

**République Algérienne Démocratique et Populaire**  
**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**  
**Université Mouloud MAMMERI, Tizi-Ouzou**



**Faculté de Génie Electrique et d'Informatique**  
**Département d'Automatique**

**Mémoire de Fin d'Etudes**

En vue de l'obtention du diplôme

*D'Ingénieur d'Etat en Automatique*

**Thème**

Commande d'un moteur à courant continu à excitation

Séparée via la carte KPCI 3102

Proposé par : M<sup>r</sup>. HASNAOUI Mohammed

Présenté par :

Lounis

M<sup>r</sup>. TIDAF

Dirigé par : M<sup>r</sup>. HASNAOUI Mohammed

M<sup>r</sup>. ZAKNOUN

Tarik

Soutenu le : 14 / 07 /2010

***Promotion 2010***

Ce travail a été préparé à : La Faculté de Génie Electrique et d'Informatique  
Département d'Automatique

## *Remerciements :*

Au terme de ce travail, nous tenons à remercier notre profonde gratitude à *Dieu* de nous avoir donné la force et la patience pour mener ce travail.

Nos premiers remerciements s'adressent à notre promoteur Mr HASNAOUI pour avoir proposé et dirigé notre travail et pour ses conseils précieux, aussi Mr LEKHLEF et l'ingénieur de laboratoire qui ont vraiment consacré pour nous aider dans le laboratoire.

Nous remercions tous ceux qui ont participé de près ou de loin enseignants ou étudiants à l'élaboration de ce mémoire et qui se sont dévoués pour nous venir en aide.

Nous tenons aussi à remercier monsieur le président et les membres de jury d'avoir acceptés notre travail.

Nous profitant de cette occasion pour remercier nos enseignants que nous avons la chance de rencontrer durant notre cursus, nous n'oublions pas nos amis et nos camarades de notre promotion.

# *Dédicaces*

*Je dédie ce modeste travail à :*

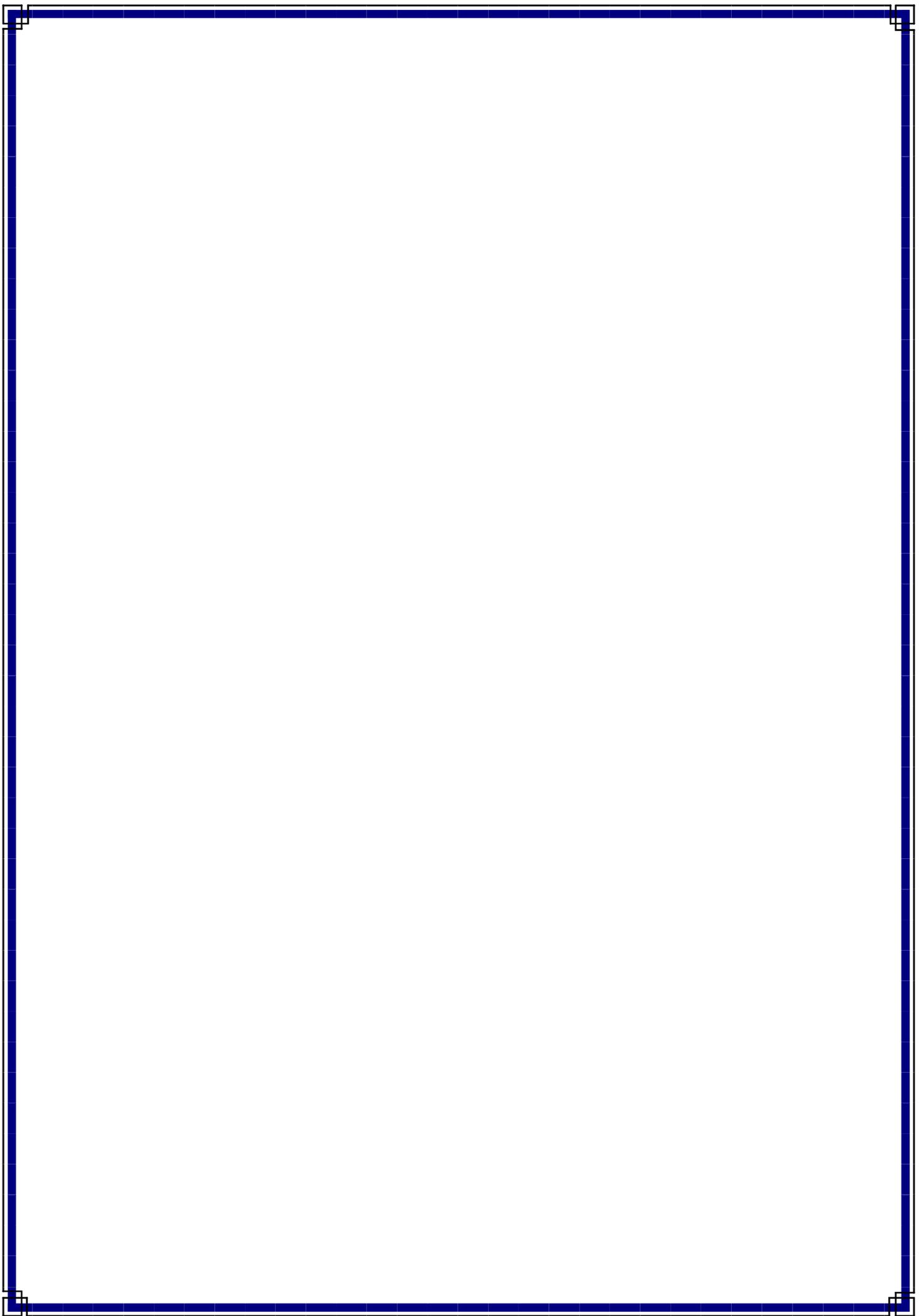
- Mes très chers parents.
- La mémoire de mes grands parents.
- Ma grand'mère Djazira .
- Mes sœurs Djamila,Souhila et Zouhra.
- Mes frères Mourad,houcine,Sofiane et ABD Elkader.
- Les familles TIDAF, TERRAB et TEMMAR.
- Mes amis Belkacem, Boualem, Malek, Hand, Chabha, Lala,Lila ,Lynda, Ghani et Kaci.
- Aux petits Kamilia, Mokrane, kensa, Foufou, Koussaila et Faiz.
- La mémoire de mon cousin Madjid.
- La mémoire de Zouina.
- Mes copains Lyes,Rabah,Idir et Ouremdane .
- Mon binôme Tarik Zak.

*Lounis TIDAF*

*Je dédie ce modeste travail à :*

- Mes très chers parents.
- Mes grand'mère Tassadit et Ouiza.
- Mes sœurs Samira et Thaniana.
- Mon frère Momouh.
- Mes oncles Arezki, Youcef, Hamid et Saïd.
- Mes tantes Fatima, Fazia et Nadia.
- Les familles Zehraoui, Yacin, Boukacem, Temer,Kareb et Kabeche.
- Mes amis Yacine, Bouzid,Moloud, Merzouk, Lynda B, lila ,Noura et à tout les autres.
- Mon binôme LounisTidaf

*Tarik ZAKNOUN*



# Sommaire

|  |  |
|--|--|
| <b>Introduction générale.....</b>  | <b>1</b>   |
| <b><u>Chapitre I :</u></b>   | <b><u>présentation des moteurs à courant continu</u></b> |
| I.1 Présentation générale .....  | 3  |
| I.1.1 Création d'une force électromotrice induite.....   | 3  |
| I.1.2 Une première machine élémentaire.....  | 4  |
| I.2 Construction d'une machine à courant continu.....  | 5  |
| I.2.1 L'inducteur (stator) .....   | 6  |
| A- Les bobines excitatrices .....  | 6  |
| B- Flux magnétique .....   | 6  |
| C- Modèle équivalent de l'inducteur .....  | 7  |
| I.2.2 L'induit (rotor) .....   | 7  |
| A- Différents types d'enroulement.....   | 8  |
| B- Modèle équivalent de l'induit.....  | 8  |
| I.2.3 Le collecteur .....  | 9  |
| I.3 Différents types de machines à courant continue.....   | 9  |
| I.3.1 moteur à excitation série.....   | 9  |
| I.3.2 moteur à excitation compound.....  | 10   |
| I.3.3 moteur à excitation shunt.....   | 10   |
| I.3.4 moteur à excitation séparée .....  | 11   |
| I.4 La force électromotrice .....  | 11   |
| I.5 Bilan des puissances .....   | 12   |
| I.6 Procédés de variation de la vitesse pour les moteurs à courant continu .....                   | 13   |
| I.6.1 Régulation à l'aide d'une résistance placé dans le circuit d'induit .....                    | 13   |
| I.6.2 Action sur le flux .....   | 14   |
| I.6.3 Actions sur la tension d'induit .....  | 14   |
| I.7 Caractéristique de base de moteur à courant continu .....                                      | 15   |
| I.7.1 Point de fonctionnement .....  | 15   |
| I.7.2 Relation entre la vitesse de rotation et la tension d'alimentation .....                     | 15   |
| I.8 Etude d'étage de puissance utilisé pour le moteur à courant continu à excitation séparée ..... | 15   |
| I.8.1 rôle du pont de diodes .....   | 16   |
| I.8.2 rôle du pont mixte asymétrique .....   | 16   |
| I.9 Conclusion .....   | 17   |

|  |    |
|--|----|
| II.1 Introduction.....   | 18 |
| II.2 Étude générale sur une chaîne d'acquisition.....  | 18 |
| II.2.1 Rôle et l'emplacement de la chaîne .....  | 18 |
| II.3 Les éléments de la chaîne d'acquisition .....   | 19 |
| II.3.1 Les Capteurs .....  | 19 |
| II.3.1.1 Les différents types de capteurs.....   | 20 |
| A- Les capteurs TOR .....  | 20 |
| B- Les capteurs numériques .....   | 20 |
| B.1- Les capteurs incrémentaux.....  | 20 |
| C- Les capteurs analogiques.....   | 21 |
| D- Capteur de vitesse .....  | 21 |
| D.1- Caractéristique de la génératrice tachymétrique .....                                     | 22 |
| II.3.2 Le conditionnement du signal .....  | 22 |
| II.3.3 Le multiplexage .....   | 23 |
| II.3.4 L'échantillonnage .....   | 23 |
| II.3.5 convertisseur analogique numérique (CAN) et convertisseur numérique analogique (CNA) .. | 24 |
| II.3.5.1 CAN.....  | 25 |
| II.3.5.2 CNA .....   | 25 |
| II.4 Acquisition de données sur ordinateur .....   | 25 |
| II.4.1 Les carte d'acquisition .....   | 26 |
| II.4.2 Cartes en format du bus PCI .....   | 26 |
| II.4.3 Les entrées analogiques (analogue input) .....  | 28 |
| II.4.4 Les sorties analogiques .....   | 29 |
| II.4.5 Entrées-sorties numériques (digital Input/output) .....                                 | 29 |
| II.4.6 Entrées-sorties de comptage et base de temps .....                                      | 30 |
| II.4.7 Options du logiciel .....   | 31 |
| II.5 Conclusion .....  | 31 |

|   |    |
|---|----|
| III.1 Introduction .....                  | 32 |
| III.2 Les instruments virtuels .....      | 32 |
| III.2.1 face avant ( <i>panel</i> ) ..... | 33 |

|   |    |
|---|----|
| III.2.2 Le diagramme .....  | 33 |
| III.3 Présentation générale de l'environnement .....                        | 34 |
| III.3.1 Lancement de l'environnement LabVIEW .....                          | 34 |
| III.3.2 Palettes .....  | 35 |
| A- Palette d'outils .....   | 35 |
| B- Palette de commande .....  | 36 |
| C- Palette de fonctionnement .....  | 37 |
| III.4 Les base de la programmation LabVIEW.....                             | 38 |
| III.4.1 Edition et exécution d'un programme VI.....                         | 38 |
| A- Edition de la « Face-avant » du programme .....                          | 39 |
| B- Edition du programme dans la fenêtre « Diagramme » .....                 | 40 |
| C- Exécution du programme .....   | 41 |
| III.5 Boucles .....   | 42 |
| III.5.1 La boucle While (tant que) .....                                    | 42 |
| III.5.2 La boucle For .....   | 42 |
| III.5.3 Réalisation des boucles .....                                       | 43 |
| III.6 Les graphes .....   | 44 |
| III.7 Application : acquisition de données et automatisation .....          | 44 |
| III.7.1 DAQmx (Assistant keithley DAQ; VI Express).....                     | 44 |
| A- Quelques exemples de Vis utilisés pour l'acquisition et la commande..... | 45 |
| III.7.2 Exemple d'application.....  | 46 |
| A- Acquisition et commande d'un système thermique.....                      | 46 |
| B- Régulation de la température TOR .....                                   | 51 |
| III.8 Conclusion .....  | 53 |

**Chapitre IV :** **Acquisition, identification et commande**

|   |    |
|---|----|
| IV.1 INTRODUCTION.....  | 54 |
| <b>PARTIE I :</b>   |    |
| IV.2 rappel sur la commande et l'identification numériques..... | 54 |
| IV.2.1 commande numérique .....                                 | 54 |
| IV.2.2 Formes échantillonnés des correcteurs classiques .....   | 56 |
| A- Action Intégrale I.....                                      | 56 |
| B- Dérivateur .....   | 57 |

|   |    |
|---|----|
| C- Correcteur PID .....   | 57 |
| IV.3 Généralité sur l'identification .....                                    | 58 |
| IV.3.1 Types de modèle .....  | 59 |
| A- Modèle de connaissance .....   | 59 |
| B- Modèle de comportement .....   | 59 |
| C- Modèle intermédiaire .....   | 59 |
| IV.4 Etapes de l'identification .....   | 59 |
| IV.5 choix de la structure des modèles .....                                  | 60 |
| IV.6 Choix de la méthode d'identification .....                               | 60 |
| IV.6.1 Méthodes graphiques .....  | 61 |
| IV.6.2 Méthodes numériques .....  | 61 |
| A- Méthode des moindres carrées .....   | 61 |
| B- Application des moindres carrées à l'identification des systèmes .....     | 61 |
| C- Méthode des moindres carrées récursives .....                              | 62 |
| IV.6.3 System Identification Toolbox™.....                                    | 63 |
| <b>PARTIE II Partie pratique</b>  |    |
| IV.7 Présentation du matériel utilisé .....                                   | 65 |
| IV.8 Etude statique du système .....  | 67 |
| IV.9 Etude dynamique .....  | 69 |
| IV.9.1 programmation avec logiciel LabVIEW .....                              | 69 |
| IV.10 Identification des paramètres (K, $\tau_1$ , $\tau_2$ ) du système..... | 71 |
| IV.10.1 Méthode des moindres carrées .....                                    | 71 |
| IV.10.2 Méthode « System Identification Toolbox™ » .....                      | 75 |
| A- Comparaison des deux méthodes.....   | 77 |
| B- Estimation d'un modèle plus proche du réel pour toutes les entrées .....   | 78 |
| C- Vérification du modèle sur simulink .....                                  | 79 |
| IV.11 PARTIE REGULATION .....   | 80 |
| IV.11.1 Action proportionnelle .....  | 80 |
| IV.11.4 Action proportionnelle intégrale .....                                | 83 |
| IV.12 Application du correcteur P et PI pour la régulation réelle .....       | 87 |
| IV.12 .1 Application du correcteur P pour la régulation réelle .....          | 87 |
| IV.12.2 Application du correcteur PI pour la régulation réelle .....          | 88 |
| A- Calcule en z du correcteur PI .....  | 88 |
| B- Application de l'action PI sous LabVIEW .....                              | 89 |
| IV.13 Conclusion .....  | 92 |

## *Introduction générale*

---

En automatique, on parle d'un système temps réel lorsque ce dernier contrôle (ou pilote) un procédé physique à une vitesse adaptée à l'évolution du procédé contrôlé.

Les systèmes temps réel se différencient des autres systèmes par la prise en compte de contraintes temporelles dont le respect est aussi important que l'exactitude du résultat, autrement dit le système ne doit pas simplement délivrer des résultats exacts, il doit les délivrer dans des délais imposés. Les systèmes temps réel sont aujourd'hui présents dans de nombreux secteurs d'activités : dans l'industrie de production, dans l'aéronautique au travers des systèmes de pilotage embarqués (avions, satellites) etc...

De nos jours, les moteurs à courant continu continuent à être utilisés dans l'industrie grâce à leurs performances statiques et dynamiques.

En effet, beaucoup d'applications nécessitent un couple de démarrage élevé. Or, le moteur à courant continu, par nature, possède une caractéristique couple/vitesse de pente importante, ce qui permet de vaincre un couple résistant élevé, et d'absorber facilement les à coups de charge la vitesse du moteur s'adapte à sa charge. D'autre part, les concepteurs trouvent dans le moteur à courant continu une solution idéale lorsqu'il recherche la miniaturisation car il présente un rendement élevé, en comparaison aux autres technologies.

Désormais la commande numérique en temps réel est une technique utilisant des données numériques pour représenter des instructions géométriques et techniques à la conduite d'une machine ou d'un procédé. D'une autre manière, elle est conçue pour piloter le fonctionnement d'une machine à partir des instructions d'un programme sans intervention directe de l'opérateur pendant son exécution.

L'objectif de notre travail est la conception et la réalisation d'une commande au temps réel du procédé constitué d'un moteur à courant continu couplé à une génératrice tachymétrique précédé d'un étage de puissance.

En effet, la commande de notre système via ordinateur nécessite tous d'abord l'acquisition des informations nécessaires à la connaissance et le contrôle fiable de notre procédé, cela est rendue possible grâce à la carte d'acquisition ou carte d'interface ayant des entrées/sorties analogiques, numériques et des compteurs/ timer implantés sur PC. Notre carte d'acquisition est une carte a bus de format PCI (Peripheral component InterConnect).

## *Introduction générale*

---

Notre carte peut être pilotée par plusieurs logiciels tels que le MATLAB, C++ et LabVIEW (Laboratory Virtual Instrument Engineering Workbench).

La particularité du logiciel LabVIEW réside dans son langage de programmation graphique, qui utilise des icônes à la place de lignes de texte pour créer des applications. Contrairement aux langages de programmation textuels où les instructions déterminent l'exécution de programme, LabVIEW utilise une programmation par flux de données dans laquelle les données déterminent l'exécution. LabVIEW fournit aussi de nombreux mécanismes permettant de se connecter à des codes ou logiciels externes grâce, entre autres, aux DLL, aux bibliothèques partagées et à ActiveX. De plus, de nombreux outils supplémentaires sont disponibles pour une grande variété d'applications dans plusieurs domaines scientifiques et techniques.

