ampoule ou une LED servant à indiquer l'état du fonctionnement du système, on peut utiliser ces voyants en respectant un code de couleur normalisé.

Couleur de voyant	Signification
 Vert	Le système est en fonctionnement normal.
 Rouge	Le système est dans une situation d'urgence.
 Jaune	Le système est dans un état anormal pouvant devenir critique, nécessitant l'intervention de l'opérateur.
 Bleu	l'action de l'opérateur est nécessaire pour la poursuite du fonctionnement normal
 Blanc	surveillance.

Tableau I-3 : Code de couleur normalisé pour les voyants lumineux selon CEI.

I-4-3 les Capteurs :

Un capteur est un organe de prélèvement et de codage d'information qui transforme une grandeur physique en une grandeur exploitable, généralement électrique (signal électrique), qui peut être interpréter par un dispositif de contrôle commande. [3]

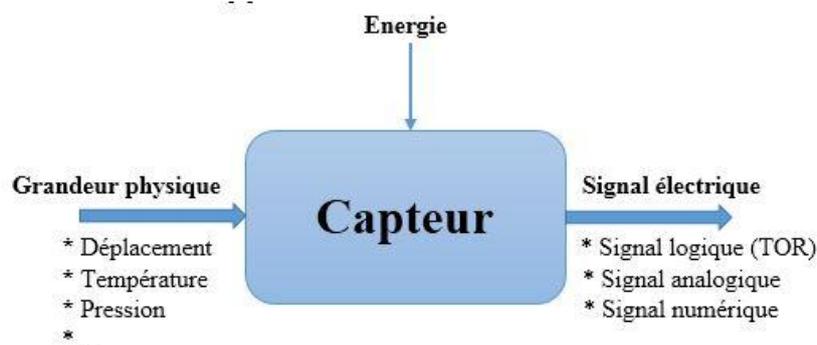


Figure I-4 : Grandeur physique et signal d'un capteur.

Les capteurs peuvent être classés selon :

a/ L'origine du signal de sortie :

La grandeur de sortie est donc un signal électrique qui pourrait varier de :

- Manière binaire (information vraie ou fausse), c'est le capteur Tout Ou Rien (TOR).
- Façon progressive (variation continue), c'est le capteur analogique.
- Échelon de tension ou de courant, c'est le capteur numérique.

b/ Le mode de détection:

On a deux types de capteurs :

- Capteurs par contact (les interrupteurs de position...).
- Capteurs sans contact (les détecteurs de proximité inductifs, capacitifs, photoélectriques...).

I-4-3-1 les Interrupteurs de position : [4]

Les interrupteurs de position ou les capteurs de fin de course sont des appareils actionnés par contact direct avec les objets. Ils transforment ce contact physique en une fermeture ou une ouverture d'un contact électrique (NO/NF).

Symbole	Désignation
	Interrupteur de position contact NO
	Interrupteur de position contact NF

Tableau I-4 : Les symboles d'interrupteur de position selon la norme CEI.

I-4-3-2 les détecteurs de proximité photoélectrique : [5]

Leur principe les rend aptes à détecter tous types d'objets, qu'ils soient opaques, réfléchissants ou même quasi-transparents. Ils sont aussi exploités pour la détection de personnes (ouvertures de portes, barrières de sécurité).

Les détecteurs photoélectriques peuvent fonctionner selon un principe transmissif. Une diode électroluminescente (LED) émet un faisceau lumineux, généralement un rayonnement infrarouge. Ce rayonnement est reçu ou non par une photodiode ou un phototransistor en fonction de la présence ou de l'absence d'un objet à détecter. Dans ces capteurs, la lumière reçue ou non provoque un changement d'état de la sortie (TOR).

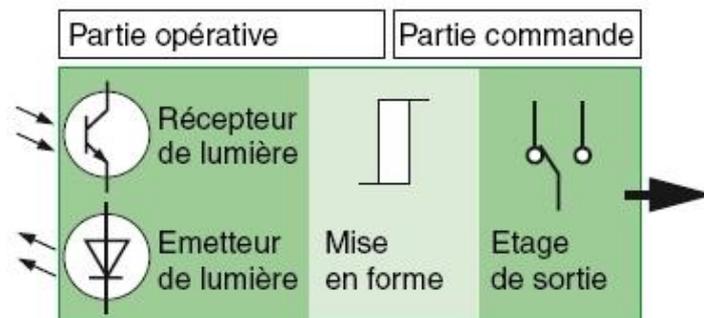


Figure I-5 : Principe d'un détecteur de proximité photoélectrique.

Ce type de détecteur peut être réalisé sur la base de trois systèmes principaux :

a/ Système barrage :

Émetteur et récepteur sont placés dans deux boîtiers séparés.

Grande portée, détection précise et fiable, adapté aux environnements difficiles

Symbole

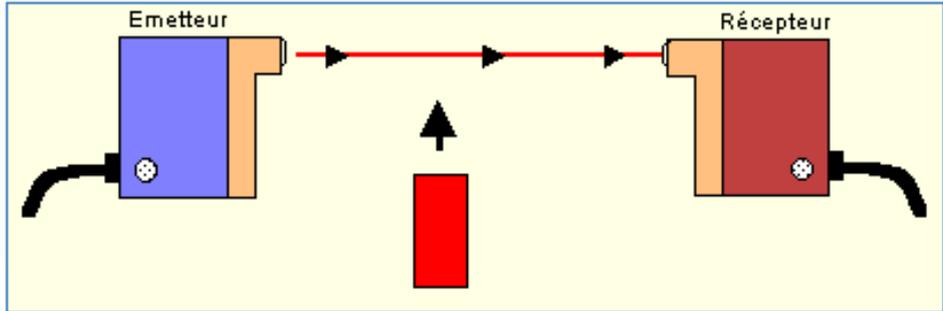
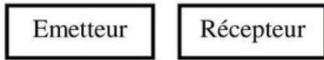
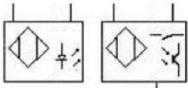


Figure I-6 : Système détection par barrage.

b/ Système réflex :

L'émetteur et le récepteur sont dans un même boîtier. Dans ce système, le faisceau réfléchi est coupé par l'objet à détecter.

Facilité de mise en œuvre, détection d'objets opaques

Symbole

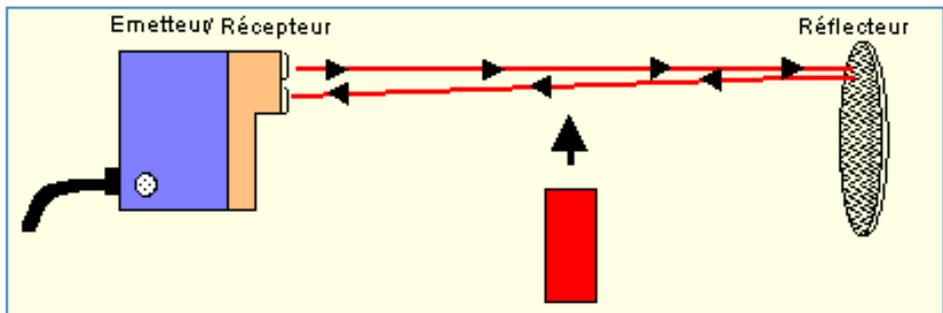
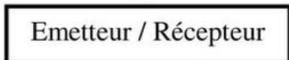
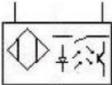


Figure I-7 : Système détection par réflex.

c/ Système proximité :

On utilise la réflexion directe de l'objet à détecter.

Détection des cibles fortement réfléchissantes

Symbole

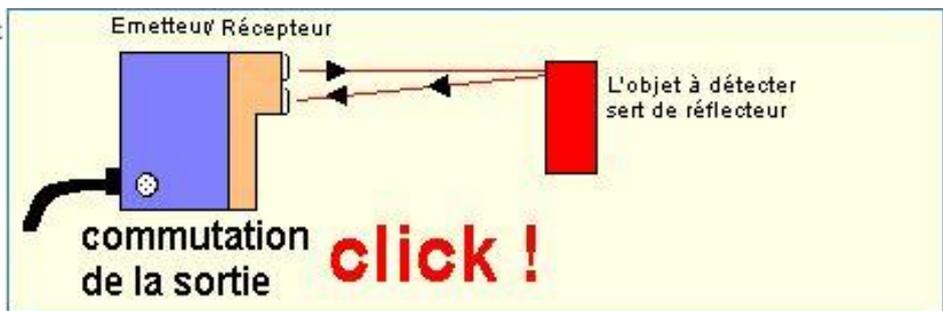
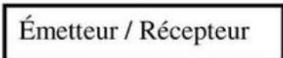
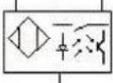


Figure I-8 : Système détection par proximité.

I-4-4 Relais électromécanique : [6]

Le relais électromécanique est un constituant de la famille des pré-actionneurs. La fonction première des relais est le plus souvent de séparer les circuits de commande des circuits de puissance à des fins d'isolement, par exemple pour piloter une tension ou un courant élevé, à partir d'une commande plus faible, et dans certaines applications, assurer aussi la sécurité de l'opérateur.

Le relais est composé principalement d'un électroaimant, qui lorsqu'il est alimenté, transmet une force à un système de commutation électrique peut être composé d'un ou plusieurs interrupteurs simple effet appelés contacts normalement ouverts (NO) ou normalement fermés (NF). Ces commutateurs sont adaptés aux courants et à la gamme de tensions à transmettre à la partie puissance (actionneur).

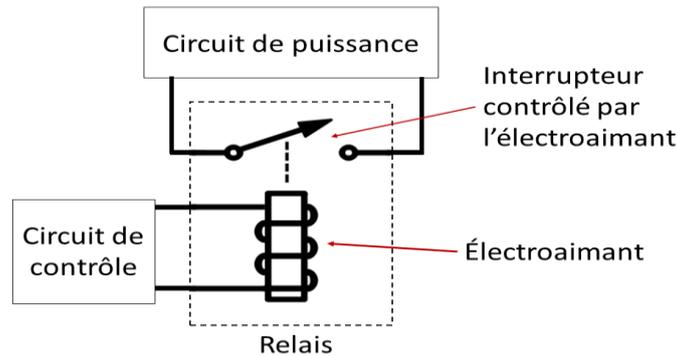


Figure I-9 : Schéma principe d'un relais électromécanique.

I-4-5 Moteur à courant continu (DC) :

Le moteur à courant continu est l'actionneur électrique le plus classique. Cette machine tournante convertit l'énergie électrique reçue (courant continu) en énergie mécanique utile.

Le changement automatique du sens de rotation d'un moteur à courant continu nécessite d'inverser la polarisation aux bornes de ce dernier à l'aide d'un circuit constitué principalement de 4 transistors de type MOSFET ou BJT, et on peut l'appeler un pont en H (H-Bridge).

Il existe plusieurs méthodes de varier la vitesse de rotation d'un moteur à courant continu. Mais le moyen le plus efficace est d'obtenir un signal PWM (Pulse Width Modulation) en utilisant le temporisateur 555.

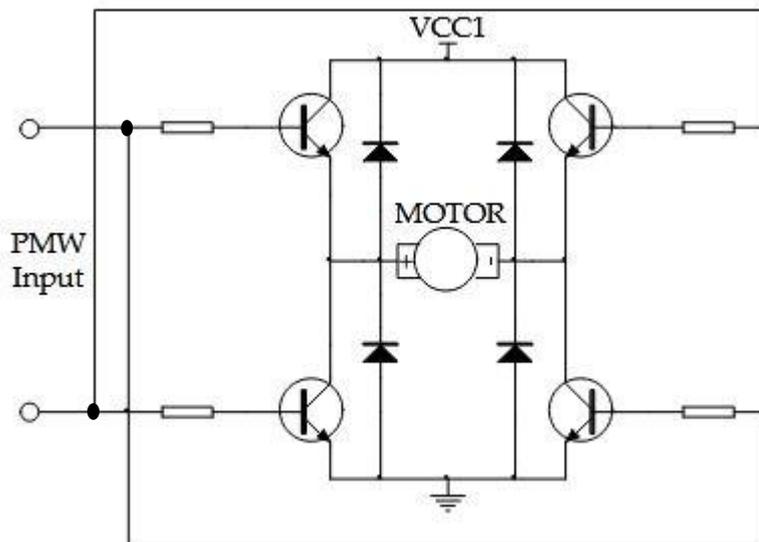
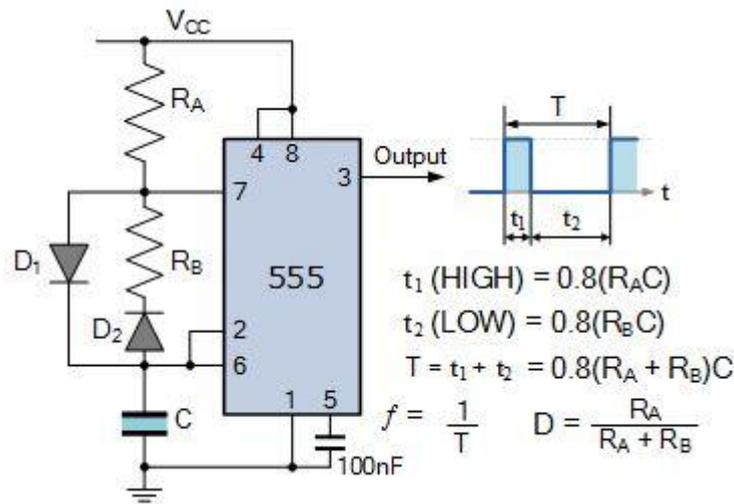


Figure I-10 : Circuit H-Bridge pour un seul moteur DC. [7]



D : Duty cycle en pourcentage %.

Figure I-11 : Circuit d'un générateur de signal PWM. [8]

I-4-6 Automates programmables industriels (API) : [6] ; [9]

Les Automates Programmables Industriels (API), ou en anglais, Programmable Logic Controller (PLC), sont apparus aux Etats-Unis vers 1968 (Modicon), à la demande de l'industrie automobile Américaine (General Motors), qui réclamait plus d'adaptabilité de leurs systèmes de commande.

Avant d'utiliser la technologie des automates pour commander les systèmes, ils ont utilisé la logique à relais câblée mais cette dernière aux inconvénients suivants:

- Grande partie du câblage
- Pas de flexibilité
- Pas de communication possible

Pour cela ils ont utilisé des systèmes à base de microprocesseur permettant une modification aisée des systèmes automatisés, c'est la logique programmée.

I-4-6-1 Définition de l'API :

Un automate programmable industriel, est un dispositif de commande électronique basé sur un microprocesseur qui utilise 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 comme :

- Logique séquentielle et combinatoire ;
- Temporisation, comptage, décomptage, comparaison ;
- Calcul arithmétique ;
- Réglage, asservissement, régulation, etc.,

Pour commander, mesurer et contrôler au moyen d'entrées et de sorties (logiques ou analogiques) différentes sortes de machines ou de processus.

