

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne démocratique et populaire

وزارة التعليم العالي و البحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique

جامعة سعد دحلب البليدة
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculté de Technologie

قسم الإلكترونيك
Département d'Électronique



Mémoire de Master

Mention Électronique
Spécialité Automatique

présenté par

LABATI SABIHA

&

DJIDJLI FATMA ZOHRA ZINEB

Asservissement des actionneurs d'une chaine de conditionnement pédagogique

Proposé par : Mr BOUALEM.KAZED

Année Universitaire 2015-2016

Remerciements

Nos remerciements Allah qui nous donne le courage pour faire ce travail

Nos remerciements les plus sincères à tout les personnes qui auront contribué de près ou de loin à l'élaboration de ce projet et plus particulièrement notre promoteur Mr KAZED.B pour l'aide et le temps qu'il a bien voulu nous consacrer.

Nous tenons à remercier les membres de jury Mr MADDI et Mr FAS qui ont accepté d'évaluer notre thèse et notre travail.

Nos remerciements à Mr FAS.L pour son aide.

Nous tenons aussi à remercier tous les professeurs qui nous ont encadrés durant notre cursus scolaire.

Labati & djidjli



Dédicaces

Je dédie ce modeste travail

A Mon cher Père qui grâce à leur encouragement je suis là

A Ma cher Mère qui ma laissée pas

A Mes chers frères et Mes chères sœurs

A Mes chères familles.

A tous mes Amis

A toute la promotion

A tous ceux qui m'ont aidé durant ma vie scolaire



Sabiha





Dédicaces



Je dédie ce modeste travail

A mon cher père qui est le plus proche et grâce a lui je suis la

A ma chère mère qui m'a aidé beaucoup

Mes chers frères Mohamed et Amer

A Ma familles la plus large

A tous mes Amis

A tous ceux qui m'ont aidé durant ma vie



Zineb



ملخص:

الهدف من هذا المشروع هو تحقيق التحكم في وضعية ثلاثة محركات ذو تيار مستمر. ويشكل هذا العمل إكمال التطور الذي قام به زملائنا العام الماضية الذي يشمل خط التعبئة والتغليف. ليستفيد منه طلاب فرع الآلية.

كلمات المفاتيح : CUBLOC, PID.

Résumé :

Le but de ce projet consiste à réaliser l'asservissement, en position, de 3 moteurs à courant continu. Ce travail constitue un perfectionnement de ce qui a été entrepris par nos collègues de l'année dernière ; à savoir la réalisation d'une station de conditionnement pédagogique permettant, à termes, de servir comme base pour effectuer des travaux pratiques pour les étudiants inscrits dans la filière automatique.

Mots clés : CUBLOC, PID.

Abstract :

The goal of this project is to implement a closed loop control system for 3 DC motors. This work follows another project that has been carried out last year by our colleagues; Their task was to design a packaging machine that could be used as a learning base for the students registered for a degree in automatics.

Keywords : CUBLOC, PID.

Listes des acronymes et abréviations

P.O : Partie Opérative.

P.C : Partie Commande.

PLC : « Programmable Logic Controller » automate programmable industriel

GPAO : Gestion de la Production Assistée par Ordinateur.

GMAO : Gestion de Maintenance Assistée par Ordinateur.

G.P : Gestion de Production.

BO : boucle ouverte.

BF : boucle fermé.

FBD: « Function Blok Diagram » schéma par blocs.

SFC: « Sequentiel Function Chart » GRAFcet

LD: « Ladder Diagram » langage a contact.

IL: « Instruction List » List d'instruction.

ST: « Structured Text » text structure.

E/S: Entrée/Sortie.

BP: Bouton poussoir.

TOR: Tout Ou Rien

LED/DEL: « Light Emiting Diode » diode émettrice lumière (diode électroluminescente).

PWM: « Pulse Width Modulation»

BAU: Bouton d'Arrêt d'Urgence.

BASIC: Beginner's All-purpose Symbolic Instruction Code.

TTL: Transistor Transistor Logique.

UART: Universal Asynchronous Receiver Transmitter.

RTC: Real Time Clock.

NC: Normally Close.

NO: Normally Open.

SRAM: Static Random-Access Memory.

EEPROM: Electrically Erasable Programmable Read-Only Memory.

Instr/sec: Instruction / seconde.

CC : courant continu.

PID : proportionnel intégrateur dérivateur.

Table des matières

Remercîments

Dédicaces

Dédicaces

Résumé

Listes des acronymes et abréviations

Table des matières

Introduction générale.....1

Chapitre 01 : les systèmes automatisée

1.1.	Introduction.....	2
1.2.	Historique.....	2
1.3.	Les systèmes automatisés	4
1.3.1.	Définition	4
1.3.2.	Objectif de l'automatisation.....	5
1.3.3.	Structure générale d'un système automatisé.....	5
1.3.4.	Les modes de commande.....	8
1.3.5.	Les technologies des automatismes.....	9
1.3.6.	Les langages de programmation.....	10
1.4.	Les actionneurs	14
1.4.1.	Définition.....	14
1.4.2.	Les type des actionneurs.....	14
1.4.3.	Les vérins pneumatiques	15
1.5.	Pré-Actionneur.....	17
1.5.1.	Définition.....	17
1.5.2.	Distributeur pneumatique.....	17
1.6.	Les capteurs.....	21
1.6.1.	Définition.....	21

1.6.2. Les type de capteurs.....	22
1.7. Conclusion.....	23

Chapitre 02 : constitution mécanique des la chaine

2.1. Introduction.	24
2.2. Constitutions de la chaine.....	24
2.2.1. Les Actionneur.....	24
2.2.2. Pré-actionneurs (Distributeur).....	29
2.2.3. Les capteurs.....	30
2.2.4. L'énergie pneumatique.....	32
2.2.5. Les mécanismes de transformation de mouvement.....	34
2.3. Conclusion	35

Chapitre 03 : partie électronique

3.1. Introduction.....	36
3.2. Carte de commande.....	37
3.2.1. Définition.....	37
3.2.2. Le microcontrôleur CUBLOC.....	37
3.2.3. Constitution de la carte de comman.....	45
3.3. Carte de puissance	46
3.3.1. Définition.....	46
3.3.2. Le circuit L298N	47
3.3.3. Notion du signale PWM.....	49
3.3.4. Diode de roue libre.....	50
3.3.5. Les relais.....	50
3.3.6. Constitution de la carte de puissance.....	52
3.4. La carte ARDUINO.....	54
3.4.1. Définition.....	54

3.4.2. Constitution de la carte ARDUINO.....	54
3.4.3. La structure de programmation.....	56
3.4.4. Principe de fonctionnement.....	58
3.4.5. But de l'utilisation.....	59
3.4.6. Les bonnes raisons de choisir ARDUINO.....	59
3.5 Conclusion	59

Chapitre 4 : programmation de la machine

4.1. Introduction	60
4.2. L'asservissement des moteurs	60
4.2.1. Régulateur proportionnel (P)	60
4.2.2. Régulateur intégrateur pur (I)	61
4.2.3. Régulateur dérivateur pur (D)	61
4.2.4. Régulateur proportionnel intégrateur dérivateur (PID).....	61
4.2.5. Identification du moteur à courant continu.....	62
4.3. Partie programmation.....	67
4.3.1. LADDER du CUBLOC	67
4.3.2. ARDUINO	70
4.4. Principe de fonctionnement de la chaine.....	73
4.4.1. Cahier des charges.....	73
4.4.2. Organigramme	74
4.4.3. Cycle de travail.....	75
4.4.4. Schéma de la chaine	78
4.5. Conclusion	80
Conclusion générale.....	81

Liste des figures

Figure1.1 .système automatisé.....	4
Figure1.2. structure générale d'un système automatisé.....	5
Figure 1.3. la décomposition PO-PC.....	6
Figure1.4. système en Bo.....	8
Figure1.5. système en BF.....	9
Figure1.6. programmation en langage FBD.....	11
Figure1.7. exemple d'un programme Grafcet.....	11
Figure1.8. programmation en langage LADDER.....	12
Figure1.9. programmation en langage IL.....	13
Figure1.10. programmation en langage ST.....	13
Figure1.11. quelque type d'actionneurs.....	15
Figure1.12. constituants de base d'un vérin.....	15
Figure1.13. vérin simple effet.....	16
Figure1.14. vérin double effet.....	16
Figure1.15. un distributeur 4/2 forme didactique.....	17
Figure1.16. représentation schématique.....	18
Figure1.17. distributeur 3/2 bistable.....	20
Figure1.18. distributeur 3/2 monostable.....	20
Figure1.19. distributeur 4/2 en état de repos et travail.....	21
Figure1.20. quelque type de capteur.....	22
Figure2.1. Caractéristique interne d'un moteur CC.....	27
Figure2.2. Moteur EMG30.....	27
Figure2.3. Pince de conditionnement.....	28
Figure 2.4. Distributeurs pneumatique.....	29
Figure 2.5. Principe de fonctionnement d'un distributeur 3/2 monostable.....	30
Figure2.6. Principe de fonctionnement d'un distributeur 5/2 bistable.....	30

Figure 2.7. Capteur infrarouge.....	30
Figure 2.8. La détection de capteur infrarouge.....	31
Figure 2.9. Manomètre analogique.....	31
Figure 2.10. Bouton d'arrêt d'urgence.....	32
Figure 2.11. Compresseur pneumatique.....	33
Figure 2.12. BIELLE – MANIVELLE.....	34
Figure 2.13. Système vilebrequin.....	34
Figure 2.14. Crémaillère.....	35
Figure 2.15. Vis - écrou.....	35
Figure 3.1. Schéma synoptique générale de la chaine de conditionnement.....	36
Figure3.2. La forme d'un PLC traditionnel et le module CUBLOC.....	37
Figure 3.3. Structure interne de CUBLOC.....	38
Figure3.4. Schéma d'un programme en LADDER.....	40
Figure3.5. Exemple d'un programme en BASIC.....	41
Figure3.6. Les possibilités de la programmation.....	42
Figure 3.7. Les deux langages de CUBLOC studio.....	43
Figure 3.8. Le module CB280.....	44
Figure 3.9. Alimentation de module CB280.....	45
Figure3.10. La carte de commande.....	46
Figure 3.11. Pont en H.....	47
Figure 3.12. Le circuit L298N.	47
Figure3.13. Schéma simplifié du pont en H.....	48
Figure 3.14. Schéma de moteur à l'arrêt, sens direct et le sens inverse.....	49
Figure 1.15. Signal PWM.....	50
Figure 3.16. Diode de roue libre.....	50
Figure 3.17. Quelques types des relais.....	51
Figure 3.18. Relais électromécanique et leur symbole.....	51

Figure 3.19. Carte de puissance.....	52
Figure 3.20. L'interface entre la carte de commande et la carte de puissance.....	53
Figure 3.21. Le symbole d'ARDUINO.....	54
Figure 3.22. Quelques cartes ARDUINO.....	54
Figure 3.23. La constitution de la carte ARDUINO(MEGA).....	55
Figure 4.1. Fenêtre de l'instruction 'ident' de Matlab.....	62
Figure 4.2. Signal d'entrée et de sortie de notre modèle superposée avec la vitesse mesurée.....	62
Figure 4.3 Signal d'entrée et de sortie de deux données.....	63
Figure 4.4. Réponse de modèle et la réponse mesure.....	63
Figure 4.5. La réponse à l'échelon du modèle.....	64
Figure 4.6. Schéma bloc du correcteur PID en régulation de vitesse et de position.....	64
Figure 4.7. Evolution de la consigne (bleu) et de la position (rouge).....	65
Figure 4.8. Evolution du signal de control (bleu) et de vitesse (rouge).....	65
Figure 4.9. Signal issu du contrôleur de vitesse (PWM (l'entée de modèle)).....	66
Figure 4.10. Programme en LADDER.....	67
Figure 4.11. Barre d'outils de CUBLOC STUDIO.....	68
Figure 4.12. Programme en LADDER actif.....	68
Figure 4.13. La tige filetée.....	71
Figure 4.14. Distance qui le bras va déplacer.....	71
Figure 4.15. Cycle de travail.....	77
Figure 4.16. Schéma pneumatique.....	78
Figure 4.17. Schéma synoptique de la chaine de conditionnement.....	79
Figure 4.18. Schéma réel de la chaine.....	80

Liste des tableaux

Tableau1.1. les éléments de PO et PC.....	7/8
Tableau1.2. les différentes commandes de distributeur.....	19
Tableau3.1. table comparative.....	41
Tableau4.1. Les registres utilisés par le LADDER.....	69
Tableau4.2. Table des mesures.	72
Tableau4.3. Table de vérité de moteur (avant/arrière)	72
Tableau 4.4. Table de vérité de moteur (descende/monte).....	72
Tableau 4.5. Table de vérité de moteur (rotation de 90).....	72

Introduction générale

On appelle système automatique un système pour lequel tout ou une partie du savoir faire est confié à une machine. Aujourd'hui, les systèmes sont conçus avec un comportement confié à une informatique locale, on les qualifie donc des systèmes automatiques et non plus des systèmes automatisés.

Notre projet représente une station utilisée dans les systèmes automatisés, qui est appelé chaine de conditionnement, elle contient plusieurs éléments mécaniques et des cartes électroniques. Le fonctionnement de cette station dépend généralement d'un module de type CUBLOC qui dispose de deux possibilités de programmation en même temps (LADDER et BASIC).

Ce travail vient compléter celui entamé par nos collègues de l'année écoulée ou Il y'a avait quelques problèmes dans les moteurs, un manque de précision dans la rotation des 3 moteurs servant à placer la pince de préhension à l'endroit désiré, ceci se traduisait par un mouvement brusque de la pince et donc des erreurs imprévisibles au niveau du fonctionnement du système dans sa globalité. Pour résoudre ce problème nous nous sommes fixées comme objectif d'implémenter 3 correcteurs PID pour les 3 moteurs impliqués dans le positionnement et la rotation de la pince de préhension des objets à conditionner.

A la suite de cette introduction nous donnerons les détails de ce travail dans les chapitres suivants, on va commencer par des généralités sur les systèmes automatisés, après on va parler de la constitution mécanique de la chaine de conditionnement et de la constitution électronique (les cartes électronique et leurs caractéristiques) ; nous terminerons à la fin avec la partie programmation de notre machine qui contient les logicielles utilisés pour le fonctionnement de cette chaine.

1.1 Introduction

L'automatique est une science qui traite de la modélisation, de l'analyse, de l'identification et de la commande des systèmes dynamiques. L'automatique permet de contrôler un système en respectant un cahier des charges (rapidité, précision, stabilité...). Les objets que l'automatique permet de concevoir pour procéder à l'automatisation d'un système (automates, régulateurs, etc.), s'appellent les automatismes ou les organes de contrôle-commande d'un système piloté.

L'automatisme est un dispositif dont le fonctionnement ne nécessite pas l'intervention de l'homme. C'est une machine qui, par le moyen de dispositifs mécaniques, pneumatiques, hydrauliques ou électriques. Un système est dit automatisé s'il exécute toujours le même cycle de travail après avoir reçu d'un opérateur. Aujourd'hui le monde fait un grand développement dans le domaine automatique qui facilite tous les moyens de travail, actuellement les usines presque ne besoin pas de main œuvre et le rôle de l'homme c'est juste la surveillance et la régularité des machines automatiques.

1.2 Historique

L'histoire des systèmes automatiques peut se diviser en trois époques.

La **première époque**, que l'on peut qualifier de préhistoire de l'automatique s'étend de l'antiquité au milieu du siècle dernier. Des inventeurs géniaux ont conçu des systèmes automatiques de manière purement intuitive. Dès 250 avant J-C, nous avons des exemples de régulation de niveau : l'horloge automatique à eau (clepsydre) de Ktesibios, la lampe à huile de Philon de Bizance et la machine à doser le vin de Héron

d'Alexandrie (50 ans avant J-C.). Plus tard, Réaumur (1683–1757), Watt (1736–1819) et son régulateur, Jacquard (1752–1793) et son métier à tisser à cartes perforées font progresser l'automatisation.

La **seconde époque**, à partir du milieu du 19ème siècle, est caractérisée par la théorie du bouclage et les applications de l'algèbre de Boole. Les premiers travaux sur le bouclage sont dus à Maxwell (1868), à Routh avec son critère algébrique (1872) et à Hurwitz (1890).

L'étude analytique de la stabilité du régulateur de Watt fut commencée par Maxwell en 1868 et complétée en 1876 par Wichnegradsky. L'étude des systèmes bouclés doit beaucoup à l'approche fréquentielle de Nyquist, Bode, Black, Nichols, Hall, Evans, qui ont laissé leur nom à des représentations et qui ont publié la plupart de leurs résultats à la fin de la seconde guerre mondiale. Les premières implémentations des systèmes de commande à cette époque reposaient sur l'utilisation de dispositifs électroniques à lampes.

La **troisième époque** débute avec les années cinquante. L'apparition des calculateurs numériques révolutionne le monde de l'automatique. La puissance de calcul disponible fait naître les méthodes dites de l'automatique« moderne » ou « avancée ». Parmi les faits marquants on peut citer :

- l'introduction de la représentation d'état, particulièrement bien adaptée à l'utilisation des calculateurs numériques pour l'étude et la commande des systèmes complexes et multi variables (Kalman 1960).
- le développement des méthodes d'étude des systèmes non-linéaires (Kockenburger, Cypkin) et des systèmes échantillonnés (Jury, Ragazzini).
- la prise en compte des phénomènes aléatoires dans les théories récentes comme celles de Kalman, de Bucy.

A ce moment-là, le développement dans le domaine des sciences en général, en particulier l'automatisation ne s'arrête pas. [1]

1.3 Les Systèmes automatisés

1.3.1 Définition

Un automatisme est un sous-ensemble d'une machine, destinée à remplacer l'action de l'être humain dans des tâches en générales simples et répétitives, réclamant précision. On est passé d'un système dit manuel, à un système mécanisé, puis au système automatisé.

L'automatisation d'un procédé consiste à en assurer la conduite par un dispositif technologique. L'intervention d'un opérateur reste souvent nécessaire pour assurer un pilotage global du procédé (spécification des consignes), pour surveiller les installations et reprendre en commande manuelle tout ou partie du système en cas de besoin. [2]

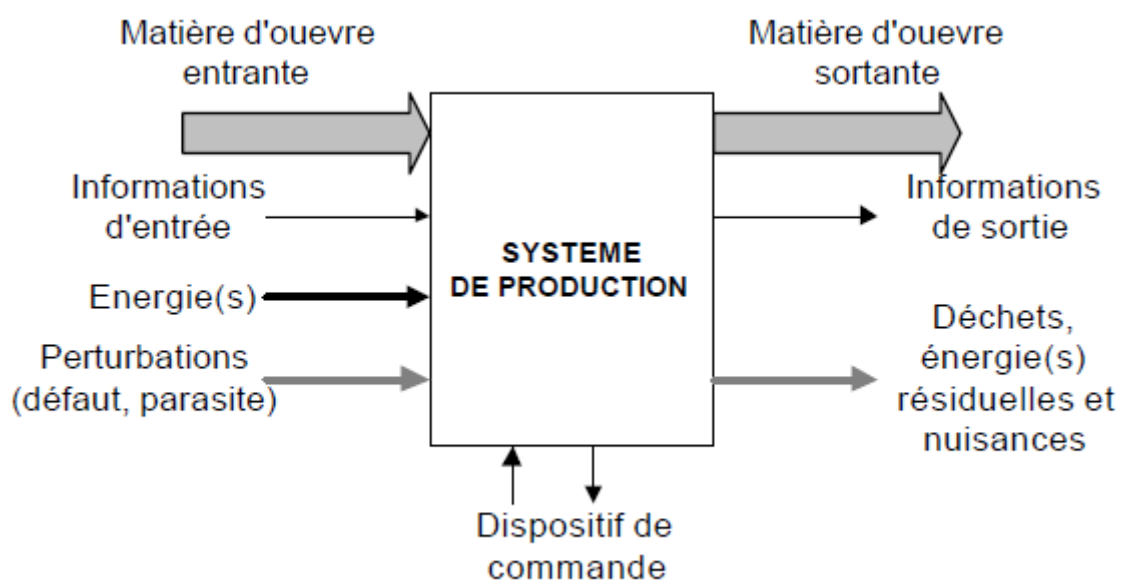


Figure 1.1. Système automatisé.

1.3.2 Objectifs de l'automatisation

La compétitivité de l'entreprise et des produits. Cette compétitivité passe par la qualité, la maîtrise des coûts et l'innovation. Cela induit une disponibilité à tous les niveaux. On cherche donc à améliorer la productivité. L'amélioration des conditions de travail, et surtout la sécurité, fait partie des objectifs de l'automatisation.

Le but de l'automatisation est donc :

- Eliminer les tâches répétitives et Simplifier le travail de l'humain.
- Réaliser les opérations impossibles à contrôler manuellement.
- Augmenter la sécurité.
- Rechercher de coût plus bas, par réduction des frais de main d'œuvre, d'économiser les matières premières et l'énergie.
- Améliorer la qualité.

Le système automatisé permet d'obtenir de façon reproductible la valeur ajoutée.

1.3.3 Structure générale d'un système automatisé

Un système automatisé peut être décomposé en trois grandes parties : tout système automatisé se décompose en 3 parties interdépendantes : la partie opérative (PO), partie commande (PC) et l'interface homme machine. Les informations échangées entre ces parties sont des informations internes au système alors que les consignes, signalisation proviennent de l'extérieur (**Figure1.2**). [2]

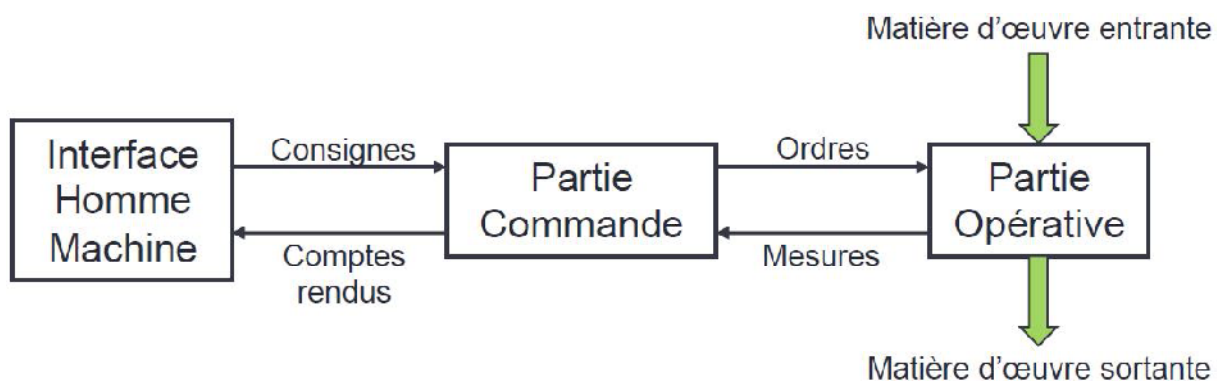


Figure 1.2. Structure générale d'un système automatisé.

- **La partie Commande PC**

Elle permet à l'opérateur d'établir le programme qui commande l'automate. La partie Commande reçoit les consignes d'un opérateur. Elle adresse des ordres à la partie opérative. Inversement le PC reçoit des comptes rendus de la partie opérative et envoie des signaux à l'opérateur.

- **La partie Opérative PO**

Reçoit les ordres de la Partie Commande. Dans la Partie Opérative il y a des actionneurs (vérins, moteurs,) qui exécutent les ordres reçus.

- **L'interface**

Elle relie Partie Opérative et Partie Commande. C'est un système de traduction d'informations entre PC et PO.

- **La décomposition PO-PC**

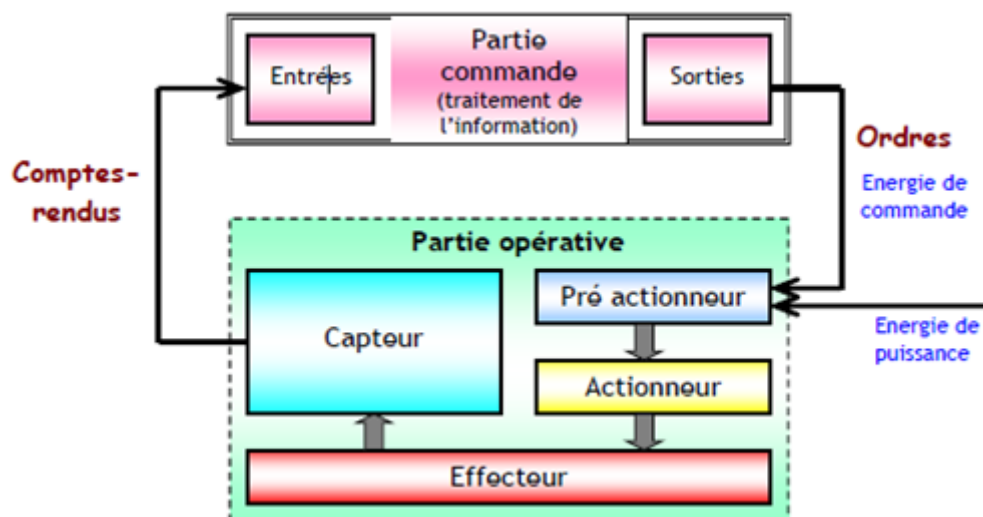


Figure 1.3. La décomposition PO-PC.