Le système d'identification TOOLBOXES de MATLAB offre la simplicité, rapidité et précision d'identification des paramètres de notre système par rapport aux méthodes textuelles classiques telle que la méthode d'identification par les moindres carrés récursifs. Le logiciel de simulation Simulink, partie aussi intégrante de MATLAB, permettra d'effectuer la linéarisation du modèle et de valider les paramètres du modèle et des régulateurs.

La réalisation pratique de notre commande en temps réel est réalisée à l'aide du logiciel LabVIEW. Nous allons tester les performances de notre système bouclés en appliquant différentes commandes.

Afin de mener à bien notre travail nous répartissons ce mémoire en quatre chapitres :

- dans le premier on verra des généralités sur le moteur à courant continu.
- Le second sera consacré pour l'étude générale d'une chaîne d'acquisition.
- Le troisième nous permettra de découvrir le logiciel LabVIEW.
- Et dans le quatrième nous donnerons quelques rappels sur la commande, la régulation et les méthodes d'identification numériques que nous terminerons par la présentation des résultats obtenus en pratique.

Et nous terminerons par une conclusion générale.

# Chapitre I

*Présentation du moteur*

*à*

*courant continu*

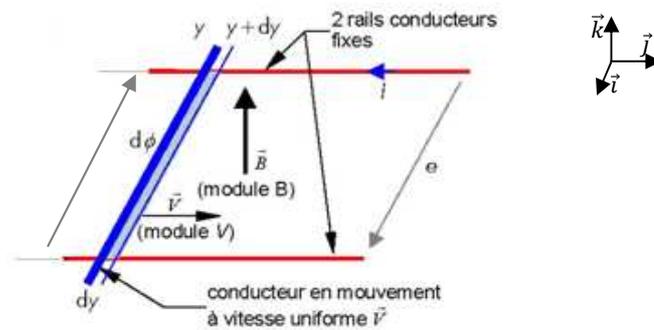
**I.1 Présentation générale :**

Les machines à courant continu (MCC) sont des machines électriques tournantes ; elles sont des convertisseurs électromécaniques permettant la conversion bidirectionnelle entre une installation électrique à courant continu et un dispositif mécanique.

La machine à courant continu (MCC) repose sur le phénomène physique de création d'une force électromotrice (f.e.m.) aux extrémités d'un conducteur en mouvement dans un champ d'induction magnétique (par la loi de Faraday). Réciproquement, la circulation d'un courant dans ce conducteur le soumet à une force qui tend à le mettre en mouvement (loi de Laplace). [4]

**I.1.1 Création d'une force électromotrice induite :**

Dans un repère, un conducteur de longueur  $L$  est en mouvement à vitesse constante  $V$  dans un champ d'induction ( $B$ ) uniforme et orthogonal au plan de mouvement du conducteur (Figure I.1). [4]



**Figure I.1 : Barre en translation dans le champ d'induction B**

Entre ( $t$ ) et ( $t+dt$ ), la barre parcourt la distance élémentaire  $dy = V \cdot dt$

Le flux ( $\phi$ ) coupé par le conducteur est alors :  $d\phi = B \cdot L \cdot dy$  (I.1)

En appliquant la loi de Faraday, on obtient l'expression de la f.e.m ( $E$ ) induite :

$$E = - \frac{d\phi}{dt} = -B \cdot L \cdot V \quad (I.2)$$

➤ Réversibilité du phénomène

Si le conducteur est maintenant parcouru par un courant constant  $I$ , la loi de Laplace donne

$$d\vec{F} = I \cdot d\vec{x} \wedge \vec{B} \quad (I.3)$$

L'effort élémentaire ( $F$ ) appliqué au conducteur :

$$d\vec{F} = -B \cdot I \cdot dx \cdot \vec{i} \wedge \vec{k} \quad (I.4)$$

Le conducteur est soumis à l'effort :

$$d\vec{F} = -B \cdot I \cdot dx \cdot \vec{j} \quad (I.5)$$

### I.1.2 Une première machine élémentaire :

Pour développer technologiquement le principe précédent, on propose les bases d'une machine industrielle. Le conducteur est enroulé sur un rotor cylindrique en fer de rayon  $R$  et de longueur  $L$ , pour former une spire mise en rotation autour de l'axe à la pulsation  $\Omega_r$  (Figure I.2). Ce circuit électrique constitue l'induit de la machine. La spire comporte un conducteur aller sous le pôle sud qui génère une f.e.m.  $+e'$  et un retour sous le pôle Nord qui fournit  $-e'$  car la vitesse est opposée à celle sous l'autre pôle. L'association en série ajoute les deux f.e.m.

Pour bénéficier de l'induction magnétique, le rotor est situé entre deux pôles qui épousent au mieux la forme du stator pour assurer un entrefer constant et minimal. Là, le champ d'induction  $\vec{B}$  est créé dans l'entrefer par les enroulements d'inducteur. Il est normal et de module constant en tout point de l'entrefer et par continuité à la surface du rotor aussi.

La Figure I.3 retrouve le principe précédent car chaque point du conducteur a une vitesse de déplacement tangentielle au rotor et par conséquent orthogonale au champ d'induction [4]

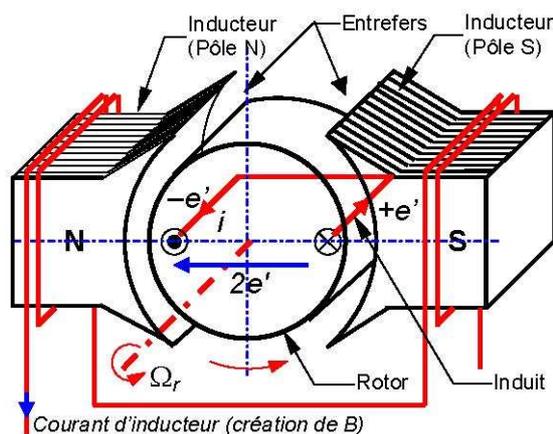


Figure I.2 : Structure industrielle.

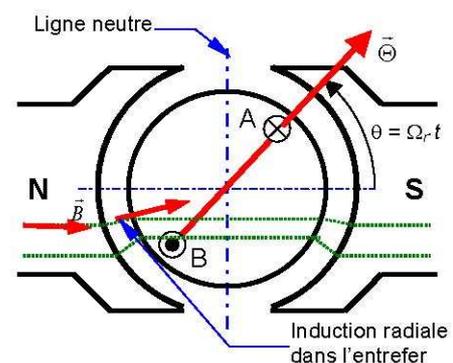


Figure I.3 : Vue de face.

Sous l'effet de la rotation, la spire voit l'induction sous la surface  $S$  :

$$S(t) = 2 \cdot R \cdot L \cdot \sin(\Omega_r t) \quad (I.6)$$

Par sa variation, le flux qui en découle,  $\Phi = B \cdot S(t)$  permet de créer la f.e.m.  $E(t)$  par spire :

$$E(t) = \frac{d\Phi}{dt} = B \cdot \frac{dS}{dt} = 2 \cdot B \cdot R \cdot L \cdot \Omega_r \cdot \cos\Omega_r t = \Phi \cdot \Omega_r \cdot \cos\Omega_r t \quad (1.7)$$

Ce résultat montre que la f.e.m. induite est proportionnelle au flux  $\Phi$  sous un pôle et à la vitesse de rotation  $\Omega_r$ . Sa représentation temporelle est précisée à la Figure I.4.

Le passage de la spire sous le flux maximal en  $(\theta = \pi/2 [k\pi])$  correspond à une f.e.m. nulle. C'est à cet endroit, appelé **ligne neutre**, que la f.e.m. est prélevée pour être transmise à la partie fixe.

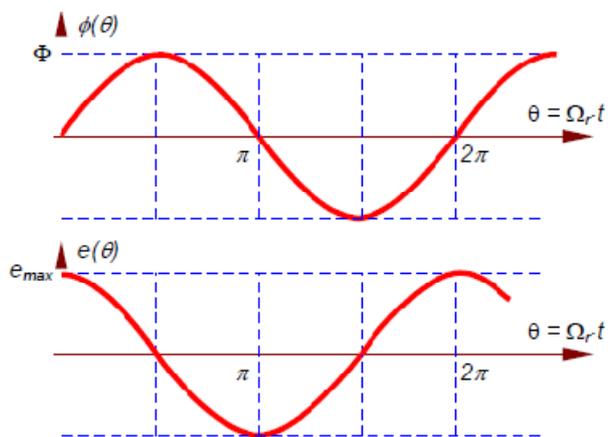


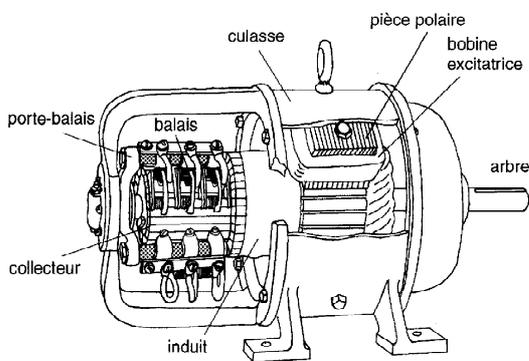
Figure I.4 : Flux et FEM

**I.2 Construction d’une machine à courant continu :**

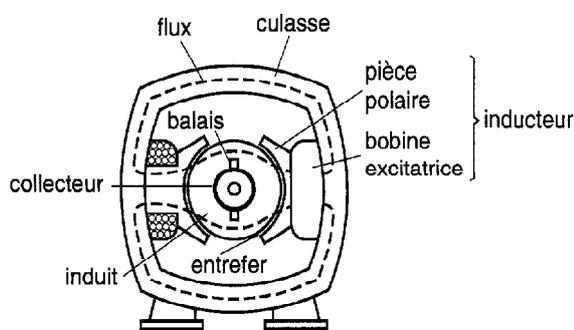
Une machine à courant continu comprend trois parties principales :

- ✓ une partie fixe appelée stator ou inducteur.
- ✓ une partie mobile appelée rotor ou induit.
- ✓ une liaison rotor-éléments extérieurs à la machine appelée collecteur.

Voire les figures suivantes, représente les parties constituants un moteur.



Vue en coupe d’un MCC



Vue d’ensemble d’un MCC

Figures I.5 : Constituant d’un moteur à courant continu

### I.2.1 L'inducteur (stator) :

L'inducteur (parfois appelé –champ) produit le flux magnétique dans la machine. Il est constitué d'un électro-aimant qui engendre la force magnétomotrice (FMM) nécessaire à la production du flux dans les machines bipolaires (à deux pôles), deux bobines excitatrices sont portées par deux pièces polaires montées à l'intérieur d'une culasse. Elle est généralement en fonte, tandis que les pièces polaires sont formées de tôles d'acier doux [1].

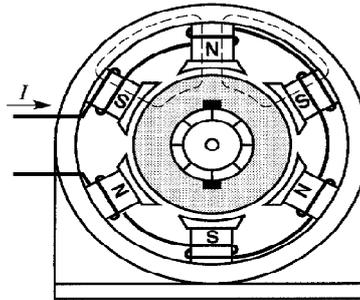


Figure I.6 : Polarités magnétiques d'une MCC à 6 pôles.

#### A- Les bobine excitatrices :

Ils sont alimentés en courant continu, et le courant qui les traverse porte le nom de courant d'excitation ( $I_e$ ). Elles sont composées de plusieurs centaines de spires et portent un courant relativement faible. Les bobines sont bien isolées des pièces polaires afin de réduire les risques de court-circuit à la terre. Dans certaines MCC, les bobines et pièces polaires sont remplacées par des aimants permanents [1].

#### B- Flux magnétique :

Le champ magnétique inducteur et son flux à travers les spires de l'induit peuvent être produits de deux façons différentes [1] :

- a. Par un bobinage alimenté d'une source de tension  $U_e$  et parcouru par un courant  $I_e$  dit courant d'excitation. Dans ce cas le flux peut-être modifié. Il est proportionnel au courant d'excitation  $I_e$ .

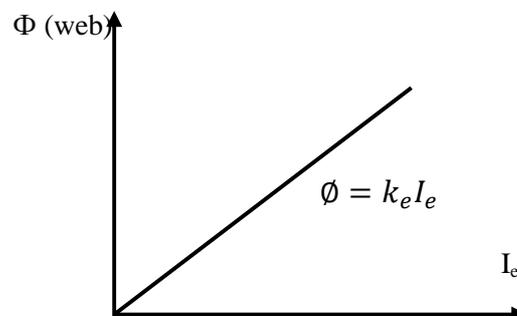


Figure I.7 : Flux et courant d'excitation.

b. Par des aimants permanents. Dans ce cas, le flux est constant et ne peut être modifié, c'est un matériau dur, dont son champ rémanent et son induction coercitive sont grands cela lui donne des propriétés particulières comme d'exercer une force d'attraction sur un matériau ferromagnétique.

### C- Modèle équivalent de l'inducteur :

Valable uniquement dans le cas où l'inducteur est bobiné Le bobinage inducteur alimenté sous tension continue ne présente que sa résistance de bobinage [1] :

$$U_e = I_e \cdot R_e \quad (I.9)$$

. $R_e$  : résistance de l'enroulement inducteur ( $\Omega$ ).

. $U_e$ : Tension d'alimentation de l'inducteur (V).

. $I_e$  : courant dans l'inducteur (A).

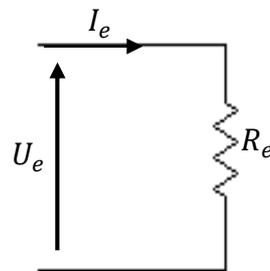


Figure I.8 Modèle équivalent de l'inducteur.

### I.2.2 L'induit (rotor):

L'induit est composé d'un ensemble de bobines identiques réparties uniformément autour d'un noyau cylindrique. Il est monté sur un arbre et tourne entre les pôles de l'inducteur. Le noyau est formé d'un assemblage de tôles (en fer doux) isolées les unes des autres par isolant comme le mica et porte des encoches destinées à recevoir les conducteurs des bobines [1].

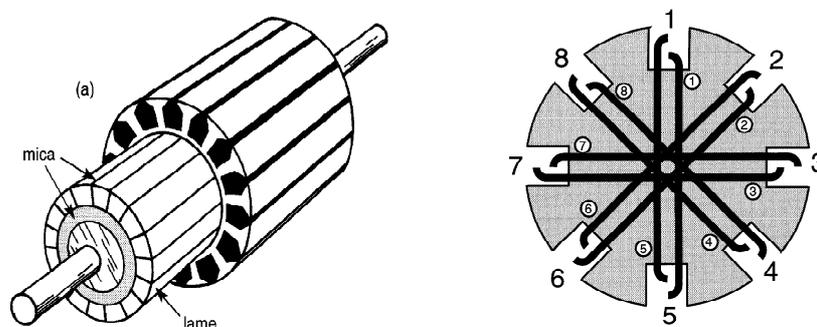


Figure I.9: L'induit et son enrôlement

**A-Différents types d'enroulement :**

Il existe deux manières de réaliser la mise en série des spires :

**a-Enroulement imbriqué ou parallèle :** cet enroulement est caractérisé par le fait qu'il y a toujours autant de voies d'enroulement que de pôles :

.a : nombre de paire de voies d'enroulement, c'est-à-dire le nombre de bobines en parallèles [1].

.p : nombre de paire de pôles.

Avec  $p=a$

Machine bipolaire : 2 pôles ( $p=1$ ), 2 voies d'enroulement ( $a=1$ ).

**b-Enroulement ondulé ou en série :** cet enroulement est caractérisé par le fait qu'il y a deux voies d'enroulement quelque soit le nombre de pôle.

Machine quadripolaire : 4 pôles ( $p=2$ ) ; 2 voies d'enroulement ( $a=1$ ).

**B- Modèle équivalent de l'induit :**

L'induit va présente une f.e.m.  $E$  et sa résistance de bobinage  $R$  [1].

$$U_a = E + R_a I_a \quad (\text{I.10})$$

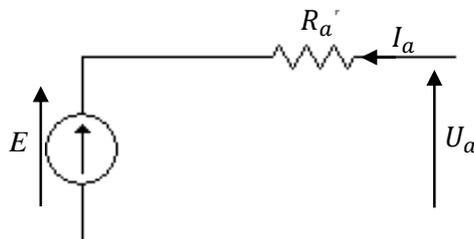
Avec :

$E$  : f.e.m. du à la rotation dans le champ inducteur (V).

$R_a$  : Résistance des enroulements ( $\Omega$ ).

$U_a$  : Tension d'alimentation de l'induit (V).

$I_a$  : Courant dans l'induit (A).



**Figure I.10 : Modèle équivalent de l'induit**

### I.2.3 Le collecteur :

Est un ensemble cylindrique de lames de cuivre isolées les unes des autres par des feuilles de mica. Le collecteur est monté sur l'arbre de la machine, mais isolé de celui-ci. Les conducteurs de l'induit sont reliés aux lames du collecteur [1].

Dans la machine bipolaire deux balais (B+ et B-) fixes et diamétralement opposés appuient sur le collecteur et assurent le contact électrique entre l'induit et le circuit externe

### I.3 Différents types de machines à courant continu :

Un moteur à courant continu comporte deux parties distinctes :

- ✓ L'une qui ne sert qu'à magnétiser (inducteur).
- ✓ L'autre qui transforme l'énergie électrique en énergie mécanique (induit).

Plusieurs possibilités de connexions électriques existent, en effet Les moteurs à courant continu ont les mêmes modes d'excitation que les génératrices et on distingue [1]:

#### I.3.1 Moteur à excitation série :

L'inducteur de ce moteur est en série avec l'induit, le courant d'induit est également le courant d'excitation.

Nous supposons que le flux utile sous un pôle est proportionnel au courant d'excitation série.