I-4-6-2 Structure des APIs :

Les automates peuvent être de type compact ou modulaire :

a/ Type compact : Il intègre l'alimentation, le processeur (CPU), les modules d'entrées / sorties. Selon les modèles et les fabricants. Ces automates, de fonctionnement simple, sont généralement destinés à la commande de petits automatismes.



Figure I-12 : Compact API (LOGO SIEMENS).

b/ Type modulaire : Processeur (CPU), l'alimentation et les modules d'entrées / sorties résident dans des unités séparées (modules) et sont fixées sur un ou plusieurs racks.



Figure I-13 : Modulaire API (SIMATIC S7-300).

De manière générale, un API est structuré autour de plusieurs modules de base que sont :

a/- L'alimentation (PS) :

Le bloc d'alimentation est convertit la tension alternative du secteur en une tension continue nécessaire à l'unité central et aux circuits des modules d'entrée et de sortie.

b/- Processeur (CPU) :

Le processeur interprète les signaux d'entrée et effectue des actions de commande selon le programme stocké en mémoire, et transmet les décisions aux sorties sous forme de signaux d'action.

c/- Modules d'entrée et de sortie :

Les APIs offrent une grande variété de module E/S comme :

* E/S T.O.R. (Tout Ou Rien).

* E/S spéciales (modules intelligents) comme les E/S analogiques, les commandes d'axes, les cartes de comptage rapide et les cartes de régulation P.I.D...

d/- Interface de communication :

Interface de communication est de plus en plus intégrée avec le processeur. Elle permet de communiquer l'API avec une console de programmation, d'autre API...

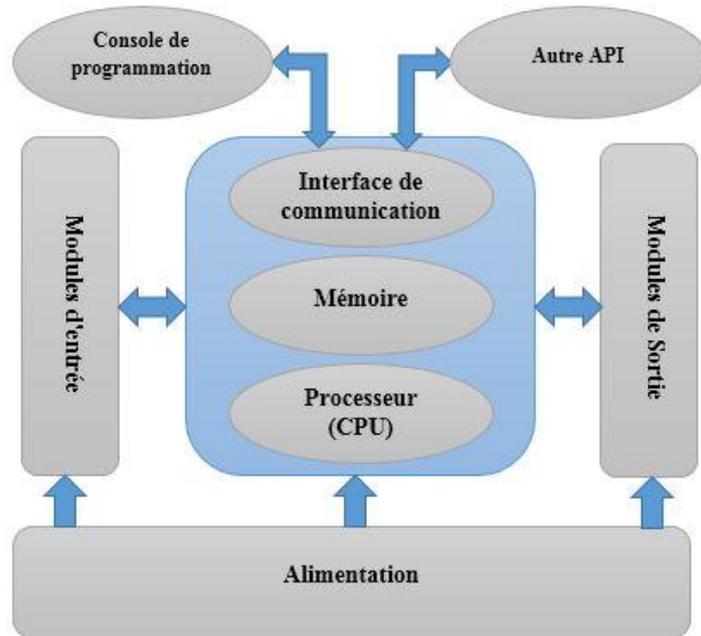


Figure I-14 : Structure générale d'un automate programmable.

I-4-6-3 Programmation des APIs :

Chaque automate possède son propre langage. Mais, les constructeurs proposent tous une interface logicielle répondant à la norme CEI 61131-3 définie par la commission électrotechnique internationale (CEI). Cette norme définit cinq langages de programmation utilisables, qui sont :

- Les langages graphiques :

LD : Ladder Diagram (langage à contacts).

FBD: Function Block Diagram (Logigrammes).

SFC: Sequential Function Chart (Grafset).

- Les langages textuels :

IL : Instruction List (Liste d'instructions).

ST: Structured Text (Texte structuré).

Ladder Diagram (langage à contacts):

Le Ladder est un langage simple qui reprend les bases du schéma à contact. Pour programmer en Ladder, il suffit de transcrire les équations logiques en schéma électrique à l'aide de symboles placés (des contacts pour les entrées, des bobines pour les sorties) entre deux barres verticales (qui représentent les lignes d'alimentations).

Un programme écrit en LADDER, appelé diagramme Ladder (en anglais, Ladder Diagram = LD), se lit de gauche à droite et du haut en bas.



Figure I-15 : Principe du langage à contacts.

I-4-6-4 Fonctionnement des APIs :

Le cycle de fonctionnement d'un automate se compose de trois étapes : l'obtention d'une entrée (lecture), le traitement des données et l'émission de commandes (écriture). Ces trois opérations sont effectuées en continu en quelques millisecondes par un automate (processus cyclique) comme indiqué sur la Figure II-12.

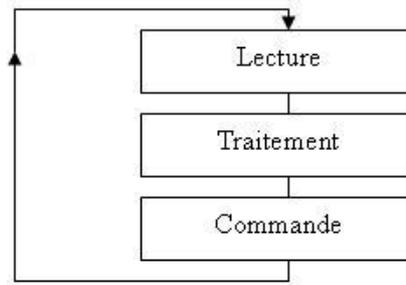


Figure I-16 : Principe de fonctionnement d'un automate programmable.

I-4-6-5 Choix d'un API :

Le choix de l'automate va se faire suivant des critères qui peuvent être différents suivant les personnes, les projets et les pays. Dans ce qui suit quelques critères à prendre en Considération.

a/ Le critère de familiarité :

On est plus familier à une certaine marque d'automate, on a donc tendance à choisir un automate car on le maîtrise déjà.

b/ La renommée :

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

c/ Le prix :

Pour un petit projet d'automatisation, il est préférable de choisir des micros ou mini automates, ce qui nous évitera des dépenses exagérées.

d/ Cahier de charge :

Le choix définitif dépendra de ce cahier de charge et des spécificités techniques du projet. Donc il faut quantifier les besoins en nombres entrées / sorties, type de processeur (la taille de mémoire, la vitesse de traitement...), fonctions ou modules spéciaux, fonctions de communication avec autre automate, supervision...

I-5 Conclusion :

Dans ce chapitre on a vu les parties principales pour l'automatisation de notre projet et on a découvert le principe de fonctionnement des équipements nécessaires pour notre système.

SIMATIC S7-300
Et
logicielle STEP 7

II-1 Introduction :

Siemens est un groupe international d'origine allemande spécialisé dans les secteurs de l'énergie, de la santé, de l'industrie et du bâtiment. Elle fabrique plusieurs gammes de produits API dans la famille SIMATIC S7. Ce sont : S7-200, S7-300 et S7-400. Siemens Step 7 est une puissante solution logicielle intégrée pour l'automatisation et comprend l'environnement de programmation pour les automates programmables Siemens. Ce logiciel fournit des outils de programmation uniques et puissants avec de multiples avantages, notamment : la création d'une logique réutilisable, une architecture de programme structurée et l'intégration d'un seul programme pour plusieurs dispositifs d'automatisation.

On va utiliser l'automate de type S7-300 qui est programmé à l'aide du logiciel Step 7 pour assurer l'automatisation de notre projet.

II-2 l'automate SIMATIC S7-300 : [10] ; [11]

SIMATIC S7-300 est un automate de type modulaire ou compact pour une gamme d'automatisation inférieure et moyenne.



Figure II-1: Automate SIMATIC S7-300.

II-2-1 Caractéristiques de l'automate S7-300 :

- Gamme diversifiée de la CPU.
- Gamme complète du module.
- Possibilité d'exécution jusqu'à 32 modules.
- Possibilité de mise en réseaux avec MPI, PROFIBUS ou PROFINET.
- Liberté de montage au différent emplacement.

II-2-2 Modularité du S7-300:

Parmi les caractéristiques essentielles du S7-300, le fait qu'il est disposé d'une vaste gamme de Modules.

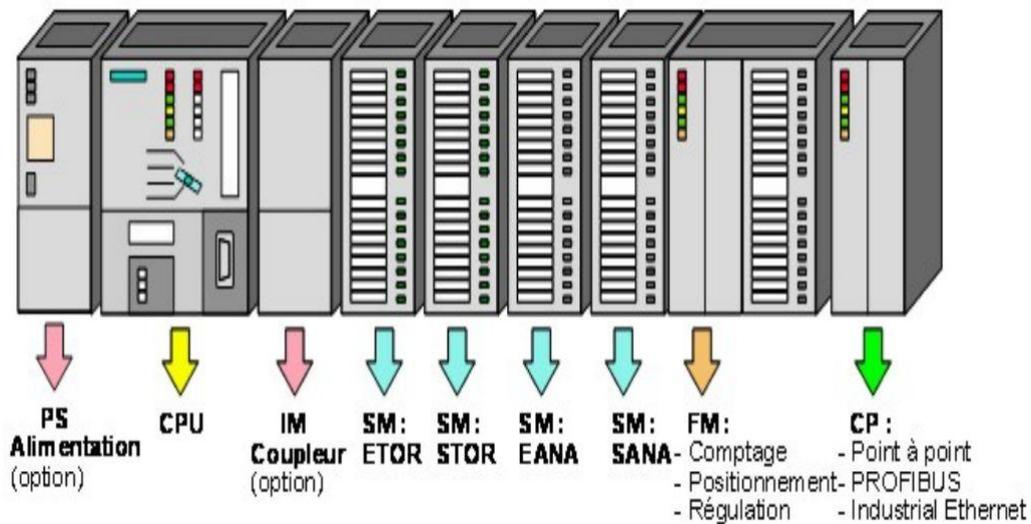


Figure II-2 : Disposition des modules de l'API S7-300.

a/ Module d'alimentation (PS) :

Le S7-300 peut être alimenté avec une tension de 24 VDC, cette dernière est assurée via le module d'alimentation par conversion de la tension Secteur 120V/230 VAC.



Figure II-3 : Différentes module d'alimentation SIMATIC S7-300.

b/ Unité centrale (CPU) :

C'est une carte électronique construite autour d'un ou plusieurs processeurs et mémoire. La CPU possède un système d'exploitation, une unité d'exécution et des interfaces de communication (MPI, PROFIBUS, PROFINET). Essentiellement la CPU lit l'état des signaux d'entrée et exécute le programme utilisateur séquentiellement.



Figure II-4 : CPU SIMATIC S-300.

- **Interfaces MPI :**

Chaque CPU est équipée d'une interface MPI (Multi Point Interface) pour la connexion de la console de programmation (PG) ou un appareil par exemple adaptateur PC.



Figure II-5 : Câble de programmation pour automate S7-300.

- **Signalisation des états :**

Sur la face avant de la CPU, on trouve des LEDS permettant à l'automate de signaler certains états, tel que :

- RUN** (LED verte).
- STOP** (LED jaune).
- SF** (Erreur matérielle ou logicielle, LED rouge).
- BF** (Erreur de bus, LED rouge).
- DC5V** (Alimentation 5 V pour CPU et bus S7-300, correcte, LED verte).
- MAINT** (Requête de maintenance, LED jaune).
- FRCE** (La LED s'allume en jaune : requête de forçage active).

- **Sélecteur de mode de fonctionnement :**

Le sélecteur de mode permet de régler le mode de fonctionnement de la CPU, tel que :

- RUN** (La CPU traite le programme utilisateur).
- STOP** (La CPU ne traite aucun programme utilisateur).
- MRES** (position dans laquelle en effacement général de la CPU peut être effectué).

- **Carte mémoire :**

La CPU peut être équipée d'une micro carte mémoire, permettant la sauvegarde du programme en cas de coupure du courant.



Figure II-6 : Micro carte mémoire SIMATIC.

c/ Coupleur (IM) :

Les coupleurs sont des cartes électroniques qui assurent la communication entre les E/S (périphéries ou autre) et l'unité centrale. L'échange de l'information entre la CPU et les modules d'E/S s'effectue par l'intermédiaire d'un câble PROFIBUS.

d/ Modules de signaux (SM):

Assurant une liaison entre la CPU de l'automate S7-300, et le processus à commandé. On a différents modules de signaux :

- **Modules d'Entrées/Sorties TOR:** Les modules entrées/sorties TOR constituent les interfaces d'entrée et de sortie pour les signaux tout ou rien de l'automate. Ces modules permettent de raccorder à l'automate S7-300 des capteurs et des actionneurs tout ou rien les plus divers, en utilisant si, nécessaire des équipements d'adaptation (conditionnement, conversion, etc...).
- **Modules d'Entrées/Sorties analogiques :** Si on a à faire à des capteurs et actionneurs analogiques, on doit faire appel à des modules d'entrées analogiques, assurant la conversion des signaux analogiques du système à commande en signaux numériques, qui peuvent être traités par l'API S7-300. Toutefois la conversion des signaux numériques en signaux analogiques destinés au processus à commande est faite, via les modules de sorties. Les modules d'Entrées/Sorties analogiques réalisent les deux fonctions, c'est-à-dire la conversion Analogique-Numérique, et Numérique-Analogique.

e/ Module de fonction (FM) :

Ces modules réduisent la charge de traitement de la CPU en assurant des tâches lourdes de calculs. On peut citer les modules suivants : Module de commande d'axe pour servomoteur, Module de positionnement pour moteur pas à pas, Module de régulation, Module de comptage...

f/ Module de communication (CP) :

Le module de communication Assurant une communication par transmission série, comme ils peuvent aussi établir des liaisons point à point avec d'autres automates SIMATIC S7-300 ou bien d'autres constructeurs...

g/ Le châssis (rack) :

Les châssis constituent des éléments mécaniques de base du SIMATIC S7-300 (Assemblage mécanique des modules).

II-3 Programmation STEP 7 : [12]

Les automates programmables industriels effectuent des tâches d'automatisation traduites sous formes de programme d'application qui définit la manière dont l'automate doit commander le système par une suite d'instructions, le programme doit être écrit dans un langage déterminé avec des règles définies pour que l'automate puisse l'exécuter, pour cela les automates de la famille SIMATIC sont programmés grâce au logiciel STEP7 via une console de programmation ou PC et sous un environnement WINDOWS.

II-3-1 Définitions du logiciel STEP7 :

Step7 est le logiciel de base qui permet la configuration et la programmation des systèmes d'automatisation SIMATIC. Il s'exécute sous un environnement Windows, à partir d'une console de programmation ou d'un PC. Il existe plusieurs versions : STEP 7 micro/ Win pour les applications S7-200 et STEP 7 pour les applications S7-300 et S7-400.

Le logiciel STEP7 offre les possibilités suivantes :

- Configuration et paramétrage du matériel et de communication.
- Création de gestion des projets.
- La création des programmes.
- Gestion des mnémoniques.
- Test d'installation de l'automatisation.
- Le diagnostic lors des perturbations dans l'installation.
- Document et archivage.