Au centre de la Partie Commande, le « TRAITEMENT » est la convergence de 3 dialogues qu'il coordonne : (**Figure 1.3**) [4]

➤ **Le Dialogue avec la machine**

- Commande des actionneurs (moteurs, vérins...) via les pré-actionneurs (Contacteurs, distributeurs, variateurs...)

- Acquisition des signaux en retour par les capteurs qui rendent compte de l'évolution de la machine.

➤ **Le Dialogue homme-machine**

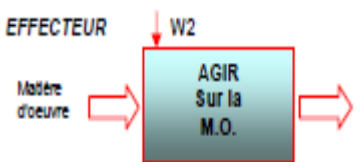
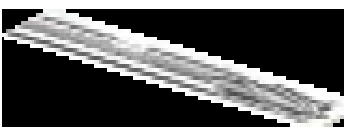
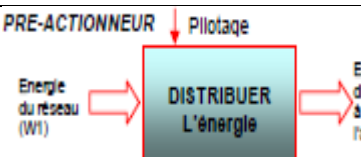
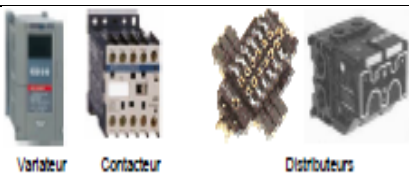
Pour exploiter, régler et dépanner la machine, le personnel émet des consignes et reçoit des informations en retour.

➤ **Le Dialogue avec d'autres machines**

Plusieurs machines peuvent coopérer dans une même production.

Leur coordination est assurée par le dialogue entre les parties commandes.

Les éléments de la partie opérative et la partie commande (tableau 1.1)

Diagramme fonctionnel	Exemples	Commentaires
EFFECTEUR 	Fraise, Foret, Mors d'étau, pince de robot.... 	Les effecteurs sont multiples et variés et sont souvent conçus spécialement pour s'adapter à l'opération qu'ils ont à réaliser sur la Matière d'œuvre. Ils reçoivent leur énergie des actionneurs
ACTIONNEUR 		Convertissent l'énergie qu'ils reçoivent des pré-actionneurs en une autre énergie utilisée par les effecteurs. Ils peuvent être Pneumatiques, Hydrauliques ou Electrique
PRE-ACTIONNEUR 	 Variateur Contacteur Distributeurs	Distribuent l'énergie aux actionneurs à partir des ordres émis par la PC.
CAPTEUR 		Renseignent la PC sur l'état de la PO, Ils peuvent détecter des positions, des pressions, des températures, des débits... Peuvent être électriques ou pneumatiques. Signaux du type TOR, Analogique ou Numérique

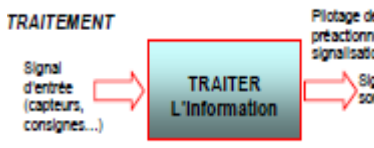
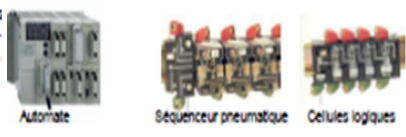
TRAITEMENT 		Dans les systèmes modernes, l'API assure de plus en plus cette fonction. Certains systèmes purement pneumatiques peuvent être contrôlés par des séquenceurs ou des fonctions logiques.
DIALOGUE 		L'unité de dialogue permet à l'opérateur d'envoyer des consignes à l'unité de traitement et de recevoir de celle-ci des informations sur le déroulement du processus

Tableau 1.1. Les éléments de PO et PC.

1.3.4 Les Modes de commande

a. Mode de commande direct (boucle ouverte)

Un système est en BO (non perturbé) lorsque la commande est élaborée sans l'aide de la connaissance des grandeurs de sortie : il n'y a pas de feedback (**Figure 1.4**).

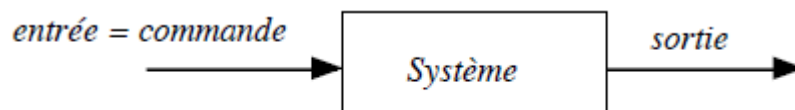


Figure 1.4. Système en boucle ouverte.

b. Mode de commande avec compte-rendu d'exécution (boucle fermé)

Le système est dit en BF. La commande est alors fonction de la consigne (la valeur souhaitée en sortie) et de la sortie. Pour observer les grandeurs de sortie, on utilise des capteurs. C'est l'information de ces capteurs qui va permettre d'élaborer la commande (**Figure 1.5**).

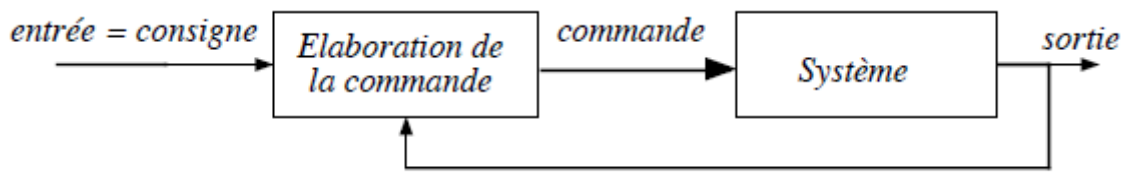


Figure 1.5. Système en boucle fermé.

1.3.5 Les technologies des automatismes

L'automaticien dispose de nombreux outils technologiques pour réaliser l'organe de commande de son système que l'on regroupe habituellement en deux catégories fondamentales : [2]

- **les solutions câblées** : l'automatisme est réalisé par des modules raccordés entre eux selon un schéma fourni par la théorie ou par l'expérience.

En électricité ou en électronique, les liaisons sont faites par câbles électriques alors qu'en fluide il s'agit de canalisations reliant les différents composants.

Chaque opérateur des équations de commande booléennes est représenté physiquement par un circuit.

Trois technologies permettent de réaliser des automatismes câblés : relais électromagnétiques, modules logiques pneumatiques, cartes ou modules électroniques.

Elles comportent des contraintes tels que le poids et le volume des composants, la rentabilité financière, la complexité de l'installation, les risques d'erreurs de câblage, la recherche des pannes. Toute modification dans le choix du fonctionnement de l'installation entraîne :

- Intervention dans le câblage (main d'œuvre).
- augmentation du nombre de relais et de fils.
- coût élevé et dialogue limité avec de telles installations (voyants, synoptique).

Les avantages sont :

- technologie simple, connue et maîtrisée.
- Conception, réalisation, mise en service et maintenance assurées par un personnel ne nécessitant pas de formation spécifique.

- **les solutions programmées** : l'automatisme est réalisé par la programmation de constituants prévus à cet effet. Le fonctionnement obtenu résulte de la programmation effectuée. L'automatisme est personnalisé par les choix matériels et par la programmation.

Trois technologies permettent de réaliser des automatismes programmés : cartes électroniques standards et spécifiques, micro et mini-ordinateurs, automates programmables.

Avantage :

- Encombrement de l'armoire électrique réduit lorsque la complexité de l'installation augmente.
- Main d'œuvre réduite lors du câblage.
- Terminal de programmation pouvant être commun à plusieurs automates.
- Modification possible sans intervention sur le câblage.
- Dialogue avec l'installation développée...

1.3.6 Les langages de programmation

Dans les systèmes automatisés il existe différentes langages qui peuvent être utilisés pour la programmation d'application d'automates programmables industriels (API), Les langages plus utilisés sont [7] :

➤ **FBD (schéma par blocs) :**

Il se présente sous forme diagramme : suite de blocs, connectables entre eux, réalisant des opérations, simples ou très sophistiquées.

Est un langage graphique qui permettant d'exprimer le comportement des fonctions, des blocs fonctionnels ou des programmes comme un ensemble de boites noires interconnectées (à la manière des portes logiques en électronique) (**Figure 1.6**).

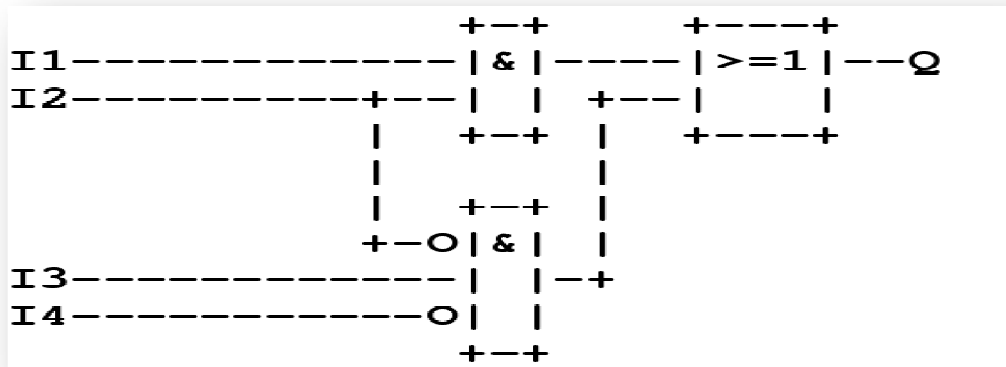


Figure 1.6. Programmation en langage FBD.

Ce programme FBD réalise l'opération suivante:

$Q := (I1 \text{ and } I2) \text{ or } (\text{not } (I2) \text{ and } I3 \text{ and not } (I4))$

➤ **SFC (GRAFcet):**

C'est un langage graphique de haut niveau, permettant de structurer tout comportement séquentiel pouvant être décrit dans l'un de quatre autres langages de la norme (**Figure 1.7**).

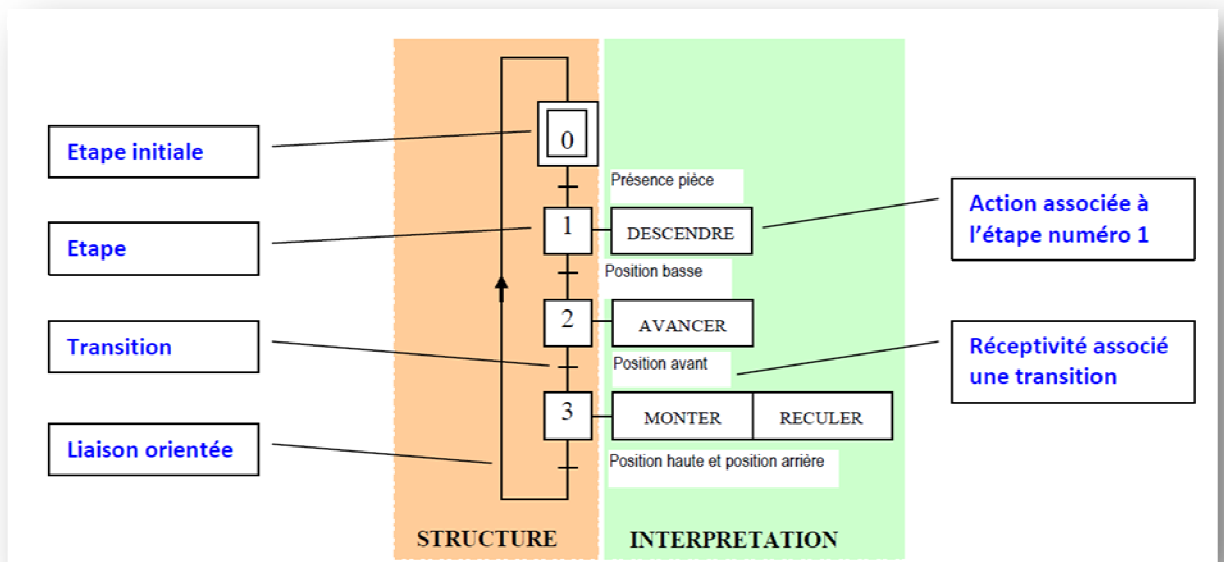


Figure 1.7. Exemple d'un programme GRAFCET.

➤ **LD (langage a contact):**

Est un langage graphique, très utilisé en milieu industriel, car il s'inspire des circuits de commande basés sur la logique électrique, les équations combinatoires étant câblées à l'aide de contacts et de relais électromécaniques. Un programme est décrit par un diagramme sous forme d'échelle (**Figure 1.8**).

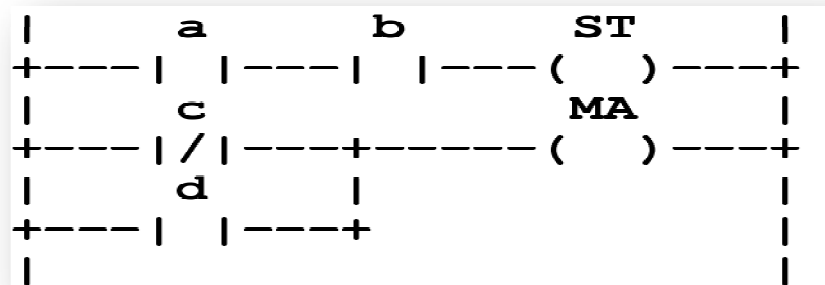


Figure 1.8. Programme en langage LADDER.

Ce programme LADDER réalise les opérations suivantes : $ST = a \text{ and } b$; $MA = \text{not}(c) \text{ or } d$

➤ **IL (List d'instruction):**

Ce langage textuel de bas niveau est un langage liste d'instruction permet de transcrire sous forme de liste :

- un schéma à contact.
- un logigramme, équations booléennes.
- un GRAFCET.

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

Ce langage est très proche du langage assembleur (LD) on travaille au plus près du processeur en utilisant l'unité arithmétique et logique, ses registres et ses accumulateurs (**Figure 1.9**).

```

PROGRAM And
VAR INPUT
I1 : BOOL;
I2 : BOOL;
END VAR

VAR OUTPUT
O : BOOL;
END_VAR

LD I1
AND I2
ST O

END_PROGRAM

```

Figure 1.9. Programme en langage IL.

➤ **ST (texte structure):**

Est un langage textuel structuré ressemble aux langages de haut niveau. Il permet la programmation de tout type d'algorithme plus ou moins complexe (**Figure1.18**).

```

IF RUN THEN XOUT := XOUT + K * (XIN - XOUT) ;
ELSE XOUT := XIN ;
      K := TIME_TO_REAL(CYCLE) / TIME_TO_REAL(CYCLE + TAU) ;
END_IF ;

```

Figure1.10. Programme en langage ST.

1.4 Les Actionneurs

1.4.1 Définition

Les actionneurs sont des convertisseurs électromécaniques conçus pour mettre en mouvement des systèmes mécanique à partir de commande électrique. Ce sont essentiellement des moteurs et des vérins. Ils produisent de l'énergie mécanique à partir d'énergie électrique, hydraulique ou pneumatique mais toujours contrôlés par des signaux de commande électrique. [2]

1.4.2 Les types des actionneurs

a. *Actionneurs électriques*

Ils utilisent directement l'énergie électrique distribuée sur les machines. Ils peuvent être sous plusieurs formes : moteur électrique, électrovanne de débit, résistance de chauffage, tête de soudure, ...

Les pré-actionneurs associés à ces actionneurs électriques sont principalement les contacteurs et variateurs de vitesse, entourés des sécurités nécessaires. [2]

b. *Actionneurs pneumatiques*

Ce sont des vérins qui utilisent directement l'air comprimé distribué sur les machines. Ils se présentent sous diverses formes et ils sont utilisés pour de nombreux mouvements : transferts, serrages, marquages, maintiens, éjections,...

Les pré-actionneurs associés à ces actionneurs sont les distributeurs qui peuvent être à commande pneumatique ou électrique (lorsqu'ils sont équipés d'une électrovanne). [2]

c. *Actionneurs hydraulique*

Ils peuvent être des vérins ou moteurs et ils ne sont utilisés qu'en cas de besoin car ils exigent l'installation sur la machine d'un groupe générateur de pression. Ils sont utiles lorsque les efforts à développer sont très importants (exemple : application de presses) ou lorsque des vitesses lentes doivent être contrôlées avec précision (exemple: avance d'outils de coupe, ...) (**Figure 1.11**). [2]



Figure 1.11. Quelque type d'actionneurs.[2]

1.4.3 Les vérins pneumatiques

a. Définition

Les vérins pneumatiques transforment l'énergie d'un fluide sous pression en énergie mécanique (mouvement avec effort). Ils peuvent soulever, pousser, tirer, serrer, tourner, bloquer, percuter, ...

Leur classification tient compte de la nature du fluide, pneumatique ou hydraulique, et du mode d'action de la tige : simple effet (air comprimé admis sur une seule face du piston), double effet (air comprimé admis sur les deux Faces du pis. (**Figure 1.12**).

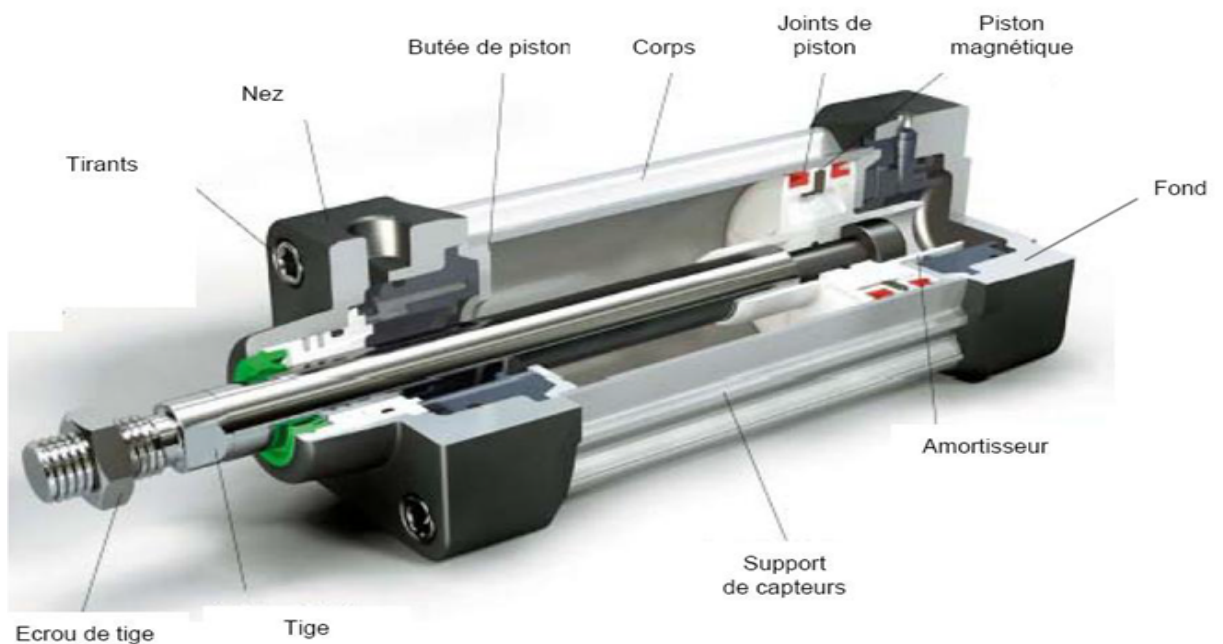


Figure 1.12. Constituants de base d'un vérin.

b. Principe de fonctionnement

C'est l'air comprimé qui, en pénétrant dans l'une des chambres, pousse le piston. La tige se déplace. L'air présent dans l'autre chambre est donc chassé et évacué du corps du vérin. Le mouvement contraire est obtenu en inversant le sens de déplacement de l'air comprimé.

c. les types des vérins

- **simple effet**

Le vérin simple effet est un composant monostable (Stable dans une seule position).

L'air comprimé qui commande le déplacement de la tige est admis d'un seul côté du piston (commande en sortie). Le retour de la tige en position initiale est généralement obtenu à l'aide d'un ressort. (**Figure 1.13**). [2]

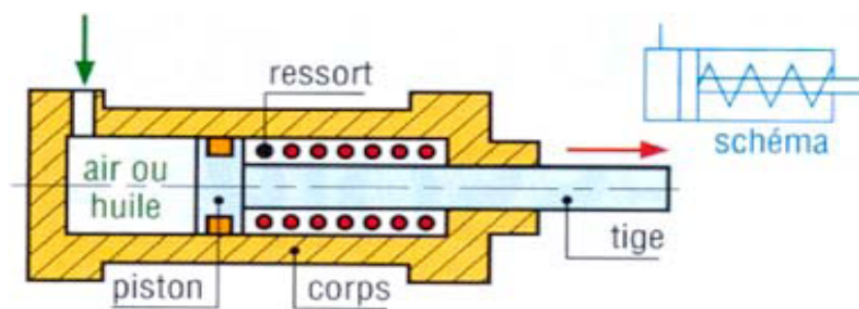


Figure 1.13. Vérin simple effet.

- **double effet**

Le vérin double effet est un composant bistable (Stable dans deux positions)

Ces vérins comportent deux orifices d'admission de l'air comprimé ce qui permet de commander la sortie ou la rentrée de la tige en fonction du besoin. Ainsi ces vérins permettent d'exercer un effort dans les deux sens. (**Figure 1.14**).

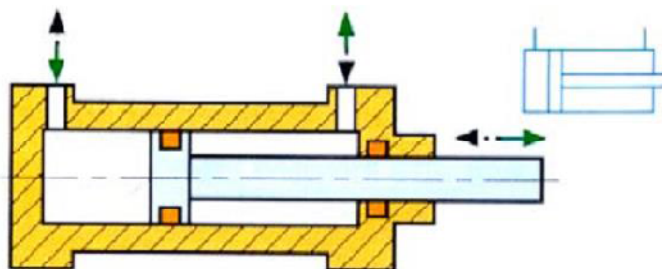


Figure 1.14. Vérin double effet.

1.5 Pré-actionneur

1.5.1 Définition

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

1.5.2 Distributeur pneumatique

Le distributeur est un composant qui va orienter (laisser passer ou non), grâce au mouvement d'une pièce interne, une énergie, que l'on appelle : énergie de PUISSANCE.

Pour obtenir le mouvement de la pièce interne, nous avons besoin d'une autre énergie, qui n'est pas nécessairement de même nature, c'est l'énergie de COMMANDE. Comme le contacteur associé à un moteur électrique le distributeur est le pré actionneur associé à un vérin pneumatique, les plus utilisés 3/2 4/2 5/2 2/2.... (Figure 1.15). [8]

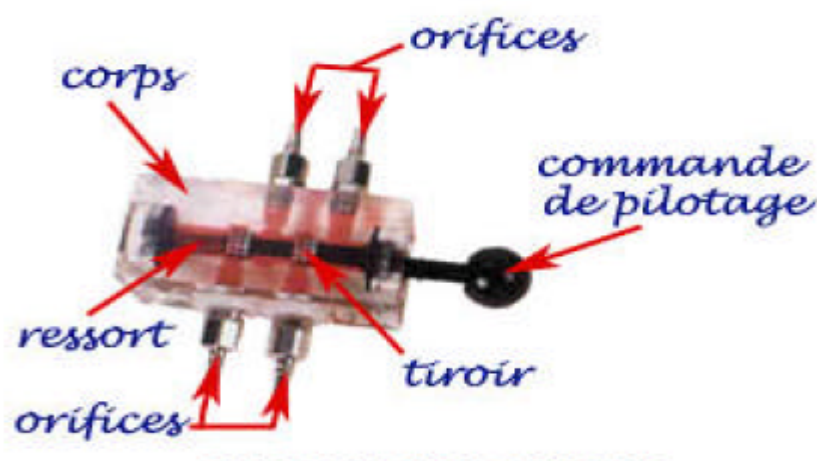


Figure 1.15. Un distributeur 4/2 forme didactique.[2]

a. Représentation schématique

Dans chaque case, les connexions provoquées par le mécanisme sont représentées par des traits et des flèches (**Figure 1.16**). [8]

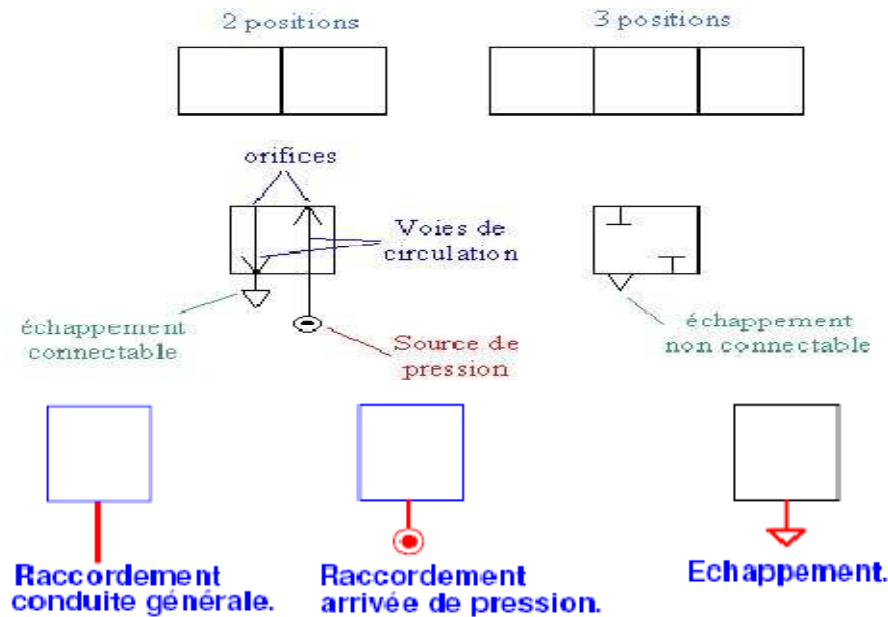
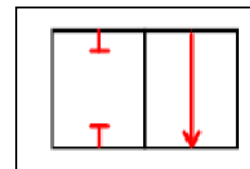


Figure 1.16. Représentation schématique. [8]

Remarque: Les canalisations aboutissent à la case représentant la position initiale.