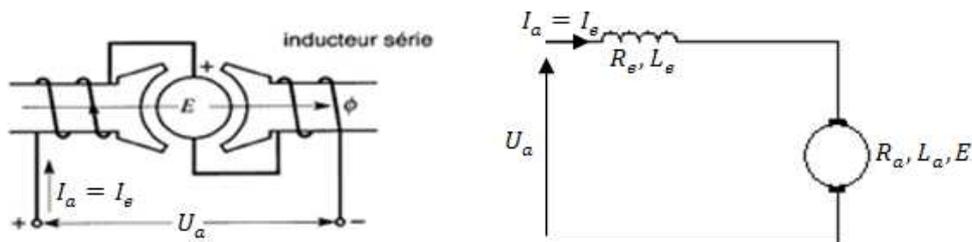


Figure I.11: Excitation série pour MCC.

➤ Remarque sur le fonctionnement :

- ✓ c'est la charge qui impose le courant d'induit I.
- ✓ si le couple augmente, l'intensité I augmente et la vitesse de rotation du moteur diminue [18].

➤ Caractéristique mécanique du moteur :

Le couple répond à la loi :  $C = H \cdot I_a^2$

Nous obtenons alors comme caractéristique couple-vitesse une hyperbole.

La (figure I.12) montre que cette hyperbole se rapproche des axes lorsque U diminue.

Nous retrouvons la limite maximale du couple, imposée par le courant d'induit maximal, qui obéit à la relation  $C_{max} = H \cdot I_a^2_{max}$  avec  $H$  champs magnétique.

En examinant la (figure I.12) nous voyons que la vitesse est devenue très sensible au couple et que le moteur s'emballe à vide [18].

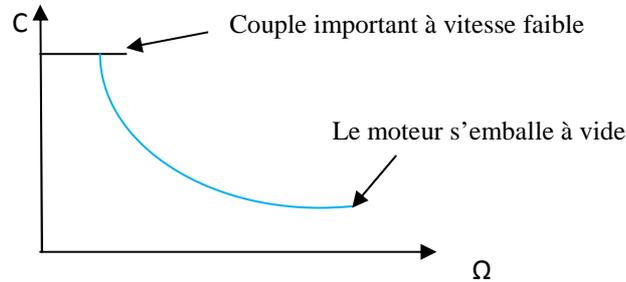


Figure I.12 : Caractéristique couple-vitesse en excitation série.

**I.3.2 Moteur à excitation compound :**

Le moteur compound porte un inducteur série et un inducteur shunt.

la FMM de l'enroulement série agit toujours dans le même sens que celle de l'enroulement shunt.

La FMM de l'enroulement shunt est habituellement plus grande que celle du champ série, même a pleine charge [1].

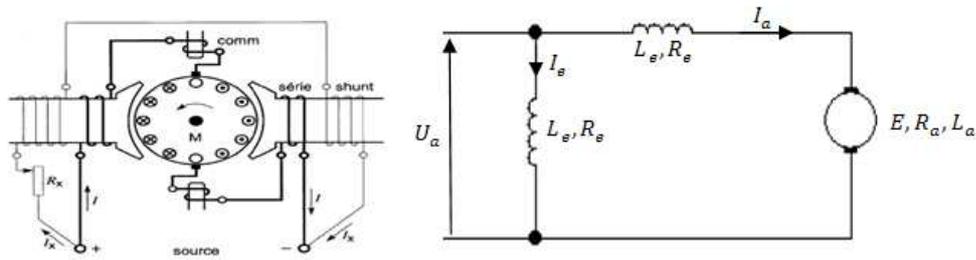


Figure I.13: Moteur à excitation compound.

**I.3.3 Moteur à excitation shunt :**

On peut utiliser une seul alimentation pour l'induit et l'inducteur. Il suffit de placer l'enroulement inducteur en parallèle avec celui d'induit. Dans ce cas la tension se présente comme suit :  $U_a = U_e = U_n$

Avec  $U_n$  : tension nominale [18].

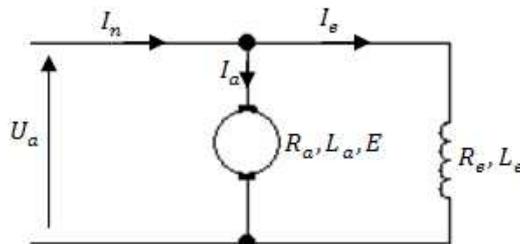


Figure I.14 : Moteur à excitation shunt

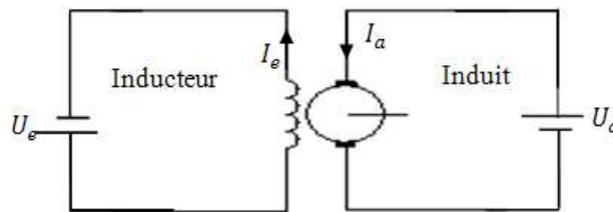
**I.3.4 Moteur à excitation séparée :**

Le moteur à courant continu est l'exemple le plus simple d'application de la loi de Laplace.

Dans la machine à courant continu à excitation séparée, le flux et le couple sont découplés et contrôlables indépendamment (voir la figure I.13).

- ✓ Le courant inducteur est producteur de flux.
- ✓ Le courant induit est producteur de couple.

Le maintien des conducteurs actifs dans le flux est réalisé par une commutation mécanique à l'aide du collecteur et des balais [18].



**Figure I.15: Excitation séparée pour MCC.**

➤ Remarque sur le fonctionnement :

- ✓ C'est la charge couplée au moteur qui impose le courant d'induit.
- ✓ La vitesse de rotation du moteur est proportionnelle à la tension d'alimentation de l'induit.
- ✓ Le réglage de la vitesse est dépendant de la charge.

➤ Caractéristique mécanique du moteur :

- ✓ Au démarrage, la vitesse de rotation est nulle ainsi la  $f_{cem}$ , l'intensité  $I$  imposée par la charge est trop importante pour les enroulements de l'induit dont on intercale un rhéostat de démarrage dans le circuit d'induit [18].

**I.4 La force électromotrice :**

Lorsque l'induit tourne, chaque conducteur se déplaçant dans le champ d'induction est le siège d'une FEM ; grâce au collecteur les FEM créent dans les conducteurs s'ajoutent et leur somme est recueillie entre les balais [18].

Pour une machine bipolaire ; la FEM induite dans une spire ( $E_{ind}$ ) est donnée par :

$$E_{ind} = \frac{2}{\pi} \Phi_p \omega \quad \text{Avec} \quad \omega = \frac{2\pi}{60} \times N$$

D'où on aura :

$$E_{ind} = \frac{4}{60} \Phi_p N \quad (\text{I.11})$$

Soit  $n$  le nombre total de conducteurs actifs dans l'induit. une spire ayant deux conducteurs, le nombre de spires sera donc égal à  $(n/2)$ . Ces spires constitueront  $(2a)$  voies d'enroulements en parallèle, le nombre de spires par voies d'enroulements est donc égal à  $[(n/2)/ (2a)]$ .

La FEM( $E$ ) induite par la machine est égale à celle d'une voie d'enroulement et sera donc donné par :

$$E = \frac{n/2}{2a} E_{ind} = \frac{n/2}{2a} \frac{4}{60} \Phi_p N \qquad E = \frac{1}{a} \frac{n}{60} \Phi_p \cdot N \qquad (I.12)$$

Dans le cas d'une machine multipolaire ( $2p$  pôles), la FEM induite dans chaque spire sera multipliée par  $p$  (en effet le flux par pôle sera plus grand). La tension induite est donc, elle aussi  $p$  fois plus grande. Elle est donnée par :

$$E = \frac{p}{a} \frac{n}{60} \Phi_p \cdot N \quad \Leftrightarrow \quad E = K_m \cdot \Phi_p \cdot \omega \qquad (I.13)$$

où  $K_m = \frac{p}{a} \cdot \frac{n}{2\pi}$

**Avec :**  $K_m$  : Constant propre de la machine

.  $p$  : nombre de paires de pôles.

.  $N$  : vitesse de rotation en tr/min.

.  $a$  : nombre de paires de voies d'enroulement

.  $\Phi_p$  : Le flux par pôle.

.  $n$  : nombre de conducteurs actifs

.  $\omega$  : vitesse de rotation en rad/s

### I.5 Bilan des puissances

Expression des puissances lorsque la tension aux bornes du moteur et l'intensité qui le traverse sont variables au cours du temps.

Puissance absorbée :  $P_a = V_a \cdot I_a \qquad (I.14)$

Pertes joule :  $p_j = R_a \cdot I_a^2 \qquad (I.15)$

Puissance électrique :  $p_e = E \cdot I_a \qquad (I.16)$

$p_e$  : (Puissance électrique transmise à la partie tournante)

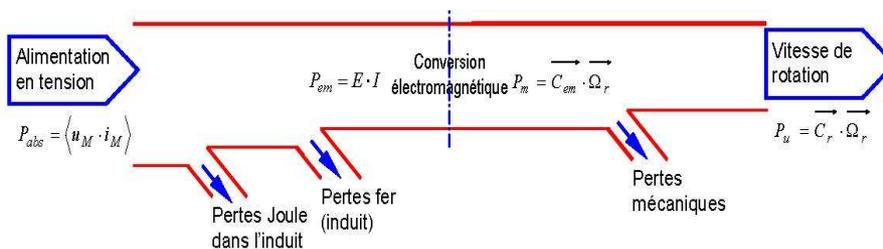
Puissance utile ( $P_u$ ): C'est la puissance mécanique fournie par le moteur pour entraîner la charge.

Elle est donc nulle en fonctionnement à vide.

Pertes constantes  $P_c$  ces pertes sont la somme des pertes mécaniques et magnétiques.

Elles sont constantes à une vitesse donnée et peuvent se déterminer à vide.

Le Rendement: 
$$h = \frac{p_u}{p_a} \qquad (I.17)$$



**Figure I.14: Arbre des puissances à l'induit du moteur à courant continu.**

### I.6 Procédés de variation de la vitesse pour les moteurs à courant continu :

Le moteur à courant continu possède par rapport aux moteurs à courant alternatif, l'avantage d'être très souple car on dispose de plusieurs procédés pour faire varier la vitesse [18].

De l'équation d'équilibre de la tension de circuit d'induit on a :  $V_a = R_a \cdot I_a + L_a \cdot \frac{dI_a}{dt} + E$  (I.18)

L'équation de la FEM :  $E = K_m \cdot \Phi \cdot \omega$  (I.19)

L'équation de couple :  $C = K_m \cdot I_a \cdot \Phi$  (I.20)

Dans le régime établi  $\frac{dI_a}{dt} = 0$

L'équation (I.18) nous donne :  $V_a = R_a \cdot I_a + E$  (I.21)

En remplaçant (I.19) dans (I.21) on aura :  $V_a = R_a \cdot I_a + K_m \cdot \omega \cdot \Phi$  (I.22)

Nous aurons :  $\omega = \frac{V_a}{K_m \cdot \Phi} - \frac{R_a \cdot I_a}{K_m \cdot \Phi}$  Equation électromagnétique (I.23)

De l'équation (I.20) on aura :  $I_a = \frac{C}{K_m \cdot \Phi}$  (I.24)

On remplaçant (I.24) dans (I.23) on aura :  $\omega = \frac{V_a}{K_m} - \frac{R_a \cdot C}{(K_m \cdot \Phi)^2}$  équation mécanique (I.25)

La variation de la vitesse s'obtient comme l'indique l'équation (I.23)

Par la variation de l'un des paramètres suivants : tension ( $V_a$ ), flux ( $\Phi$ ) et résistance de l'induit ( $R_a$ ).

#### I.6.1 Régulation à l'aide d'une résistance placée dans le circuit d'induit :

On fixe le flux d'excitation et la tension d'induit à leurs valeurs nominales, on peut varier la vitesse en agissant sur la résistance d'induit avec un rhéostat monté en série avec l'enroulement induit [18].

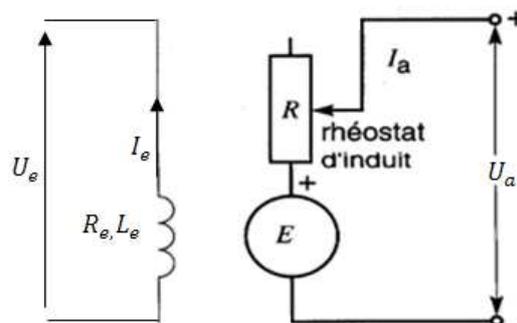


Figure I.15 Réglage de la vitesse à l'aide de Rhéostat en série avec l'induit.

De cette façon, on peut obtenir plusieurs valeurs de vitesses qui dépendent de la valeur de la résistance auxiliaire qui est placée dans le circuit induit, cette méthode est généralement utilisée pour la régulation du courant au démarrage [1].

### I.6.2 Action sur le flux :

Lorsque la tension ( $V_a$ ) est constante, la régulation de la vitesse s'obtient par la variation du flux.

Lorsque le courant inducteur diminue, le flux magnétique du moteur diminue aussi. Cette diminution provoque une augmentation de la vitesse, pouvant aller jusqu'à l'emballement du moteur et sa destruction [18].

Autre risques, la diminution du flux conduit aussi à diminuer le couple du moteur qui est directement proportionnelle au flux selon la relation:

$$C = K_m \cdot \Phi \cdot I_a \quad (I.26)$$

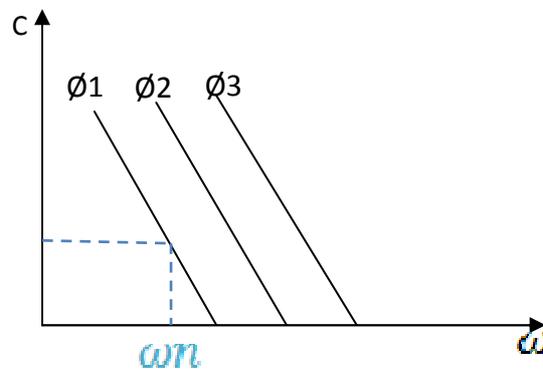


Figure I.16: Action sur le flux avec une tension constante  $\Phi_1 < \Phi_2 < \Phi_3$ .

### I.6.3 Actions sur la tension d'induit :

La relation (I.23) montre que la vitesse est directement proportionnelle à la tension d'alimentation ( $V_a$ ) lorsque le flux constant [18].

La variation de la tension est obtenue par un montage électronique convertisseur de thyristors. Cette variation est obtenue par la variation de l'angle d'amorçage des thyristors  $\alpha$ .

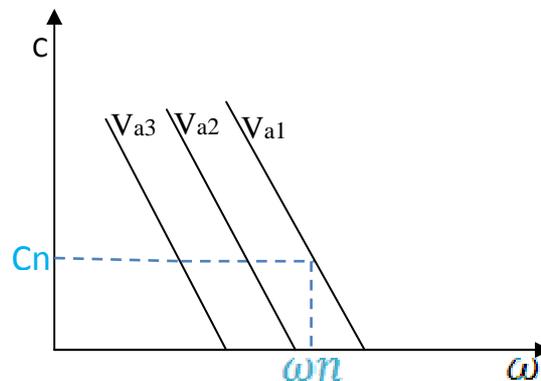


Figure I.17: Action sur la tension avec un flux constant  $V_{a1} > V_{a2} > V_{a3}$ .

## I.7 Caractéristique de base de moteur à courant continu :

### I.7.1 Point de fonctionnement :

Une charge oppose au moteur un couple résistant  $C_r$ . Pour que le moteur puisse entraîner cette charge,[1] il doit fournir un couple utile  $C_u$  tel que :

$C_u = C_r$  C'est le point de fonctionnement de l'ensemble moteur + charge.

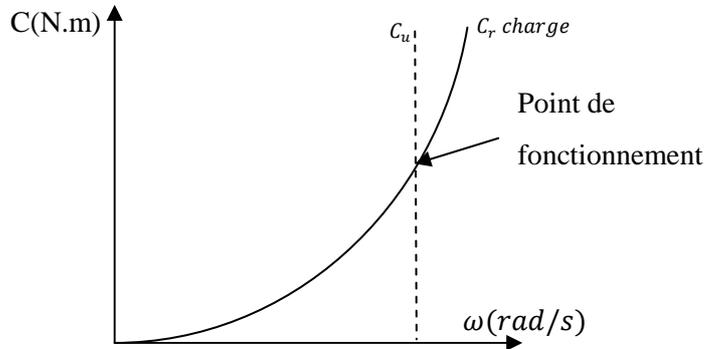


Figure I.18: Point de fonctionnement d'un moteur à courant continu.

### I.7.2 Relation entre la vitesse de rotation et la tension d'alimentation :

La vitesse de rotation est directement proportionnelle à la tension d'alimentation comme est montré sur la figure I.19:

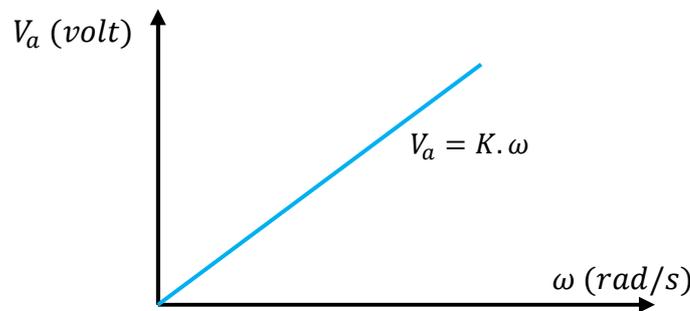


Figure I.19: La tension en fonction de la vitesse  $\Omega$

## I.8 Etude d'étage de puissance utilisé pour le moteur à courant continu à excitation séparée:

Dans notre travail le flux crée par l'inducteur sera constant et on agit sur la variation de la tension de l'induit.

L'étage de puissance permettant d'alimenter un MCC à partir d'un réseau électrique alternatif est constitué d'un redresseur à pont de diode suivi d'une capacité en parallèle pour alimenter l'inducteur en tension constante continue et un pont mixte asymétrique suivi par une self pour la variation de la tension de l'induit.