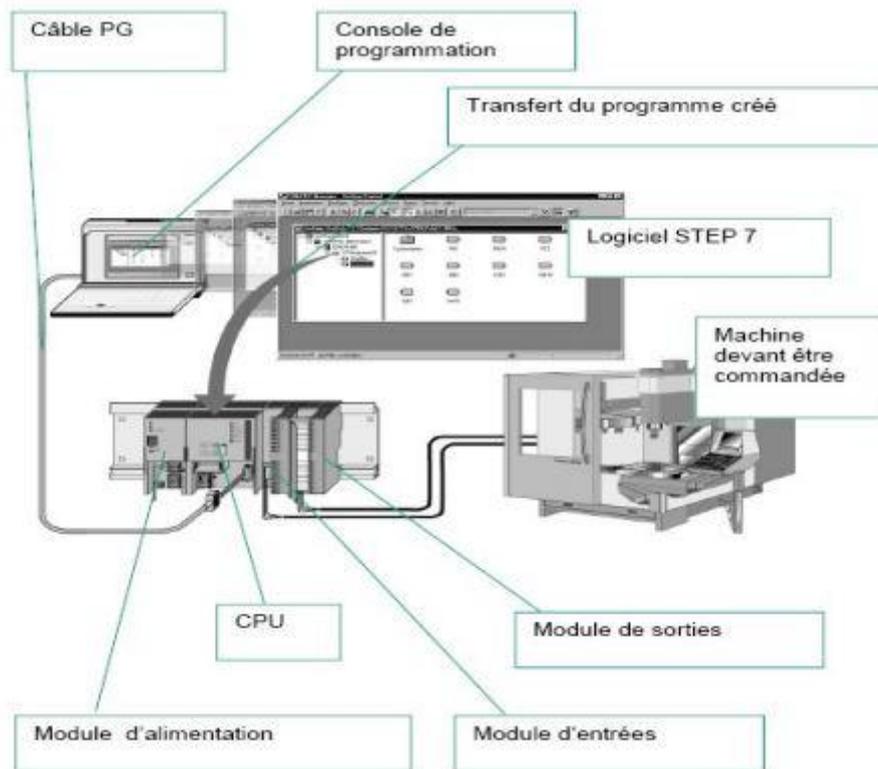


Figure II-7 : Vue d'ensemble de l'automatisme.

Le STEP7 dispose de quatre langages de programmation :

- Langage liste (LIST)** : image textuelle proche du comportement interne de l'automate.
- Langage logigramme (LOG)** : langage graphique, utilisant les symboles de l'électronique numérique (portes logiques).
- Langage à contacts (CONT)** : suite de réseaux parcourus séquentiellement dont les entrées sont représentées par des interrupteurs et les sorties par des bobines.
- Graph** : utilise le GRAFCET comme outil.

STEP 7 offre les blocs utilisateur suivants pour la programmation structurée :

-Bloc d'organisation (OB) :

Un OB est appelé cycliquement par le système d'exploitation et constitue donc l'interface entre le programme utilisateur et le système d'exploitation. L'OB contient des instructions d'appels de blocs indiquant à l'unité de commande de l'automate l'ordre dans lequel il doit traiter les blocs.

-Bloc fonctionnel (FB) :

Le FB dispose d'une zone de mémoire qui lui est affectée en propre. Il est possible d'affecter un bloc de données (DB) au FB à l'appel du bloc. Il est possible d'accéder aux données du DB d'instance via les appels contenus dans le FB. On peut affecter plusieurs DB à un FB. Il est possible d'appeler d'autres FB et FC dans un bloc fonctionnel via des instructions d'appels de blocs.

-Fonction (FC) :

Une FC contient des routines pour les fonctions fréquemment utilisées. Elle est sans mémoire et sauvegarde ses variables temporaires dans la pile de données locales. Cependant elle peut faire appel à des blocs de données globaux pour la sauvegarde de ses données.

-Bloc de données (DB) :

Les DB sont utilisés pour la mise à disposition d'espace mémoire pour les variables types données. Il existe deux types de blocs de données. Les DB globaux dans lesquels tous les OB, FB et FC peuvent lire les données enregistrées ou écrire des données et les DB d'instance qui sont affectés à un FB donné.

II-3-2 L'adressage absolu des modules de signaux :

Chaque entrée et sortie du module de signaux possède une adresse absolue déterminée par la configuration matérielle.

a) Adressage des modules TOR :

L'adressage d'une entrée ou d'une sortie est constitué d'une adresse d'octet et d'une adresse de bit.

-L'adressage d'octet dépend de l'adresse de début de module.

-L'adressage de bit est indiqué sur le module.

La figure (II-8) illustre un exemple d'adressage absolu ou direct.

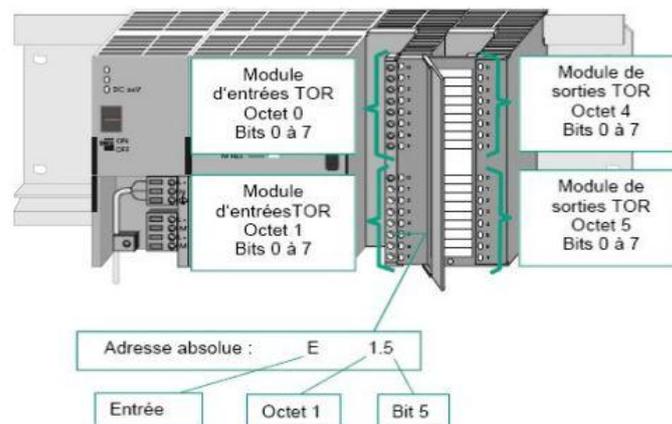


Figure II-8 : Exemple d'adressage absolu d'un module TOR.

b) Adressage des modules analogiques :

L'adresse d'une voie d'entrée ou de sortie analogique est toujours une adresse de mots. L'adresse de voie est basée sur l'adresse initiale des modules. L'adresse initiale de chaque module analogique suivant augmente de 16 bits par emplacement.

c) Mémentos :

Les mémentos sont utilisés pour les opérations internes à l'automate pour lesquelles l'émission d'un signal n'est pas nécessaire. Les mémentos sont des bistables servant à mémoriser les états logiques 0 ou 1. Chaque automate programmable dispose d'un grand nombre de mémentos. On programme ces derniers comme des sorties. En cas de panne de la tension de service, le contenu sauvegardé dans les mémentos est perdu.

II-3-3 Création d'un projet STEP7 :

Les procédures qui permettent la création de projet sous logiciel STEP7 sont comme suit :

1- Double clic sur l'icône SIMATIC Manager; ceci lance l'assistant de STEP7.



Figure II-9 : SIMATIC Manager qui lance l'assistant de STEP7.

2- La fenêtre illustrée en (figure II.10) apparaît, elle permet la création d'un nouveau projet.

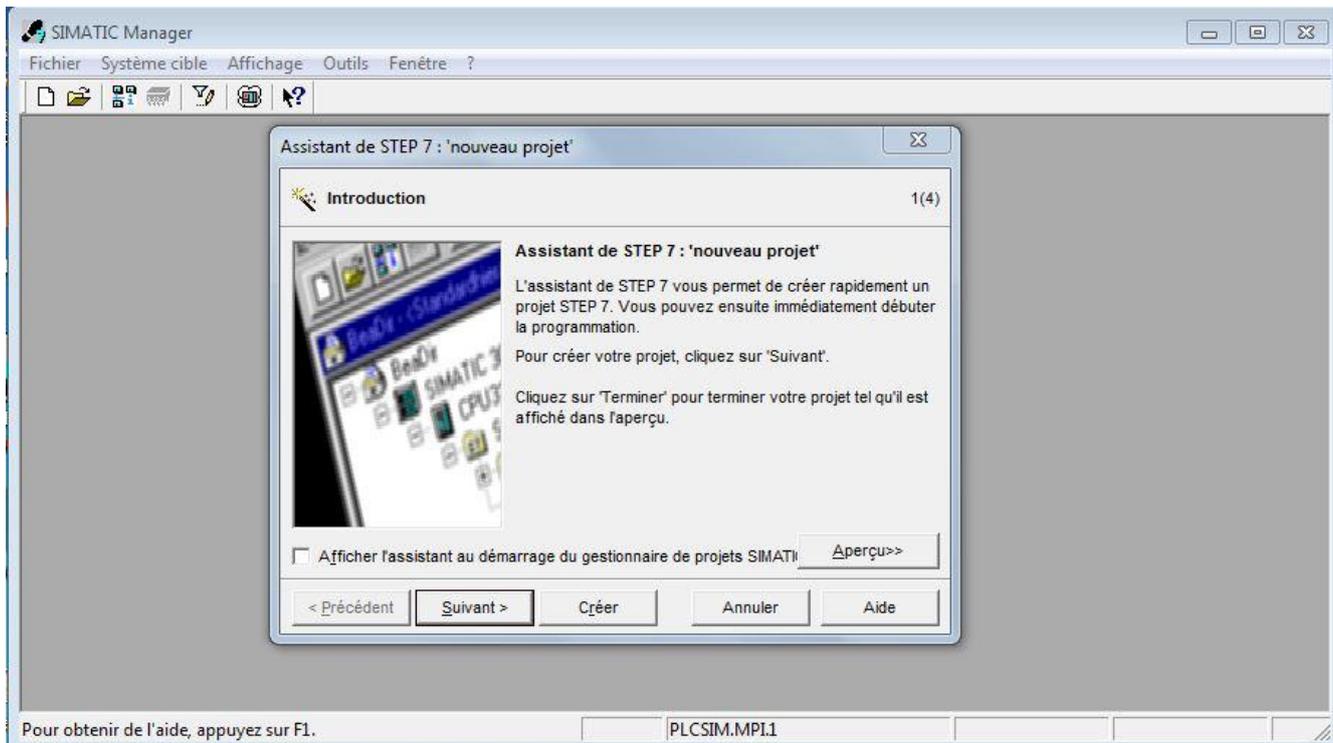


Figure II-10 : Assistant de STEP 7 : nouveau projet.

3- En cliquant sur l'icône suivant, la fenêtre suivante, nous permet de choisir la CPU.

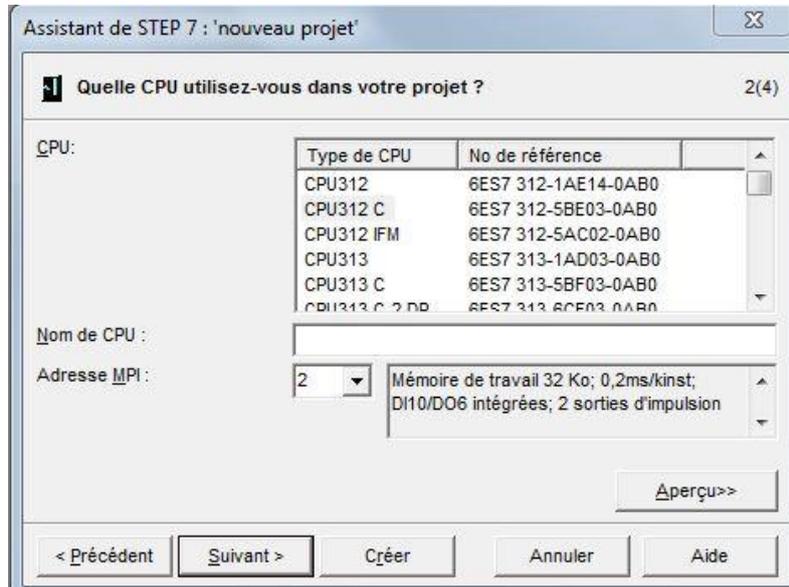


Figure II-11 : Fenêtre de choix de la CPU.

4- Après validation de la CPU, la fenêtre qui apparaît permet de choisir les blocs à insérer, et choisir le langage de programmation (LIST, CONT, LOG).

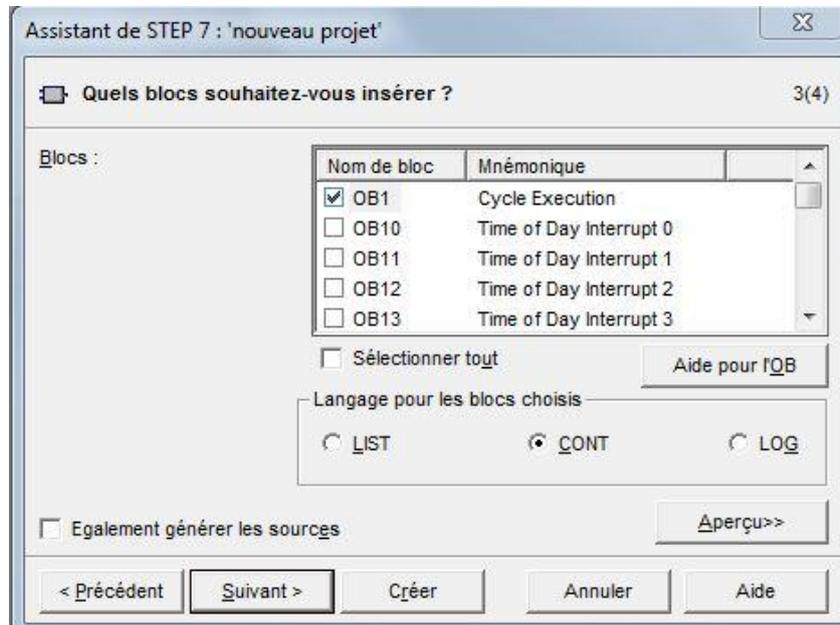


Figure II-12 : Choix des Blocs à utilisés et de langage.

5- En cliquant sur suivant, la création de projet apparaît pour le nommer.

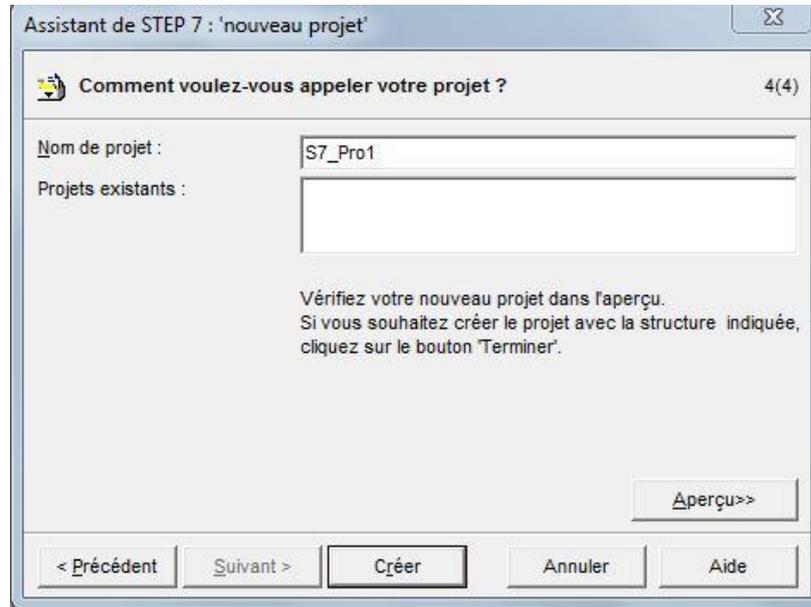


Figure II-13 : Nomination du projet.