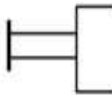
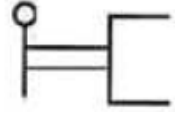
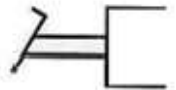
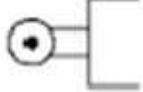
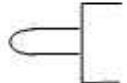
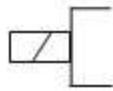
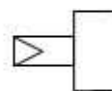
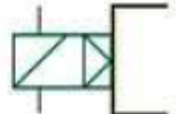
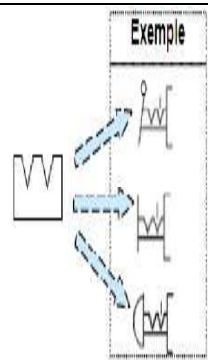
- Une possibilité de passage du fluide est symbolisée Par une flèche indiquant le sens de circulation.
- Un blocage du fluide est symbolisé par un "T".



• La symbolisation des distributeurs pneumatiques intègre trois fonctions [9]

- **Les orifices** : Ce sont les raccords nécessaires au passage de l'air comprimé, les flèches indiquent le sens de circulation de l'air.
- **Les positions**: Elles sont représentées par des cases, il y a autant de cases que de position du distributeur.
- **Les organes de commandes** : Ils déterminent la façon dont le distributeur est piloté.

b. Les types de commandes des distributeurs

Commande manuelle				Commande mécanique	
Commande manuelle : symbole général	Commande manuelle par levier	Commande manuelle par pédale	Commande manuelle par bouton poussoir	Commande mécanique par galet	Commande mécanique par poussoir
					
					
Commande indirecte					
Commande Electrique	Commande Pneumatique	Commande Electropneumatique	Rappel par Ressort	Dispositif de maintien en position «verrouillage mécanique »	
					
					

Tableaux 1.2. Les différentes commandes de distributeur. [9]

Le choix de la commande sera aussi effectué avec soin à partir de donnée constructeur pour vérifier que le temps de commutation soit compatible avec la performance attendue.

c. *Stabilité des distributeurs*

- ***Distributeur MONOSTABLE***

Si le distributeur possède un rappel par ressort on dit qu'il est MONOSTABLE. En présence du signal de pilotage, le tiroir bascule et reste dans sa position si le pilotage est maintenu. En l'absence de signal de pilotage le tiroir retrouve sa position repos sous l'action du ressort (**Figure 1.17**). [8]

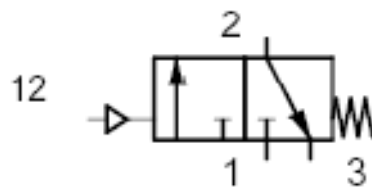


Figure 1.17. Distributeur 3/2 monostable.[8]

- ***Distributeur BISTABLE***

Si le distributeur possède deux pilotages il est dit BISTABLE. En l'absence de signal de pilotage, le tiroir ne bouge pas et occupe la position qu'il avait précédemment (**Figure 1.18**). [8]

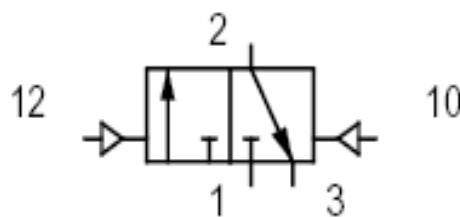


Figure 1.18. Distributeur 3/2 bistable.[8]

d. Principe de fonctionnement

Schématiquement, un distributeur sera constitué d'un tiroir percé de canalisations, Ce tiroir pourra occuper (en glissant) deux positions différentes à l'intérieur du corps.

Le corps est, lui-même, percé d'orifices qui sont mis en communication deux à deux selon la position occupée par le tiroir(**Figure1.19**). [9]



Figure1.19. Distributeur 4/2 en état de repos et travail.

Ils permettent de commuter et contrôler la circulation des fluides sous pression et assurent diverses fonctions :

- Contrôle du mouvement de la tige d'un vérin.
- Choisir le sens de circulation d'un fluide.
- Exécuter des fonctions logiques.
- Démarrer ou arrêter la circulation d'un fluide.
- Être des capteurs de position (pressostat, vacuostat).

1.6 les capteurs

1.6.1 définition

Un capteur est un organe de prélèvement d'information qui élabore à partir d'une grandeur physique, une autre grandeur physique de nature différente. Cette grandeur représentative de la grandeur prélevée est utilisable à des fins de mesure ou de commande. Il est composé de 2 éléments : corps d'épreuve et détecteur (élément sensible) (**Figure 1.20**).



Figure 1.20. Quelque type de capteur.

1.6.2 Les types des capteurs

a. Capteur analogique

La grandeur de sortie est en relation directe avec la grandeur d'entrée. Dans ce cas le capteur doit être linéaire (voir plus loin) sinon nous aurions un signal déformé. L'avantage est que nous avons la possibilité de mesurer sur toute une plage et non pas simplement un seuil.

Dans la pratique industrielle, on donne à ce type de matériel le nom de capteur, type de signal de sortie : 0-10v ou 4-20mA [11]

b. Capteur numérique

Ce type de capteur délivre en sortie une information électrique à caractère numérique, image de la grandeur physique à mesurer, c'est à dire ne pouvant prendre qu'un nombre limité de valeurs distinctes (comme tout signal numérique...). Le signal de sortie peut être récupéré en série ou en parallèle.

Selon le cas portent les noms en pratique de codeurs ou de compteurs. Type de signal de sortie : 0011 ou 0001.[11]

c. Capteur logique (TOR)

Il génère un signal de type binaire (0 ou 1). L'avantage est qu'ils sont peu chers mais ils sont spécialisés dans un type précis de mesure. Par exemple ils peuvent dire si une pièce est présente ou non.....etc.

Ils ne permettent pas de mesurer sur toute une plage.

Ils portent le nom de signal de sortie : 0v ou 5v. [11]

1.7 Conclusion

Nous avons données dans ce chapitre les notions importantes et principales sur les systèmes automatisés et des informations sur les actionneurs, pré-actionneurs et les capteurs.

Chapitre 2 : Constitution mécanique de la chaîne

2.1 Introduction

La règle principale dans la construction d'une machine est de bien concevoir sa partie mécanique car la précision des tâches à accomplir et du déplacement de bras de la machine dépendent de la précision de sa structure mécanique.

Nous allons voir dans ce chapitre les différents matériels et leurs caractéristiques qui constituent notre machine.

2.2 Constitution de la chaîne

2.2.1 Les Actionneurs

Nous avons utilisé dans notre système trois types d'actionneurs qui sont :

- 3 vérins pneumatiques.
- 3 moteurs CC (EMG30).
- Pince d'emballage.

a. Vérin pneumatique

C'est un actionneur qui permet, à partir d'air comprimé, de faire déplacer de manière rectiligne (droite un mécanisme).

Dans notre machine il y a trois vérins pneumatiques, deux simples effets et un vérin double effet.

- **Vérin simple effet**

Le rôle de ce vérin dans notre système est de tamponner les objets lorsque le moteur M3 s'arrête.

Force statique développée

Il faut tenir compte de la force R_c du ressort comprimé, d'où :

$$F_s = p \times S - R_c$$

Avantage

Les vérins simple effet sont économiques, et la consommation de fluide est réduite.

Inconvénients

Ils sont plus longs que les vérins double effet.

La vitesse de la tige est difficile à régler en pneumatique et les courses proposées sont limitées (jusqu'à 100 mm).

- **Vérin double effet**

Le rôle de ce vérin dans notre système est l'ouverture/fermeture de pince d'emballage lorsque le capteur indique qu'il y a un objet.

1. Force statique

En faisant agir l'air comprimé sur une face immobile, on obtient une force statique F_s proportionnelle à la pression p et à sa surface d'action S : **$F_s = p \times S$**

2. Force dynamique

Si la face est mobile en translation, la force dynamique F_d obtenue pendant le mouvement est plus faible car elle dépend des forces qui s'opposent à son déplacement : force liée à la pression opposée (dite contre-pression), force de frottement, force d'inertie : **$F_d = p \times S - \sum F_r$**

Avantage

Plus grande souplesse d'utilisation.

Réglage plus facile de la vitesse, par contrôle du débit à l'échappement.

Amortissements de fin de course, réglables ou non, possibles dans un ou dans les deux sens.

Ils offrent de nombreuses réalisations et options.

Inconvénients

Ils sont plus coûteux.

b. Moteur à courant continu [12]

• Définition

Les moteurs courant continus sont des convertisseurs de puissance :

Soit ils convertissent l'énergie électrique absorbée en énergie mécanique lorsqu'ils sont capables de fournir une puissance mécanique suffisante pour démarrer puis entraîner une charge en mouvement. On dit alors qu'ils ont un fonctionnement en moteur.

Soit ils convertissent l'énergie mécanique reçue en énergie électrique lorsqu'ils subissent l'action d'une charge entraînant. On dit alors qu'ils ont un fonctionnement en générateur.

• Caractéristique d'un moteur à courant continu

Le moteur à courant continu se compose de (**Figure 2.4**):

- l'inducteur ou du stator.
- l'induit ou du rotor.
- collecteur et des balais.

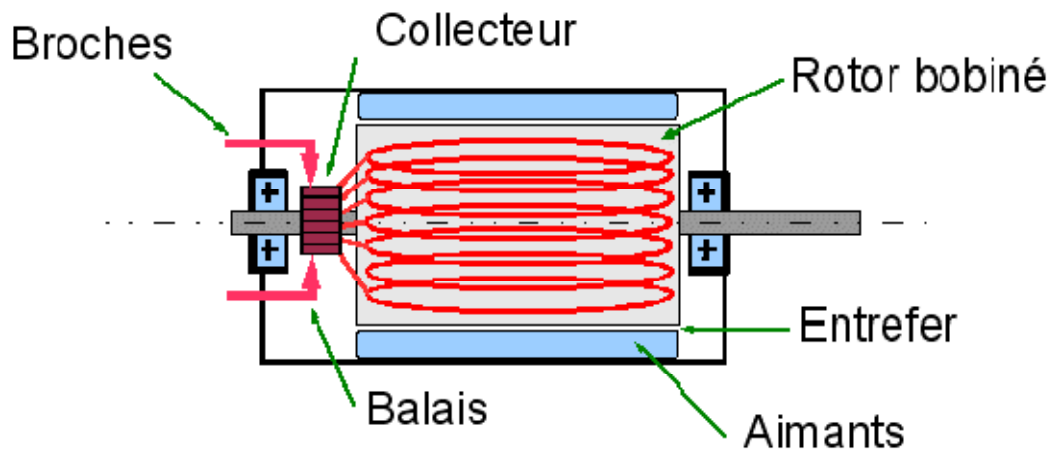


Figure 2.1. Caractéristique interne d'un moteur CC.

La plupart des machines d'ascenseur sont configurées en excitation parallèle ou indépendante. L'inversion du sens de rotation du moteur s'obtient en inversant soit les connexions de l'inducteur soit de l'induit.

- **Moteur DC (EMG30) [13]**

L'EMG30 (moto réduction) est un moteur de 12v totalement équipé avec encodeur et une boîte à engrenages à réduction : 30.1.

Il est idéal pour les applications robotiques, fournissent un retour d'informations et une commande efficaces à l'utilisateur.

Il comprend également un condensateur de suppression de bruit.

Caractéristique de moteur EMG30

- Tension : 12v.
- Démultiplication : 30 ,1.
- Couple nominale : 1,5Kg/cm.
- Vitesse : 170tr/mn (216 hors charge).
- Courant : 530mA (150 hors charge).
- Courant de décrochage : 2,5A.
- Plage de sortie : 4,22w.
- Nombre d'encodeur par tour : 360 impulsions.
- Dimensions : $\Phi 28, 5 \times 86,6$ mm (axe de 10mm inclus).



Figure 2.2. Moteur EMG30.

c. Pince d'emballage

Une pince est un dispositif mécanique ou un outil, ayant pour fonction de pincer quelque chose d'autre ; c.-à-d. serrer pour saisir ou maintenir.

Est un outil à deux branches qui sert à prendre ou à tenir quelque chose comme des pièces.

Dans le domaine de l'usinage (tournage, fraisage...), c'est une pièce qui sert à maintenir en position une autre pièce (à usiner) ou un outil.

La pince de notre machine est illustrée par la **Figure 2.3**, et leur rôle est de serrer l'objet pendant le déplacement de M2.

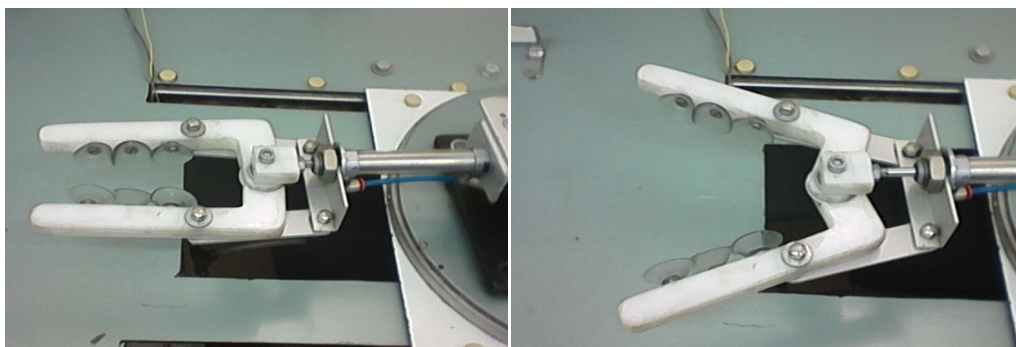


Figure 2.3. Pince de conditionnement.

2.2.2 Pré-actionneurs (Distributeur)

Les distributeurs pneumatiques sont des éléments de la chaîne d'énergie. Ils distribuent de l'air comprimé aux actionneurs pneumatiques (vérins, générateurs de vide, moteurs à palettes...) à partir d'un signal de commande (pilotage). (**Figure 2.4**).



Figure 2.4. Distributeurs pneumatique.

- Dans notre système nous avons :
 - 1- Pour le vérin simple effet on utilise deux distributeurs 3/2 monostable (3 orifices et 2 positions) a commande Electropneumatique/rappel par ressort puisque le vérin simple effet ne comporte qu'un seul orifice à alimenter. (**Figure 2.5**).

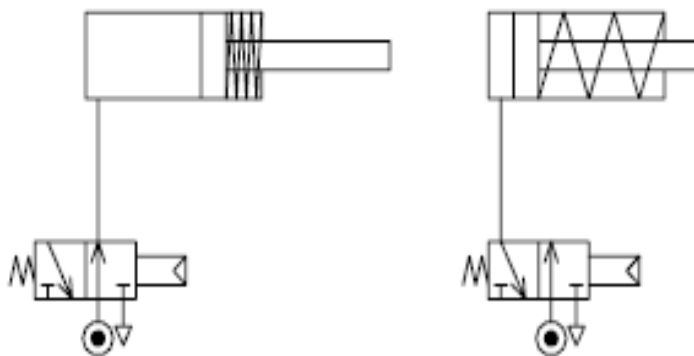


Figure 2.5. Principe de fonctionnement d'un distributeur 3/2 monostable.

- 2- Pour le vérin double effet on utilise un distributeur 5/2 bistable a commande Electropneumatique/manuelle puisque ce vérin comporte 2 orifices de sortie sur lequel il faut alterner les états de pression et d'échappement. (**Figure 2.6**).

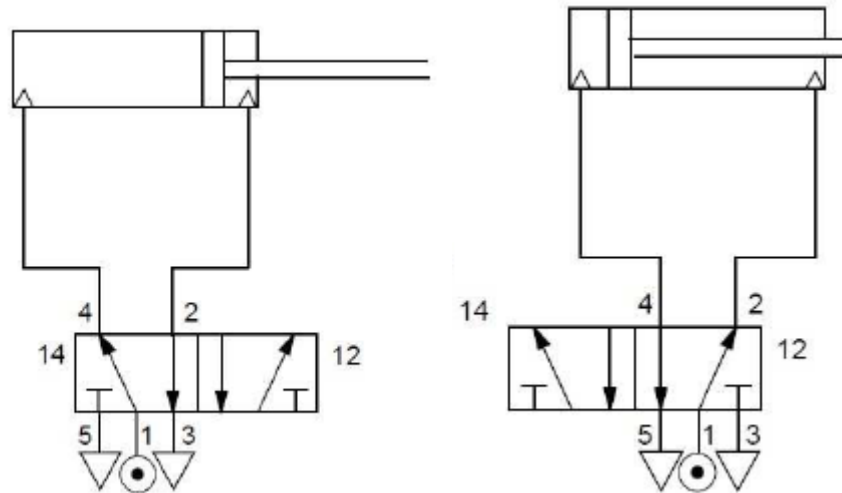


Figure 2.6. Principe de fonctionnement d'un distributeur 4/2 bistable.

2.2.3 Les capteurs

a. Capteur infrarouge (IR)

Le capteur laser est basé sur l'émission d'une onde électromagnétique structurée. Ce capteur permet donc d'obtenir un faisceau d'ondes très concentré, contrairement aux capteurs ultrasons.

Les systèmes laser possèdent de nombreux avantages qui en font des capteurs souvent utilisés dans les applications des systèmes pour l'obtention des mesures précises de distance (**Figure 2.7**).[14]



Figure 2.7. Capteur infrarouge.

- **Principe de fonctionnement**

Un capteur IR est l'œil électronique qui se trouve sur de nombreux appareils qui viennent avec une télécommande. La commande à distance transmet un faisceau infrarouge, invisible à l'œil humain, sur une distance fixe de l'appareil, qui suit les instructions de transmission codées de chaque bouton de la télécommande. Par exemple, on appuyant sur "pause" sur la télécommande du lecteur de DVD provoque l'arrêt de moteur tourner le disque dans le lecteur DVD (**Figure 2.8**).

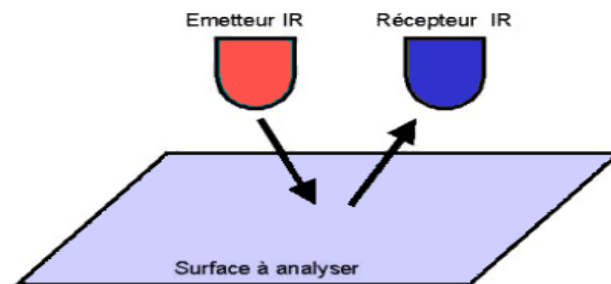


Figure 2.8. La détection de capteur infrarouge.

b. Manomètre

Un manomètre est un appareil qui permet de mesure une pression, qu'il s'agisse de la pression d'un gaz ou d'un fluide.

Cet appareil est appeler aussi capteur de pression, le rôle de cet capteur c'est le réglage de pression à partir d'un régulateur de pression l'augmentation/diminution de pression dépend de notre système (**Figure 2.9**).[14]



Figure 2.9. Manomètre analogique.

c. Bouton d'arrêt d'urgence

En automatique, l'arrêt d'urgence provoque une mise hors énergie des actionneurs, et permet de protéger ou d'empêcher l'aggravation des opérateurs de système.

Le BAU permet de détecter une forte pression de doigt, ce capteur est généralement utilisé sur des machines dangereuses. Il permet à l'utilisateur, en cas de danger, de stopper l'arrêt de la machine en donnant un « coup de poing » sur la partie rouge du capteur (**Figure 2.10**).

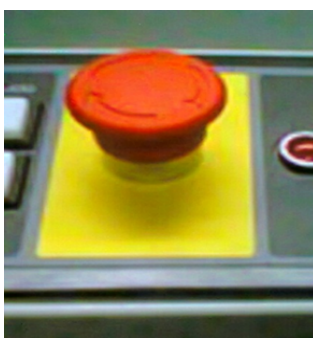


Figure 2.10. Bouton d'arrêt d'urgence.

2.2.4 L'énergie pneumatique

Pour agir sur la matière d'œuvre, tout système a besoin d'une énergie qui sera convertie puis adaptée au besoin.

L'énergie pneumatique utilise l'air comprimé comme fluide pour le transport de l'énergie et sa transformation en énergie mécanique.

Pour réaliser une installation pneumatique il y a des actionneurs (vérin, moteur....), ils transforment l'énergie pneumatique en énergie mécanique et des prés actionneurs (distributeur), des organes de commandes (capteurs, bouton poussoir...), des composants d'automatisme.

a. Compresseur

Un compresseur est un organe mécanique destiné à augmenter la pression d'un fluide, et donc son énergie. Il mesure le déplacement des fluides compressibles tels les gaz (*Figure 2.11*).

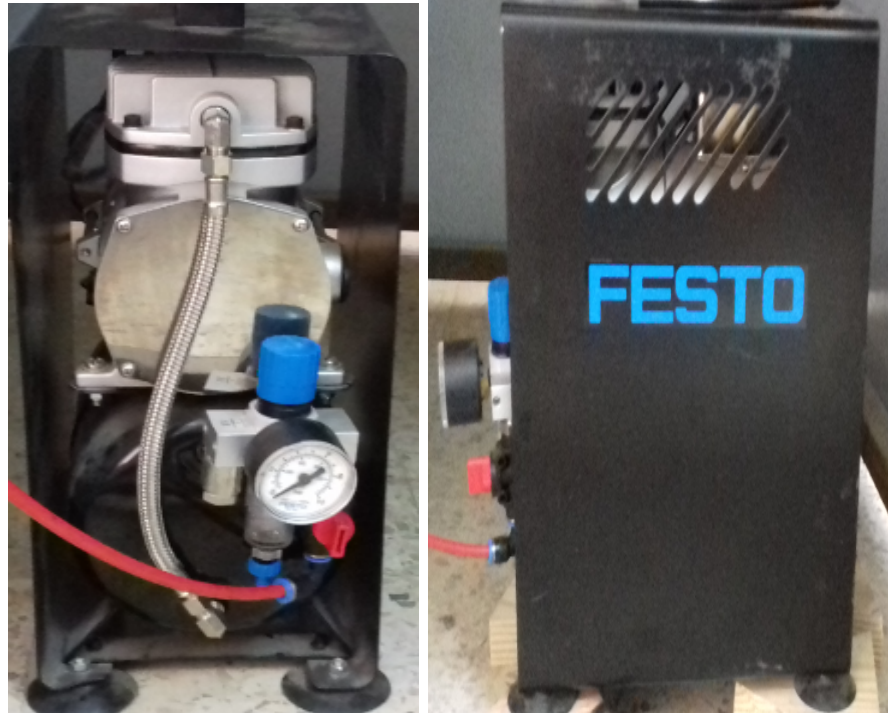


Figure 2.11 compresseur pneumatique..

2.2.5 Les mécanismes de transformation de mouvement

Les mécanismes de transformation de mouvement transmettent le mouvement de l'axe d'entrée à l'axe de sortie mais en changeant le type de mouvement, un mouvement de rotation. Ces mécanismes transforment le mouvement de rotation en mouvement de translation. Ils sont :[15]

- LE SYSTÈME BIELLE – MANIVELLE.
- LE VILEBREQUIN.
- LE MÉCANISME ROUE DENTÉE-CRÉMAILLÈRE.
- LE MÉCANISME VIS – ÉCROU.

a. LE SYSTÈME BIELLE – MANIVELLE

Le système bielle-manivelle permet la transformation d'un mouvement circulaire continu en un mouvement rectiligne alternatif de va-et-vient, ou réciproquement. La bielle est une pièce reliant deux articulations d'axes mobiles dans le but de transmettre une force, alliée au vilebrequin (manivelle), elle donne le système qui nous intéresse (**Figure 2.12**).