Le schéma synoptique suivant indique ce branchement :

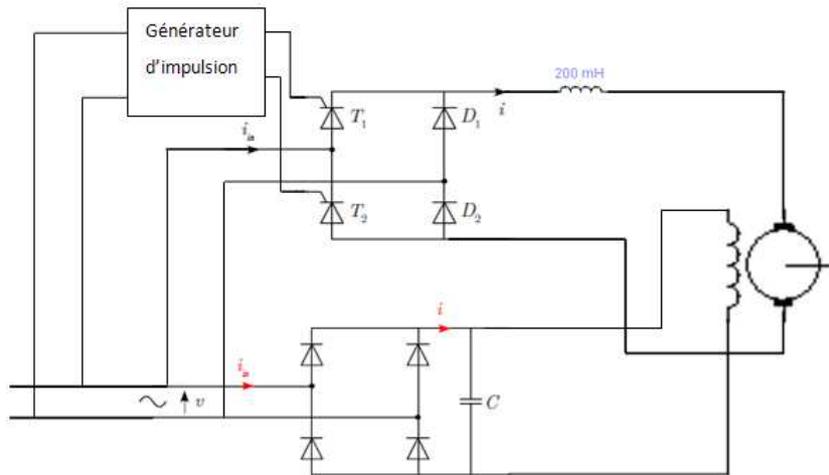


Figure20 : alimentation et commande en vitesse du MCC

### I.8.1 Rôle du pont de diodes :

Pour alimenter l'inducteur du moteur, on a besoin d'une *tension d'excitation* la plus constante possible. On est alors amené à placer un condensateur en parallèle du pont.

Le condensateur tend à s'opposer aux variations de la tension de sortie. Plus fort est le condensateur et plus lisse est la tension de sortie comme il est montré dans cette figure :

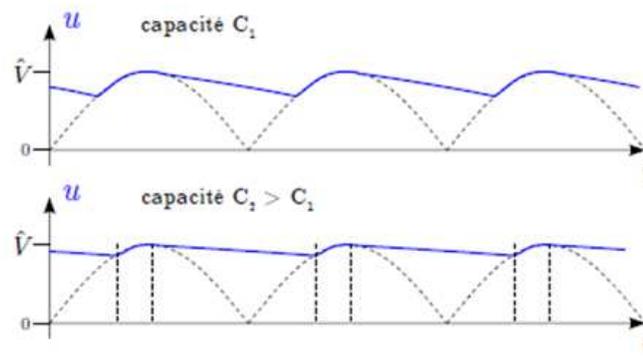


Figure21 : Redressement double alternance à diode.

### I.8.2 Rôle du pont mixte asymétrique :

La variation d'une vitesse d'un moteur à courant continu est réalisable grâce à ce pont, et cette variation est due au changement de la tension moyenne à la sortie de ce pont et cela est effectué à l'aide des thyristors qui sont commandables à la fermeture en agissant sur les angles de retard à l'amorçage.

Un changement sur l'angle de retard à l'amorçage entraîne avec lui une variation de la tension moyenne à sa sortie comme on peut le constater par les calculs suivants:

Si l'angle d'amorçage  $\alpha = 0$ , la valeur moyenne de la tension d'alimentation de l'induit est égale à celle d'un pont redresseur double alternance c'est à dire  $2 V_{\max} / \pi = 2 \times 311 / \pi = \mathbf{198 \text{ V}}$

Dans le cas n°1 : Angle d'amorçage  $\alpha = \pi / 2 \implies V_{\text{moy}} = (1 + \cos \alpha) \times V_{\max} / \pi$   
 $= (1 + 0) \times 311 / \pi = \mathbf{99 \text{ V}}$

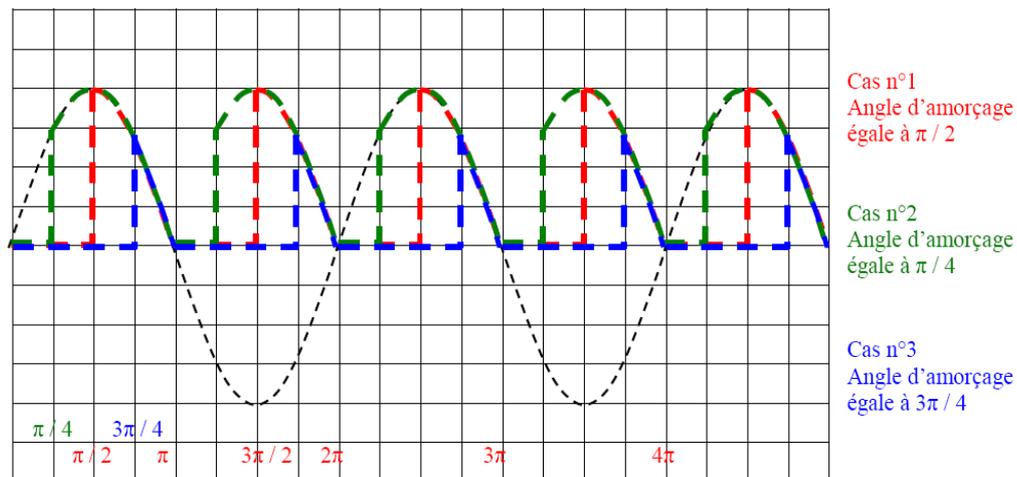
Dans le cas n°2 : Angle d'amorçage  $\alpha = \pi / 4 \implies V_{\text{moy}} = (1 + \cos \alpha) \times V_{\max} / \pi$   
 $= (1 + 0,7) \times 311 / \pi = \mathbf{168 \text{ V}}$

Dans le cas n°3 : Angle d'amorçage  $\alpha = 3\pi / 4 \implies V_{\text{moy}} = (1 + \cos \alpha) \times V_{\max} / \pi$   
 $= (1 - 0,7) \times 311 / \pi = \mathbf{29 \text{ V}}$

On peut visualiser le tracé sur le chronogramme ci dessous l'allure de la tension d'alimentation de l'induit :

- dans le cas 1 où l'angle d'amorçage est  $\alpha = \pi / 2$
- dans le cas 2 où l'angle d'amorçage est  $\alpha = \pi / 4$
- dans le cas 3 où l'angle d'amorçage est  $\alpha = 3\pi / 4$

La tension d'alimentation du réseau est 220 V 50 Hz.



**Figure I.22 : Le chronogramme d'allure de la tension d'alimentation de l'induit.**

### I.9 Conclusion:

Dans cette partie nous avons étudié le moteur à courant continu, dont son fonctionnement, les différents modes d'excitation ainsi que leurs caractéristiques et les ponts de la commande en vitesse et dans le chapitre suivant nous intéresserons au constituant de la chaîne d'acquisition des données pour ce procédé.

# Chapitre II

*Etude générale de la  
chaîne  
d'acquisition*

## II.1 Introduction:

Pour agir efficacement sur un processus physique ou biologique, naturel ou industriel, il importe, au préalable, de bien fournir au décideur (homme ou machine) les informations permettant d'orienter son action et de valider ses décisions, d'où l'utilisation d'un système de mesure qui est constitué d'un micro ordinateur interfacé avec le système sur lequel on travaille.

Etablir un dialogue entre un micro ordinateur et un système extérieur au PC est possible par l'intermédiaire de cartes d'interface [2].

Ces **cartes dites d'Entrées/Sorties** sont directement implantées à l'intérieur du PC et connectées sur le bus PC via des Slots. Elles sont généralement équipées de compteurs (timer), de circuits convertisseurs offrant des entrées et des sorties analogiques et des circuits à base de portes Logiques offrant des entrées et des sorties logiques à l'utilisateur.

Cet ensemble PC et carte d'interface est complété par un logiciel (généralement appelé driver) permettant la gestion de l'ensemble des fonctions de la carte Entrées/Sorties par l'intermédiaire d'un programme, d'un coût modéré et surtout de vitesses élevées de transfert des données vers la mémoire, permettant un traitement en temps réel.

## II.2 Etude générale sur une chaîne d'acquisition :

### II.2.1 Rôle et l'emplacement de la chaîne :

Une chaîne d'acquisition recueille les informations nécessaires à la connaissance et au contrôle d'un procédé ; elle délivre ces informations sous une forme appropriée à leur exploitation.

L'assignation d'une valeur à une grandeur physique ou chimique est une opération de mesure, la grandeur objet de la mesure étant appelée le mesurande [2].

L'analyse pour ses grandeurs, de l'ensemble complexe d'interaction qui constitue un procédé s'obtient par une succession d'opérations instrumentales ayant chacune sa fonction propre.

La chaîne d'acquisition est formée par un ensemble ordonné et coordonné des divers dispositifs assurant ces fonctions (voire la figure II.1)

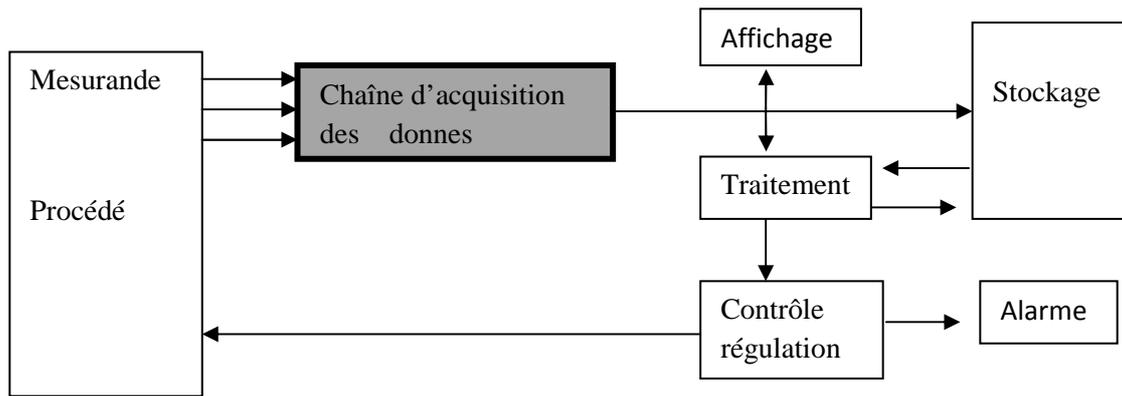


Figure II.1 : Placement de la chaîne d'acquisition dans un ensemble de mesure-contrôle de procédé.

Dans cette structure, la chaîne doit pourvoir assurer au moyen des dispositifs appropriés, les fonctions suivantes :

- extraction de l'information** concernant chacune des grandeurs physiques à connaître et traduction en signal électrique au moyen des capteurs et conditionneurs.
- traitement analogique du signal** destiné en particulier à en éviter la dégradation par le bruit et les parasites: amplification, filtrage [2]
- sélection** parmi l'ensemble des signaux disponibles du seul signal requis, à l'aide de multiplexeur.
- conversion du signal sous forme numérique** adaptée au calculateur chargé de l'exploité, au moyen de l'échantillonneur- bloqueur et du convertisseur analogique numérique.
- la coordination des opérations précédentes** est assurée par un calculateur qui, en outre, peut être chargé de divers traitement sur les signaux numériques qu'il acquis, traitement destiné à faciliter leur exploitation(linéarisation) ou corriger des défauts de l'appareillage.

## II.3 Les éléments de la chaîne d'acquisition :

### II.3.1 Les Capteurs :

Les capteurs sont les premiers éléments d'une chaîne d'acquisition des données. Ils sont sensibles aux phénomènes physiques ou chimiques. Leur rôle consiste à produire un signal électrique, le plus souvent une tension analogique, mais aussi parfois une intensité, une fréquence ou une série de pulsations.

Ces signaux sont liés au milieu dans lequel ils sont placés ou au phénomène qu'ils doivent détecter [2].

### II.3.1.1 Les différents types de capteurs :

La grandeur électrique de sortie d'un capteur peut varier de 3 manières différentes [10]:

- Binaire (Information vraie ou fausse)                   ⇒ Capteur tout ou rien (TOR)
- Progressive (Variation continue)                       ⇒ Capteur analogique
- binaire qui représente la fréquence                   ⇒ Capteur numérique

#### A- Les capteurs TOR :

Dans la structure d'un système automatisé, le capteur fait partie de la chaîne d'acquisition des capteurs qui délivrent un signal binaire (0 ou 1) à la partie commande qui traduit soit la présence d'une pièce ou non, ou bien le seuil de température atteint, vérin sortie...etc. [5]

Pour ça il existe différentes familles technologiques de détecteurs soit mécanique, optique, inductif ou capacitif...etc.

#### B- Les capteurs numériques :

La sortie est une séquence d'états logiques qui, en se suivant, forment un nombre [10]. La sortie peut prendre une infinité de valeurs discrètes. Le signal des capteurs numériques peuvent être du type :

- train d'impulsions, avec un nombre précis d'impulsions ou avec une fréquence précise
- code numérique binaire
- bus de train

Quelques capteurs numériques typiques :

- les capteurs incrémentaux
- les codeurs absolus

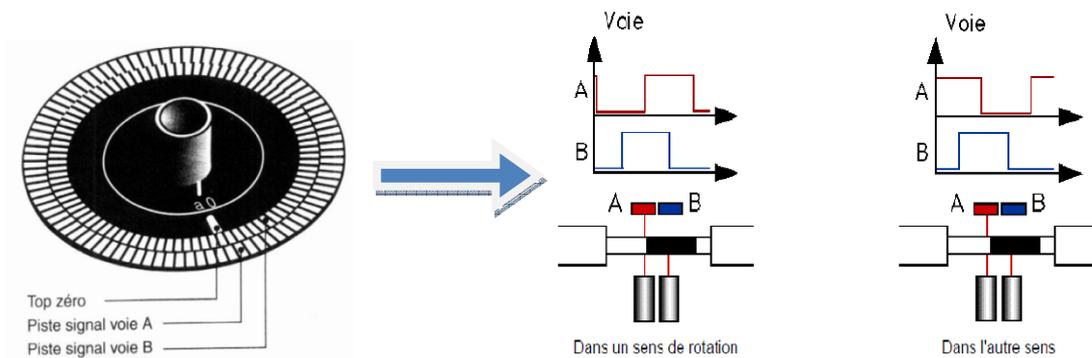
#### B.1- Les capteurs incrémentaux :

Les codeurs incrémentaux sont destinés à des applications de positionnement et de contrôle de déplacement d'un mobile par comptage et décomptage des impulsions qu'ils délivrent.

Le disque d'un codeur incrémental comporte 2 types de pistes :

Les piste extérieures : (voie A et B) sont divisées en « n » intervalles d'angles égaux alternativement opaques et transparents « n » s'appelle la résolution ou nombre de périodes. C'est le nombre d'impulsions qui sera délivré par le codeur pour un tour complet du tambour supportant le disque codé. Derrière les deux pistes extérieures sont installées deux diodes photosensibles décalées délivrant des signaux carrés (A et B) en quadrature [10].

On utilise un deuxième faisceau lumineux qui sera décalé par rapport au premier. Le premier faisceau qui enverra son signal indiquera aussi le sens de rotation du codeur.

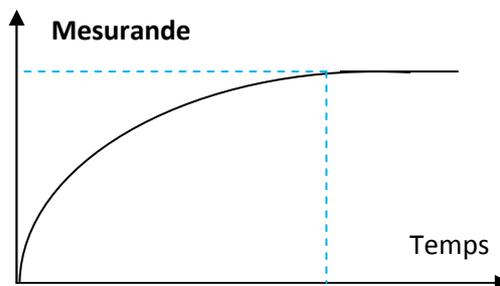


**Figure II.2 : Codeur incrémental.**

### C- Les capteurs analogiques :

La sortie est une grandeur physique dont la valeur est proportionnelle à la grandeur physique mesurée par le capteur. La sortie peut prendre une infinité de valeurs continues [10]. Le signal des capteurs analogiques peuvent être du type :

- sortie tension
- sortie courant
- règle graduée, cadran, jauge (avec une aiguille ou un fluide)



**Figure II.3 : Signal analogique.**

### D- Capteur de vitesse :

Certaines machines automatiques, t'elles que les machines à commande numérique, exigent une mesure précise de la vitesse [11]. La mesure de la vitesse linéaire peut se ramener à celle de la mesure de la vitesse angulaire. Comme c'est le cas de notre projet en utilise une :

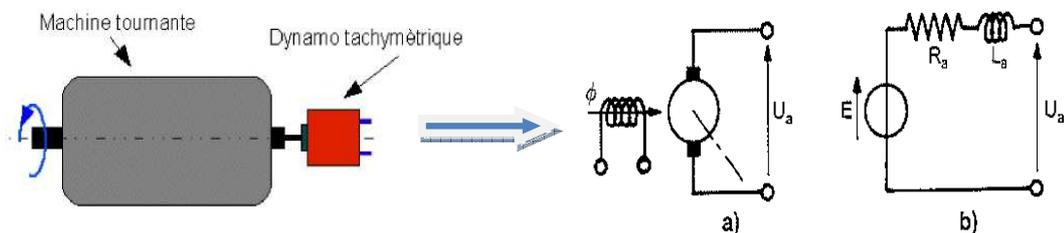
**Tachymétrie (génératrice tachymétrique):**

Elle délivre une tension proportionnelle à sa vitesse de rotation. Son principal domaine d'application se situe dans la régulation de vitesse d'un moteur électrique.

Le système évolue de sorte que l'erreur de vitesse soit nulle [11].

**D.1 - Caractéristique de la génératrice tachymétrique**

- vitesse maximale de rotation (en tours par minute),
- constante de f.e.m. (en volts à 1000 trs/mn ou en v/tr/mn),
- linéarité (en %),
- ondulation crête à crête (en %),
- courant maximal.



**Figure II.4: Dynamo tachymétrique.**

**Principe de fonctionnement :**

Le principe de fonctionnement d'une dynamo tachymétrique repose sur le principe inverse du moteur c'est-à-dire elle génère une tension qui est produite par une transformation d'une énergie mécanique produite par la machine tournante et la tension générée représente la vitesse de cette dernière selon ça caractéristique de faite que 1V généré présente 1000tr/min de la machine tournante.

**II.3.2 Le conditionnement du signal :**

Les signaux issus du capteur ne sont pas toujours directement utilisables par le dispositif de CNA. Par exemple, les cartes de conversion les plus répandues ont un calibre d'entrée de 0 - 10 volts, avec une résistance de quelques centaines de k $\Omega$ . Souvent, il faudra [2] :

- Amplifier les signaux
- Adapter leur impédance
- Décaler leur origine de manière à exploiter au mieux l'amplitude acceptée par l'organe de mesure électrique.
- Transformer des courants ou des fréquences en tensions

- Filtrer pour éliminer des interférences, peut-être aussi la linéarité de manière à obtenir un signal proportionnel à la grandeur étudiée.

**II.3.3 Le multiplexage :**

Le multiplexage est une méthode permettant d'examiner successivement plusieurs voies d'entrée dans un système d'acquisition. Bien entendu, cela allonge d'autant l'intervalle de temps qui s'écoule entre deux mesures sur un même canal.