6- On clique sur créer, la fenêtre suivante apparaît.

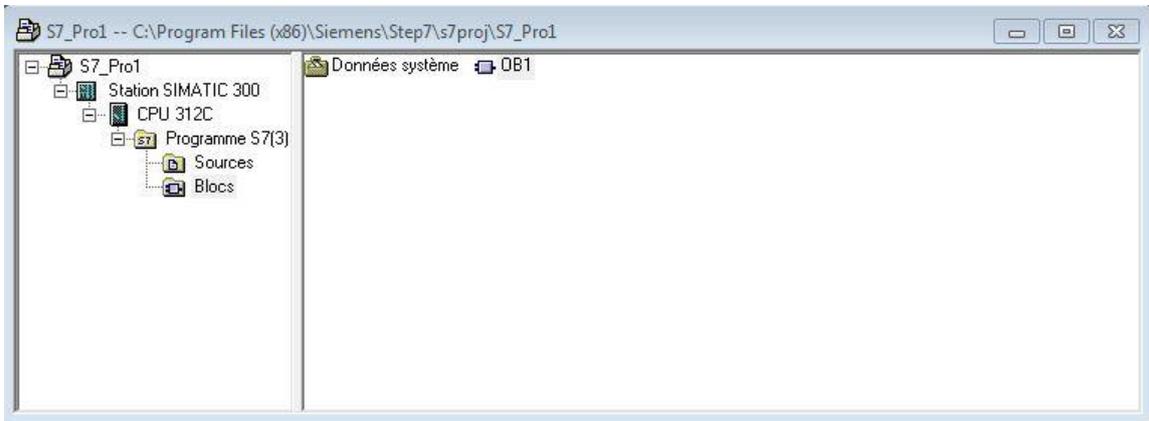


Figure II-14 : Répertoire de la station SIMATIC et de la CPU.

7- Configuration matérielle : La configuration matérielle est une étape importante. Elle consiste à disposer les châssis (rack), les modules et les appareils de la périphérie centralisée. Les châssis sont représentés par une table de configuration dans laquelle on peut placer un nombre définis de modules comme dans les châssis réels.

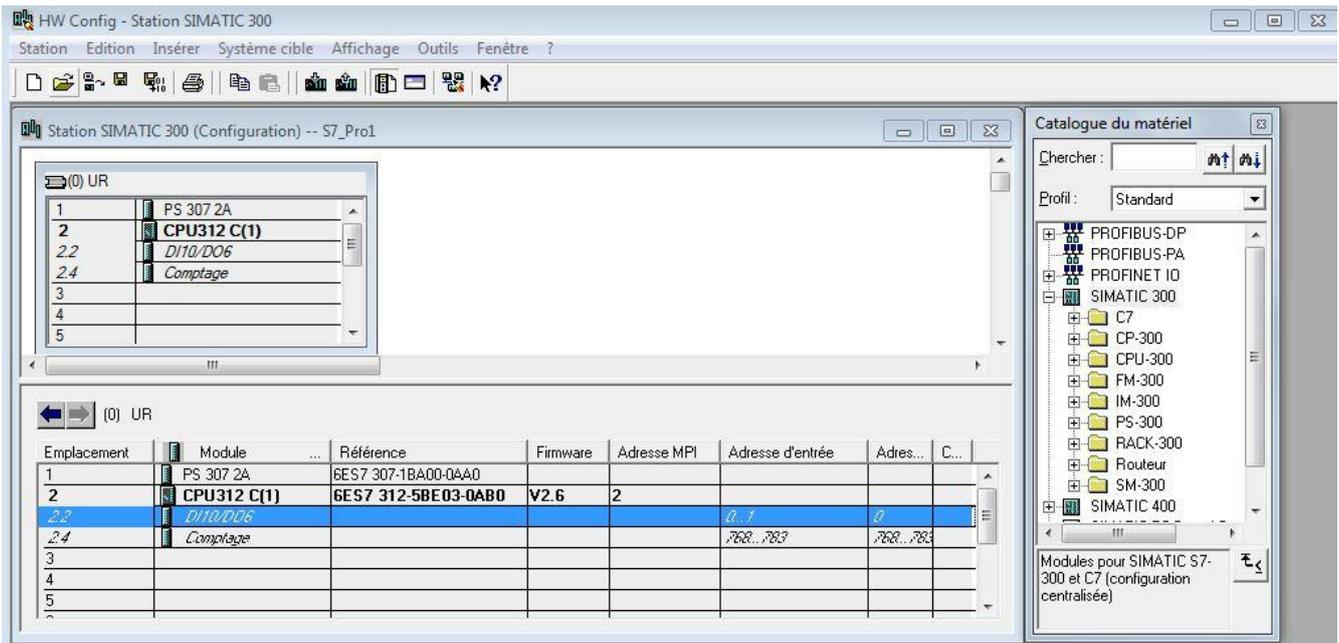


Figure II-15 : Configuration du matériel.

8- Table des Mnémoniques : Une mnémonique est un nom que l'utilisateur définit en respectant les règles de la syntaxe imposée. Il est destiné à rendre le programme lisible et aide donc à gérer facilement le grand nombre de variables couramment rencontrées dans ce genre de programme. Ce nom qu'on a donné à l'adresse pourra être utilisé directement dans le programme une fois les affectations terminées. La figure suivante (Figure II-16). Illustre une partie de la table de mnémoniques.

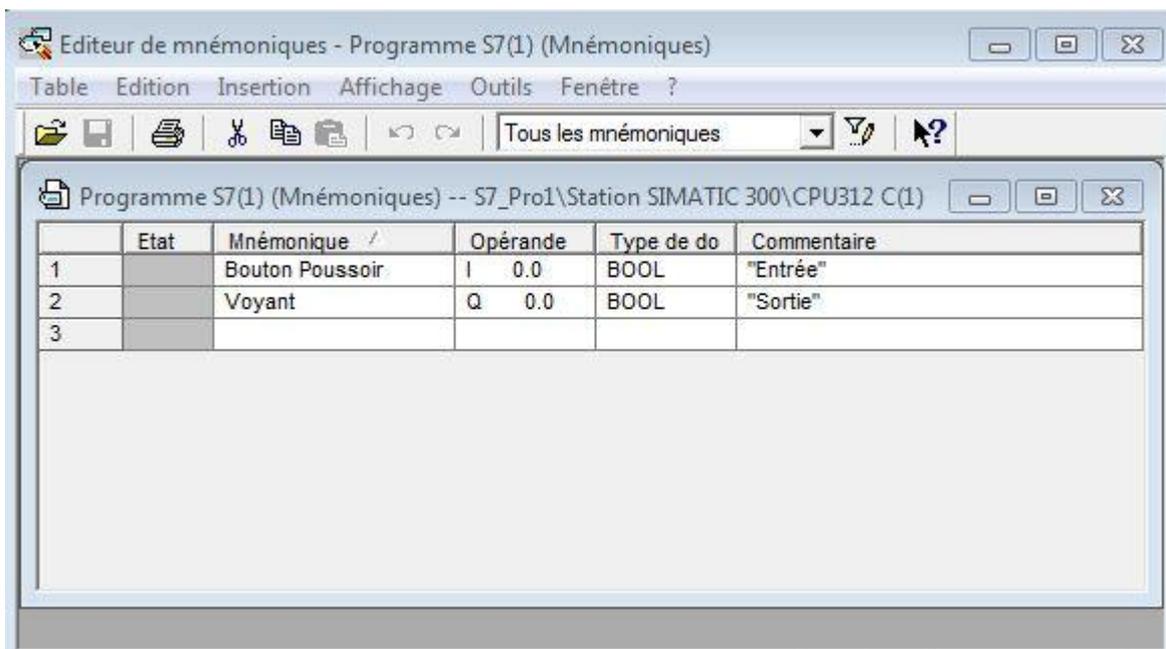


Figure II-16 : La table de mnémoniques (pour activer et désactiver d'un voyant).

9- création du programme sur le bloc OB1 :

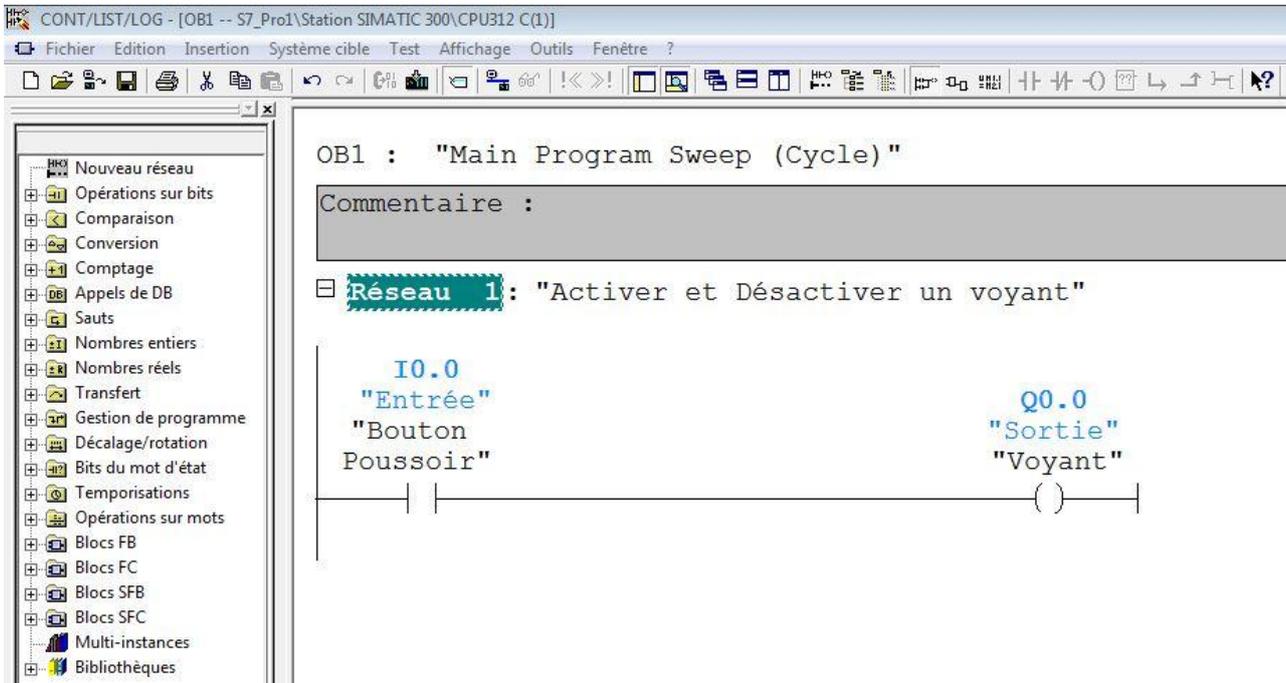


Figure II-17 : Création du programme (langage à contacts, pour activer et désactiver d'un voyant.).

II-4 Simulation S7-PLCSIM :

Après l'élaboration du programme de commande d'un système automatisé, on arrive à l'étape décisive du travail effectué. Cette étape est la validation du programme par simulation et vérification de son bon fonctionnement. Pour cela on utilise le logiciel S7 PLCSIM qui est un logiciel optionnel de STEP 7. L'application de simulation de modules S7-PLCSIM nous permet d'exécuter et de tester notre programme dans un automate programmable. La simulation étant complètement réalisée au sein du logiciel STEP 7, il n'est pas nécessaire qu'une liaison soit établie avec un matériel S7 quelconque (CPU ou modules de signaux). L'objectif de ce logiciel est le test des programmes STEP 7 pour les automates S7-300 et S7-400 qu'on ne peut pas tester immédiatement sur le matériel et ceci pour différentes raisons, l'application est critique, car elle peut occasionner des dommages matériels ou blessures corporelles en cas d'erreurs de programmation, mais la simulation permet de corriger ces erreurs pendant le test de la simulation.

II-4-1 Présentation du S7-PLCSIM :

L'utilisation du simulateur de modules physiques S7-PLCSIM nous permet d'exécuter et de tester le programme dans un automate de simulation que nous simulons dans un ordinateur ou dans une console de programmation. La simulation étant complètement réalisée au sein du logiciel STEP7.

Le S7-PLCSIM dispose d'une interface simple nous permettant de visualiser et de forcer les différents paramètres utilisés par le programme (comme activer ou désactiver des entrées.). Tout en exécutant le programme dans l'API de simulation, nous avons également la possibilité de mettre en œuvre les diverses applications du logiciel STEP7 comme, par exemple, le test de bloc afin de visualiser les variables d'entrées et de sorties.

II-4-2 Mise en route du logiciel S7-PLCSIM :

Le mode de simulation est disponible à partir du gestionnaire de projet SIMATIC.
On peut suivre la procédure suivante pour la mise en route du logiciel S7-PLCSIM.

1- Ouvrir le gestionnaire de projet SIMATIC.

2- Cliquez sur  ou sélectionnez la commande Outils > simulation de modules.
Cela lance l'application S7-PLCSIM et ouvre une fenêtre CPU

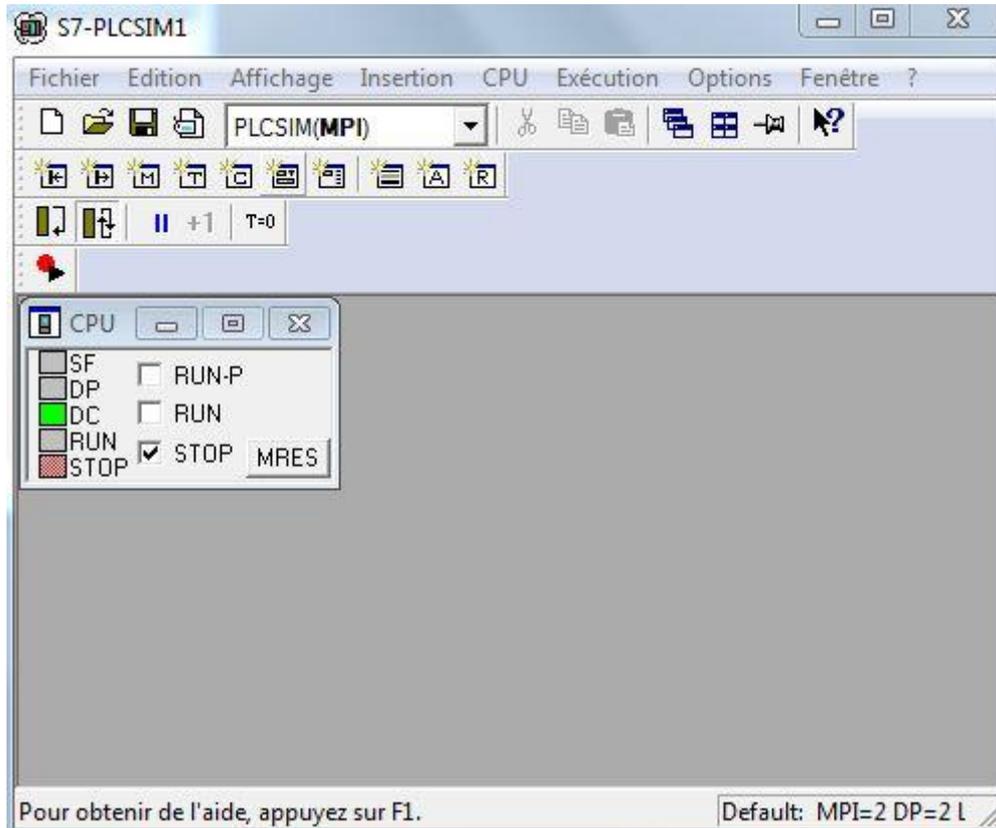


Figure II-18 : Fenêtre du S7-PLCSIM.