Rotation continue --- Translation alternative.

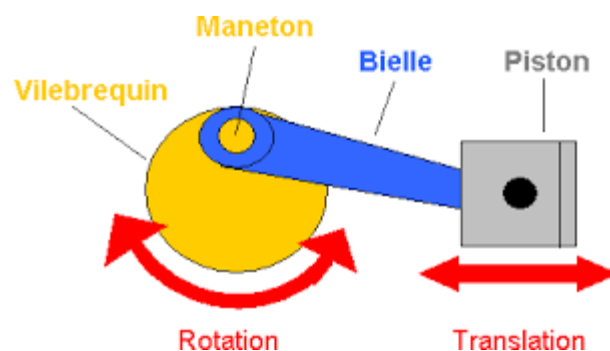


Figure 2.12. BIELLE – MANIVELLE.

b. LE VILEBREQUIN

Le vilebrequin est une pièce de moteur à combustion interne.

Elle est l'élément principal du dispositif bielle-manivelle. Il permet la transformation du mouvement linéaire alternatif du ou des piston(s) en un mouvement de rotation continu (**Figure 2.13**).

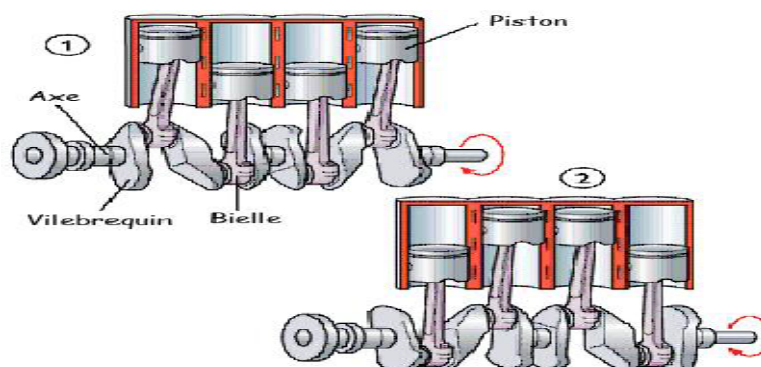
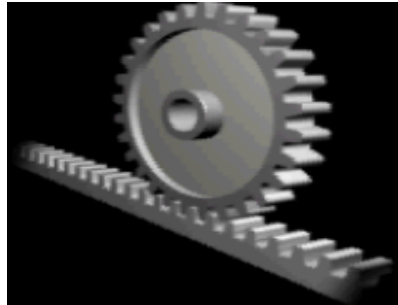


Figure 2.13. Système vilebrequin.

c. LE MÉCANISME ROUE DENTÉE-CRÉMAILLÈRE

Lorsque que le pignon fait une rotation, la crémaillère fait une translation (*Figure 2.14*).

Rotation continue --- Translation continue.



*Figure 2.14.*Crémaillère.

d. LE MÉCANISME VIS – ÉCROU

Le mécanisme de vis-écrou est un mécanisme de transformation de mouvement. En général il permet la transformation d'un mouvement de rotation en un mouvement de translation. C'est par exemple le dispositif le plus communément adopté sur les étaux, et autres presseoirs mais aussi les crics (*Figure 2.15*).

Rotation continue --- Translation continue.

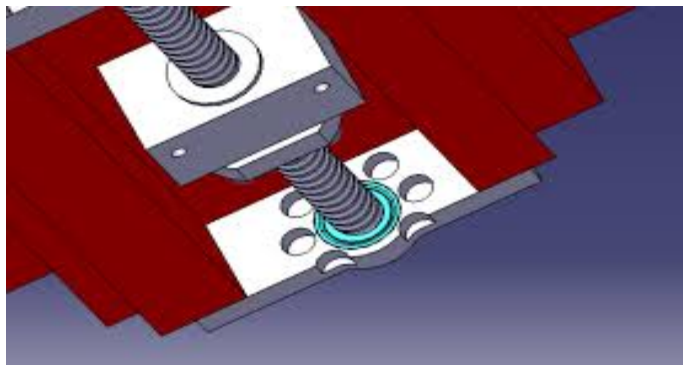


Figure 2.15. Vis écrou.

2.3 Conclusion

On a vu dans ce chapitre tous les éléments mécaniques qui constituent notre chaîne de conditionnement, leurs fonctionnements et le rôle de chaque élément.

Chapitre 3: Constitution électronique de la chaine

3.1. Introduction

Dans la conception de la chaine de conditionnement, la partie électronique joue un rôle très important dans le fonctionnement du système. En effet pour l'automatisation de cette chaine il est nécessaire de réaliser des cartes telles que la carte de Commande, la carte de Puissance et la carte Arduino. Dans ce chapitre on verra les caractéristiques de chaque carte. La structure électronique de notre chaine peut être représentée par le schéma ci-dessous :

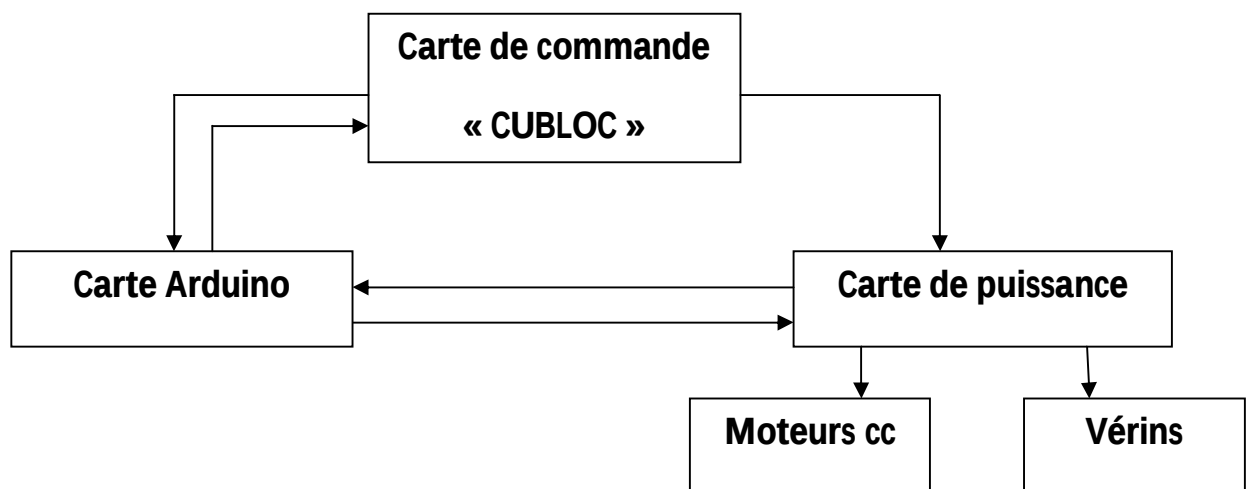


Figure 3.1 Schéma synoptique générale de la chaine de conditionnement.

3.2. carte de commande

3.2.1. Définition

La carte de commande permet de contrôler la gestion de toutes les mécanismes (moteurs, capteurs...etc.) par l'intermédiaire du microcontrôleur (CUBLOC) qui contient le programme principale.

3.2.2. Le microcontrôleur CUBLOC

a. Définition

Le microcontrôleur CUBLOC est un mini-automate programmable PLC traditionnel qui en peut le programmer par deux langages (LADDER, BASIC), la programmation se fait soit en langage LADDER, soit en langage BASIC, ou avec les deux langages simultanément.

Il se présent sous forme de modules hybrides, lesquels vous permettront de pouvoir les intégrer au sien de votre produit finale en vous laissant ainsi une plus grande flexibilité sur leur taille petite et les spécificités de votre application comme un microcontrôleur traditionnel (**Figure 3.2**).

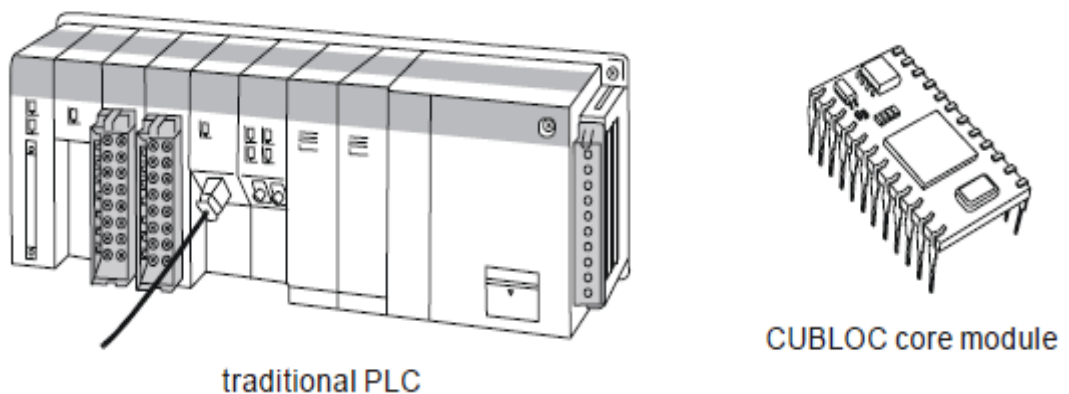


Figure 3.2 La forme d'un PLC traditionnel et le module CUBLOC.

Il existe différents module de CUBLOC, lesquels se distingue par leur capacité mémoire, nombre d'E/S etc..... et le tableau ci-dessous présente quelque modules.

b. constitution d'un CUBLOC

- ***Constitution interne***

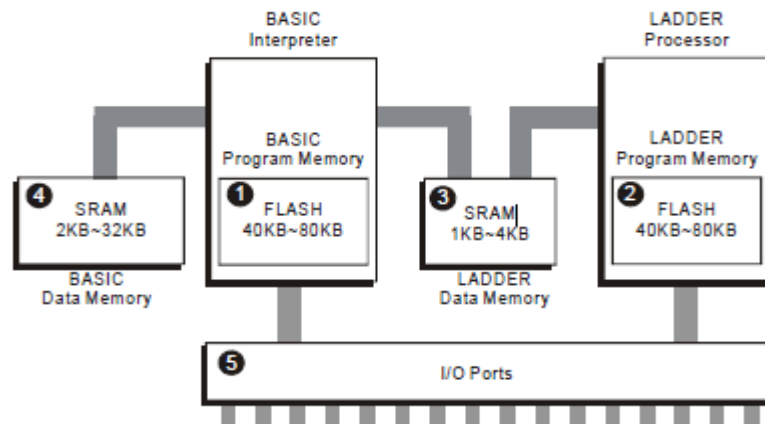


Figure 3.3 Structure interne de CUBLOC.

L'interpréteur BASIC dispose d'une mémoire Flash dédiée au programme BASIC. Le processeur de gestion du LADDER dispose également de sa mémoire Flash pour l'exécution du programme LADDER.

Les E/S peuvent être partagées librement entre les programmes BASIC et LADDER.

La mémoire de données du BASIC ne pourra être accédée que par l'interpréteur BASIC tandis que la mémoire des données du LADDER pourra être accédée à la fois par l'interpréteur BASIC comme par le processeur LADDER.

La mémoire programme du BASIC (1) et du LADDER (2) se partage la même ressource de mémoire Flash. Cette ressource mémoire est de **80 K**. une application entièrement programmée en BASIC peut utiliser l'intégralité de ces **80 k**. de même, une application entièrement programmée en LADDER peut également utiliser l'intégralité de cette mémoire. Une application programmée en BASIC(4) et en LADDER(3) pourra être développée dès lors que le total du programme BASIC et LADDER ne dépasse pas les **80k**. Les modèles CB2XX disposent actuellement de **80k**. Les futures versions de module CUBLOC pourront disposer davantage de mémoire.

Les ports d'E/S (5) peuvent être partagés entre le BASIC et le LADDER. L'utilisateur doit spécifier les ports d'E/S utilisés dans le BASIC et ceux utilisés dans le LADDER. Il est possible d'utiliser tous les ports uniquement pour le BASIC ou uniquement pour le LADDER.

- **Constitution matériel**

- **Caractéristiques matériel**

Les CUBLOC de la série CB2XX ont les caractéristiques suivantes :

- (BASIC et LADDER) 80K de mémoire Flash.
- Vitesse d'exécution BASIC : 36.000 Instr/sec.
- Vitesse d'exécution LADDER : scan time 10 ms (turbo mode ~= 100 micro seconde)
- Mémoire données pour le BASIC : 2~24 k (sauvegarde batterie sur modèle cb290°).
- Mémoire données pour le LADDER : 1~4 k.
- Mémoire EEPROM : 4 k
- 16 à 91 broches d'E/S.
- 8 canaux de conversion Analogique/ Numérique 10 bits.
- 3 ou 6 canaux PWM (8~ 16 bits).
- Gestion matérielle UART (RS232C)
- Interface PC via port RS232C.
- Circuit d'horloge RTC intégré (sur cb290 uniquement).

- **Caractéristiques électriques**

- Tension de fonctionnement : 4.5 à 5.5 Vcc.
- Fréquence de cadencement : 18.432 MHz.
- Source de courant (port E/S) : 20 mA.
- Puits de courant (port E/S) : 25 mA.
- Températures limites de fonctionnement : +10 à +65°C.
- Fonctionnement taux d'humidité : 10 à 80 % RH.

C. Les langages utilisés

- **Langage LADDER**

la programmation en LADDER est de permettre le traitement de l'ensemble des tâches avec la même durée. Ainsi quelque-soit la complexité de vos « circuits », il sera toujours prêt à activer les sorties lorsque les entrées seront sollicitées. Ceci est pourquoi il est courant d'avoir recours à ce type de langage pour le pilotage de machines diverses en automatisme. Ce langage exécute les tâches d'une manière parallèle (**Figure 3.4**).

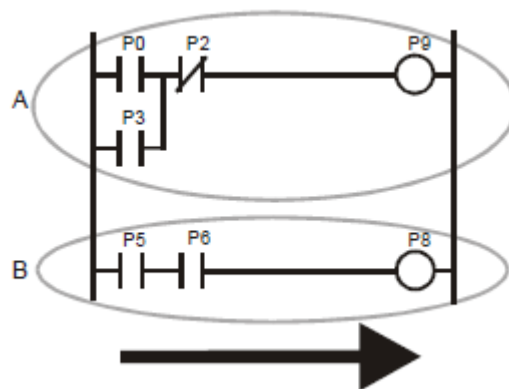


Figure 3.4 Schéma d'un programme en LADDER.

Comme vous le voyez ci-dessus, à la fois les circuits A et B sont en état d'attente et prêts à activer les sorties lorsqu'ils seront sollicités.

- **Langage BASIC**

Le pré processeur BASIC est un macro processeur automatiquement utilisé par le compilateur pour transformer votre programme avant la compilation. Il est appelé « macro processeur » car il vous permet de définir des « macros » qui vous permettent d'alléger la structure de programme de grande envergure.

La programmation des circuits en langage BASIC s'effectue selon un déroulement séquentiel (**Figure 3.5**).

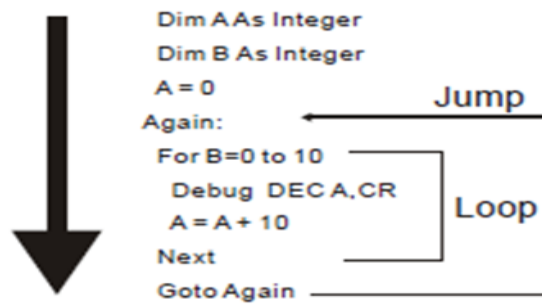


Figure 3.5 Exemple d'une programmation en BASIC.

- **LADDER et BASIC du CUBLOC**

Ces 2 types de langages sont utilisés depuis très longtemps dans des secteurs d'activités et pour des applications très différentes. Le LADDER est principalement exploité en automatisme par le biais d'automates programmables. Pour sa part, le langage BASIC est couramment utilisé pour la mise en œuvre d'applications diverses et variées sur compatible PC et sur de nombreux microcontrôleurs.

La comparaison entre LADDER et BASIC

	LADDER	Programmation en BASIC
Module	PLC (automate)	PC et microcontrôleur
Application	Automatisme	Toute application
Avantage	Gestion logique et séquentielle, gestion de bit, timers, compteurs	Gestion calculs mathématiques, communication de données, affichage sur écran LCD divers
Gestion structurelle	En parallèle	De façon séquentielle

Tableau 3.1 table comparative.

d. l'utilisation des deux technologies en même temps

La structure multitache des modules CUBLOC leur permettront ainsi de pouvoir a la fois exécuter un programma LADDER et un programme BASIC en même temps en vous permettant ainsi de pouvoir bénéficier de la précision du « **timing** » d'exécution du LADDER, tout en concernant les atouts d'une application programmée en langage BASIC. Dans tous les cas, vous conservez également la possibilité de pouvoir programmée toute votre application uniquement en langage BASIC ou uniquement en langage LADDER ou avec les 2 technologies a la fois. (Figure 3.6 & Figure 3.7).

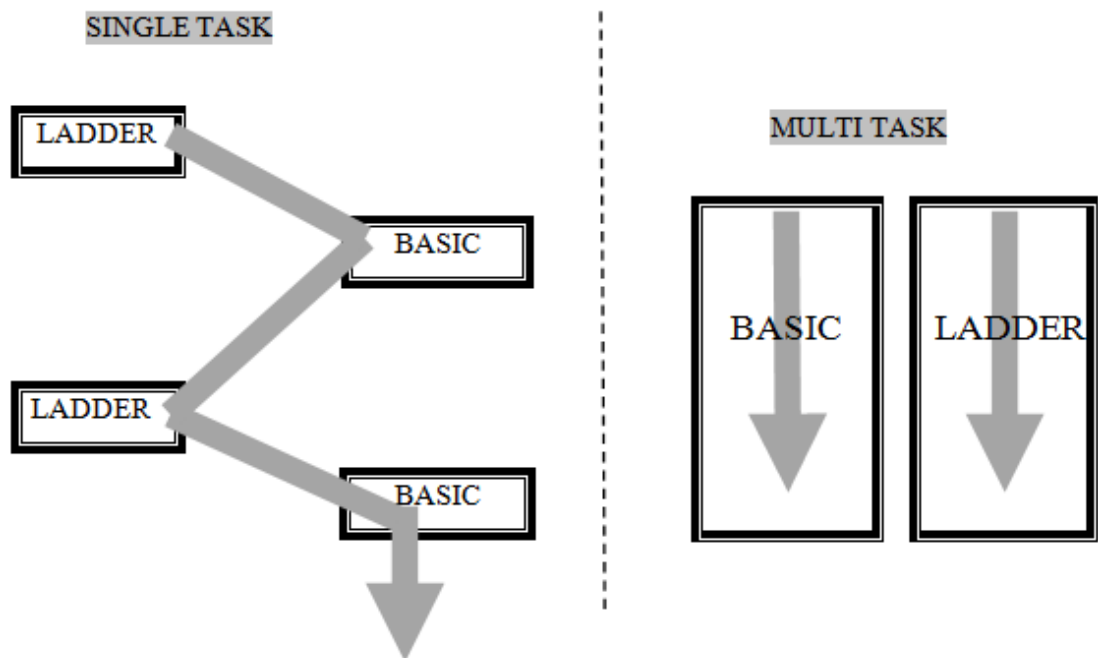


Figure 3.6 Les possibilités de laprogrammation.

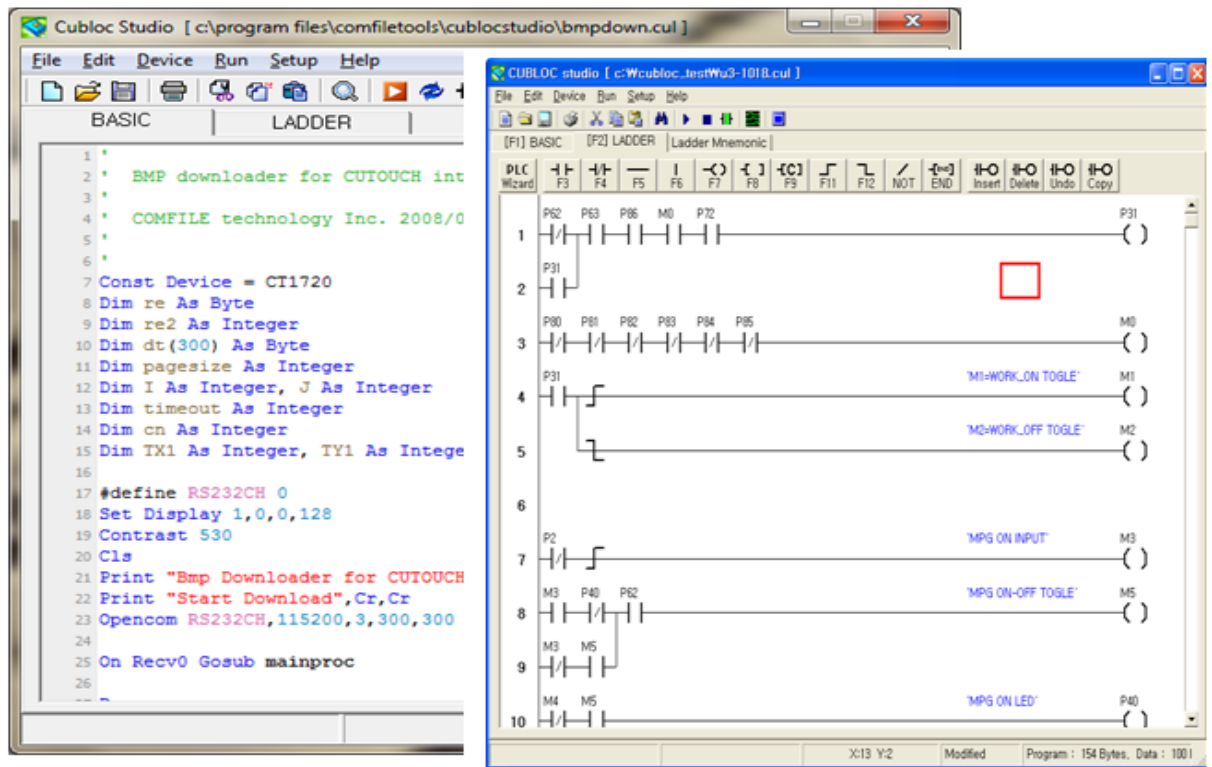


Figure 3.7. Les deux langages de CUBLOC studio.

Comme vous pouvez le constater, les modules CUBLOC s'apparentent ainsi à de nouveaux types de contrôleurs. En vous permettant de réaliser des applications que des mini-automates traditionnels ne pourraient pas effectuer sans l'apport du langage BASIC, les CUBLOC ouvrent de nouvelles perspectives de développement et d'applications.

e. les avantages de CUBLOC

- Il se présente sous forme de modules hybrides.
- Leur petite taille par rapport au PLC ne nécessite pas un câblage pour réaliser les applications.
- Il gère à la fois un programme écrit en langage BASIC et/ou en langage LADDER grâce à leur cœur multitâche.
- Les deux technologies intégrées au CUBLOC (LADDER et BASIC) soient totalement indépendantes.
- Les modules CUBLOCs sont moins chers que les PLC traditionnels.

- les automates traditionnels ne sont pas adaptés pour les tâches plus complexes, pour ce la le langage BASIC a été ajoutée sur les modules CUBLOC.
- Le CUBLOC contient certaines limitations propres à la programmation en langage LADDER par rapport aux autres modèles.

f. le module CB280 :

Le cb280 se présente sous la forme d'un module de dimension 35x25.4x11, et de 64 broches réparties sur 2 connecteurs mâles 32 broches au pas de 2 mm. Sa petite taille eu égard à ses grandes capacités techniques et 49 des ces broches pouvant être utilisé comme des "E/S". Il dispose une mémoire programmable d'une capacité totale de 80KB, et mémoire de donnée 2KB de BASIC et 1KB de LADDER (**Figure 3.8**).

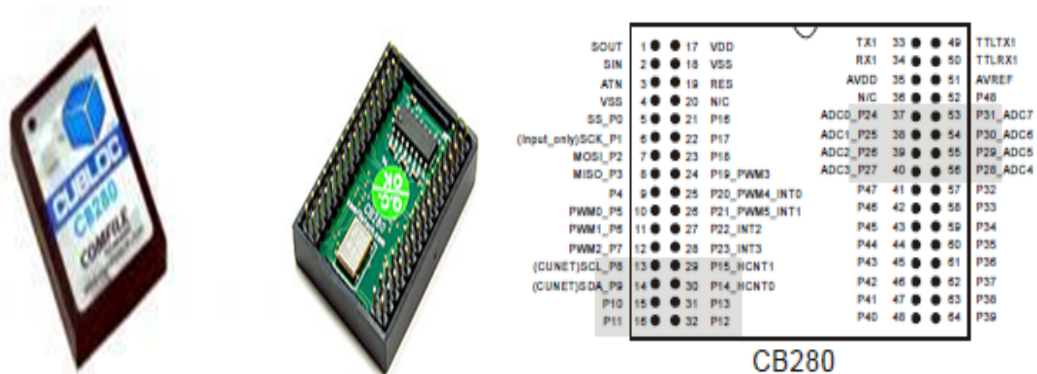


Figure 3.8 Le module CB280.