Le multiplexage peut consister simplement en une commutation sur l'une ou l'autre voie par l'intermédiaire de relais électromagnétiques pilotés par le PC lui-même, mais beaucoup de dispositifs de mesure comportent en entrée un système de multiplexage réalisé entièrement en semi-conducteurs, et bien entendu programmable.

Lorsque l'on désire une acquisition plus rapide, ou une mesure quasi-simultanée sur plusieurs voies, on préfère consacrer un convertisseur analogique numérique à chaque voie [2].

**II.3.4 L'échantillonnage :**

Pour transformer un signal analogique en un signal numérique, il faut le discrétiser.

On va donc prélever régulièrement des échantillons du signal analogique pour le rendre discret et permettre ainsi sa numérisation [12]:

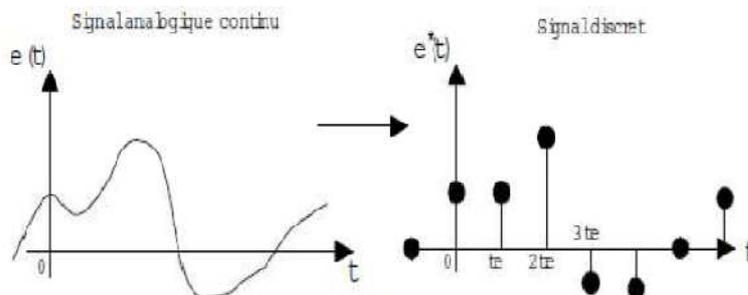


Figure II.5: Allure d'un signal échantillonné.

**II.3.4.1 L'échantillonneur bloqueur :**

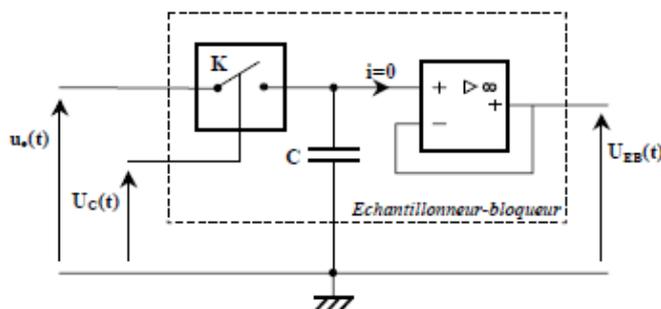


Figure II.6 : Schéma d'un bloqueur.

**II.3.4.1.1 Principe de fonctionnement**

- Phase d'échantillonnage :

L'échantillonnage consiste à prélever périodiquement la valeur de la tension  $u_e(t)$ .

Cette opération non linéaire est réalisée en utilisant un interrupteur électronique K commandé au rythme d'un signal d'horloge  $u_c(t)$  dont la période  $T_e$  est la période d'échantillonnage [13].

- Phase de maintien :

A l'instant  $kT_e$ , le condensateur C se charge avec la tension  $u_e(kT_e)$ .

Entre les instants  $kT_e$  et  $(k+1)T_e$ , le condensateur ne se décharge pas ( $i = 0$ ) et maintient constante la tension ( $u_{EB}(t) = u_e(kT_e)$ )

La présence de l'amplificateur suiveur permet d'avoir  $i = 0$ .

A l'instant  $(k+1)T_e$ , le condensateur C se charge avec la tension  $u_e[(k+1)T_e]$ .

Le signal  $u_{EB}(t)$  est appelé signal échantillonné bloqué.

- Conversion

A l'entrée du convertisseur analogique-numérique, la tension échantillonnée bloquée est maintenue constante pendant la période d'échantillonnage  $T_e$  [13].

La conversion est possible dans la mesure où le temps de conversion  $T_C$ , est inférieur à  $T_e$ .

**II.3.4.1.2 Fréquence d'échantillonnage :**

La valeur maximale de la fréquence d'échantillonnage  $F_e$  ( $F_e = \frac{1}{T_e}$ ) dépend surtout de la rapidité de conversion du CAN.

La valeur minimale de  $F_e$  dépend du signal à échantillonner.

- Théorème de Shannon :

La reconstitution d'un signal analogique  $u(t)$  à partir d'échantillons prélevés à la fréquence  $F_{max}$  n'est possible que si  $F_e$  est au moins deux fois supérieure à la plus grande fréquence  $F_{max}$  contenue dans le signal d'où  $F_e > 2F_{max}$  [13].

**II.3.5 Convertisseur analogique numérique (CAN) et convertisseur numérique analogique (CNA) :****II.3.5.1 CAN:**

Un convertisseur numérique analogique (CNA) reçoit une information numérique (mot de n bits) et la transforme en un niveau analogique [13] (tension ou courant).

Schéma :



### II.4.1 Les cartes d'acquisition :

Tout ce mécanisme d'acquisition, de conditionneur de signaux de multiplexeurs de CAN/CNA...etc., Constitue la carte d'acquisition.

L'ordinateur reçoit l'information de cette carte dont il fait le contrôle de processus, traitement de signal, gestion de liaisons séries ou parallèles, visualisation des résultats...etc.

En fonction des applications auxquelles sont destinées, ces cartes comportent divers accès désignés comme entrées/sorties selon qu'ils reçoivent ou délivrent des signaux électriques.

Suivant la nature et la fonction de ces signaux on distingue :

- les E/S analogiques, les E/S numériques.
- les E/S de comptage et base de temps (timer).

### II.4.2 Cartes en format du bus PCI :

L'acquisition de données par carte a pris une nouvelle dimension par la généralisation de bus PCI (Peripheral Component Interconnect) :

Performances élevées, facilité d'intégration logicielle, indépendance du type de processeur, configuration automatique des cartes périphériques, arbitrage fiable, etc. via le bus PCI, un périphérique peut envoyer et recevoir des données en mémoire à 132M octet /s, sur 32 bits .cette explosion des vitesses de transfert est particulièrement favorable aux cartes d'acquisitions de données. Aussi le bus PCI a la possibilité d'effectuer des transferts de données sans utiliser le processeur du PC ; d'où vient le nom « bus master » c'est-à-dire carte PCI maître de bus et ça grâce au mode DMA (Direct Memory Access) ainsi l'utilisateur se trouve avec des applications très rapide, puisque le processeur n'est utilisé que pour des opérations de traitement et d'affichage.il est important de noter que les cartes d'acquisition PCI ne sont pas toutes capable d'être maître de bus ,elle sont alors « esclaves »(bus slave) et il faut recourir au principe des interruptions pour initier le transfert des données, ce qui oblige à mettre le processeur à contribution .

Notre carte d'acquisition est une carte a bus de format PCI réalisée par Keithley (Peripheral component InterConnect) qui a plusieurs caractères internes qui rendent l'acquisition plus rapide et plus facile qui sont les suivants [15] :

- Bus de donnée PCI
- Taux d'échantillonnage maximal : 225 kHz
- Nombre d'entrées analogiques : 16 référentiels à la masse  $\implies$  8 différentiels.
- Gain des entrées [1 2 4 8]
- Intervalles des tensions : 0 à 10,  $\pm 10$  V
- Résolution des entrées et des sorties : 12 bits
- Nombre de sorties analogiques : 2

➤ Entrées/sorties numériques : 16 référentiels à la masse  $\Rightarrow$  8 différentiels.

Voire sur la figure suivante la manière d'insertion de la carte dans le slot du PC [15] :

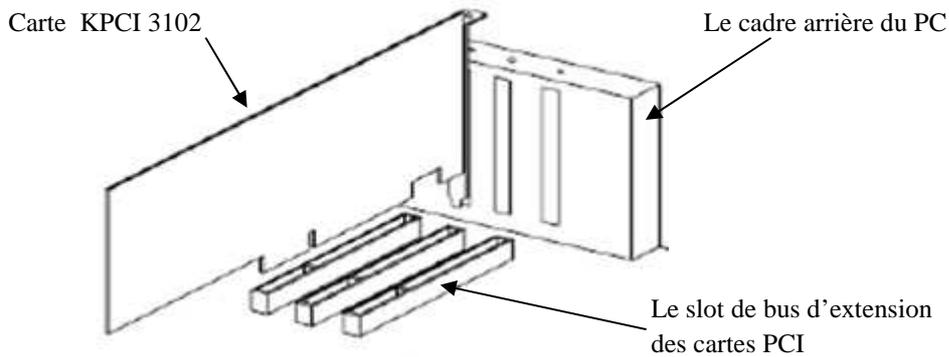


Figure II.10 : Insertion de la carte dans le PC.

Et pour faciliter la manipulation des entrées pour l'acquisition, et les sorties pour la commande on connectant la carte (KPCI 3102) à l'aide de câble (CAB-305) à un le panneau terminal appelle (STA-300 screw terminal panel) pour former la chaîne qui est représentée sur le schéma synoptique suivant :

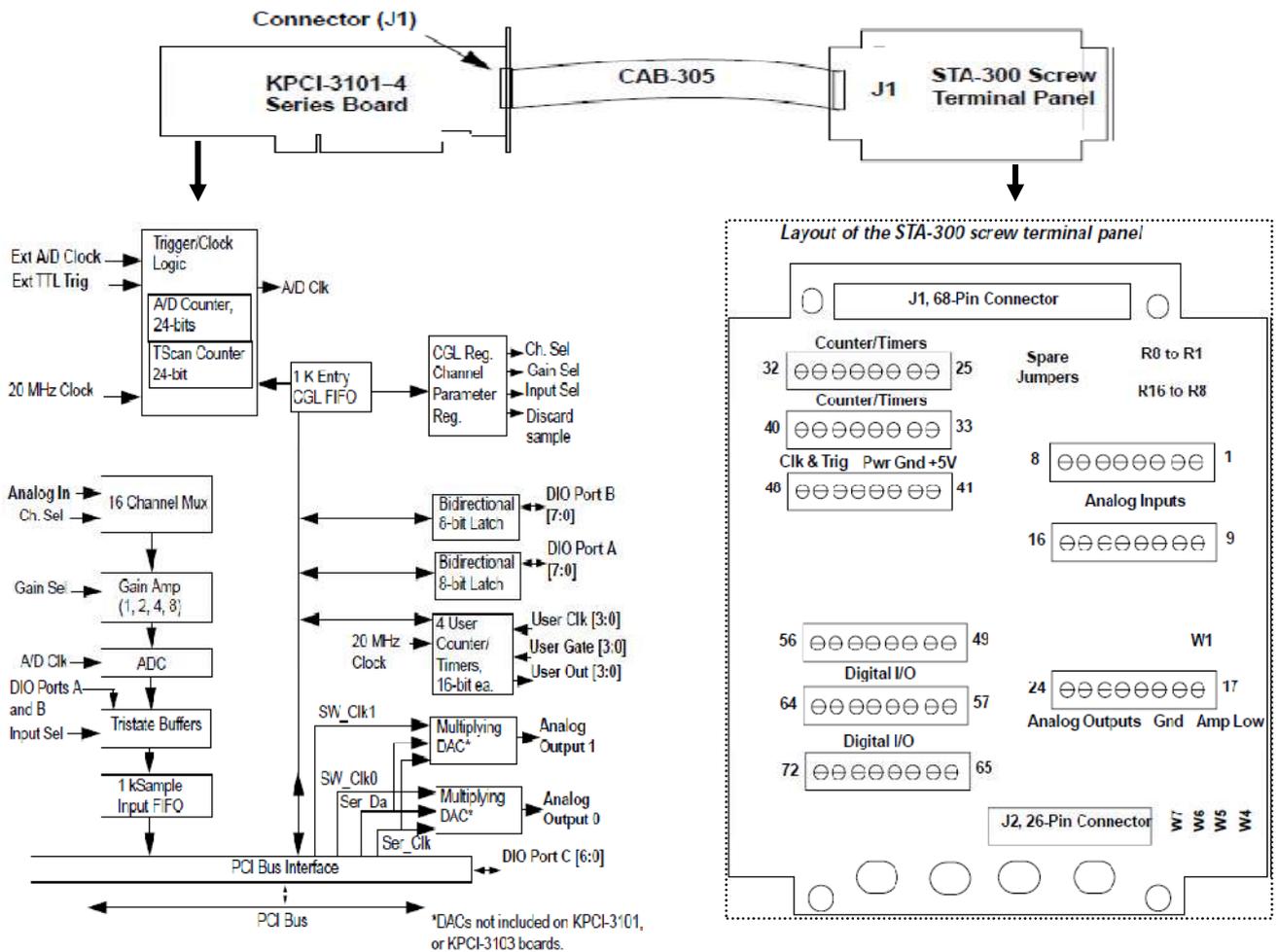


Figure II.11: la carte KPCI et son adaptateur STA 300.

**II.4.3 Les entrées analogiques (analogue input) :**

Ces entrées reçoivent les signaux issus des capteurs ou de leurs conditionneurs associés et les délivrent aux dispositifs de la chaîne d'acquisition organisés pour fournir la donnée numérique correspondante avec la résolution et la précision spécifiée [15].

Le tableau suivant va préciser l'emplacement de chacune des entrées analogiques sur le panneau terminal :

*Analog input screw terminal assignments on the STA-300*

| TB # | J1 Pin # | Description                               | Resistor Use                                |                        |
|------|----------|---|---|------------------------|
|      |          |   | Bias Return Resistor                        | Current Shunt Resistor |
| 1    | 68       | Analog Input 0                            | -   | R9                     |
| 2    | 67       | Analog Input 8/<br>Analog Input 0 Return  | R1  |                        |
| 3    | 34       | Analog Input 1                            | -   | R10                    |
| 4    | 33       | Analog Input 9/<br>Analog Input 1 Return  | R2  |                        |
| 5    | 66       | Analog Input 2                            | -   | R11                    |
| 6    | 65       | Analog Input 10/<br>Analog Input 2 Return | R3  |                        |
| 7    | 32       | Analog Input 3                            | -   | R12                    |
| 8    | 31       | Analog Input 11/<br>Analog Input 3 Return | R4  |                        |
| 9    | 64       | Analog Input 4                            | -   | R13                    |
| 10   | 63       | Analog Input 12/<br>Analog Input 4 Return | R5  |                        |
| 11   | 30       | Analog Input 5                            | -   | R14                    |
| 12   | 29       | Analog Input 13/<br>Analog Input 5 Return | R6  |                        |
| 13   | 62       | Analog Input 6                            | -   | R15                    |
| 14   | 61       | Analog Input 14/<br>Analog Input 6 Return | R7  |                        |
| 15   | 28       | Analog Input 7                            | -   | R16                    |
| 16   | 27       | Analog Input 15/<br>Analog Input 7 Return | R8  |                        |
| 17   | 26       | Amp Low                                   | Jumper W1 Connects Amp Low to Analog Ground |                        |
| 18   | 25       | Analog Ground                             |   |                        |

#### II.4.4 Les sorties analogiques :

Associées aux entrées analogiques, cette carte d'acquisition disposent souvent des sorties analogiques qui délivrent des signaux utiles ou complémentaires à la fonction d'acquisition et qui permettent :

-Soit de générer une forme d'onde (stimulus) appliquée à un dispositif à tester afin d'étudier ses réactions sur les entrées analogiques de la carte [15].

-soit de commander un actionneur en fonction des entrées analogiques. Il peut s'agir par exemple; l'augmentation de la température d'un four ou bien de commander la vitesse d'un moteur. Le tableau suivant va préciser l'emplacement de chacune des sorties analogiques sur le panneau terminal:

#### *Analog output and power screw terminal assignments on the STA-300*

| TB # | J1 Pin # | Description     |
|------|----------|-----------------|
| 19   | 58       | DAC0 Output     |
| 20   | 57       | DAC0 Return     |
| 21   | 60       | DAC0 Reference  |
| 22   | 23       | DAC1 Return     |
| 23   | 24       | DAC1 Output     |
| 24   | 59       | DAC1 Reference  |
| 41   | 1        | +5V Output @ 1A |
| 42   | 35       | Power Ground    |

#### II.4.5 Entrées-sorties numériques (digital Input/output):

Les paramètres à prendre en compte sont le nombre de lignes numériques disponibles, la vitesse d'acquisition et de génération des données numériques, et la quantité de courant disponible sur chaque ligne, en effet les Lignes de l'I/O numériques de la carte KPCI 3102 abordent le support 23 lignes de l'I/O numériques à travers les entrées numériques (DIN) et les sorties numérique (DOUT) du sous-système; Les sous-systèmes (DOUT, DIN) utilisent les mêmes lignes de l'I/O numériques [15].

Ces lignes sont divisées en ports suivants:

- Port A, règle 0 à 7.
- Port B, règle 0 à 7.
- Port C, règle 0 à 6.

On peut résumer tout les E/S numériques dans le tableau suivant qui va préciser l'emplacement de chacune de ces E/S sur le panneau terminal :

| TB # | J1 Pin # | Description                |
|------|----------|----------------------------|
| 50   | 16       | Digital I/O Port A, Line 1 |
| 51   | 49       | Digital I/O Port A, Line 2 |
| 52   | 15       | Digital I/O Port A, Line 3 |
| 53   | 48       | Digital I/O Port A, Line 4 |
| 54   | 14       | Digital I/O Port A, Line 5 |
| 55   | 47       | Digital I/O Port A, Line 6 |
| 56   | 13       | Digital I/O Port A, Line 7 |
| 57   | 46       | Digital I/O Port B, Line 0 |
| 58   | 12       | Digital I/O Port B, Line 1 |
| 59   | 45       | Digital I/O Port B, Line 2 |
| 60   | 11       | Digital I/O Port B, Line 3 |
| 61   | 44       | Digital I/O Port B, Line 4 |
| 62   | 10       | Digital I/O Port B, Line 5 |
| 63   | 43       | Digital I/O Port B, Line 6 |
| 64   | 9        | Digital I/O Port B, Line 7 |

| TB # | J1 Pin # | Description                |
|------|----------|----------------------------|
| 65   | 51       | Digital I/O Port C, Line 0 |
| 66   | 20       | Digital I/O Port C, Line 1 |
| 67   | 53       | Digital I/O Port C, Line 2 |
| 68   | 19       | Digital I/O Port C, Line 3 |
| 69   | 52       | Digital I/O Port C, Line 4 |
| 70   | 18       | Digital I/O Port C, Line 5 |
| 71   | 51       | Digital I/O Port C, Line 6 |
| 72   | 17       | Digital Ground             |

**II.4.6 Entrées-sorties de comptage et base de temps :**

Ces E/S servent à compter des occurrences d'événements binaires, à synchroniser de l'impulsion ou encore à générer des signaux carrés et des trains d'impulsions.