3- Dans le bloc OB1 cliquez sur  ou choisissez la commande Système cible (charger) pour charger le dossier blocs dans l'API de simulation.

4- Dans l'application S7-PLCSIM, on crée de nouvelles fenêtres pour visualiser les informations provenant de l'API de simulation :

- Cliquez sur  ou choisissez la commande Insertion (Entrée) pour créer une fenêtre dans laquelle vous pouvez visualiser et forcer des variables dans la zone de mémoire des entrées. Cette fenêtre s'ouvre avec l'adresse de mémoire par défaut IB0. Mais on peut modifier l'adresse (IB1, IB2...)

- Cliquez sur  ou choisissez la commande Insertion (Sortie) pour créer une fenêtre dans laquelle vous pouvez visualiser et forcer des variables dans la zone de mémoire des sorties. Cette fenêtre s'ouvre avec l'adresse de mémoire par défaut QB0. Mais on peut modifier l'adresse (QB1, QB2...).

- Cliquez sur  ou choisissez la commande Insertion (Mémento) pour créer une fenêtre dans laquelle vous pouvez visualiser et forcer des variables dans la zone de mémoire des mémentos. Cette fenêtre s'ouvre avec l'adresse de mémoire par défaut MB0. Mais on peut modifier l'adresse (MB1, MB2...).

- Cliquez sur  ou choisissez la commande Insertion (Temporisation) pour créer une fenêtre dans laquelle vous pouvez visualiser et forcer les temporisations utilisées par le programme. Cette fenêtre s'ouvre avec l'adresse de mémoire par défaut T0.

5- Choisir le menu CPU dans la fenêtre du s7-PLCSIM et vérifier que la commande à Mettre sous tension est activée.

6- Choisir la commande Exécution (Mode d'exécution) et vérifier que la commande cycle continue est activée.

7- Mettre la CPU de simulation en marche en cliquant sur l'une des cases à cocher RUN ou RUN-P

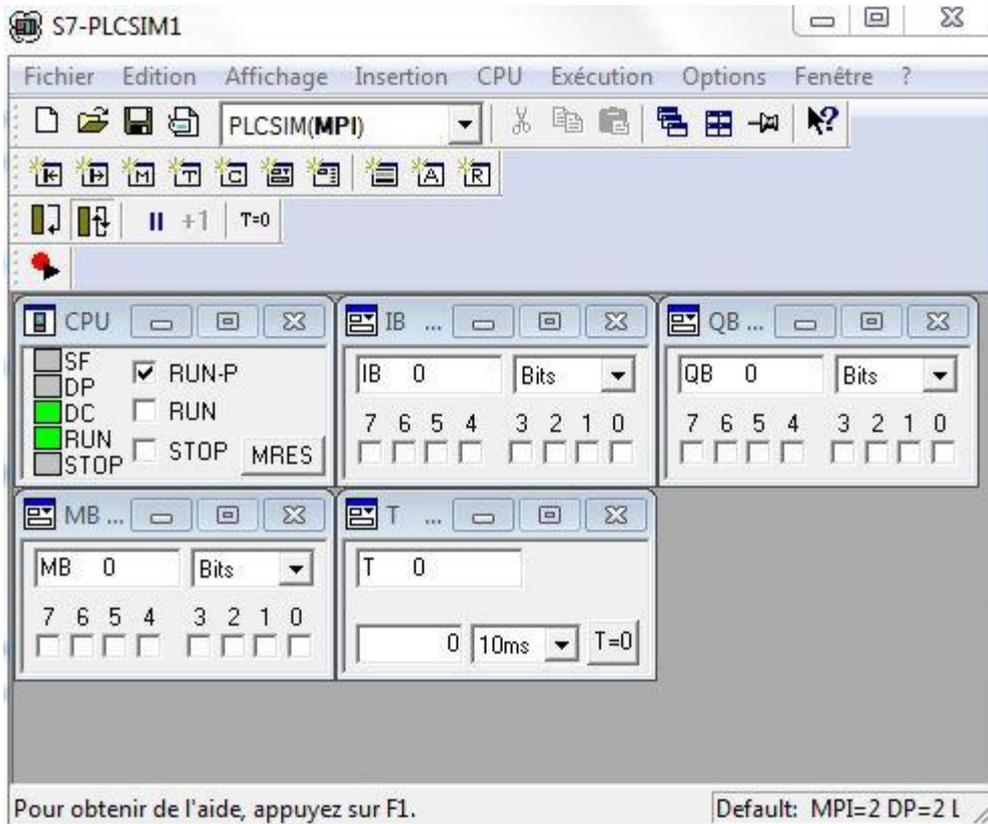


Figure II-19 : Simulateur S7-PLCSIM.

8- Après le chargement du programme dans la CPU du simulateur et la mise de cette dernière en mode « RUN » le STEP 7 nous permet de visualiser l'état du programme soit en cliquant sur l'icône  ou on sélectionnant la commande Test > Visualiser.

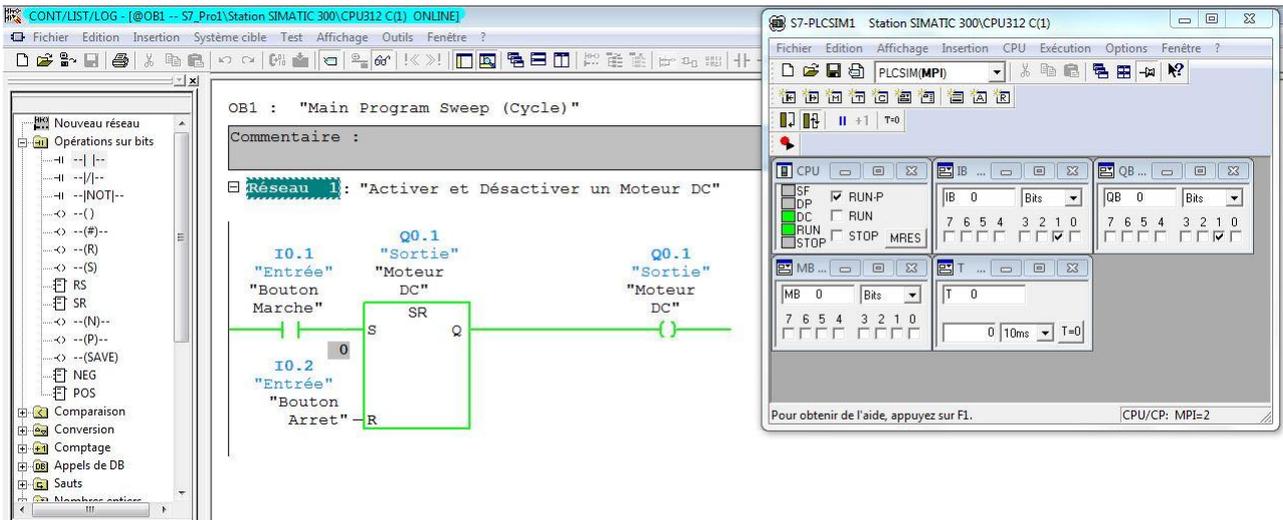


Figure II-20 : Visualisation de l'état du programme (activer et désactiver un moteur DC).

II-5 Conclusion :

Dans ce chapitre, on a présenté les différents principaux modules de l'API S7-300. Puis, on a vu comment programmer ce type d'automate en utilisant logiciel STEP 7.

Partie Pratique

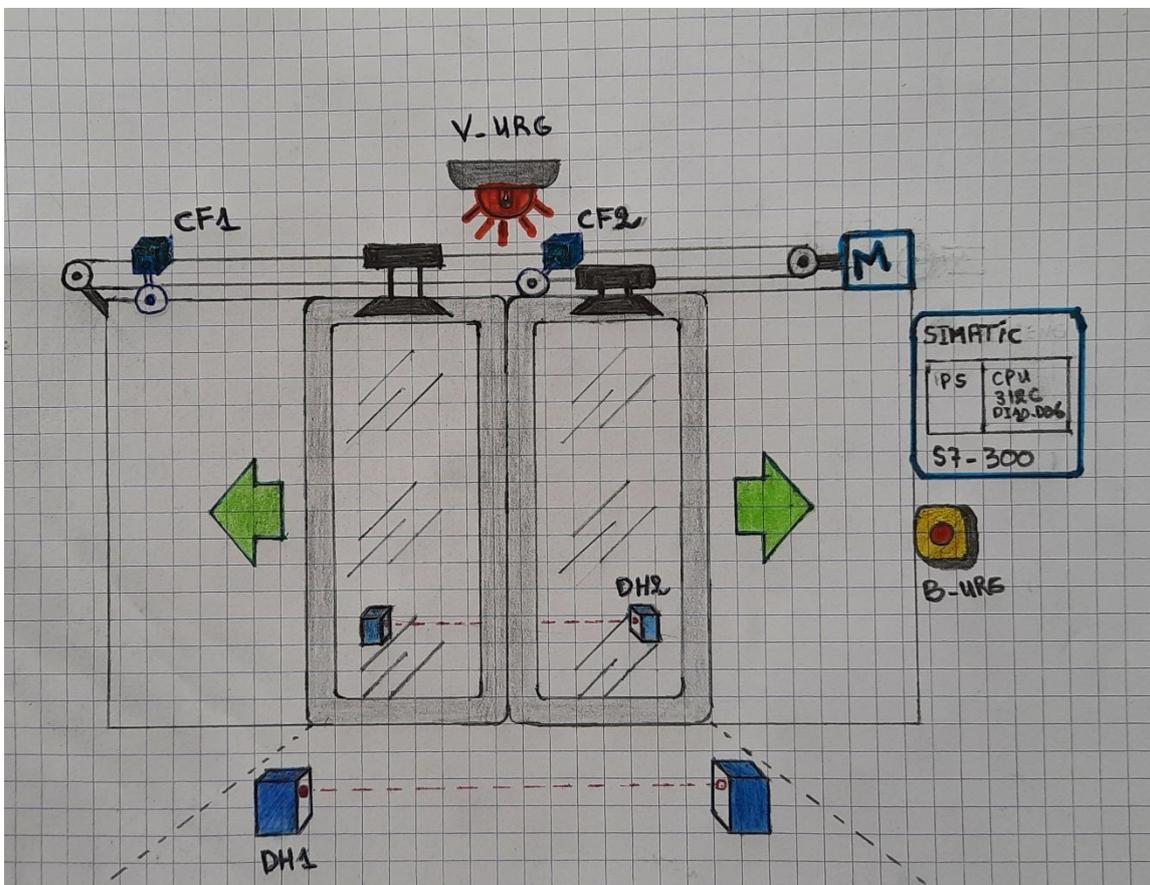
I-1 Présentation du cahier des charges :

Notre projet consiste à mettre en œuvre une porte coulissante automatique à deux vantaux, en utilisant la liste d'équipements électriques et électroniques suivante :

I-1-1 liste d'équipements :

équipement	Caractéristique	Rôle	Quantité
Automate Programmable Industriel (API)	SIEMENS CPU312C DI10/DO6	Fonctionnement et contrôle la porte.	1
Actionneurs Moteur DC 2 sens de marche	Moteur DC 24VDC	Assure le déplacement de la porte.	1
H-Bridge	H-Bridge DC MOTOR DRIVE	Assure le changement du sens de rotation d'un moteur DC	1
Générateur d'un signal PWM	Circuit 555 générer un signal PWM 5VDC	Contrôle la vitesse du moteur (contrôle la vitesse de déplacement de la porte)	1
Capteurs Photoélectrique	Détecteur de proximité photoélectrique système réflex 3 fils 24VDC	Détection humaine pour ouvrir la porte.	2
Capteur fin de course	Détecteur de position avec levier à galet NO	arrêter le déplacement de la porte.	2
Bouton poussoir	Bouton arrêt d'urgence NO	Pour l'arrêt d'urgence.	1
Voyant rouge	Voyant rouge 24VDC	Signalisation d'état d'urgence.	1
Relais électromécanique	Relais SPDT 24VDC	Pour séparer circuit puissance au circuit commande (pour la sécurité d'API contre les courts circuits).	2

Tableau I-1 : La liste des équipements.



- DH1 - DH2** : Détecteur de proximité photoélectrique système réflech (Détecteur Humain 1 et 2).
CF1 - CF2 : Détecteur de position avec levier à galet 1 et 2.
M : Moteur DC.
B-URG : Bouton arrêt d'urgence.
V-URG : Voyant Rouge d'urgence.

Figure I-1 : Porte coulissante automatique à deux vantaux.

I-1-2 Description du cycle de fonctionnement :

Lorsque quelqu'un s'approche de la porte, il coupe le faisceau lumineux de détecteur photoélectrique DH1 ou DH2, cela permet d'ouvrir la porte jusqu'à le capteur fin de course CF1 soit détecté, dans ce cas la porte reste ouverte pendant 5 secondes, avant l'expiration de cette temporisation, elle sera redémarré à chaque fois si le capteur photoélectrique DH1 ou DH2 a détecté une autre présence. Après l'expiration de temporisation, la porte commence à se fermer jusqu'à le capteur fin de course CF2 soit détecté. Avant la fermeture complète de la porte lorsqu'il y a une autre détection sur le capteur photoélectrique DH1 ou DH2, la porte commence à se ré ouvrir jusqu'à le capteur fin de course CF1 soit détecté, la temporisation précédente a été créé, après l'expiration de cette dernière, la porte commence à se refermer jusqu'à le capteur fin de course CF2 soit détecté (fermeture complète).

Pour assurer la sécurité du système on utilise un bouton d'urgence B-URG coupant l'alimentation sur les sorties et un voyant rouge V-URG signalant l'état d'urgence, Le moteur s'arrête à ce moment, la porte n'est pas ré ouverte ou refermée.