Remarque

Les broches 20 et 36 ne sont pas utilisées (pas de connecteur sur ces broches).

Alimentation du module CUBLOC CB280

Le module cb280 ne dispose pas de régulateur interne et vous devrez l'alimenter uniquement sous une tension régulée et filtrée de +5 VCC comme indiqué ci-dessous :

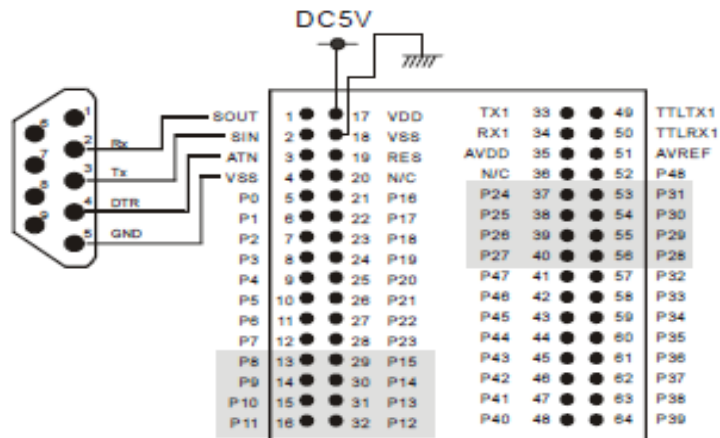


Figure 3.9 Alimentation de module CB280.

3.2.3. Constitution de la carte de commande

La carte de commande (**Figure 3.10**) de notre système est constitué de :

- Boitier de CUBLOC cb280 (partie essentielle de cette carte).
- Régulateur 5v (7805CT).
- Diode (N4007).
- Condensateur de filtrage (100nF).
- Résistance de pull-up (10K Ω).
- Bouton pour Reset.
- Fiche d'alimentation.
- Fiche de communication avec le PC.
- Une interface de communication (plaque d'essai).



Figure 3.10 La carte de commande.

3.3. la carte de puissance

3.3.1. Définition

C'est l'interface entre la carte de commande et la machine, elle comporte des composants mécaniques et permet la commande des actionneurs (moteurs, vérins). La carte doit aussi fournir la puissance pour les moteurs à partir de commandes envoyées par la carte mère.

Notre carte a deux fonctions principales :

- elle doit fournir à la carte électronique ARDUINO, leur tension d'alimentation (0/5V), pour alimenter les moteurs qui permettent à se déplacer le bras.
 - Elle fournir la puissance aux les vérins pneumatiques, pour fait son rôles.
- Donc elle constituée par deux pont H pour asservi les moteurs et des relais électromécanique pour fait passé la commande aux vérins.

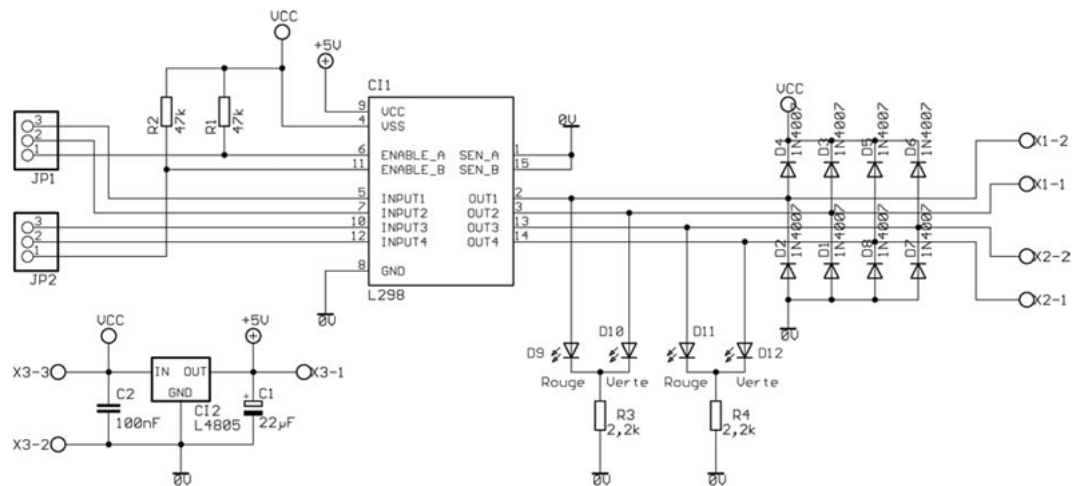


Figure 3.10. Pont en H.

3.3.2. Le circuit L298N

le L298N contient deux étages de puissance configuré en pont, chacun commandé par deux entrées logique (A, B et C, D) ainsi que deux entrées de validation (ENA1 et ENA2). De plus deux broches sont connectées en interne aux émetteurs des transistors (paires inférieur) qui permettent la connexion des résistances palpites de courant.

Celui-ci permet de disposer d'un courant important (2.5A) sous une tension élevée (45V) la puissance obtenue peut atteindre environ 200W, ce qui permet l'alimentation des moteurs puissants présentant des résistances de bobinage faible. Le L298N possède deux broches d'alimentation l'une pour les moteurs (Vs), l'autre pour la logique interne (Vss) (**Figure 3.12**).

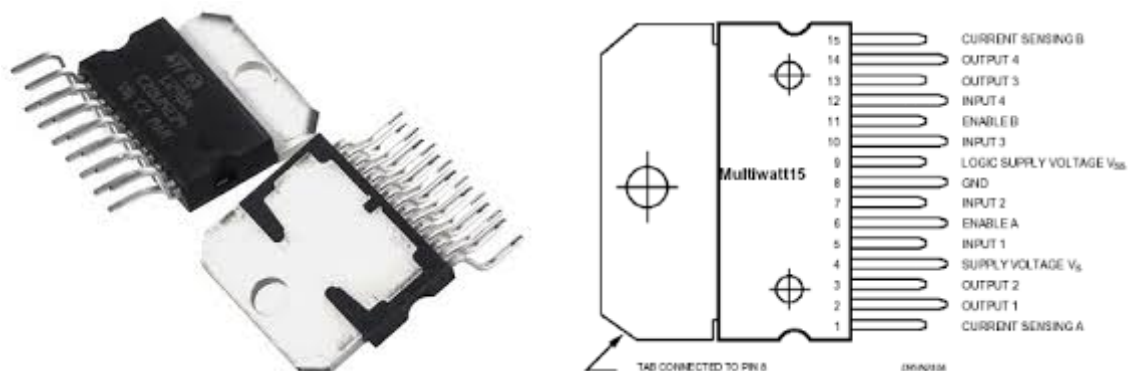


Figure 3.12 Le circuit L298N.

Le moteur peut être alimenté simplement par un relais électromécanique ou par un transistor associé à une diode de roue libre.

Le plus souvent, le moteur à besoin de tourner dans les deux sens de rotation, on utilise alors un dispositif nommé pont en H.

- **Le pont H**

Le pont en H est une structure électronique servant à contrôler la polarité aux bornes d'un dipôle. Il est composé de quatre éléments de commutation généralement disposés schématiquement en une forme de H d'où le nom. Les commutateurs peuvent être des relais, des transistors, ou autres éléments de commutation en fonction de l'application visée. Cette structure se trouve dans plusieurs applications de l'électronique de puissance incluant le contrôle des moteurs, les convertisseurs et Hacheurs, ainsi que les onduleurs, il se présente sous différentes formes passant par les circuits intégrés pour les applications de faibles et moyenne puissances, les circuits discrets ainsi que les modules intégrés pour les moyennes et haute puissance (**Figure 3.13**).

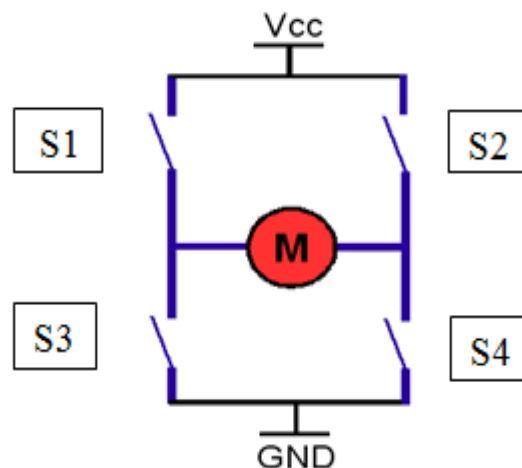


Figure 3.13 Schéma simplifié du pont en H.

- L'utilisation avec les moteurs

Le **pont H** permet de contrôler la polarité de la tension aux bornes du moteur, ou de ne les soumettre à aucune tension (moteur arrêté). Les commutateurs sont actionnés deux par deux soit **S1-S4** ou **S2-S3** pour faire tourner le moteur dans un sens ou dans l'autre. Les paires de relais peuvent avoir des puissances différentes selon que l'utilisation ne requiert pas le même couple dans un sens ou dans l'autre. De plus le pont H permet d'effectuer un freinage magnétique s'il est capable d'en dissiper la puissance générale. Cette opération s'effectue en actionnant soit les deux commutateurs supérieur ou inférieur en temps même, ce qui court-circuite les bornes du moteur. Et le fait par conséquent freiné mieux encore, il est possible avec un peu d'électronique et un contrôleur perfectionner d'effectuer un freinage régénératif (*Figure 3.14*).

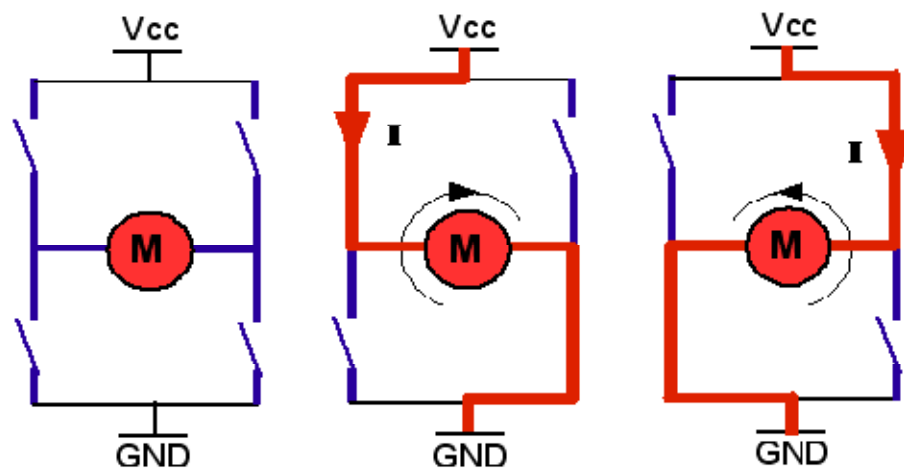


Figure 3.14 Schéma de moteur à l'arrêt, sens direct et le sens inverse.

3.3.3. Notion du signal PWM

Le PWM au MLT est un signal au quel on fait varier la tension sans modifier ni l'amplitude ni la fréquence mais la largeur de l'impulsion. En fait le signal PWM n'est pas continu, c'est un signal à impulsion (ressemble au signal carré), il possède état haut (1) et un état bas (0). Plus le signal reste à l'état haut, plus la tension sera élevée, et vice versa, le pourcentage représentant la fréquence ou la tension est à l'état haut et appelé « rapport cyclique », ou « duty cycle » en anglais (*Figure 1.15*).

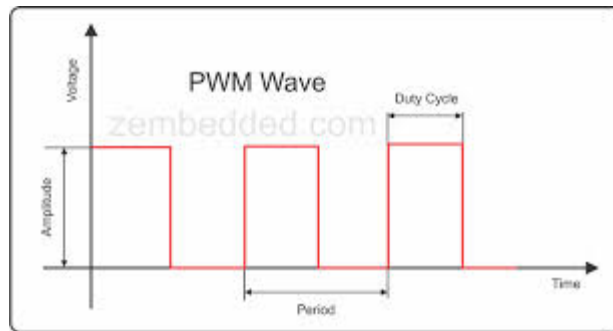


Figure 1.15 Signal PWM.

3.3.4. Diode de roue libre

La DRL simple description, il s'emploie comme un dispositif d'antiparasitage ou de protection des circuits électrique. Il se connecte en parallèle d'une charge inductive pour une continuité du courant électrique dans l'inductance. Il utilisé avec le pont en H pour protégé les moteurs (**Figure 3.16**).

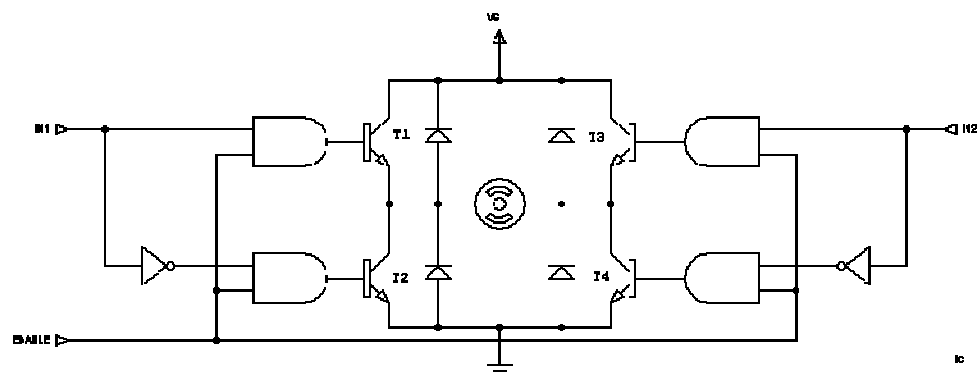


Figure 3.16 Diodes de roue libre

3.3.5. Les relais

Un relais est un *pré-actionneur* constitué au moins :

- d'un électroaimant (bobine circuit ferromagnétique).
- d'une palette mobile supportant un contact mobile.
- ainsi qu'un contact fixe.
- d'un ressort de rappel du contact mobile.



Figure 3.17. Quelques types des relais.

a. Relais électromécanique

Le relais électromécanique, est un organe électrique s'apparente à un interrupteur mécanique non pas effectuée manuellement mais à partir de la mise en alimentation d'une bobine appelée bobine d'excitation ou bobine de commande. Cette dernière crée un champ magnétique même lorsqu'elle est parcourue par un faible courant. Elle provoque alors l'ouverture ou la fermeture d'un interrupteur mis en série dans un circuit contenant un moteur ou une lampe, consommant un courant élevé.

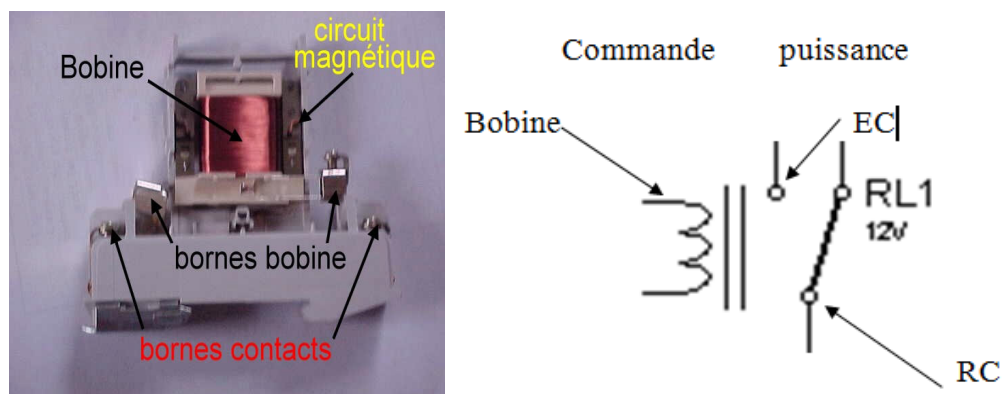


Figure 3.18 Relais électromécanique et leur symbole.

EC : Interrupteur à établissement de circuit ; RC : Interrupteur à rupture de circuit.

b. Câblage des relais électromécanique

Pour éviter des pics de tension provoqués par un effet de self lors de la mise hors tension de la bobine du relais, nous proposons une gamme de relais équipés de résistances ou de diodes en parallèle à la bobine, qui permettent d'absorber ou bloquer les courants résiduels.

- ***Tension nominale :***

12 V : pour les machines agricoles et engins de chantier etc.

24 V : pour les vérins, bus, véhicules communaux etc.

3.3.6. Constitution de notre carte de puissance

- Régulateur (7405).
- 2 L298N.
- Inverseur (74HC14N).
- 3 relais électromécaniques (SRD-06VDC-SL-C).
- 19 Diode de roue libre (N4003).
- 10 LEDs.
- 7 Résistances (3 de 10K et 4 de 2.2.K).
- 3 Condensateur (100nf).
- 8 Bornier électronique (SH134R ou bien KF301-2P).
- 5 Connecteur (1x3).
- Connecteur (1x7).
- Connecteur (1x8).

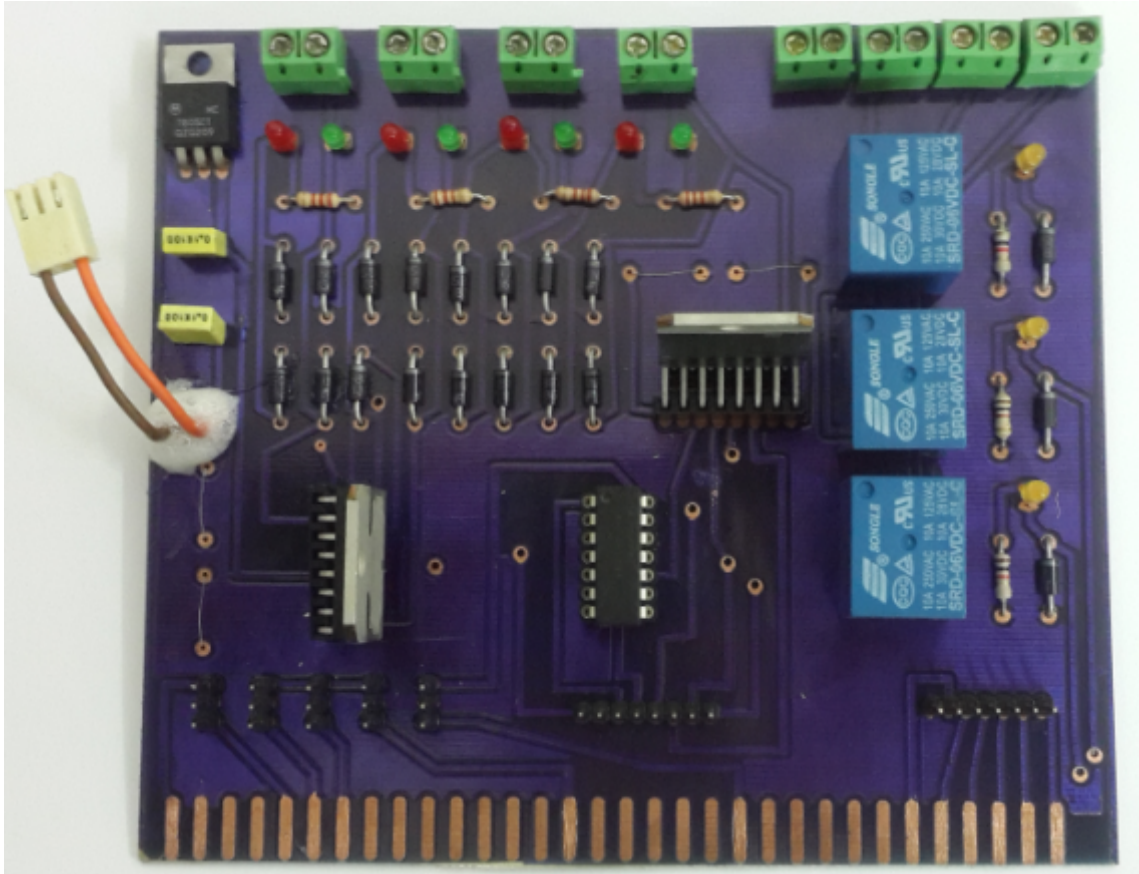


Figure 3.19 Carte de puissance.

L'interface entre la carte de commande et la carte de puissance :

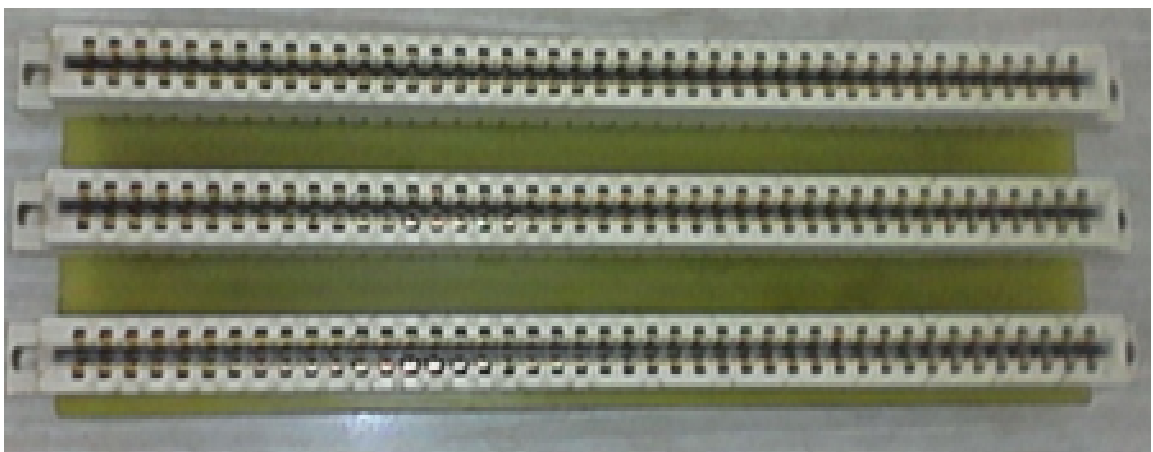


Figure 3.20 l'interface entre la carte de commande et la carte de puissance.

3.4. la carte ARDUINO :



Figure 3.21. Le symbole d'ARDUINO.

3.4.1. Définition

Le système Arduino est une plateforme open-source d'électronique programmée et basée sur une simple carte à microcontrôleur (de la famille AVR) et un logiciel. Ce dernier est un véritable environnement de développement intégré pour écrire, compiler et transférer le programme vers la carte à microcontrôleur.

Le microcontrôleur permet, à partir d'événements détectés par des capteurs, de programmer et commander des actionneurs ; la carte ARDUINO est donc une interface programmable (**Figure 3.22**).



Figure 3.22. Quelques cartes ARDUINO.

3.4.2. Constitution de la carte ARDUINO :

La carte ARDUINO repose sur un circuit intégré (un mini-ordinateur appelé également microcontrôleur) associée à des entrées et sorties qui permettent à l'utilisateur de brancher différents types d'éléments externes.

On a travaillé dans ce projet avec la carte Arduino MEGA (**Figure 3.23**).

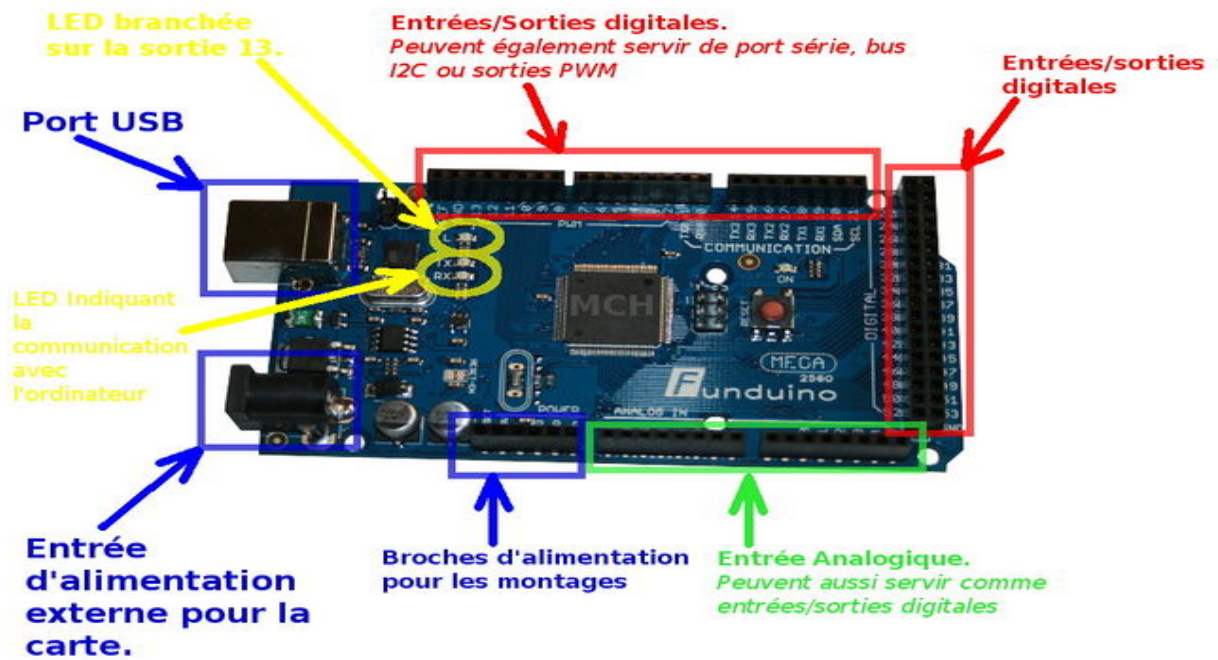


Figure 3.23. La constitution de la carte ARDUINO(MEGA).