Selon la fonction des signaux on distingue trois types d'accès : la porte, la source et la sortie.

La porte est une entrée numérique utilisée pour déclencher ou inhiber le fonctionnement du compteur, la source est une entrée numérique qui fournit la base de temps du compteur ; quand a la sortie, elle délivre des signaux carrés et des impulsions [15].

Le tableau suivant va préciser l'emplacement de chacune des E/S de comptage de base de temps sur le panneau terminal :

| TB # | J1 Pin # | Description           |
|------|----------|-----------------------|
| 25   | 42       | Digital Ground        |
| 26   | 41       | User Clock Input 0    |
| 27   | 40       | User Counter Output 0 |
| 28   | 39       | External Gate 0       |
| 29   | 8        | Digital Ground        |
| 30   | 7        | User Clock Input 1    |
| 31   | 6        | User Counter Output 1 |
| 32   | 5        | External Gate 1       |
| 33   | 8        | Digital Ground        |
| 34   | 36       | User Clock Input 2    |
| 35   | 37       | User Counter Output 2 |
| 36   | 38       | External Gate 2       |
| 37   | 42       | Digital Ground        |
| 38   | 2        | User Clock Input 3    |
| 39   | 3        | User Counter Output 3 |

| TB #       | J1 Pin # | Description                     |
|------------|----------|---------------------------------|
| 40         | 4        | External Gate 3                 |
| 41         | 1        | +5V Output @ 1A                 |
| 42         | 35       | Power Ground                    |
| 43, 44, 45 | 55       | Digital Ground                  |
| 46         | 56       | External A/D Trigger            |
| 47         | 21       | Digital Ground                  |
| 48         | 22       | External A/D Sample Clock Input |
| 49         | 50       | Digital I/O Port A, Line 0      |

**II.4.7 Options du logiciel :**

Les utilisateurs de cartes Série KPCI-3101.4 ont les deux options du logiciel suivantes.

Dans les deux cas, le logiciel connecte avec un système par le logiciel DriverLINX fourni avec la série:

- L'utilisateur peut utiliser un progiciel d'acquisition complètement intégré tel que Test Point ou Labview.
- L'utilisateur peut écrire et exécuter un programme sur commande dans C / C++ Visuel, Visuel

Basic ou Delphi, utiliser le support de la programmation fourni dans le logiciel DriverLINX.

La Série KPCI-3101.4 a le support du conducteur complètement utilitaire pour usage sous Windows 95/98/NT [15].

**II.5 Conclusion :**

L'acquisition des données avec les cartes PCI et l'ordinateur forment un couple très intéressant c'est bien que l'idée d'utiliser un PC portable a pris sa naissance depuis le début des années 1990, beaucoup de recherches ont été consacrées pour l'évolution de cette technologie tellement qu'elle offre une très grande facilité aux ingénieurs et la recherche scientifique, pour créer leurs propres systèmes d'instrumentation, économiques, évolutifs et très fiables.

# Chapitre III

*Présentation*

*du*

*logiciel LabVIEW*

### III.1 Introduction :

LabVIEW (Laboratory Virtual Instrument Engineering Workbench) est un langage de programmation dédié au contrôle d'instruments et l'analyse de données. Contrairement à la nature séquentielle des langages textuels, LabVIEW est basé sur un environnement de programmation graphique utilisant la notion de flot de données pour ordonnancer les opérations [7]

L'idée de LabVIEW est de remplacer les instruments de mesures et d'analyse d'un laboratoire par un ordinateur muni de cartes spécifiques et d'un logiciel approprié, en effet un seul ordinateur muni d'une carte d'acquisition analogique et de LabVIEW est capable de remplacer un voltmètre, un fréquencemètre ou un oscilloscope. De plus, on pourra acquérir, analyser, traiter et présenter des données et les archiver sur disque automatiquement les mesures effectuées.

### III.2 Les instruments virtuels :

Si l'on réalise qu'un programme de mesure est un instrument de mesure que l'on contrôle à partir d'un ordinateur au lieu d'un ensemble de boutons, cela conduit logiquement à la notion d'instrument virtuel (instrument réel simulé sur ordinateur). un instrument virtuel est un module de programme qui présente une interface sous forme graphique pour l'apparenter à un instrument physique. Dans LabVIEW on manipule les instruments virtuels comme s'il s'agissait d'instruments réels. En effet, une application développée sous LabVIEW est appelée **Instrument Virtuel** (*Virtual Instrument : VI*) [3].

Un programme ou un **VI** se compose principalement de deux parties associés et regroupés sous le même nom « nom\_application.vi » (l'extension **.vi** permet la reconnaissance immédiate par l'environnement LabVIEW). Ainsi nous avons :

**III.2.1 Face avant (*panel*) :** La face avant d'un VI est avant tout une combinaison de commandes et d'indicateurs. Les commandes sont les entrées des VIs, elles fournissent les données au diagramme. Les indicateurs sont les sorties des VIs et affichent les données générées par le diagramme. On peut utiliser plusieurs types de commandes et d'indicateurs tels que les commandes et les indicateurs numériques, à curseur, booléens, chaîne de caractères, les tables et les graphes [3].

Un exemple de face avant d'un VI est présenté ci-dessous:

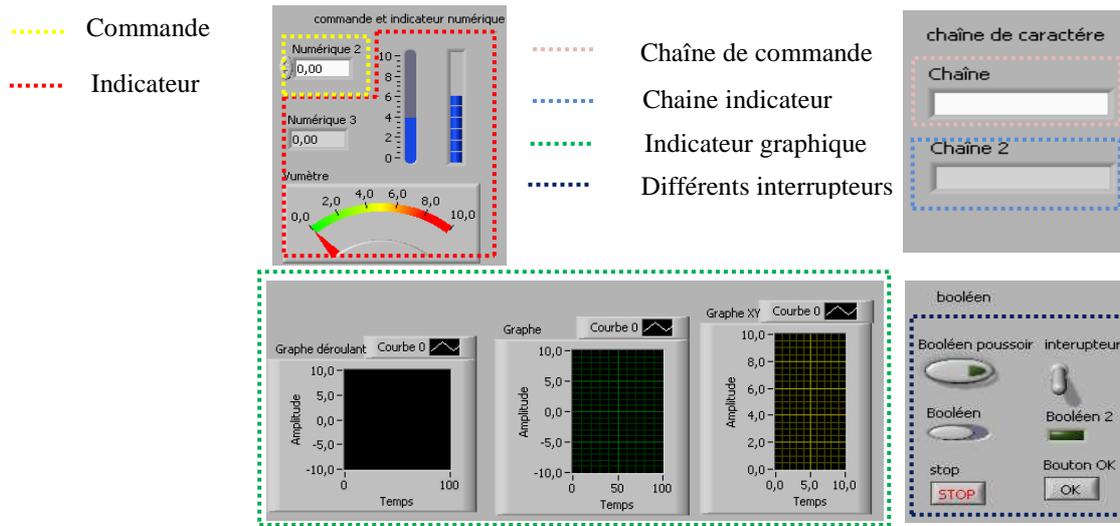


Figure III.1 : Quelques exemples sur la face avant.

On construit la face avant en plaçant des éléments graphiques accessibles dans une palette. Nous disposons des boutons, des indicateurs numériques et des chaînes, des graphes, des LEDs, des listes déroulantes, des menus...

**III.2.2 Le diagramme :** il contient les terminaux qu'ils sont générés par les objets déposés dans la face avant. Ce terminal contient la valeur de l'objet graphique correspondant. Le symbole représente le type de la donnée (par ex. DBL pour double), le sens du flot, (maigre avec une flèche à droite s'il s'agit d'un indicateur, gras avec une flèche à gauche s'il s'agit d'une commande). Le diagramme contient aussi les sous VIs, les fonctions, les constantes, les structures ainsi que les fils qui relient les différents objets pour leur transmettre les données [7].

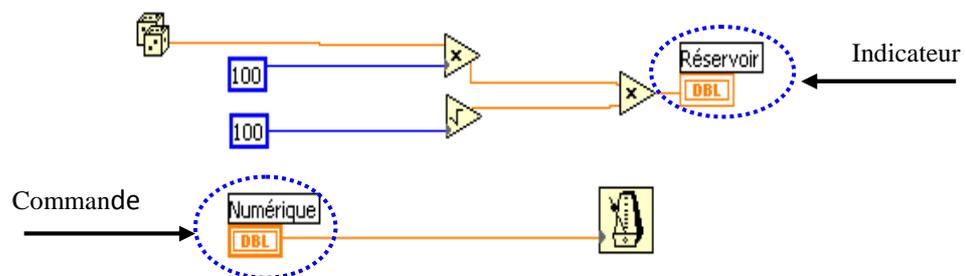


Figure III.2.a : Exemple d'apparence d'un programme sur la face diagramme.

Les fils transfèrent les données dans le diagramme, ils sont ainsi analogues aux variables dans les langages textuels. Chaque fil provient d'une seule source mais peut être réuni à beaucoup de VIs ou de fonctions destinées à les traiter [7].

| Type de la donnée | Scalaire  | Tableau 1D  | Tableau 2D   | Couleur                        |
|-------------------|---|---|--|--------------------------------|
| Numérique         |  |  |  | Orange (réel)<br>Bleu (entier) |
| Booléen           |  |  |  | Vert                           |
| Chaîne            |  |  |  | Rose                           |

Figure III.2.b: Les différents types de fils de transfert de données.

### III.3 Présentation générale de l'environnement :

#### III.3.1 Lancement de l'environnement LabVIEW :

Après avoir effectué l'installation telle, on peut passer au lancement de LabVIEW on double cliquant sur  l'icône

Alors la boîte de dialogue suivante s'apparaît.



Figure III.3 : Boîte de dialogue de lancement de LabVIEW.

Elle permet les actions suivantes:

- ✓ **Nouveau VI** permet d'ouvrir une face avant et un diagramme vierge. Le fait de cliquer sur la flèche de la liste déroulante; permet de créer d'autres types d'objets LabVIEW (des contrôles personnalisés, des variables globales, des VIs polymorphes...).
- ✓ **Ouvrir VI** permet d'ouvrir un VI existant, la flèche de la liste déroulante donne accès aux VIs récemment ouverts. [7]
- ✓ **Solutions DAQ** charge un utilitaire destiné à simplifier le développement d'applications dédiées à l'acquisition de données par cartes d'entrées sorties.
- ✓ **Exemples** ouvre une boîte de dialogue permettant de trouver des exemples classés par thème.
- ✓ **Tutorial LabVIEW** ouvre le guide d'apprentissage.
- ✓ **Quitter** ferme LabVIEW.

Comme nous l'avons vu précédemment, le travail de développement avec LabVIEW s'effectue dans deux fenêtres différentes : interface utilisateur ou «face avant » et programme ou « Diagramme».

Au lancement du logiciel LabVIEW on cliquant sur nouveau VI de la fenêtre de démarrage présentée sur la figureIII.1, une fenêtre vierge active «face avant »s'apparaît et la deuxième fenêtre « diagramme » cachée en arrière plan et non active [3] comme le montre la figureIII.2 suivante :

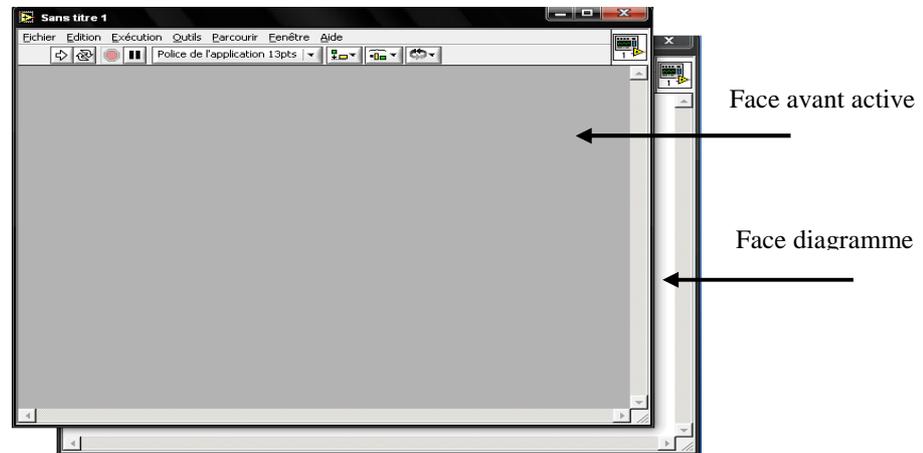


Figure III.4 : Fenêtre après le lancement de LabVIEW avec la commande *Nouveau VI*

### III.3.2 Palettes :

LabVIEW possède trois palettes flottantes respectivement nommées: **Outils**, **Commandes**, et **Fonctions**.

#### A- Palette d'outils

La palette d'outils existe sur le diagramme et sur la face avant. Elle permet de modifier des valeurs, des couleurs, mais aussi de câbler les entrées et les sorties des icônes entre elles, de poser des points d'arrêt, des sondes [3] On y accède par Fenêtre  $\Leftrightarrow$  Afficher la palette d'outils.

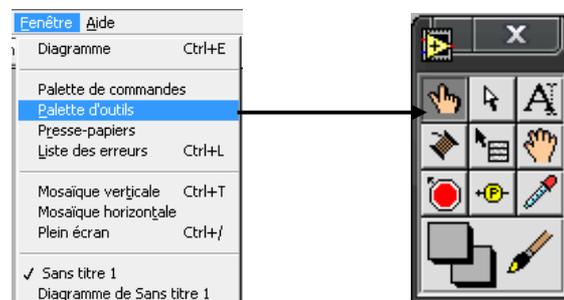
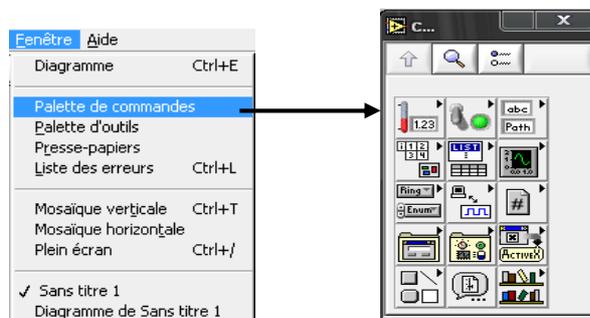


Figure III.5 : Ouverture de la palette d'outils

-  L'outil «doigt » change la valeur d'une commande, sélectionne un texte, Le curseur adopte la forme lorsqu'il est placé dans une zone contenant des caractères.
-  L'outil « flèche » sélectionne, déplace, redimensionne les objets. Il adopte formes suivantes lorsqu'il est sur l'angle d'un objet redimensionnable.
-  L'outil « édition de texte » permet de changer les étiquettes, d'éditer des objets de type caractère et de placer du texte libre dans une fenêtre.
-  L'outil « bobine » sert au câblage des VIs.
-  Accède au menu contextuel par un clic à gauche (utilité douteuse !).
-  Pour se déplacer dans une fenêtre sans les ascenseurs.
-  L'outil « point d'arrêts » définit l'emplacement sur le diagramme où le programme passera en pause pour permettre le débogage.
-  L'outil « sonde » visualise la valeur de connections particulières.
-  L'outil « pipette » mesure la couleur d'un point.
-  Le pinceau colorie un objet.

**B- Palette de commande :**

La palette de commandes est disponible dans la fenêtre de face avant. Elle apparaît fugitivement lors d'un clic droit dans un endroit vierge de la fenêtre, ou par **Fenêtre** ⇨ **Afficher la palette de commandes**. Elle contient tous les éléments graphiques disponibles pour créer l'interface utilisateur. Ceux-ci sont hiérarchisés par type de données ou par grandes familles d'objets [3]. Nous apercevons ci-dessous la palette de commandes.

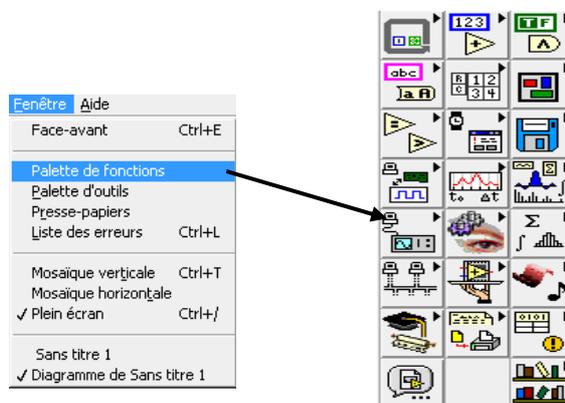


**FigureIII.6: Ouverture de palette de commande.**

-  Objets pour définir des entrées ou des sorties de données de type numérique entier ou réel.
-  Objets pour définir des entrées ou des sorties de données de types booléen.
-  Objets pour définir des entrées ou des sorties de données de types chaîne de caractères ou des chemins d'accès à des fichiers.
-  Objets pour définir des entrées ou des sorties de données de types tableaux ou de type cluster.
-  Objets pour définir des entrées ou des sorties de données homogènes en type, structurées et représentées en liste ou table.
-  Objets pour définir uniquement des sorties de données sous forme de courbes à deux ou trois dimension.
-  Objets pour définir des entrées ou des sorties de données homogènes présentées sous la forme de menu déroulant.
-  Objets pour définir des entrées ou des sorties faisant référence à des données de type « mesures » spécifiques à l'environnement LabVIEW.
-  Objets pour définir des entrées ou des sorties faisant référence unique à un objet tel qu'un fichier.
-  Objets pour définir des entrées ou des sorties destinées à des boîtes de dialogue créées dans la fenêtre « diagramme ».
-  Il est possible de sauvegarder un ensemble de commande personnalisée et d'intégrer l'accès à ces commandes dans un sous –menu identique aux autres commande de base de LabVIEW.

**C- Palette de fonctionnement :**

Cette palette fonctions est uniquement accessible dans la fenêtre « diagramme ». Elle permet d'accéder à l'ensemble des objets représentant les instructions du langage permettant de constituer le programme LabVIEW [3]. L'affichage permanent de cette palette peut se faire par l'accès menu suivant :



**FigureIII.7 : Palette des fonctions disponibles dans la fenêtre « diagramme ».**

Nous pouvons décrire brièvement quelques types de fonctions qu'on peut trouver :



Structures de programmation, constantes numériques, variables locales et global.