I-2 Réalisation hardware et software :

Effectivement, la réalisation hardware et software de notre système est représenté en câblant nos appareils électriques et électroniques, en créant et en testant le programme qui gère notre système à l'aide du logiciel STEP 7.

Le câblage de nos équipements simplifié sur les schémas électriques suivant :

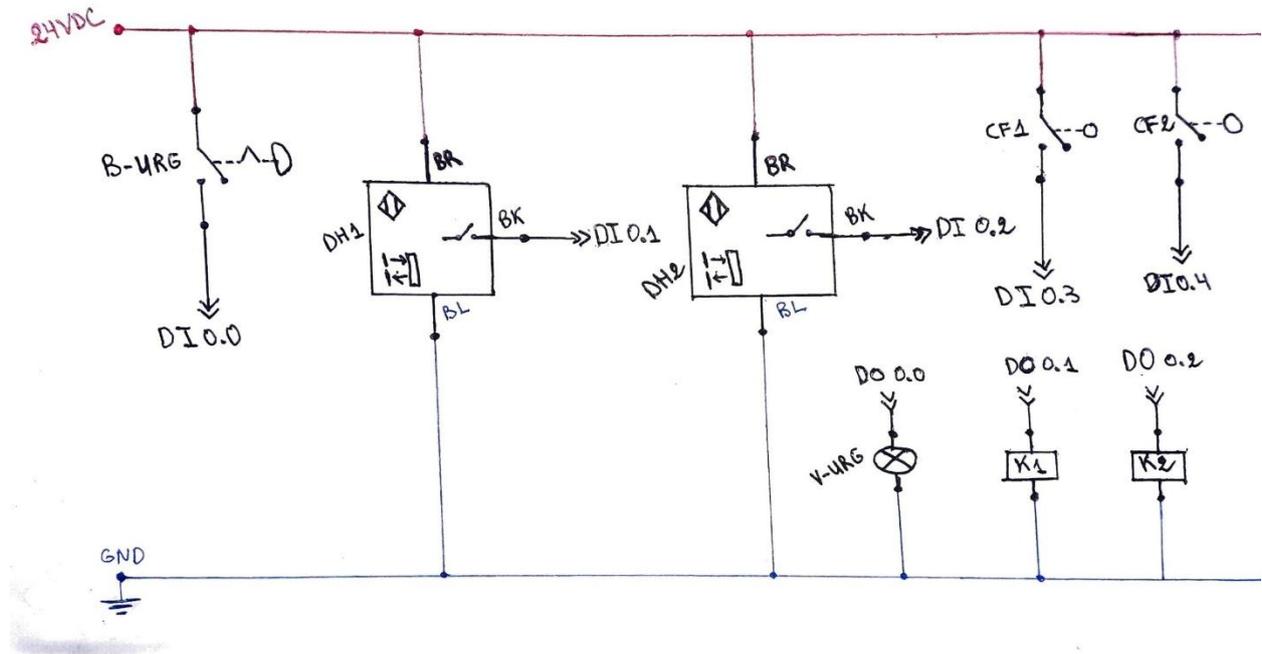


Figure I-2 : Schéma électrique circuit commande.

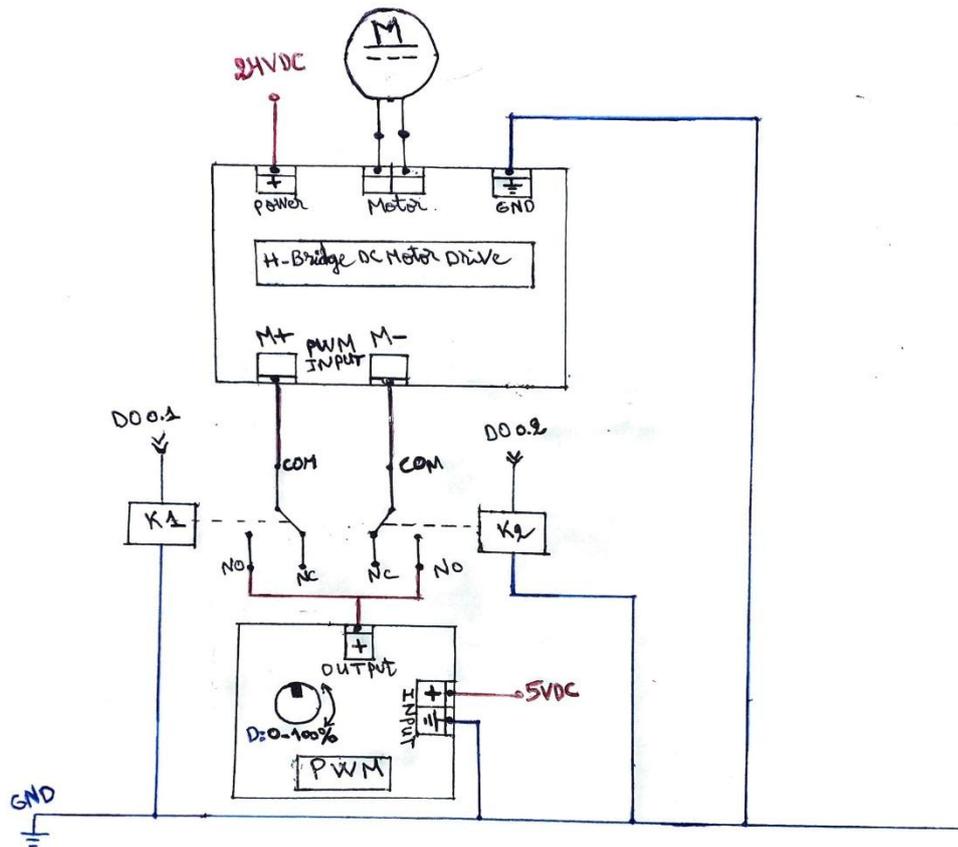


Figure I-3 : Schéma électrique circuit puissance.

Liste des symboles :

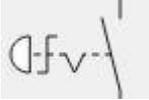
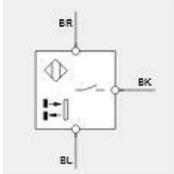
Symbole	Désignation
 <p>CF1 et CF2</p>	<p>Détecteur de position avec levier à galet NO</p>
 <p>B-URG</p>	<p>Bouton arrêt d'urgence NO</p>
 <p>DH1 et DH2</p>	<p>Détecteur de proximité photoélectrique système réflex 3 fils 24VDC</p>
 <p>V-URG</p>	<p>Voyant rouge d'urgence 24VDC</p>
 <p>K1 et K2</p>	<p>Bobine de relais SPDT 24VDC</p>
<p>DI 0.X</p>	<p>Les entrées de l'API SIEMENS CPU312C</p>
<p>DO 0.X</p>	<p>Les sorties de l'API SIEMENS CPU312C</p>
	<p>Moteur DC 24VDC</p>
<p>M+</p>	<p>Moteur tourne à gauche</p>
<p>M-</p>	<p>Moteur tourne à droite</p>

Tableau I-2 : Symbole des équipements.

Après le câblage des appareils, on crée le programme en langage à contacts sur le bloc OB1 du logiciel STEP 7 :

1- Configuration matérielle :

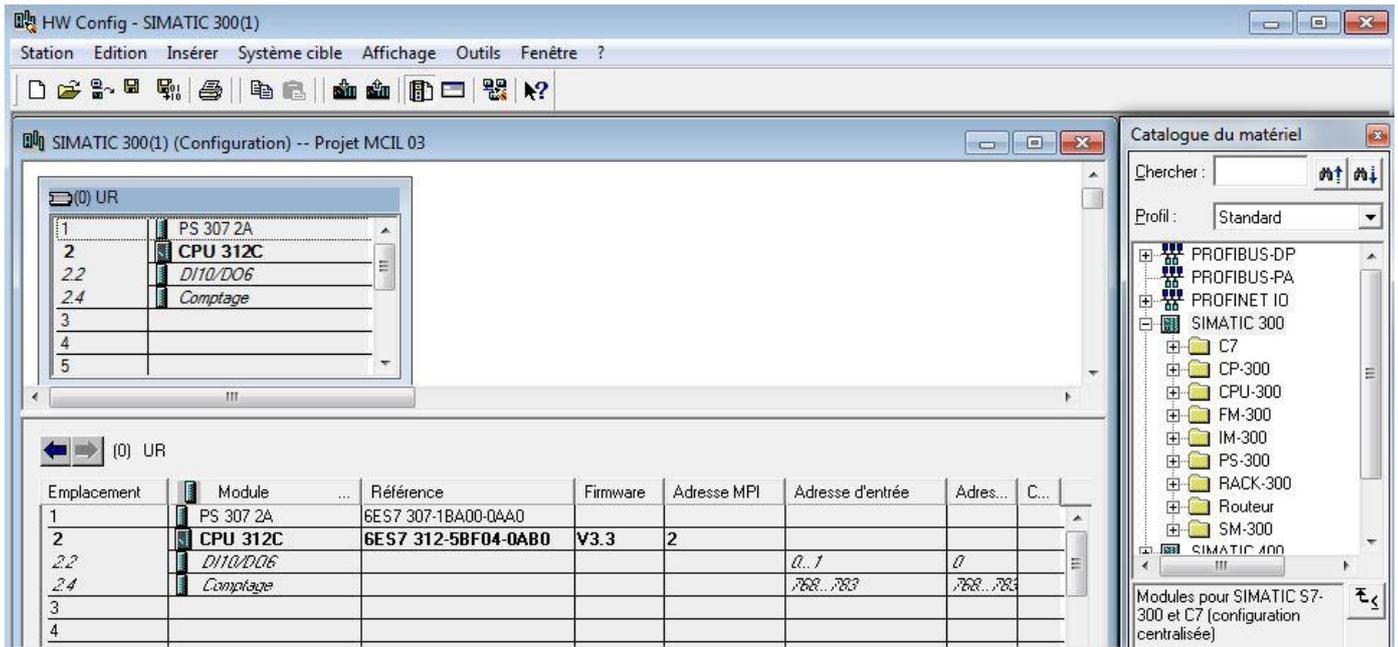


Figure I-4 : Configuration matérielle.

2- Table des mnémoniques :

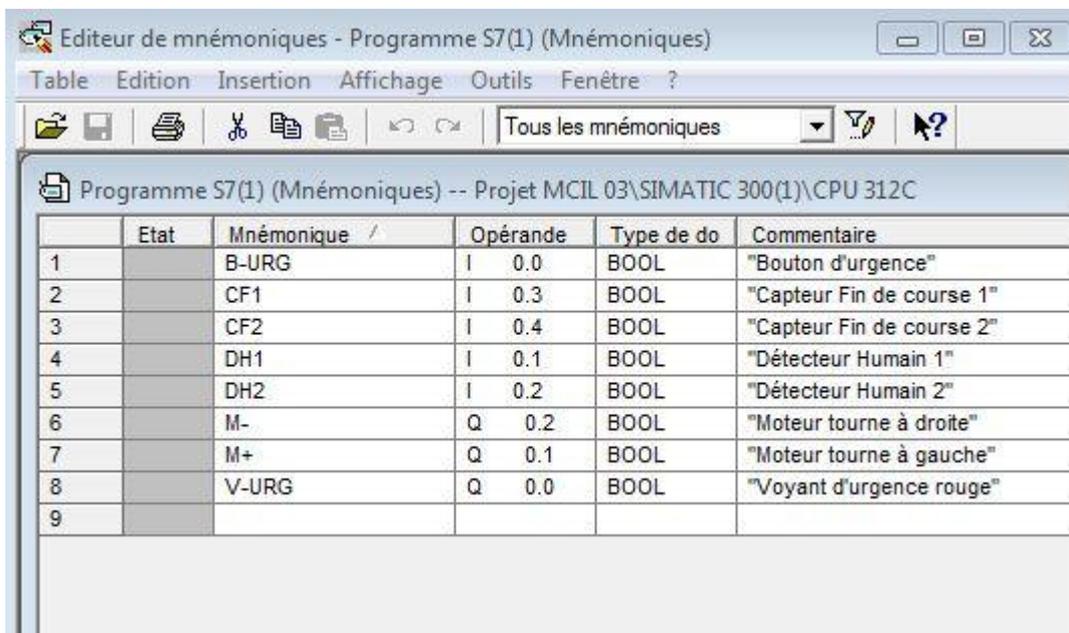


Figure I-5 : Table des mnémoniques.

3- Programme en langage à contacts sur le bloc OB1 :

☐ Réseau 1: "l'ouverture de la porte"

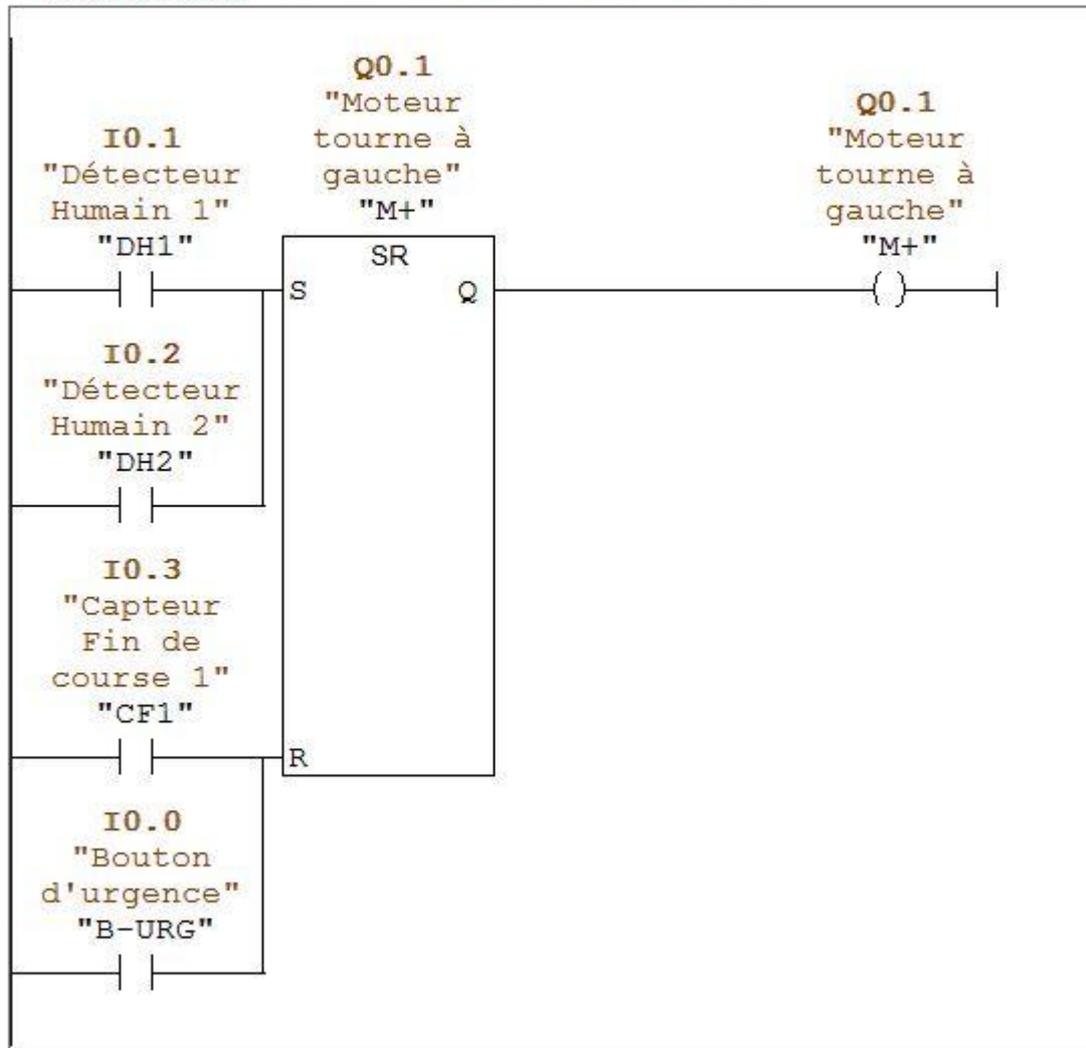


Figure I-6 : Réseau 1, l'ouverture de la porte.