- **Entrées et sorties numériques**

Chacune des 54 broches numériques de la carte Mega peut être utilisée soit une entrée numérique soit comme une sortie numérique, en utilisant les instructions `pinMode()`, `digitalWrite()` et `digitalRead()` de langage Arduino. Ces broches fonctionnent en 5V. Chaque broche peut fournir ou recevoir un maximum de 40mA d'intensité et dispose d'une résistance interne de "rappel au plus" (pull-up) (déconnectée par défaut) de 20-50KOhms. Cette résistance interne s'active sur une broche en entrée à l'aide de l'instruction `digitalWrite(broche, HIGH)`.

- **Broches analogiques**

La carte Mega dispose de 16 entrées analogiques, chacune pouvant fournir une mesure d'une résolution de 10 bits (c.-à-d. sur 1024 niveaux de 0 à 1023) à l'aide de la très utile fonction `analogRead()` de langage Arduino. Par défaut, ces broches mesurent entre 0V (valeur 0) et le 5V (valeur 1023), mais il est possible de modifier la référence supérieure de la plage de mesure en utilisant la broche AREF et l'instruction `analogReference()` de langage Arduino.

Remarque : les broches analogiques peuvent être utilisées en tant que broche numérique.

3.4.3. La structure de programmation

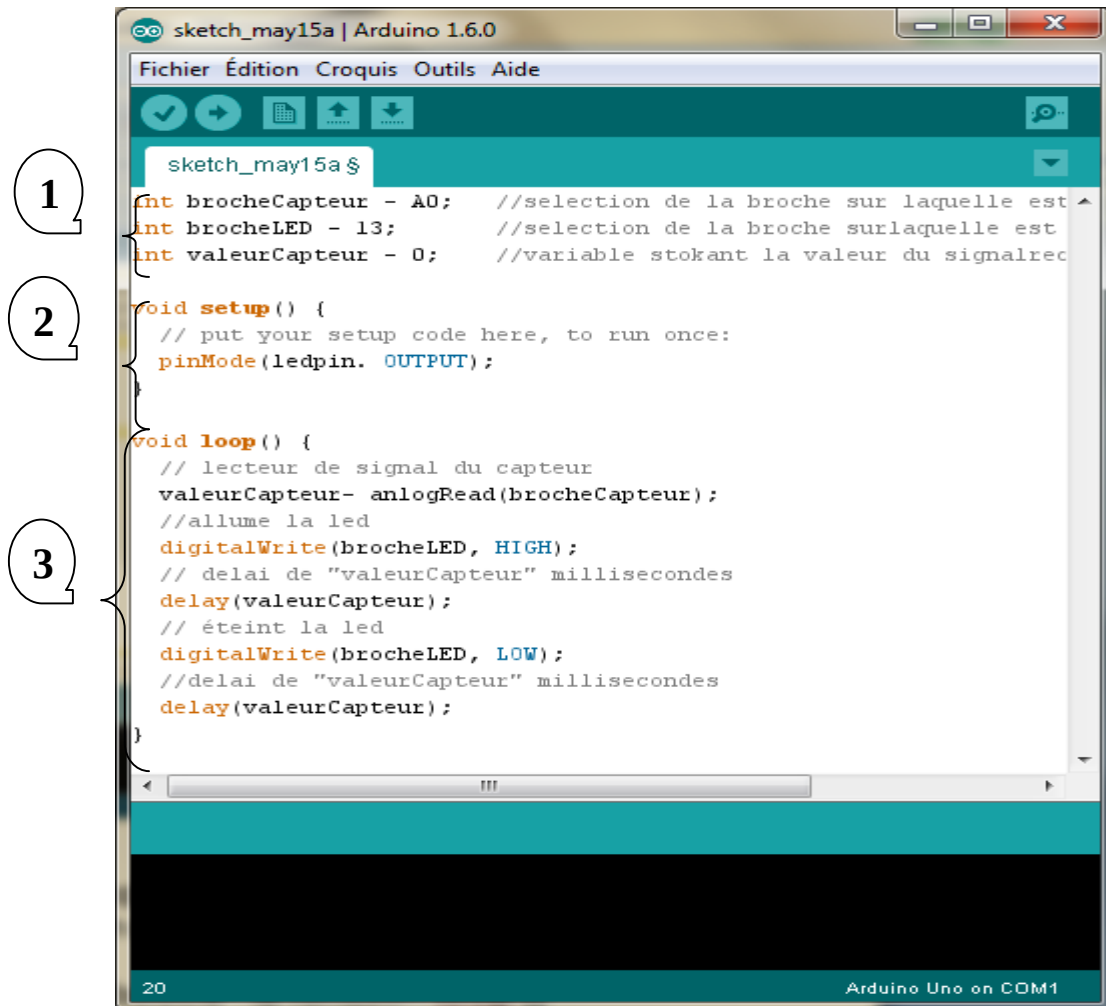


Figure 3.24. La structure de programmation.

1. la partie déclaration des variables (optionnelle).
2. la partie initialisation et configuration des entrées/sorties : la fonction **setup ()**.
3. la partie principale qui s'exécute en boucle : la fonction **loop ()**.

Dans chaque partie d'un programme sont utilisées différentes instructions issues de la syntaxe du langage ARDUINO.

Toujours en écrire des commentaires sur le programme soit :

En multi ligne : en écrire des `/**/`.

Sur une ligne de code en se séparant du code avec `//`.

- **Barre d'actions :**

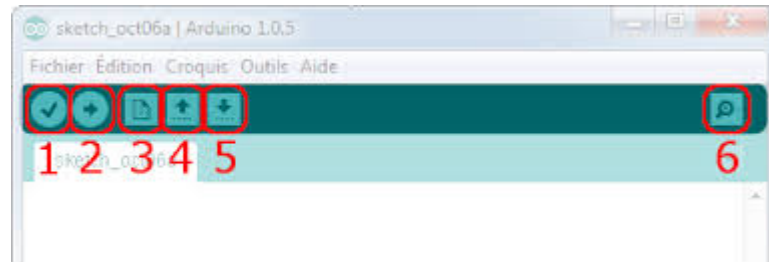


Figure 3.25. Barre d'actions.

1- Bouton « Verify » (Vérifier) : il permet de compiler votre programme et de vérifier si des erreurs s'y trouvent. Cette procédure prend un certain temps d'exécution et lorsque est terminée, elle affiche un message de type « Binary sketch size : ... » Indiquant la taille du sketch téléversé.

2- Bouton « Upload » (Téléverser) : ce bouton permet de compiler et téléversé votre sketch sur la carte Arduino.

3- Bouton « New » (Nouveau) : ce bouton permet de créer un nouveau sketch.

4- Bouton « Open » (Ouvrir) : il fait apparaître un menu qui permet d'ouvrir un sketch qui figure dans votre dossier de travail ou des exemples de sketches intégrés au logiciel.

5- Bouton « Save » (Sauvegarder) : il permet de sauvegarder votre sketch.

6- Bouton « Serial Monitor » (Moniteur sériel) : il est utilisé pour afficher l'information qui est envoyée par la carte ARDUINO vers l'application (habituellement par le câble USB). Il permet aussi d'envoyer de l'information à la carte ARDUINO.

3.4.4. Le principe de fonctionnement

- On conçoit ou on ouvre un programme existant avec le logiciel ARDUINO.
- On vérifie ce programme avec le logiciel ARDUINO (compilation).
- Si des erreurs sont signalées, on modifie le programme.
- On charge le programme sur la carte.
- On câble le montage électronique.
- L'exécution de programme est automatique après quelques secondes.
- On alimente la carte soit par le port USB, soit par une source d'alimentation Autonome (pile 9 volts par exemple).
- On vérifie que notre montage fonctionne.

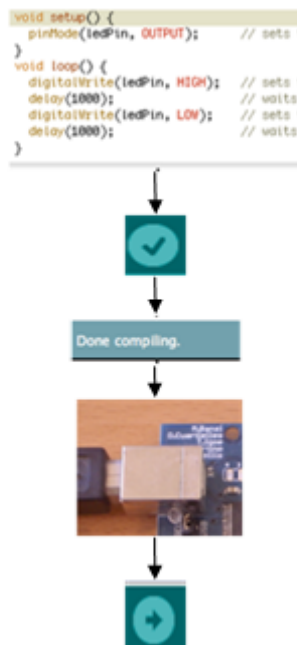


Figure 3.26. Principe de fonctionnement d'un ARDUINO.

3.4.5. Le but de l'utilité

Le système ARDUINO, nous donne la possibilité d'allier les performances de la programmation à celles de l'électronique.

Plus précisément, nous allons programmer des systèmes électroniques. Le gros avantage de l'électronique programmée c'est qu'elle simplifie grandement les schémas électroniques et par conséquent, le coût de la réalisation, mais aussi la charge de travail à la conception d'une carte électronique.

L'utilité est sans doute quelque chose que l'on perçoit mal lorsque l'on débute, mais une fois que vous serez rentré dans le monde de l'ARDUINO, vous serez fasciné par l'incroyable puissance dont il est question et des applications possibles.

3.4.6. Les bonnes raisons de choisir ARDUINO

Tout d'abord, si vous voulez commencer à programmer à moindre coût, alors l'UNO est idéale. Si vous préférez brancher une multitude de capteurs, utiliser plusieurs périphériques en série, alors la Méga est parfaite pour ça, si vous voulez les mêmes propriétés que la Méga mais avec une puissance de calcul supérieure alors la DUE votre demande, Si vous cherchez à miniaturiser votre système, optez pour une Mini. Enfin, si vous cherchez un véritable ordinateur embarqué comparable à une Raspberry, la Yun est le choix le plus évident.

Remarque : presque toutes cartes ARDUINO fonctionnent en 5V sauf l'ARDUINO DUE et l'Arduino Zéro PRO qui fonctionnent en 3.3V.

3.5. Conclusion

La gestion des différents systèmes de la chaîne de conditionnement se fait par le module CUBLOC CB280 (carte de commande), la commande des moteurs à courant continu se fait par les ponts en H (carte de puissance) et leurs asservissement se fait par la carte Arduino (Mega).

Chapitre 4 Programmation de la machine

4.1. Introduction

A ce niveau de notre étude, on arrive à la partie la plus importante, c'est la programmation de microcontrôleur (CUBLOC) et l'interface programmable (Arduino) pour communiquer entre le PC et la chaîne.

On a utilisé un régulateur PID pour asservir la vitesse de rotation des moteurs, pour donner un système plus précis.

4.2. L'asservissement des moteurs

Pour pouvoir avoir un bon fonctionnement des moteurs il faut réaliser un asservissement. Cela consiste à récupérer une information sur la vitesse du moteur, et s'en servir pour ajuster la tension de commande. Il existe plusieurs méthodes d'asservissement, nous utilisons dans notre système le correcteur PID.

4.2.1. L'action proportionnelle (P)

La loi de commande $u(t)$ est proportionnelle à l'écart $\xi(t)$:

$$U(t) = K_p \xi(t) = K_p (U_d - Y) \dots\dots\dots(1)$$

- **L'effet**

- Permet de corriger les effets d'une perturbation.
- L'augmentation du gain entraîne une diminution de l'erreur mais elle ne l'annule pas, et peut déstabiliser le système.

4.2.2. L'action intégrateur (I)

La loi de la commande est de la forme :

$$U(t) = (1/T_i) \int e(t) dt \dots\dots\dots(2)$$

La constante de temps T_i est appelée la constante de temps d'intégration.

- **L'effet**

- Corrige les effets d'une perturbation.
- Annule l'erreur statique.
- Introduit un dépassement.
- Mais elle n'est pas très rapide.

4.2.3. L'action dérivateur (D)

La loi de la commande est de la forme :

$$U(t) = T_d (de(t)/d(t)) \dots\dots\dots(3)$$

L'effet

- Accélère la correction,
- Stabilise plus rapidement le système,
- Mais n'annule pas l'erreur statique et est sensible aux vibrations.

4.2.4. Régulateur proportionnel intégrateur dérivateur (PID)

L'intérêt de correcteur PID est d'intégrer les effets positifs des trois correcteurs précédents. La détermination des coefficients K , T_i et T_d du correcteur PID permet d'améliorer à la fois la précision, la stabilité et la rapidité.

La loi de la commande est de la forme :

$$U(t) = K e(t) + (1/T_i) \int e(t) dt + T_d (de(t)/d(t)) \dots\dots\dots(4)$$

4.2.5. Identification du moteur à courant continu

L'identification des systèmes se fait par la « Toolbox » d'identification dont l'accès s'effectue avec l'instruction 'ident', on a utilisé cette instruction dans notre travail pour trouver un bon modèle qui va asservir les moteurs qui implique une bonne précision de la station.

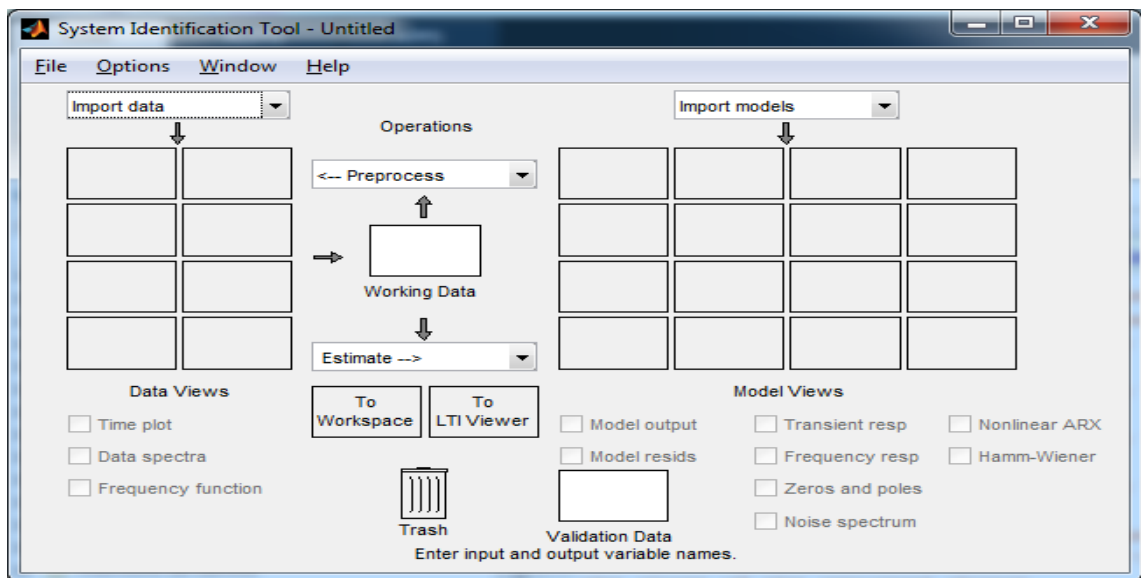


Figure 4.1. Fenêtre de l'instruction 'ident' de Matlab.

Le signal d'entrée (PWM) et de sortie (vitesse) de notre modèle représenté par la figure 4.2

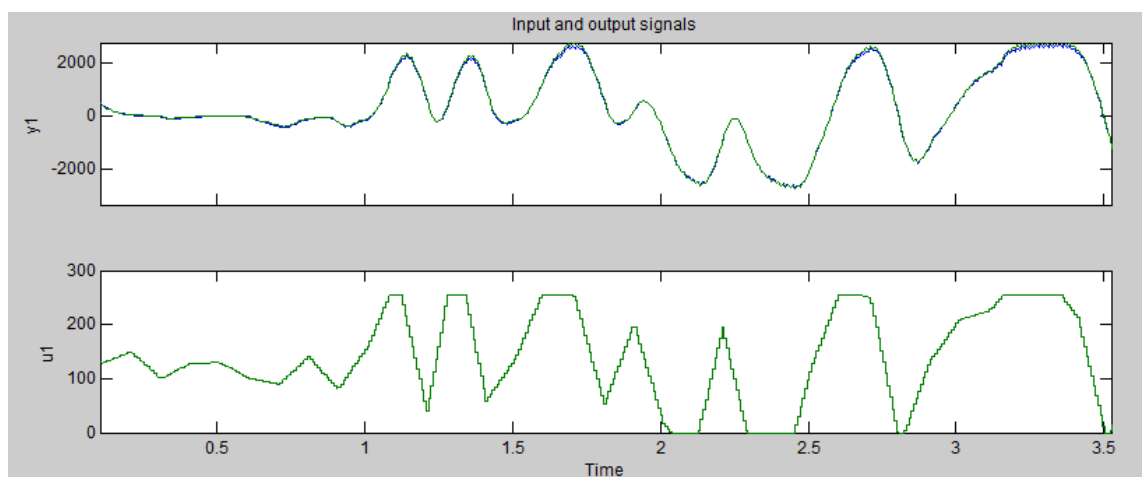


Figure 4.2. Signal d'entrée et de sortie de notre modèle superposée avec la vitesse mesurée.

On remarque que le signal de sortie suivre le signal d'entrée

Pour vérifier notre modèle on a testé par un autre signal et on a remarqué que la sortie suivre l'entrée de même manière que le premier signal, ce résultat illustré par la figure 4.3 qui représente deux régions (signal mauve pour estimer les paramètres du modèle et le signal vert pour la validation)

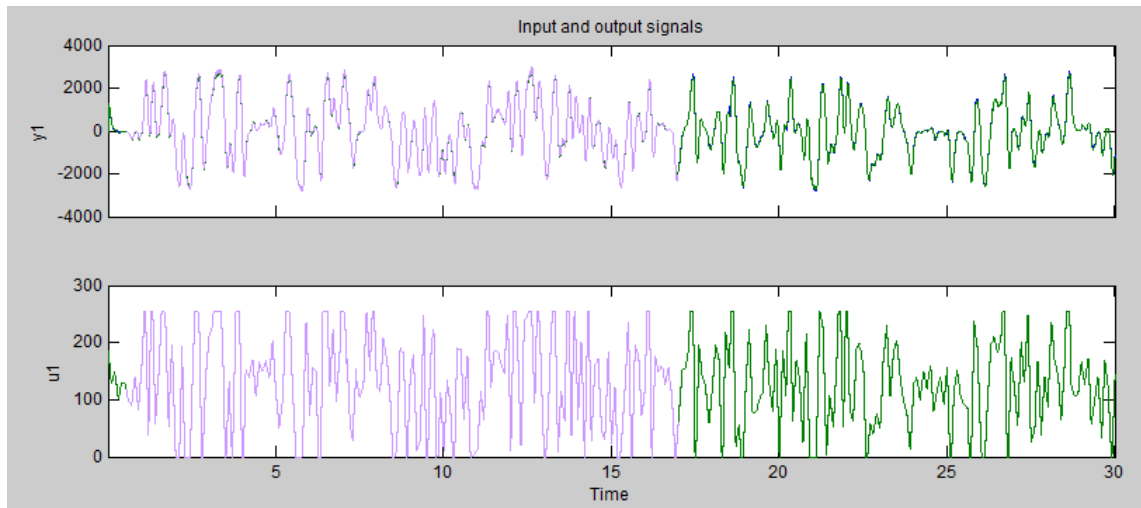


Figure 4.3 Signal d'entrée et de sortie de deux données.

La figure 4.4 représente la réponse de modèle et la réponse mesurée.

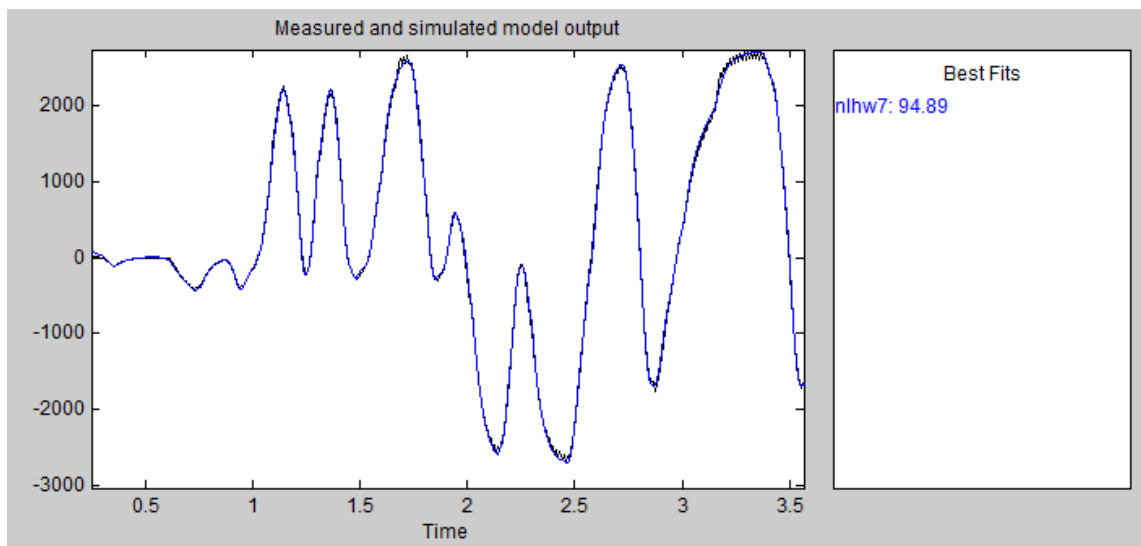


Figure 4.4. Réponse de modèle et la réponse mesure.

La réponse de modèle et la réponse mesurée presque identique avec un pourcentage de 94.89%.

Notre modèle est bon, puisque la réponse de modèle et la réponse mesure est presque identique avec une erreur presque nulle. Alors le but recherché a été atteint.

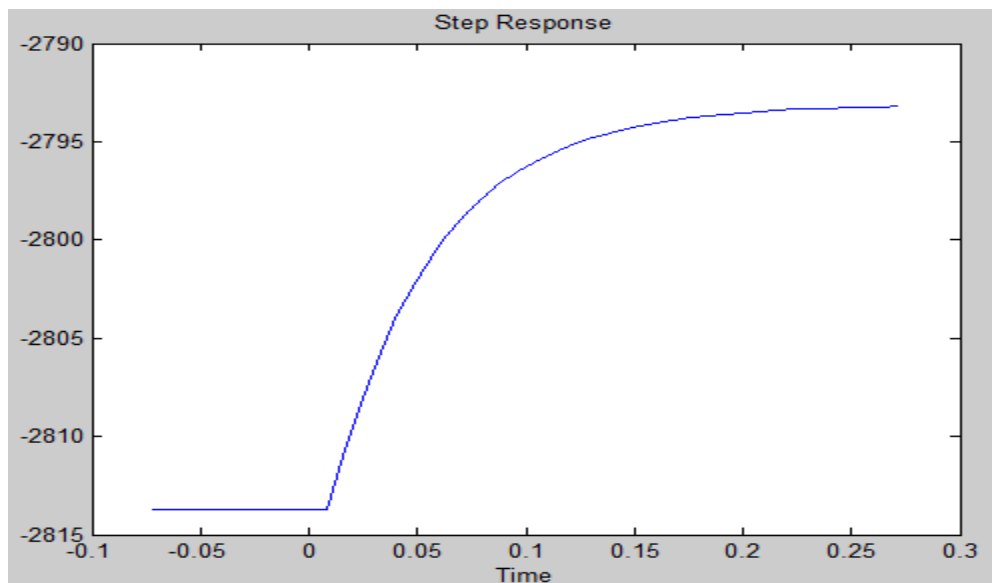


Figure 4.5. La réponse à l'échelon du modèle

On remarque que le temps de réponse du modèle est relativement faible.