Fonction de traitement de données de type tableau.



Fonctions de traitement des données de type cluster.



Ce sous-menu offre toutes les fonctions pour gérer les entrées/sorties de type analogique ou numérique, les compteurs, pour étalonner et configurer les cartes et enfin faire du conditionnement des signaux.



Fonctions de gestion des fichiers (format tableur, texte, ou format binaire).



Fonctions de traitement du signal, de mathématique et fonction de traitement adaptées pour le type waveform.



Fonctions mathématiques avancées (lissage, interpolations, opération sur les matrices, etc.) optimisation, détermination de zéros de fonctions, etc.



Fonctions permettant de contrôler l'application en mode exécution (arrêt, impression de la FA, etc.)



Cette sous-palette intègre des fonctions de haut niveau comme l'appel d'une fonction DLL

### III.4 Les bases de la programmation LabVIEW:

#### III.4.1 Edition et exécution d'un programme VI:

Afin de mettre en œuvre pas à pas l'environnement de développement LabVIEW nous allons choisir un exemple simple de l'instrumentation.

Pour cela l'exemple choisi se propose de simuler l'acquisition d'une valeur de température T en degré Celsius puis faire une conversion en Fahrenheit et en Kelvin selon les équations suivantes

Conversion de °C en K :  $K = °C + 273,15$ .

Conversion de °C en °F :  $°F = ((9 \times °C) / 5) + 32$ .

La réalisation complète de ce programme VI va donc s'effectuer selon les règles du génie logiciel en réalisant successivement les trois étapes suivantes :

- ✓ Edition de l'interface utilisateur dans la fenêtre « Face-avant » pour la mise en place des E/S
- ✓ Edition du programme dans la fenêtre « Diagramme ».
- ✓ Exécution et test du programme.

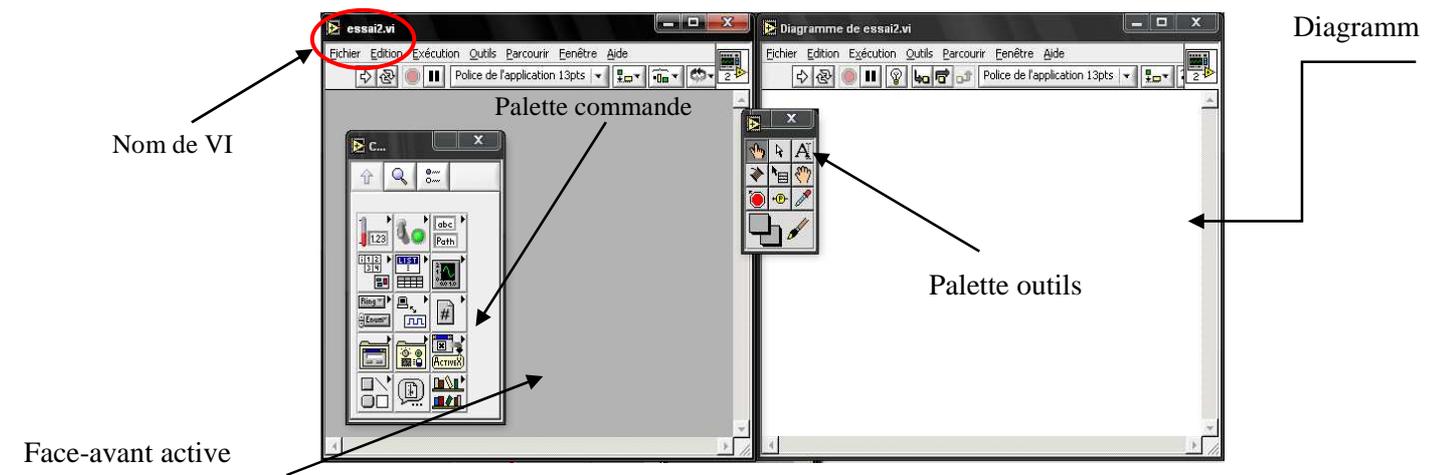
Ainsi pour finaliser complètement le programme développé, il est intéressant de procéder à la création de l'instrument virtuel encapsulé ou sous VI, où ce dernier sera réutilisé dans d'autres programmes.

**A- Edition de la « Face-avant » du programme :**

Nous allons nous placer dans le contexte où les deux fenêtres de développement sont ouvertes, mais avant de commencer l'édition de la face avant [3], nous allons enregistrer le programme sous le nom « essai2.vi » afin de pouvoir sauvegarder régulièrement le travail effectué pour cela on fait appel à la fonction sauvegarder

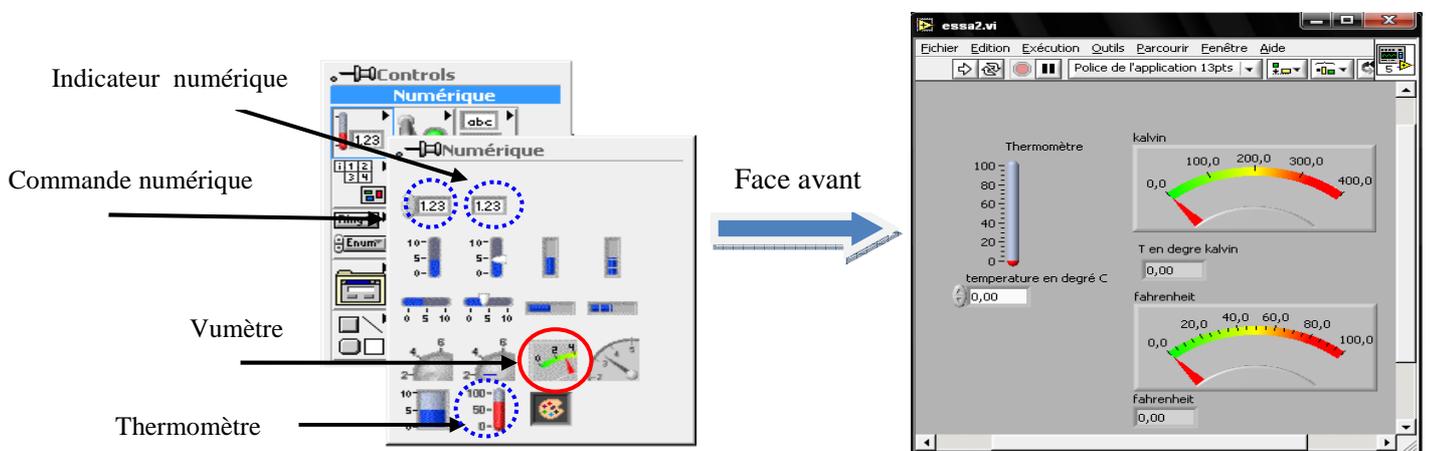


Ainsi nous obtenons l'écran de l'ordinateur qui est présenté sur la figure suivant :



**Figure III.8 : Face avant active et diagramme non active.**

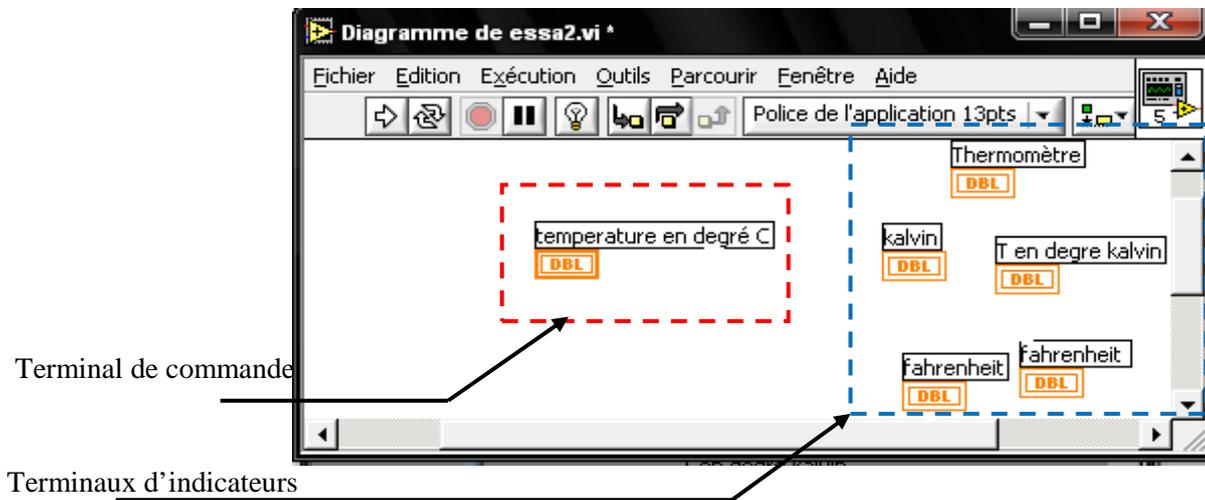
En suivant les spécifications des E/S nécessaires à ce programme, nous avons besoins d'une entrée numérique est de deux sorties numériques, et trois indicateurs de température (vumètre et thermomètre). Ces objets vont être sélectionnés de la sous palette numérique de la palette commande et le choix d'un objet s'effectue par un clic sur l'objet, puis le faire glisser dans la face avant.



**Figure IV.2 : Choix des objets pour réalisé les E/S sur la Face Avant.**

**B- Edition du programme dans la fenêtre « Diagramme » :**

Passant dans la fenêtre « diagramme » qui est non active jusqu'à présent, alors on peut remarquer que tous les objets de type variables (E/S), ainsi créés dans la fenêtre « face-avant » apparaissent dans la fenêtre « diagramme » sous forme de terminaux d'entrées ou de sorties



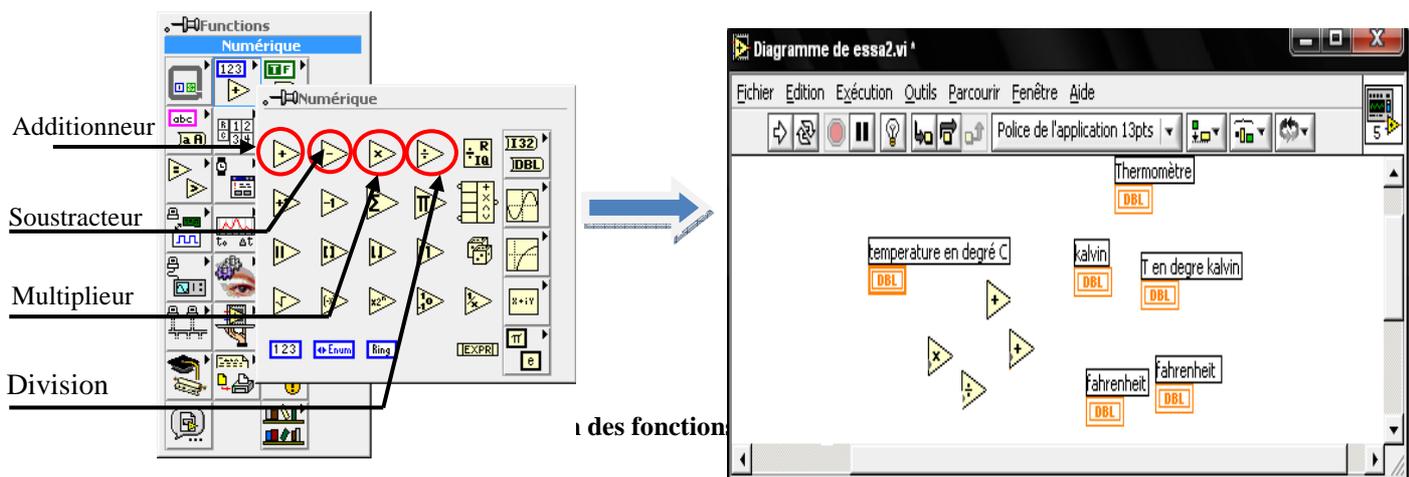
**Figure III.9: Représentation de la fenêtre « diagramme » après la première phase.**

Afin de réaliser notre programme, nous allons maintenant sélectionner les différentes fonctions dont nous avons besoin pour constituer les nœuds de traitement du programme flux de données.

Le programme nécessite des fonctions qui sont accessibles dans la palette de fonction :

- Operations sur les nombres, appelées un additionneur (add), un soustracteur (substarct), multiplieur (multiply)

La figure suivante montre comment sélectionner les fonctions nécessaires :



**Figure III.10: Sélection des fonctions nécessaires au programme.**

Avant de passer à l'exécution de ce programme, il faut commencer par faire le câblage de tous les terminaux sélectionnés, et pour le réaliser, nous utilisons l'outil « bobine » de la palette outil.

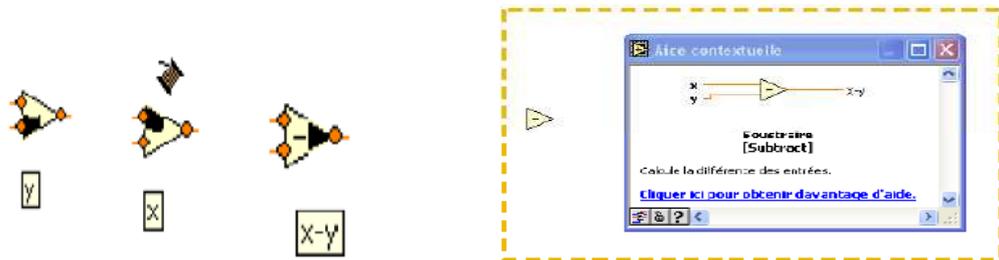


Figure III.11 : Visualisation des terminaux d'un nœud de traitement, et la fenêtre d'aide contextuelle.

Le câblage du programme flux de données étant terminé, nous obtenons la fenêtre « Diagramme » présentée sur la figure IV.5. Notons que le bouton d'exécution doit avoir repris son aspect initial (flèche non brisée).

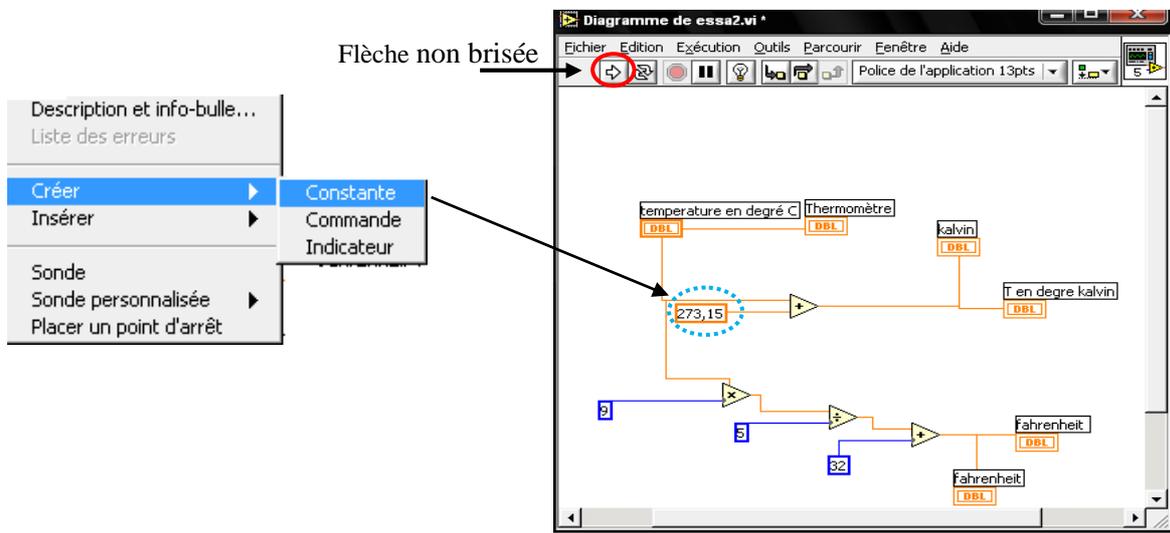


Figure III.12: Fenêtre « Diagramme » obtenue après l'édition complète du programme

**C- Exécution du programme :**

L'exécution du programme doit être faite tout naturellement à partir de la fenêtre « face-avant » à l'aide du bouton d'exécution 

Ou bien par la fonction « RUN » du menu exécution.

Cette exécution produit une valeur qui s'affiche sur les objets indicateurs, et cette valeur correspond aux résultats des équations de la conversion de la température en C° vers les K° et F°.

AN : cas ou C°=20  $\Rightarrow$  K°=20+273,15=293,15

C°=20  $\Rightarrow$  F°= ((9 x 20) / 5) + 32=68

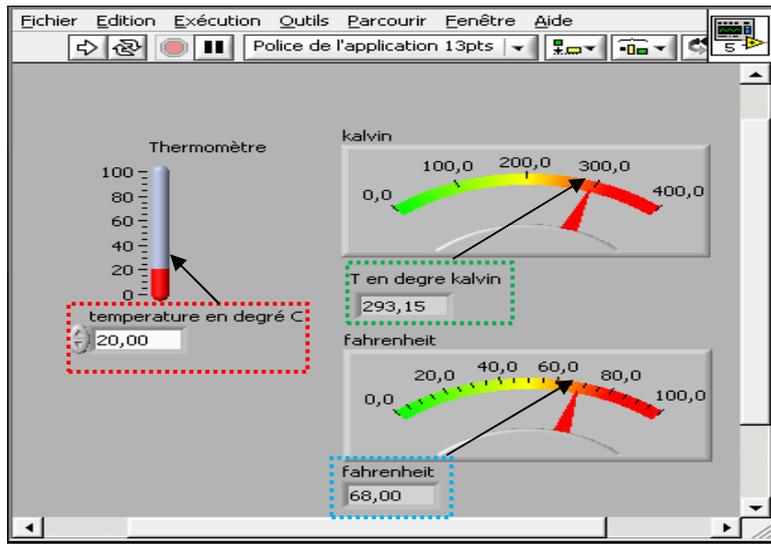


Figure III.13 : Fenêtre « face-avant » obtenue après une exécution du programme.

### III.5 Boucles :

Les structures de répétitions sont représentées graphiquement par un cadre, englobant un certain nombre de VIs et fonctions [9]. Selon le type de cadre, il s’agira de répétitions conditionnelles, d’itération ou de choix, en effet on peut distinguer 2 types de bouclage :

#### III.5.1 La boucle While (tant que)

- Possède un compteur d’itération.
- S’exécute toujours au moins une fois.
- Continue de s’exécuter tant que l’on n’a pas appuyé sur le bouton Stop [14].