☐ Réseau 2 : "La temporisation 5s"

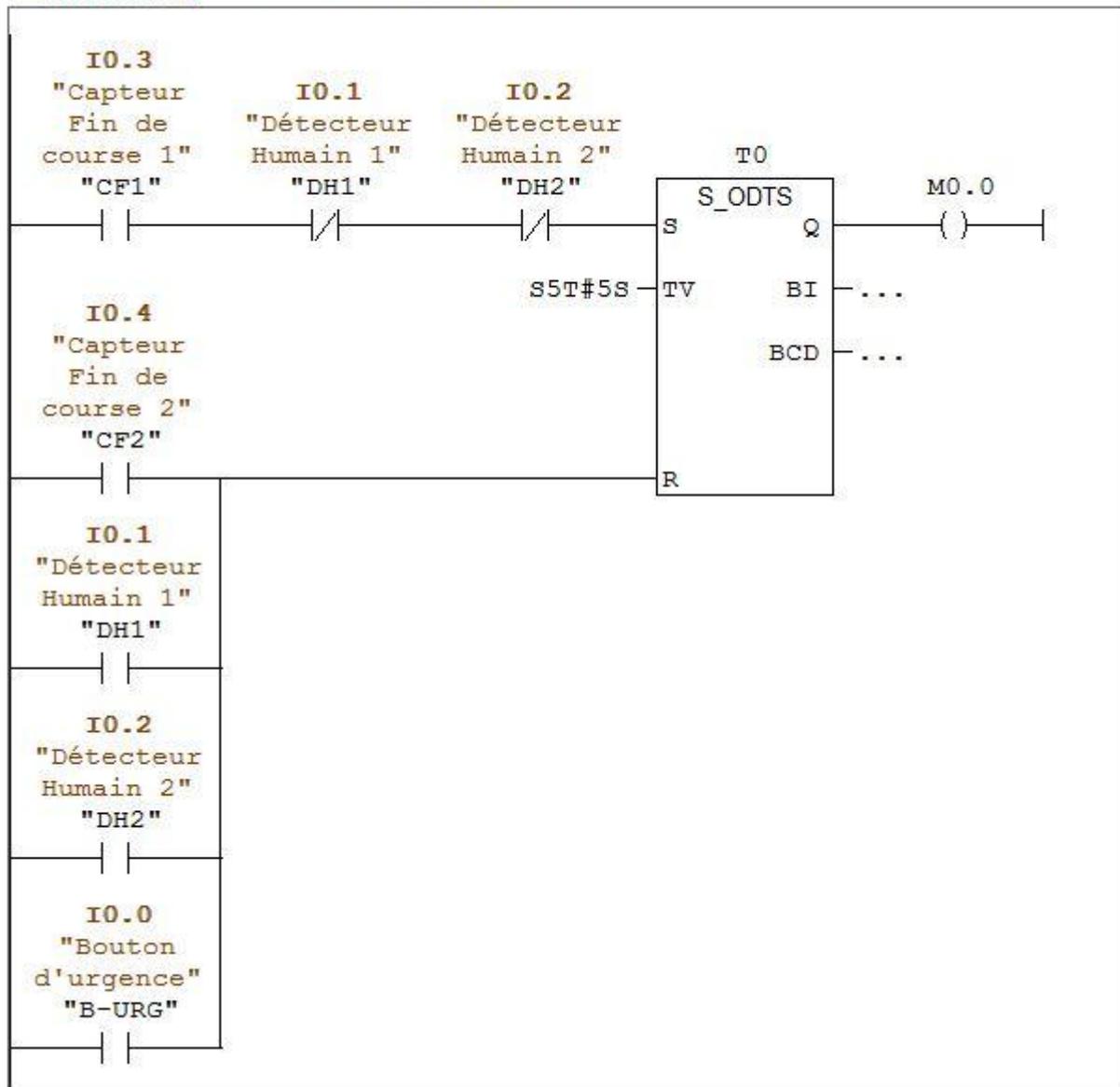


Figure I-7 : Réseau 2, la temporisation 5secondes.

☐ Réseau 3: "fermeture de la porte"

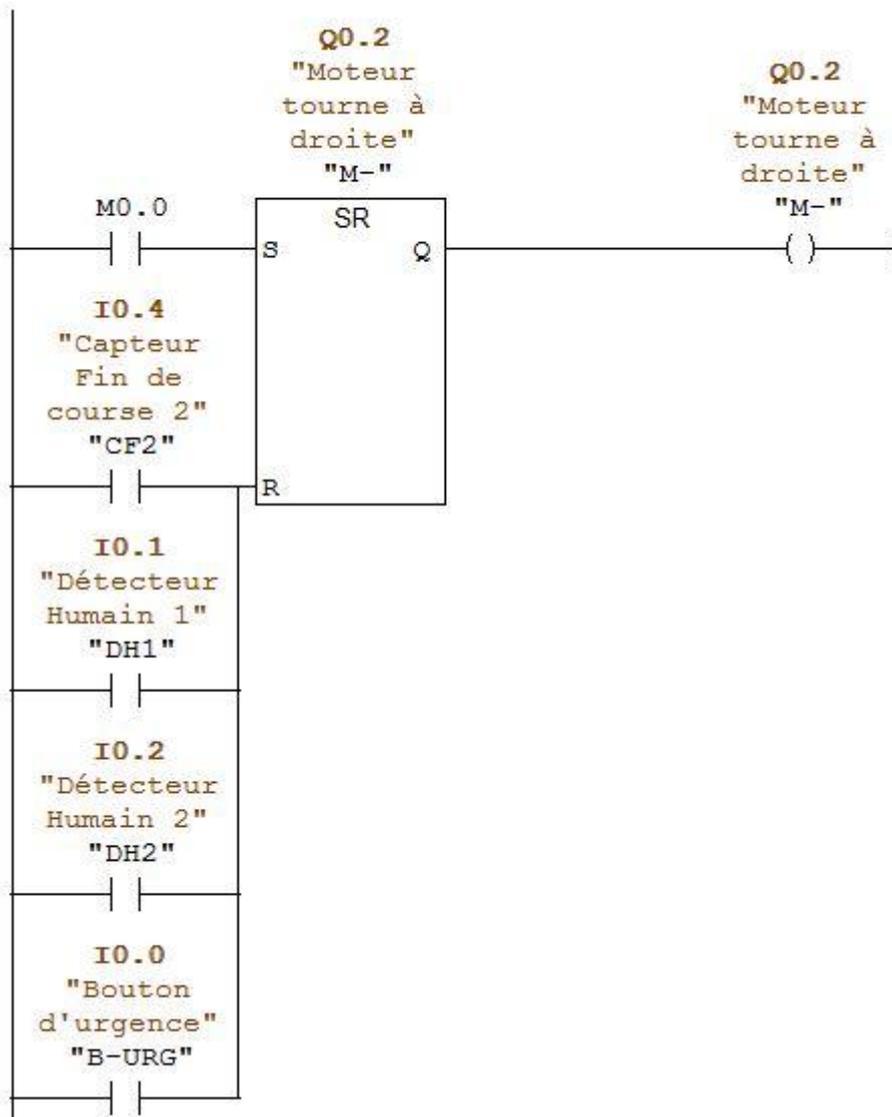


Figure I-8 : Réseau 3, fermeture de la porte.

☐ Réseau 4: "cas d'urgence"

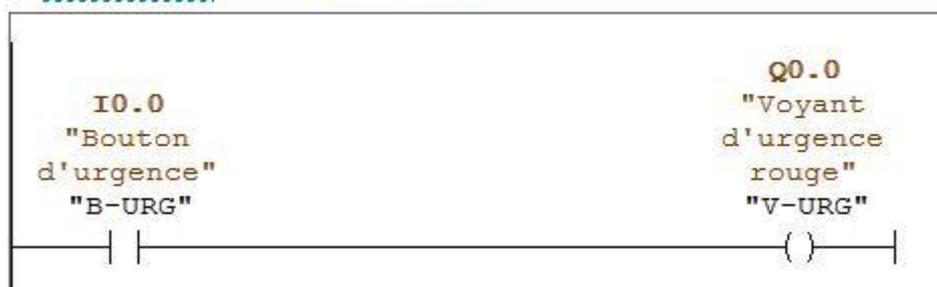


Figure I-9 : Réseau 4, cas d'urgence.

I- Simulation :

Pour valider le programme précédent, on utilise le simulateur S7-PLCSIM. Les résultats de simulation sont représentés dans les figures suivantes :

1- lorsqu'on active le détecteur de proximité photoélectrique DH1 ou DH2 (détection humaine), le moteur commence à tourner vers la gauche (l'ouverture de la porte).

▣ Réseau 1: "l'ouverture de la porte"

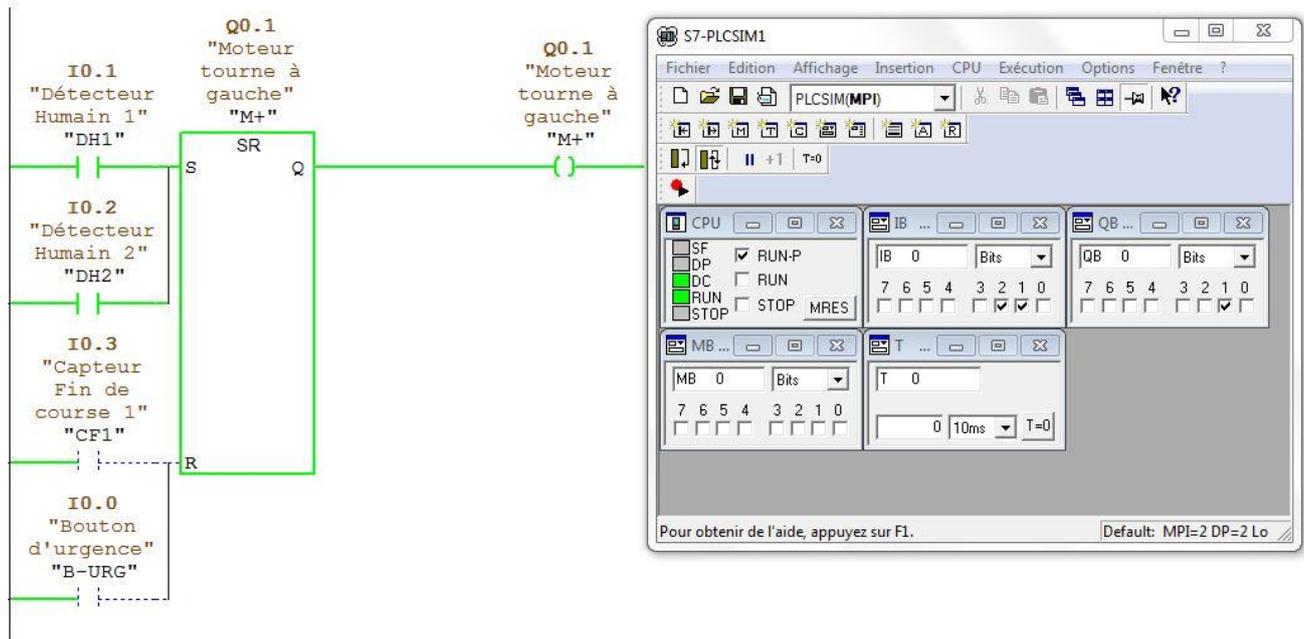


Figure I-10 : Simulation d'ouverture de porte.

2- La porte s'ouvre jusqu'à l'activation du capteur fin de course CF1, dans ce cas la temporisation de 5 secondes démarre à condition que le détecteur DH1 et DH2 ne soit pas activé.

Chaque fois qu'on active le détecteur DH1 ou DH2 pendant la temporisation, cette dernière redémarrera à nouveau.

▣ Réseau 2: "La temporisation 5s"

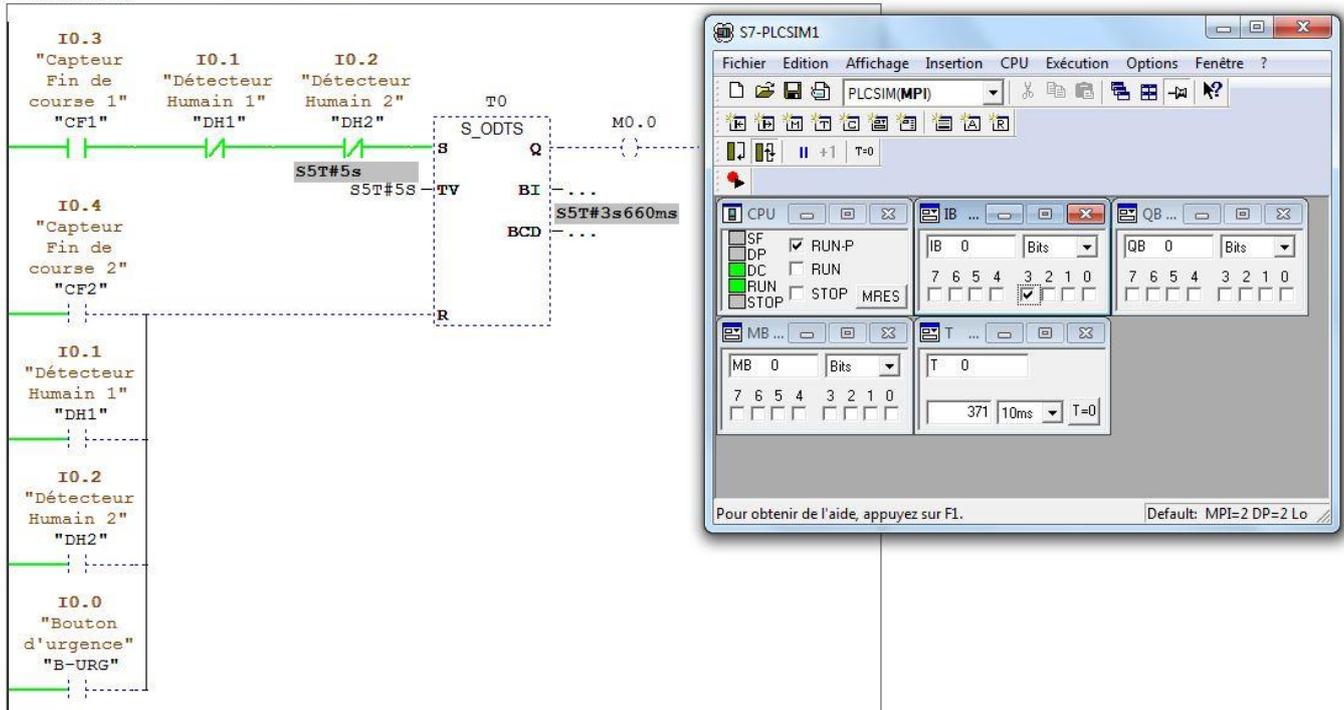


Figure I-11 : Simulation de temporisation 5s.