Pour asservir la position du bras nous avons travaillé par deux contrôleurs PID respectivement le 1^{er} pour asservir la vitesse des moteurs et la 2^{ème} pour la position. La position angulaire de l'axe du moteur a été obtenue après une intégration de la vitesse, ceci est représenté par le schéma simulink de la figure 4.6

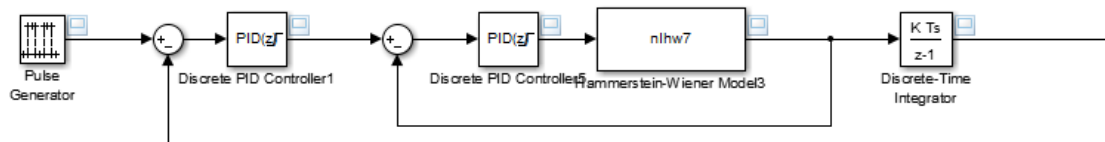


Figure 4.6. Schéma bloc du correcteur PID en régulation de vitesse et de position.

Un exemple de consigne (360°) et l'angle de rotation du moteur sont représentés par la figure 4.7

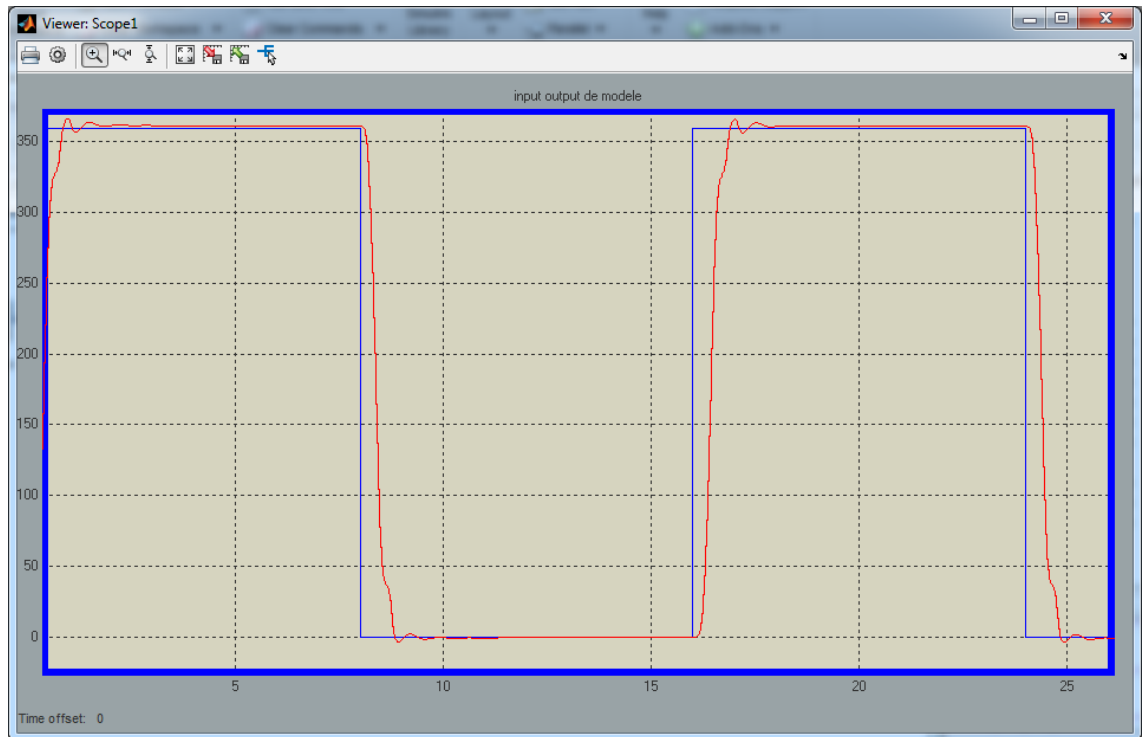


Figure 4.7. Evolution de la consigne (bleu) et de la position (rouge).

On remarque que le signal de la position suit le signal d'entrée, sauf au niveau de changement d'état on a petit retard pour obtenir la position désirée.

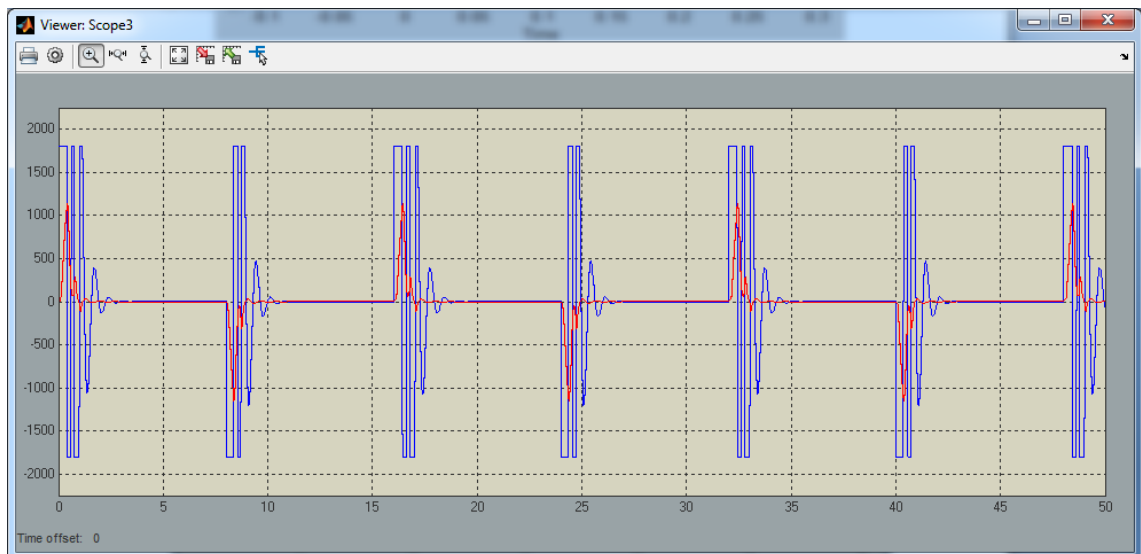


Figure 4.8. Evolution du signal de control (bleu) et de vitesse (rouge).

Le signal PWM (l'entrée du modèle) qui commande la vitesse des moteurs est représenté par la figure 4.9

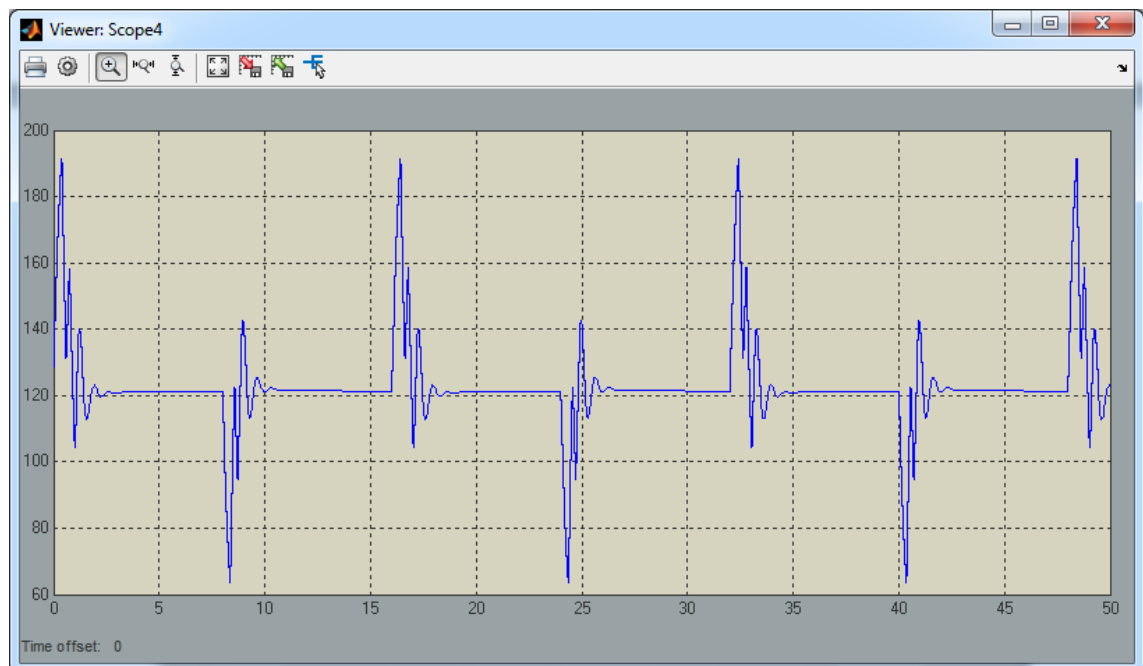


Figure 4.9. Signal issu du contrôleur de vitesse (PWM (l'entrée de modèle)).

Finalement on peut conclure que notre but d'asservir la position à partir de l'asservissement de vitesse nous a donné de très bons résultats.

4.3. Partie programmation

4.3.1. LADDER du CUBLOC [16] & [17]

a. Programmation sous LADDER

La fenêtre ci-dessous montre un exemple de développement d'une application en langage « LADDER » via le CUBLOC STUDIO.

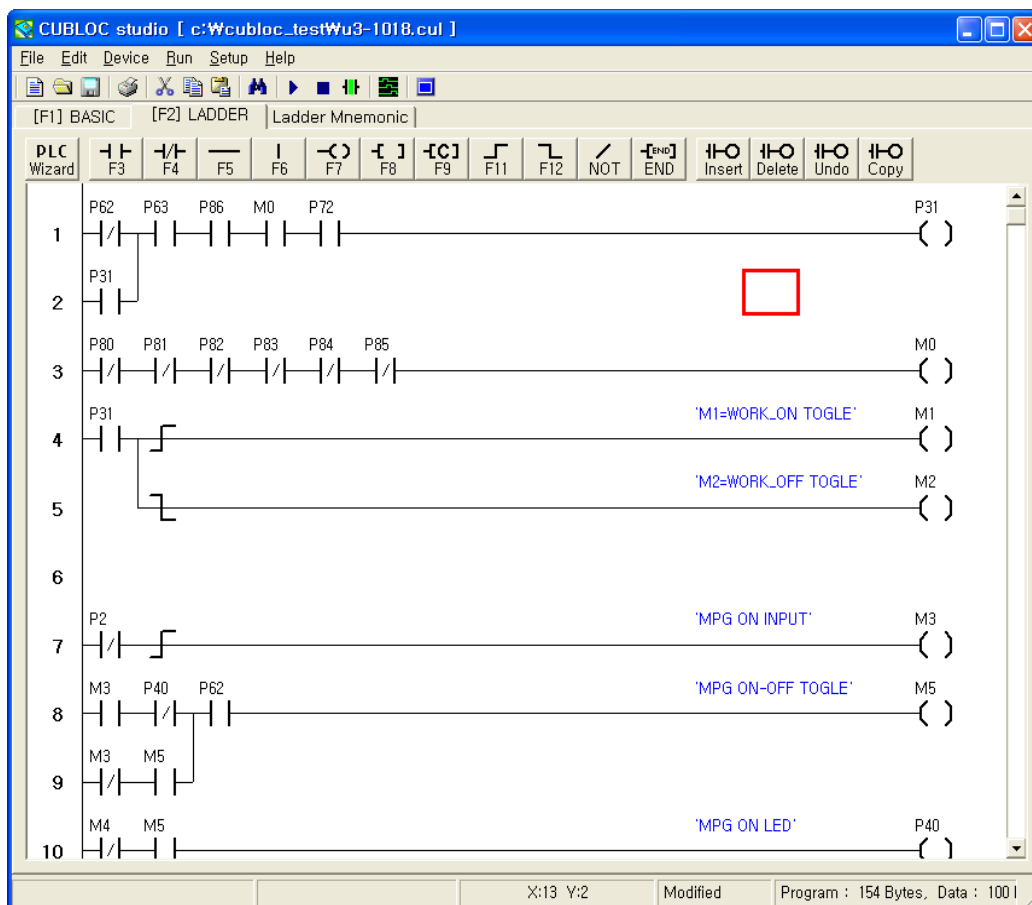


Figure 4.10. Programme en LADDER.

Le carré rouge représente le curseur de sélection de LADDER. Vous pouvez utiliser les touches de directions (haut/bas/droite/gauche) ou la souris pour déplacer ce curseur.

Une fois positionné à l'endroit voulu, vous pourrez utiliser les touches de fonctions F3 à F12 pour appliquer le symbole associé à l'écran.

Vous pourrez également saisir du texte pour les symboles qui le nécessitent

- 1- Pressez F3 pour créer un « contact »
- 2- Tapez « START » et pressez la touche « ENTER »
- 3- Pressez F5 plusieurs fois afin de continuer la ligne
- 4- On bout de ligne pressez F7 et tapez « RELAY »
- 5- Allez à la ligne suivante (line) et pressez le bouton « END »

Pensez à presser la touche « ENTER » à la fin de chaque texte que vous aurez à saisir.

De même pour rappel. La fin de votre programme LADDER doit impérativement se finir par la commande « END ».

- **Monitoring en LADDER**

Le CUBLOC STUDIO supporte le monitoring temps réel du LADDER

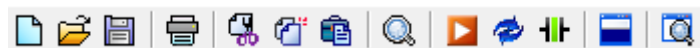


Figure 4.11. Barres d'outils de CUBLOC STUDIO.

Pour se faire, cliquez sur le bouton prévu à cet effet en haut de l'écran.

L'état des contacts actifs (ON) est affiché en VERT. Les valeurs des timers et compteurs seront affichées en valeurs décimales.

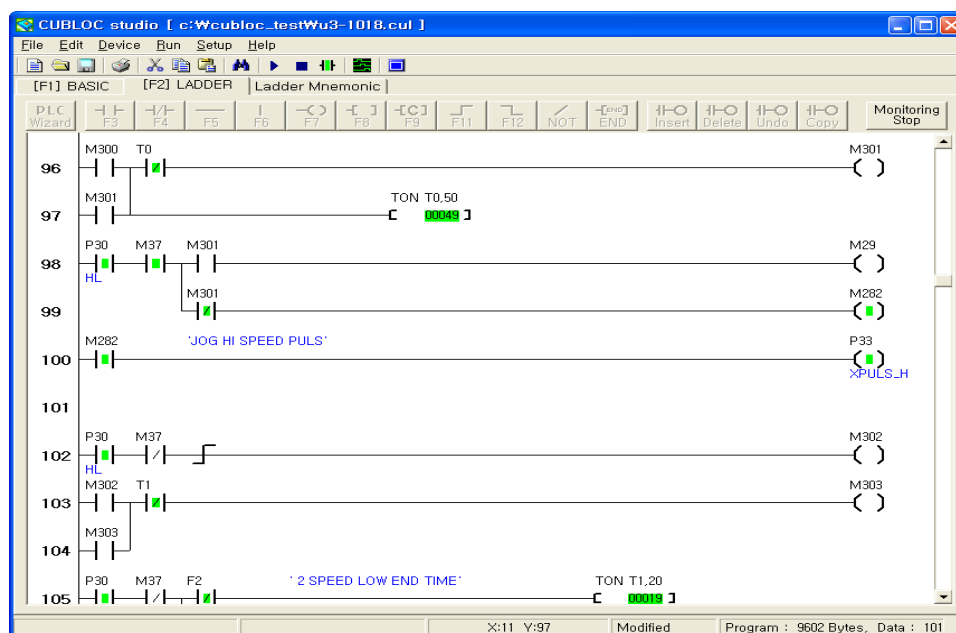


Figure 4.12. Programme en LADDER actif.

Note : assurez-vous impérativement que le monitoring est stoppé avant de modifier à nouveau votre programme ou de télécharger ce dernier dans le module CUBLOC.

- **Utilisation des Registres du LADDER**

Ce menu accessible depuis : Run ->View Register Usage vous permet de visualiser la dénomination associée à chaque « Registres » du LADDER afin que vous puissiez faciliter encore d'avantage la phase de développement de vos applications.

Liste des Registres utilisés par le LADDER

Le tableau ci-dessous donne la liste des Registres des modules CB220, CB280

Nom de Registre	Gamme	Unité	Possibilités
Registre P E/S	P0~P127	1 bit	Interface avec dispositifs externes
Registre internes M	M0~M511	1 bit	Registres internes
Registre spéciaux F	F0~F127	1 bit	Etat systèmes
Timer T	T0~T99	16 bits (1 Word)	Pour Timers
Capteur C	C0~C49	16 bits (1 Word)	Pour Compteurs
Step Enable S	S0~S15	256 steps (1 octet)	For Step Enabling
Mémoire donnée D	D0~99	16 bits (1Word)	Mémorisation de données

Tableau 4.1. Les registres utilisés par le LADDER.

- **Utilisation des E/S**

Les ports d'E/S de CUBLOC peuvent être utilisés à la fois par le programme BASIC et LADDER. Sans aucune déclaration particulière, tous les ports d'E/S sont attribués au programme BASIC. Pour utiliser des ports d'E/S en LADDER, vous devrez au préalable utiliser la commande 'Usepin' afin de déclarer les ports qui seront utilisés en LADDER.

Les commandes USEPIN 0, IN et USEPIN 1, OUT permettent de configurer les ports P0 en entrée et P1 en sortie afin qu'ils puissent être utilisés en LADDER.

- **Utilisation de « alias »**

Lorsque vous concevez votre programme en LADDER et que vous utilisez des Registres du type P0, P1 ou encore M0, il est plus simple de leur attribuer un nom (alias) qui vous permettra de faciliter la lecture et la rédaction de votre application.

Pour déclarer vos alias vous pouvez également utiliser la commande USEPIN.

- **Démarrage de LADDER**

A la mise sous tension, le programme BASIC de CUBLOC est toujours exécuté en premier. Vous pourrez alors dans ce programme BASIC ordonner l'exécution du programme LADDER à l'aide de la commande « SET LADDER ON ». Si vous ne placez pas la commande « SET LADDER ON » dans le programme BASIC, le programme LADDER ne sera pas exécuté.

4.3.2. ARDUINO

On a fait des mesures pour organiser la programmation d'ARDUINO :

La longueur de la tige filetée est 25 cm (comme illustré la figure 4.9), et pour le moteur fait un tour le bras va descendre/monter d'une distance de 2.4 cm, donc

2.4 cm → 1 Tr → 360 impulsions

25 cm → 10.4 Tr → 3750 impulsions

La distance parcourue par le support du bras est de 18 cm (comme illustré par la figure 4.10).

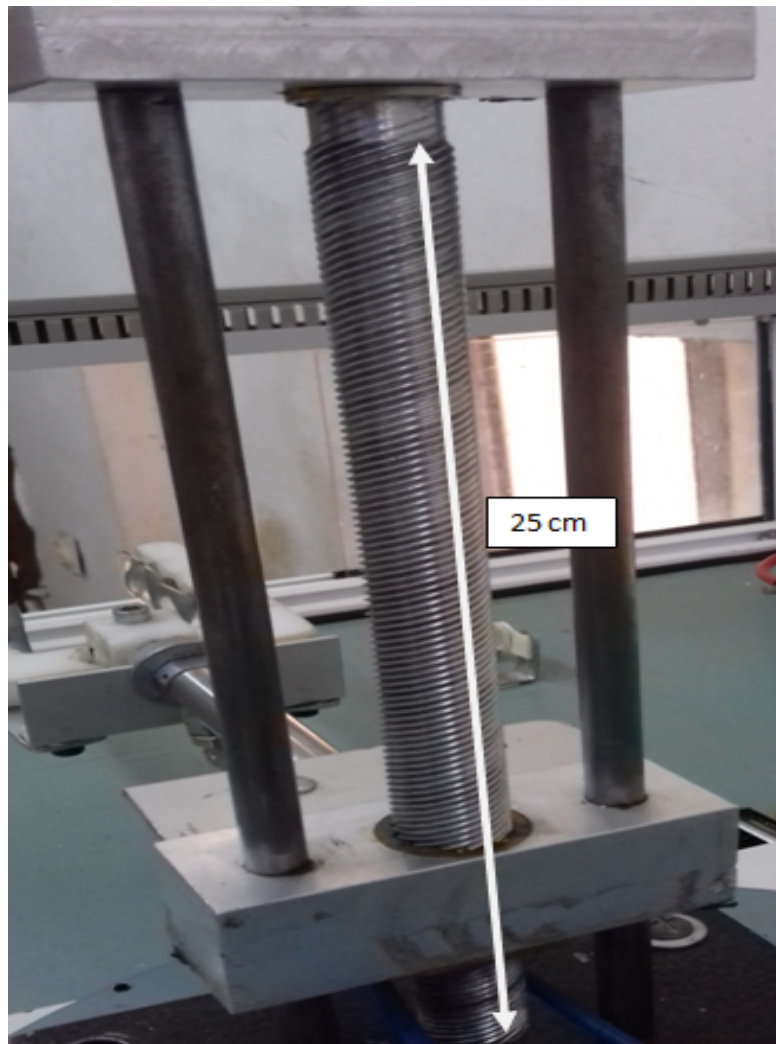


Figure 4.13. La tige filetée.

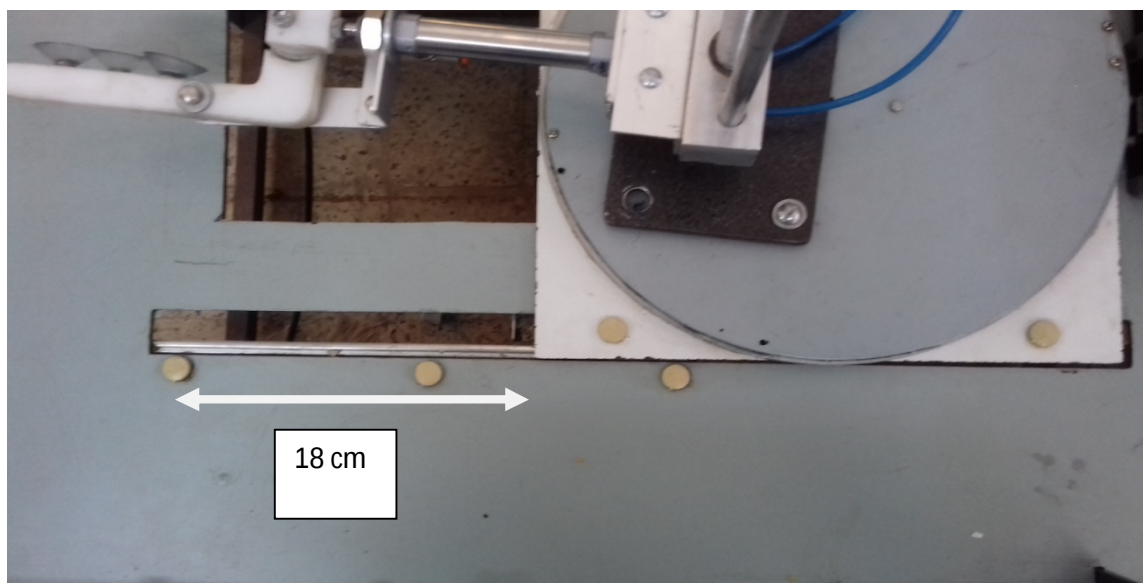


Figure 4.14. Distance de parcours du bras.

Moteurs	Pins	Impulsions	Distance
M1	Pin52, pin50	136	18 cm
M2	Pin46, pin48	3200	20cm
M3	Pin44, pin46	90	90°

Tableau 4.2. Table des mesures.

Pin50 (ENABLE)	Pin52	Moteur (M1) (impulsions)
0	0	0
1	1	136
1	0	0

Tableau 4.3. Table de vérité de moteur (avant/arrière).

Pin46 (ENABLE)	Pin48	Moteur (M2) (impulsions)
0	0	0
1	1	3200
1	0	0

Tableau 4.4. Table de vérité de moteur (descende/monte).

Pin42 (ENABLE)	Pin44	Moteur(M3) (impulsions)
0	0	0
1	1	90
1	0	0°

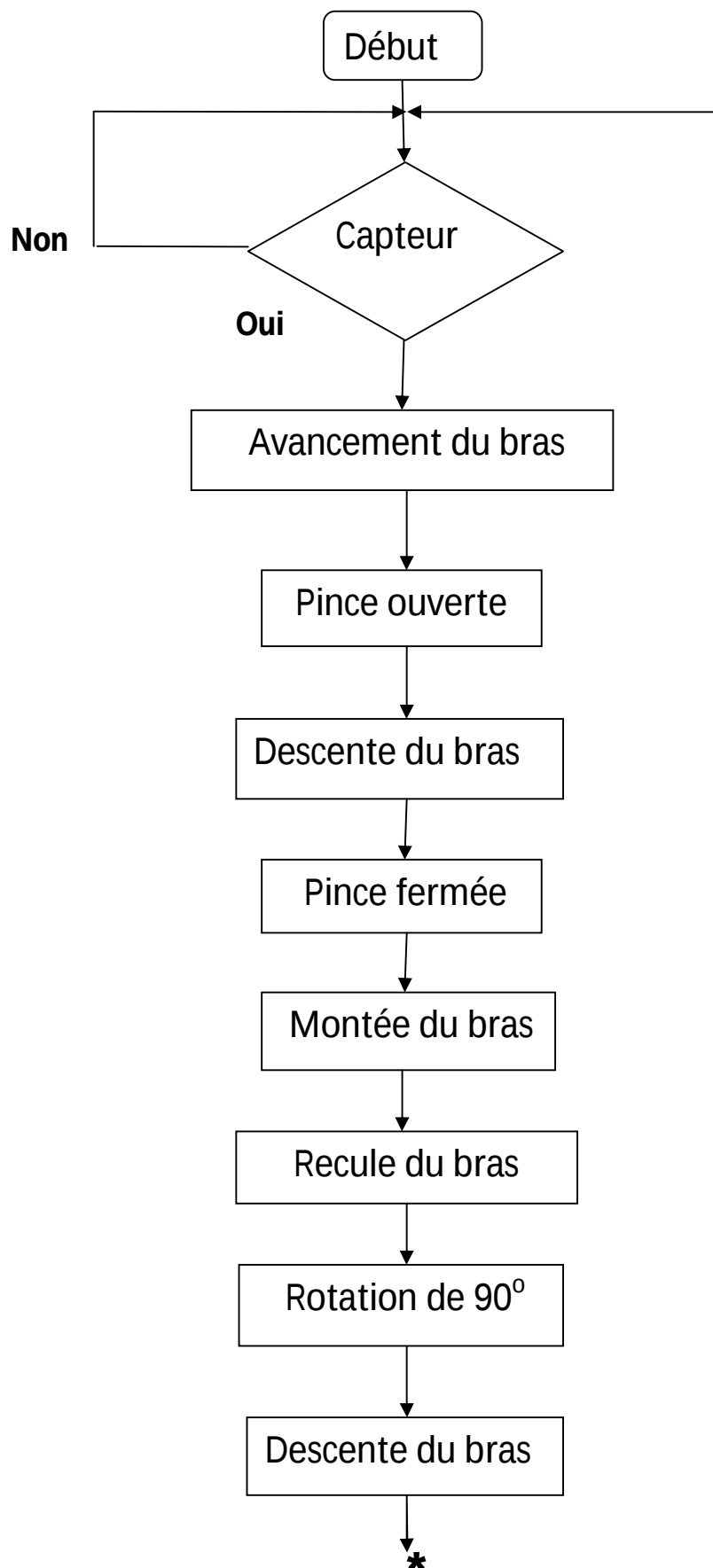
Tableau 4.5. Table de vérité de moteur (rotation de 90°).

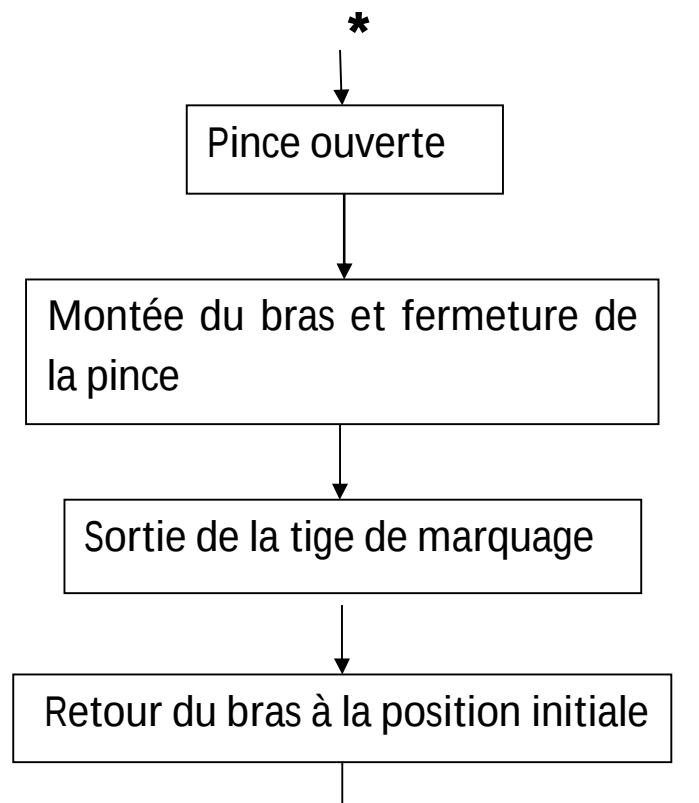
4.4. Principe de fonctionnement de la chaine

4.4.1. Cahier des charges

Nous avons travaillé sur le cahier des charges suivant : au repos le bras est à l'arrière de la station, la pince étant fermée et il se trouve à la position haute. Le cycle commence si l'utilisateur donne un ordre, et a condition de la présence d'un objet. Lorsque ces conditions sont vérifiées le moteur démarre pour avancer le bras, la pince va attraper l'objet, ensuite le bras fait une rotation de 90° dans l'un des deux sens pour déposer ce dernier à proximité d'un vérin pneumatique qui fait le marquage. La suite des opérations dépendra du choix final de l'utilisateur.

4.4.2. Organigramme

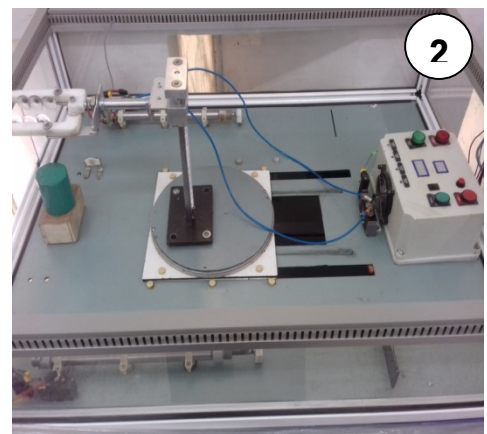
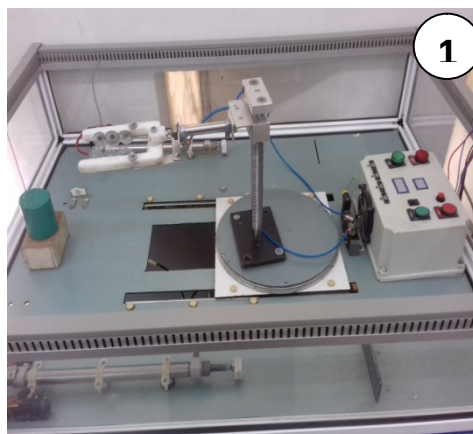


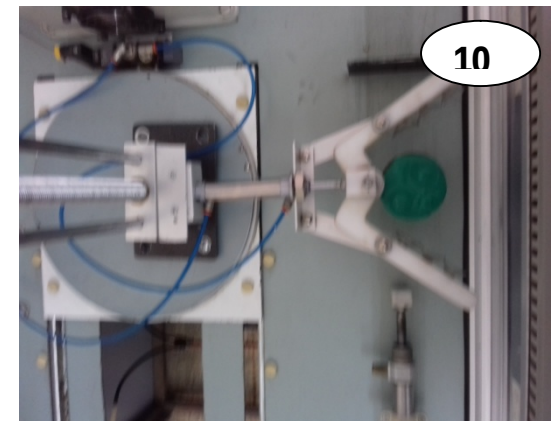
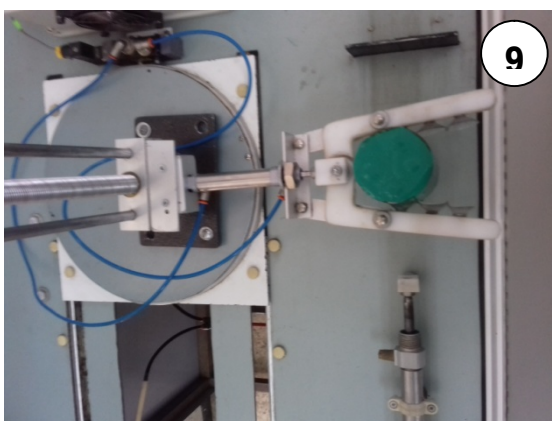
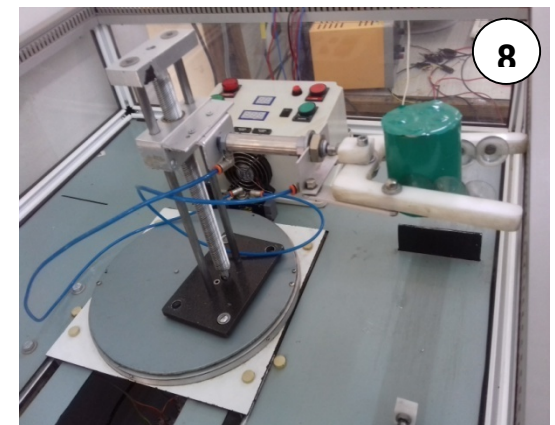
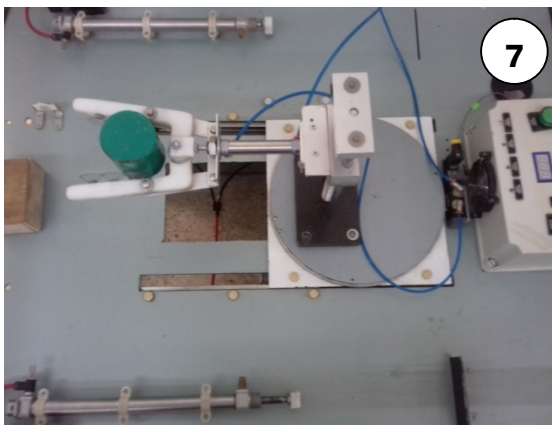
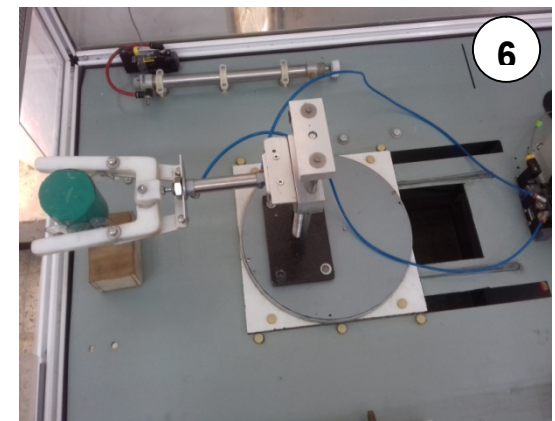
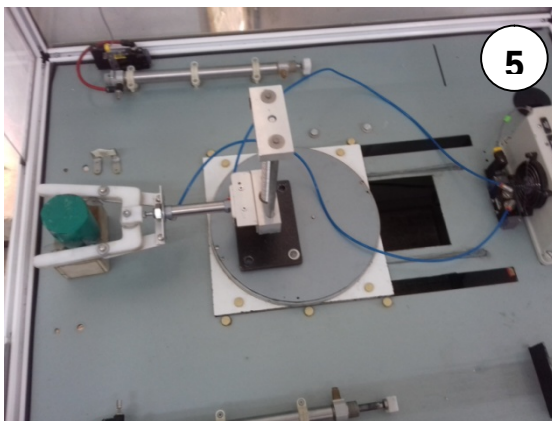
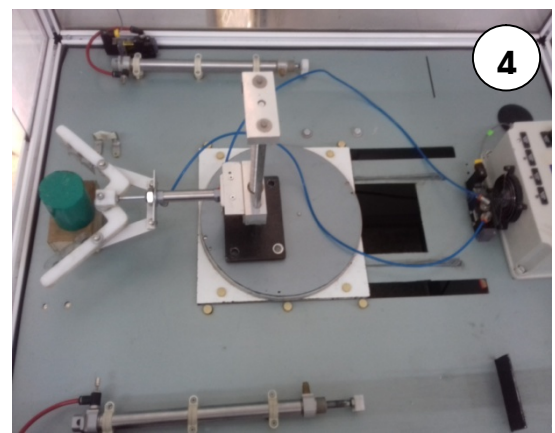
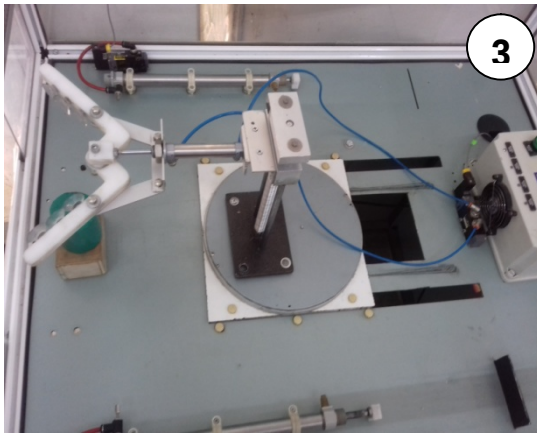


Remarque

C'est le même cycle pour l'autre coté, sauf que le moteur M3 tourne dans le sens inverse.

4.4.3. Cycle de travail





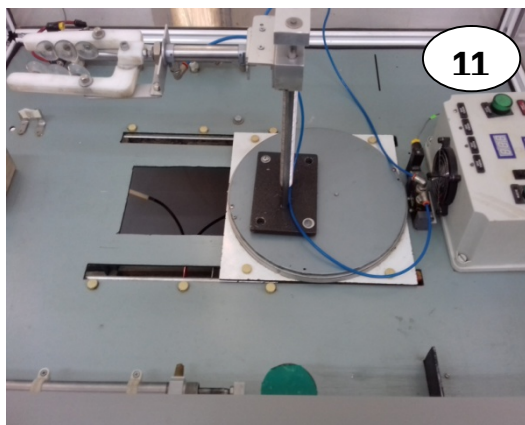


Figure 4.15 cycle de travail.

4.4.4. Schéma de la chaîne

a. Schéma pneumatique

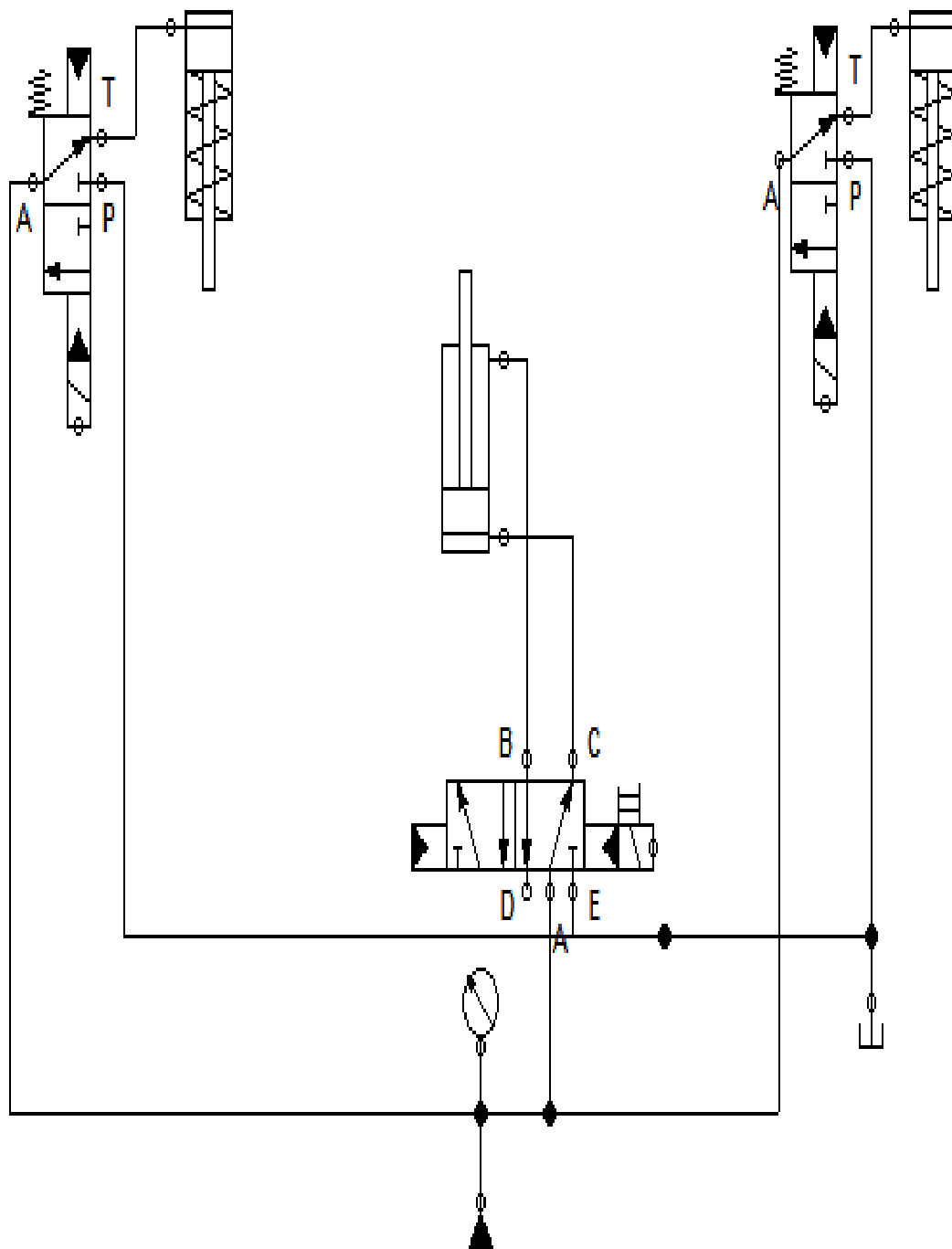


Figure 4.16. Schéma pneumatique.

b. Schéma synoptique

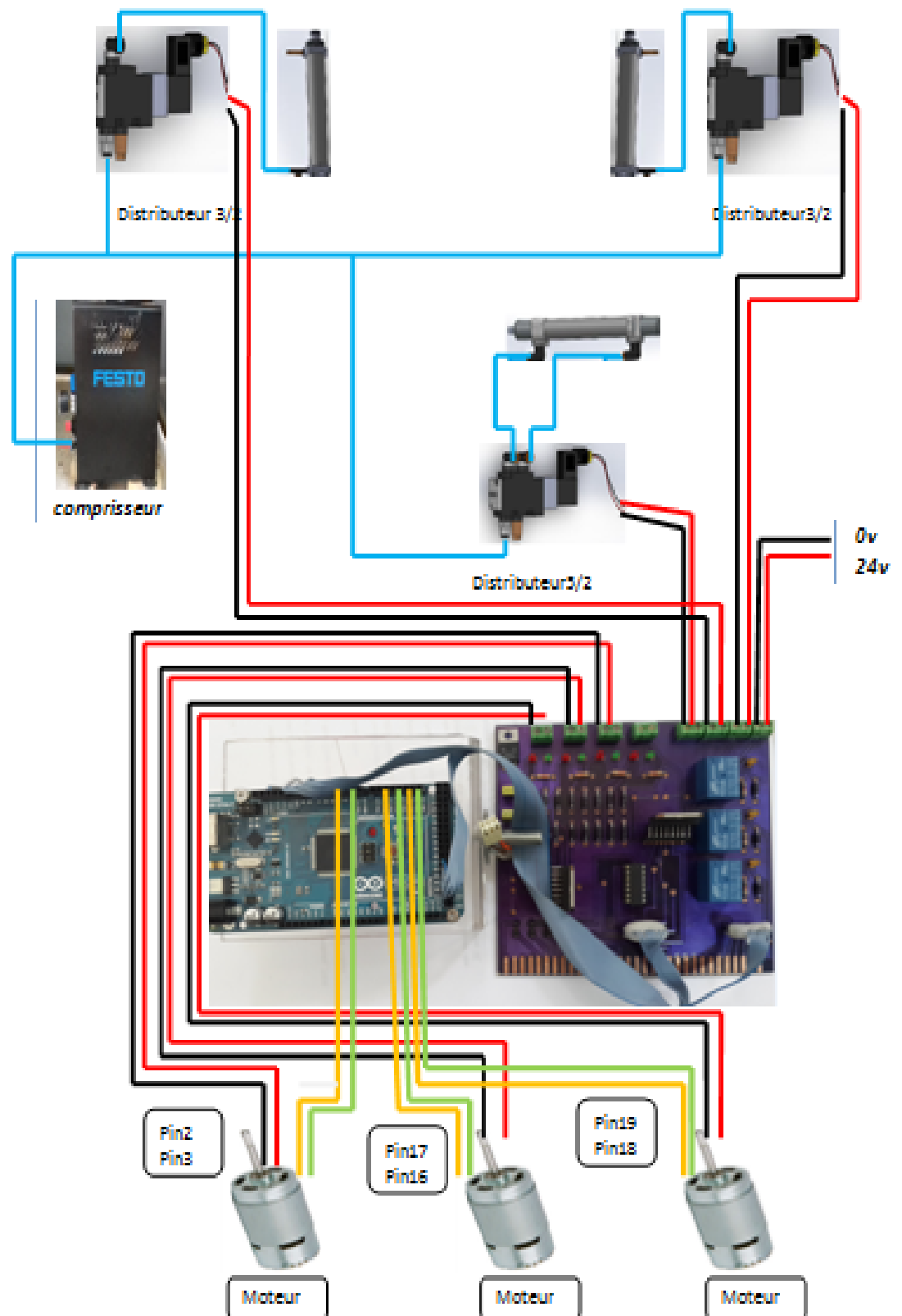


Figure 4.17. Schéma synoptique de la chaîne de conditionnement.

c. schéma réel de la chaine

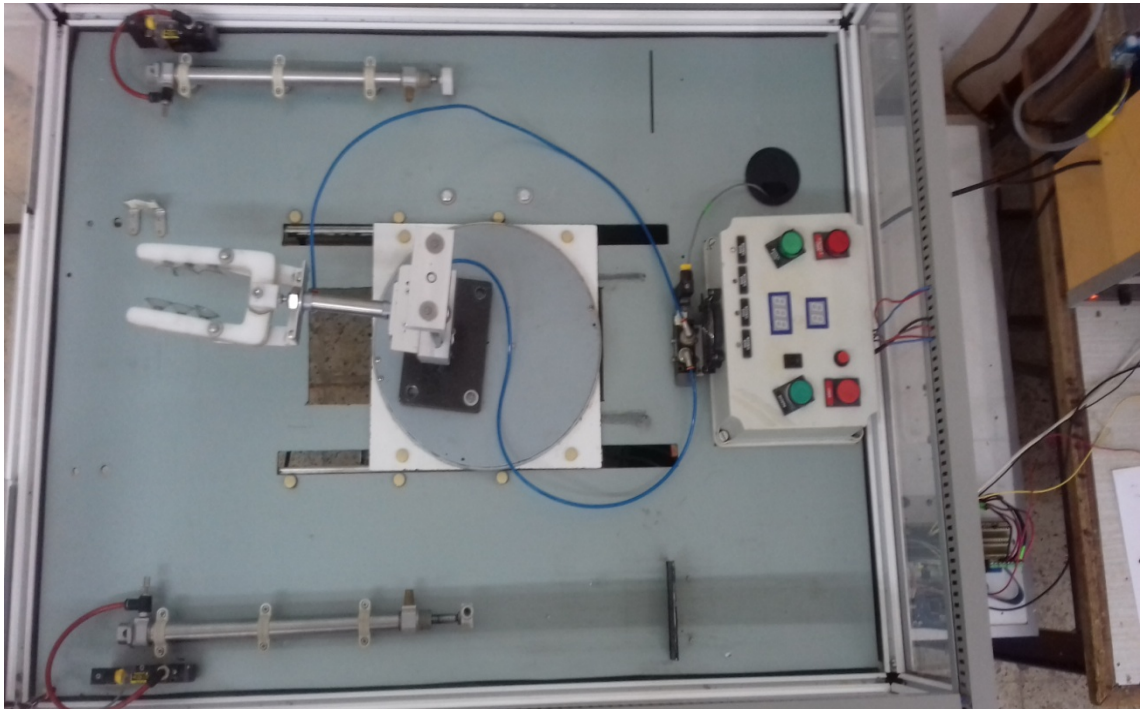


Figure 4.18. Schéma réel de la chaine.

4.5. Conclusion

La réalisation de la chaine de conditionnement a été réalisée partie par partie, dans le but d'obtenir un bon fonctionnement. Nous nous occupés de toutes les parties avec le même degré d'importance pour nous assurer du bon fonctionnement. Les tests effectués ont donné des bons résultats.

Conclusion générale

A partir de ce projet nous avons enrichi notre connaissance dans le domaine automatique et nous avons appris plusieurs principes dans le domaine mécanique.

Malgré les difficultés qu'on a eu dans les deux filières (électronique/mécanique) nous avons acquis de l'expérience dans le domaine de l'application et appris des nouveaux logiciels que nous ne connaissions pas avant qui nous a permis de réaliser notre idée qui consiste à appliquer l'asservissement des moteurs pour avoir une précision sûre de la chaîne de conditionnement.

Dans le côté électronique nous avons pris un bon moment dans la fabrication de la carte de puissance et de la programmation. Et dans le côté mécanique, nous avons rencontré quelques difficultés dans l'installation de quelques-unes des pièces.

Nous avons fait des bancs d'essais pour avoir des bons résultats, finalement notre but est de réussir.