3- Après l'expiration de temporisation, le moteur commence à tourner vers la droite (fermeture de la porte) jusqu'à l'activation du capteur fin de course CF2.

Lorsqu'on active le détecteur DH1 ou DH2 pendant la fermeture de la porte, le moteur commence à tourner vers la gauche (ré ouvrir la porte).

▣ Réseau 3 : "fermeture de la porte"

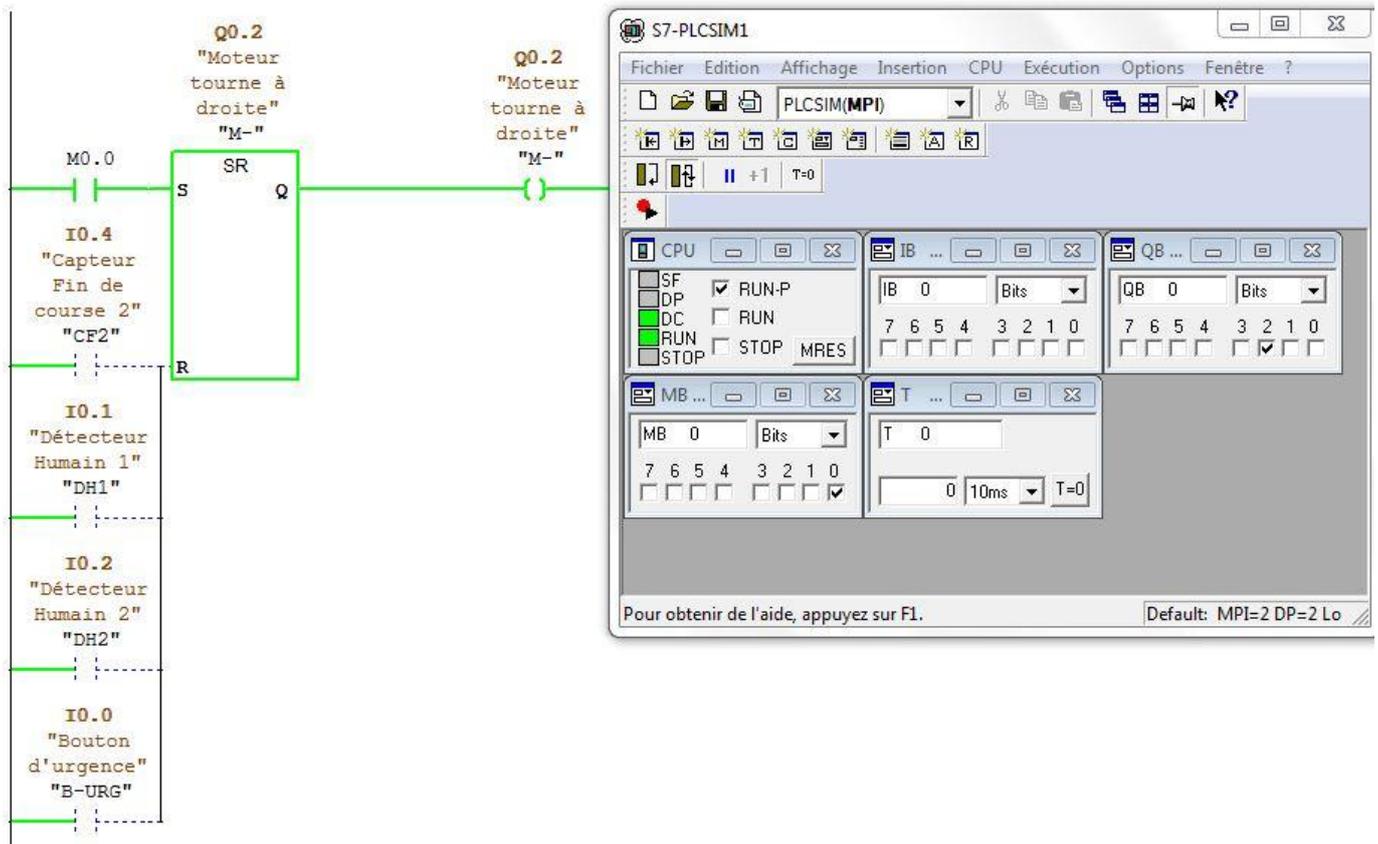


Figure I-12 : Simulation de fermeture de porte.

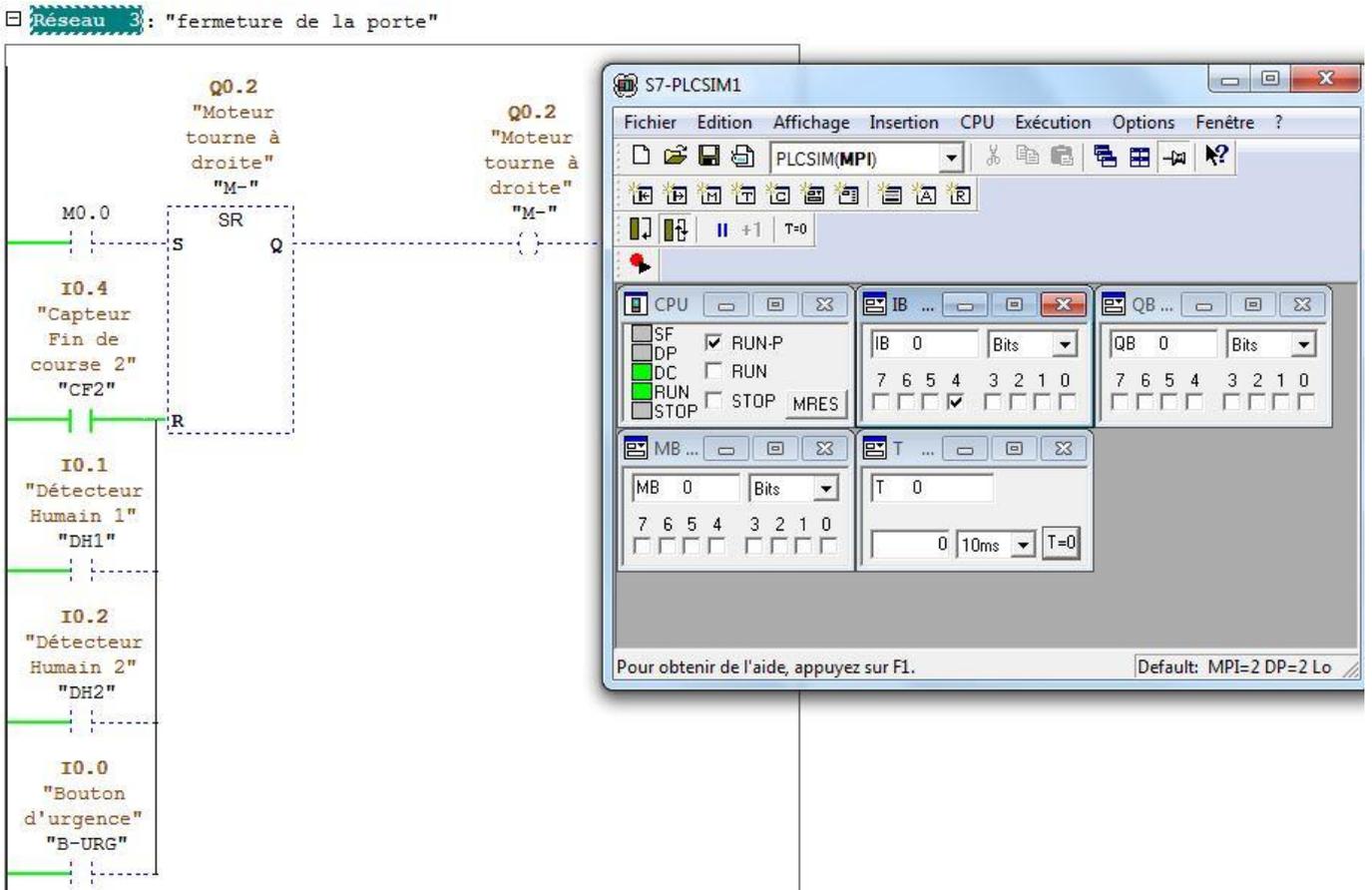


Figure I-13 : Simulation de la porte qui est complètement fermée.

4- lorsque on active le bouton d'urgence B-URG, le voyant rouge V-URG s'allume et le moteur s'arrête à ce moment (la porte n'est pas ré ouverte ou refermée). C'est le cas d'urgence.

Pour désactiver l'urgence, nous devons désactiver le bouton d'urgence.

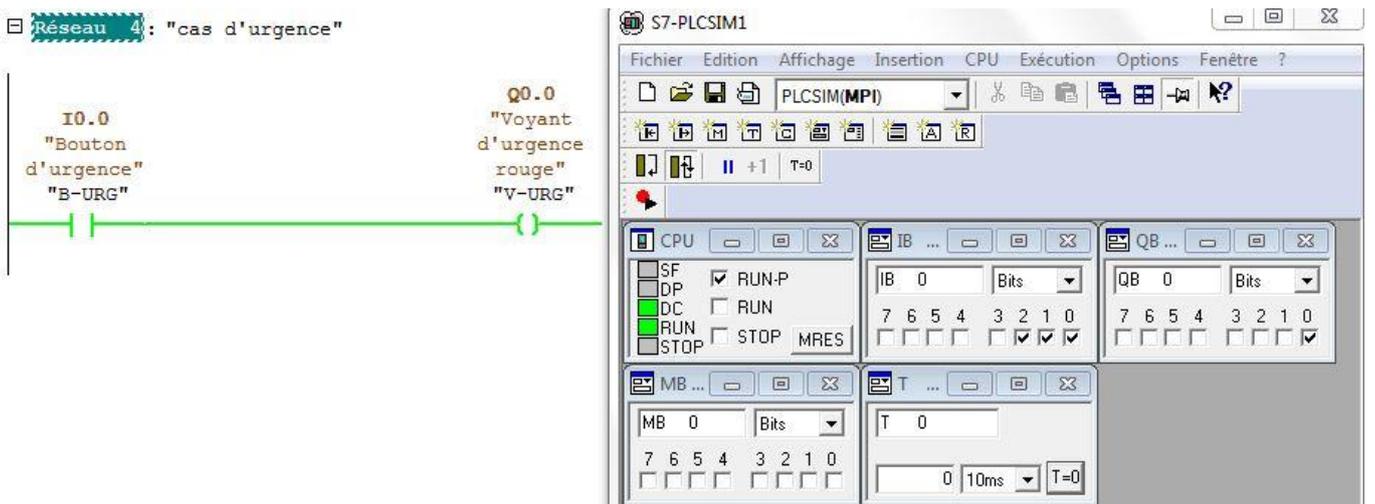


Figure I-14 : Simulation du cas d'urgence.

Conclusion générale :

Notre projet a pour objectif d'automatiser une porte coulissante en utilisant l'automate SIMATIC S7-300 qui est programmé à l'aide du logiciel STEP 7.

Dans la première partie on a présenté une étude sur le système automatisé qui utilise des équipements électriques et électroniques. Cette étude permet la possibilité de sélectionner les équipements nécessaires à la mise en œuvre de ce projet. Dans la deuxième partie on a simplifié les étapes réalisées ce projet.

Enfin, nous espérons que notre travail sera une meilleure solution à la problématique posée et servira comme une base de départ pour notre vie professionnelle et être bénéfique aux promotions futures.

Références bibliographiques :

- [1]: ADPS SARL. Entreprise d'installation et de motorisation des portes coulissante automatique. Consulté le 02 août 2021. L'adresse URL : <https://www.adps-automatisme.com/>
- [2]: Dr. Kebabla Mebarek. Cour Architecture des Systèmes Automatisés. Consulté le 03 août 2021. L'adresse URL : <http://staff.univ-batna2.dz/kebabla-mebarek>
- [3]: Dr. Bendjerad Adel. Cour sur les capteurs. Présenté en 2020. Consulté le 04 août 2021. L'adresse URL : http://staff.univ-batna2.dz/bendjerad_adel/classes
- [4]: Gareth Conner. Scenic Automation Handbook. New York, NY: Routledge, 2018.
- [5]: Gary Frigyes, Ed Myers and Jeff Allison. Fundamentals of Photoelectric Sensors. June 13, 2010. Pepperl + Fuchs GmbH. Consulté le 06 août 2021. L'adresse URL : <https://www.automation.com/en-us/articles/2014-1/fundamentals-of-photoelectric-sensors>
- [6]: William Bolton. Automates Programmables Industrielles. Traduction de Herve Soulard. 2eme édition. 2015 Dunod.
- [7]: Marco Ceccarelli. Water Dancer II-A: A Non-Tethered Tele controllable Water Strider Robot. Article in International Journal of Advanced Robotic Systems. September 2011.
- [8]: Electronics tutorials. 555 Timer Tutorial. Consulté le 15 aout 2021. L'adresse URL : <https://www.electronics-tutorials.ws/waveforms/555-circuits-part-1.html>
- [9]: Frank D. Petruzella. Programmable logic controllers. Fifth edition.
- [10]: SIEMENS. SIMATIC Système d'automatisation S7-300 Caractéristiques des modules Manuel. 02/2013.
- [11]: SIEMENS. SIMATIC Système d'automatisation S7-300 Installation et configuration CPU 312 IFM-318-2 DP. Edition 06/2003 A5.
- [12]: SIEMENS. SIMATIC Programmer avec STEP 7. Manuel 05/2010.
- [13]: Chahrazed BENMERZOUG. Automatisation & Supervision d'une porte coulissante. UNIVERSITE BADJI MOKHTAR ANNABA. Mémoire Master. 2018.