#### III.5.2 La boucle For

- Possède un compteur d’itération.
- S’exécute N fois (N paramétrable).
- Pour N = 4, i = 0, 1, 2, 3.

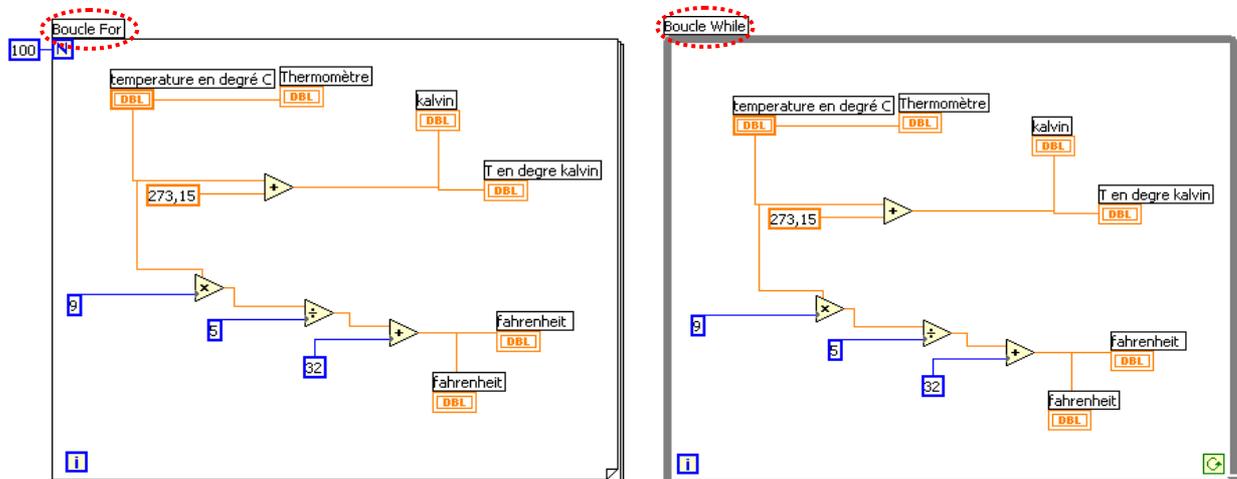


Figure III.17: Présentation des deux bouclages.



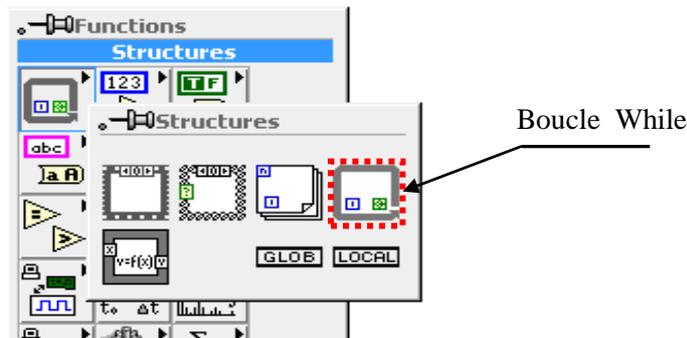
La condition d'arrêt, présentée ci-contre, permet de contrôler l'exécution de la boucle. La condition par défaut est **Continuée si vrai**. D'un clic droit il est possible d'en inverser cette condition. L'icône prend alors l'allure présentée à gauche.

**i** Le nombre d'itérations effectuées est contenu dans l'icône présentée à gauche [7]. Cette variable est initialisée à 0 lors de la première itération.

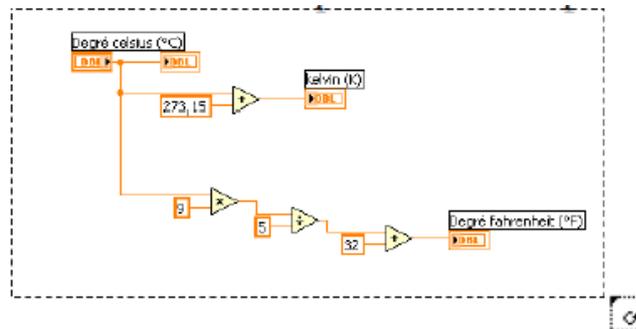
### III.5.3 Réalisation des boucles :

La création d'une boucle est basée sur 3 étapes, prenant l'exemple d'une boucle « While » :

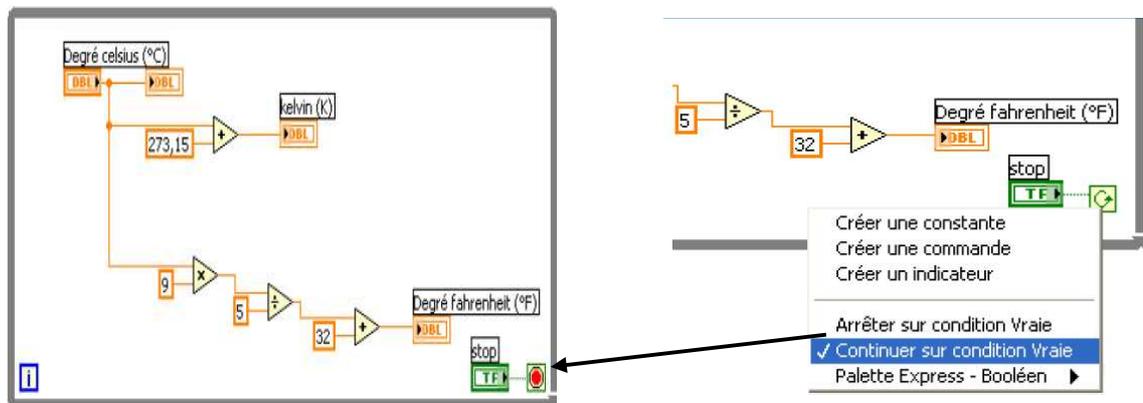
- ✓ Choisissons notre boucle : on peut effectuer ce choix comme le montre cette figure :



- ✓ Encadrer le code qui doit être répété.

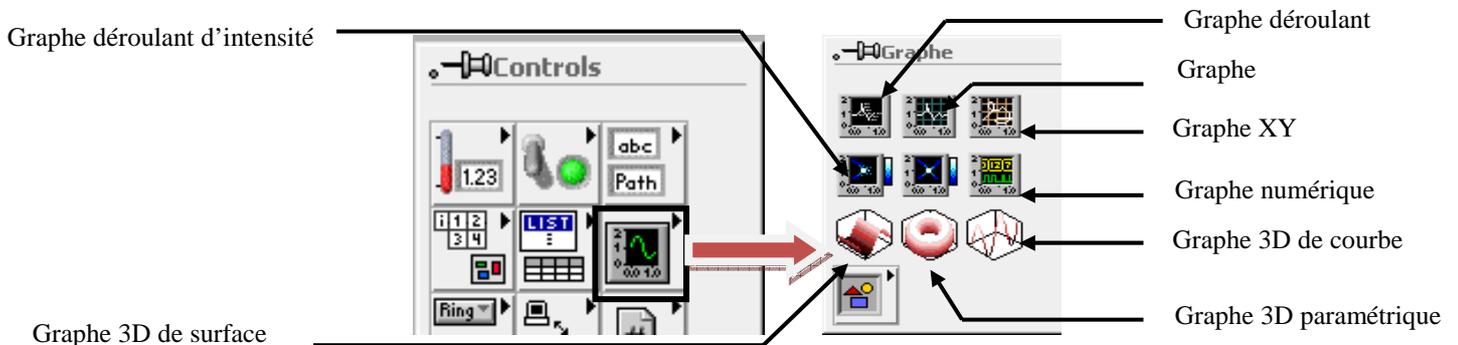


- ✓ Placer les nœuds additionnels (contrôle booléen : bouton stop par exemple) et les relier.



**III.6 Les graphes :**

Ils sont des indicateurs numériques destinés à afficher l'évolution d'une (ou plusieurs) variable(s) sur une (ou plusieurs) courbe(s). L'axe des X a donc pour échelle un numéro d'ordre d'arrivée du point. Les graphes déroulants sont situés dans la palette **Graphes** [3]. L'exemple suivant vous présente un graphe multi courbes.

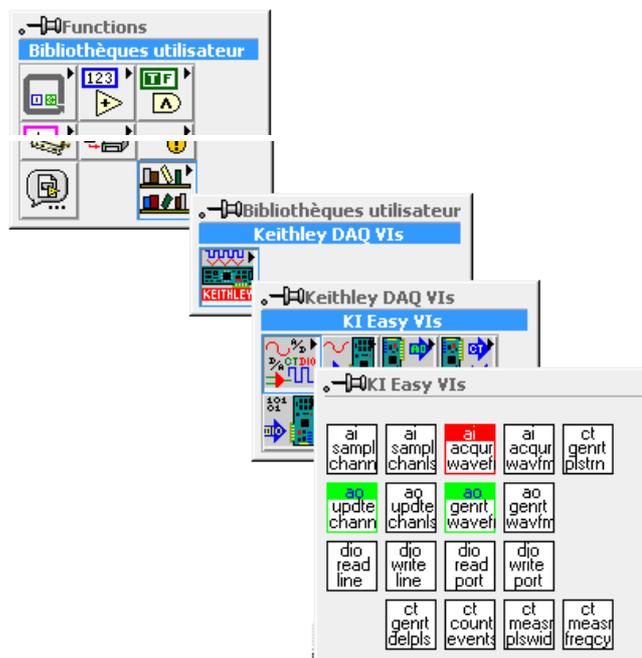


**III.7 Application : acquisition de données et automatisation :**

**III.7.1 DAQmx (Assistant keithley DAQ; VI Express):**

L'installation du driver LINX pour LabVIEW de notre carte permet l'ajout d'une bibliothèque de sous Vis, d'entré(s)/sortie(s) analogique et numérique et des compteurs; qui va enrichir la palette de fonctionnement de LabVIEW, et facilité l'acquisition, la commande et le traitement de plusieurs signaux. En outre ces sous Vis peuvent être modifiés selon nos besoins.

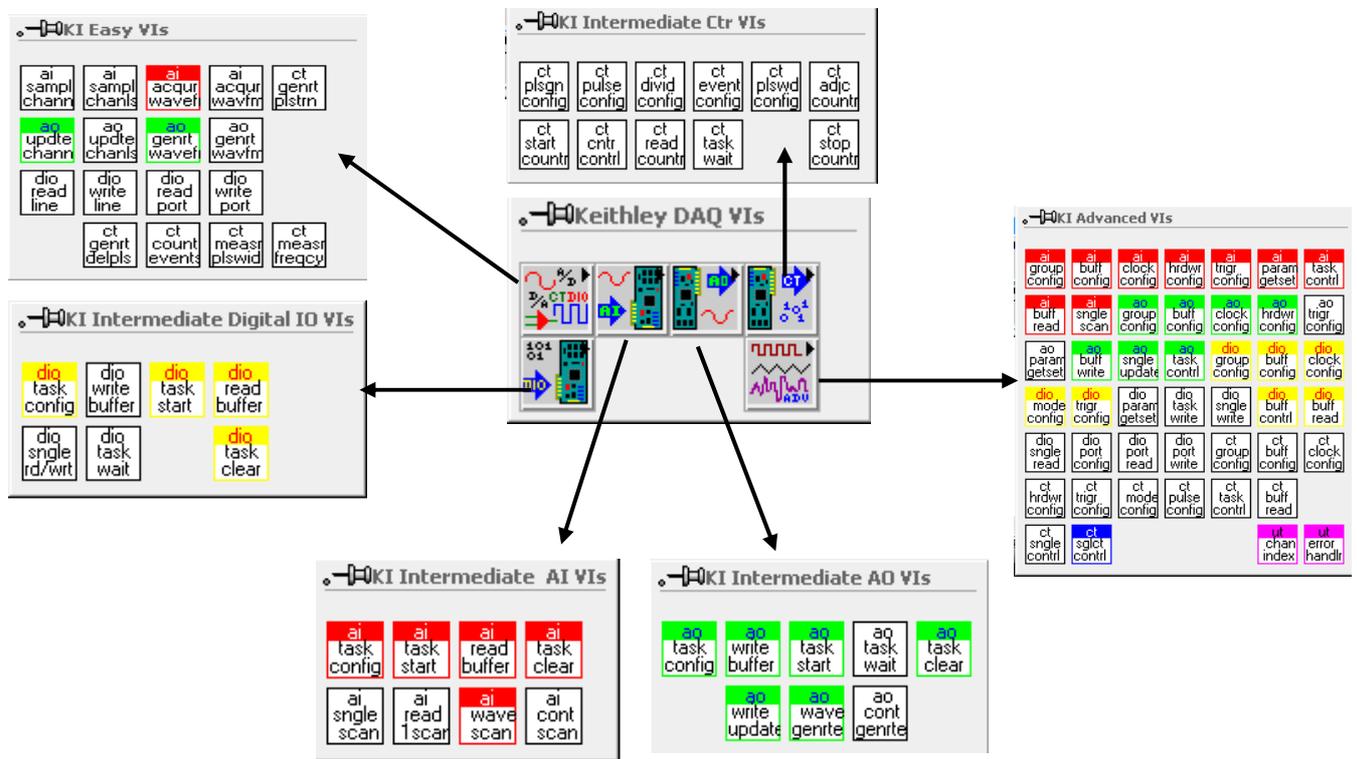
Le schéma suivant montre le chemin d'accès à cette bibliothèque.



**Figure III.23 : L'accès à la bibliothèque keithley DAQ Vis.**

**A- Quelques exemples de Vis utilisés pour l'acquisition et la commande :**

L'avantage de la bibliothèque résidant dans la diversité des blocs de Vis conçus et prêts à l'utilisation, [14] comme on le voit sur les figures suivantes :

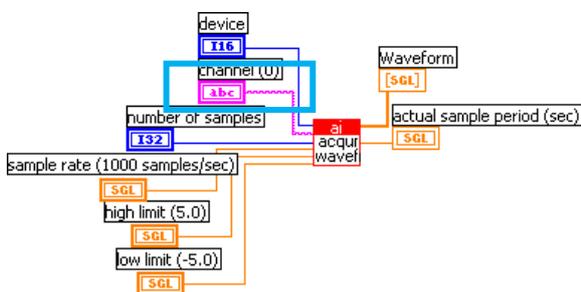


**Figure III.24 : Les différents blocs d'acquisition et de commande.**

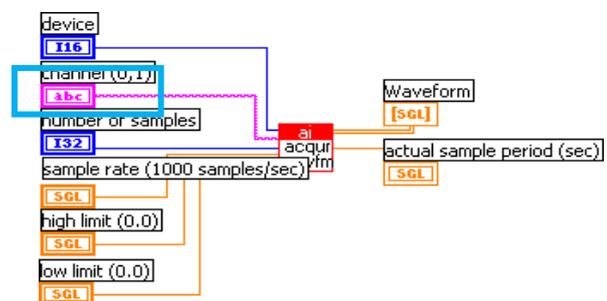
**a- Blocs d'entrée analogique (analogue input « ai »):**

Selon le type de signal qu'il soit ondulatoire ou bien constant on a :

**a.1/Il existe deux types de blocs d'acquisition analogique sinusoïdale (ai), l'un nous permet d'acquérir un seul signal (figure III.25) et l'autre au minimum deux signaux (figure III.26).**



**Figure III.25 : Bloc (ai) d'un seul signal sinusoïdal**



**figure III.26 : Bloc (ai) de plusieurs signaux sinusoïdaux**

a.2/ là aussi il existe deux types de blocs d'acquisition analogique constant (ai), l'un nous permet d'acquérir un seul signal (figure III.27) et l'autre au minimum deux signaux (figure III.28).

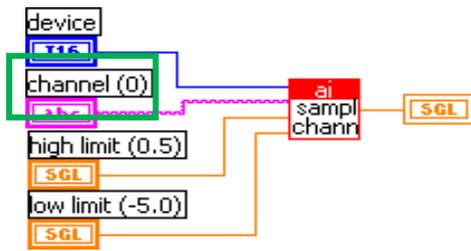


Figure III.27 : Bloc (ai) d'un seul signal constant.

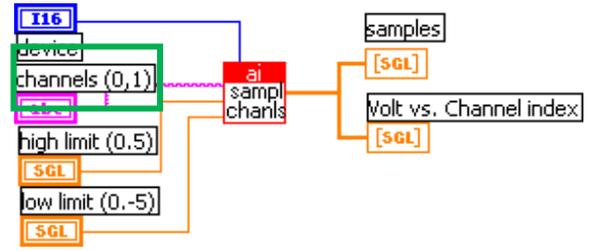


Figure III.28 : Bloc (ai) de plusieurs signaux constant

**b Blocs de sortie analogique (analogue output «ao») :**

Selon le type de signal qu'il soit ondulatoire ou bien constant on a :

b.1/ Il existe deux types de blocs de sortie (commande) analogique sinusoïdale (ao), l'un nous permet de commander par un seul signal (figure III.29) et l'autre au minimum deux signaux (figure III.30).

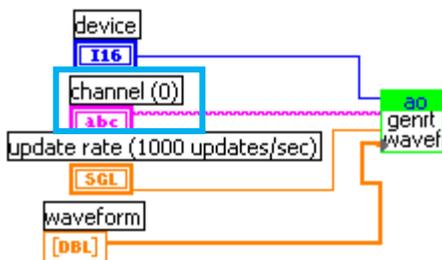


Figure III.29 : Bloc (ao) d'un seul signal

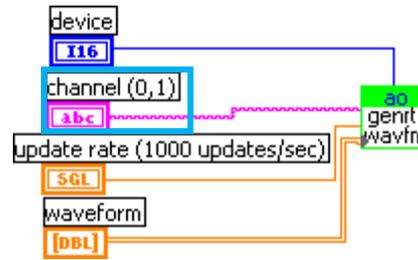


figure III.30 : Bloc (ao) de plusieurs signaux sinusoïdaux

b.2/ là aussi il existe deux types de blocs de sorties (commande) analogique constant (ao), l'un nous permet de commander par un seul signal (figure III.31) et l'autre au minimum deux signaux (figure III.32).

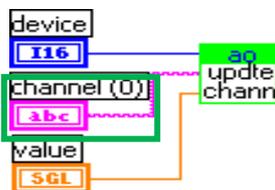


Figure III.31 : Bloc (ao) d'un seul signal constant.

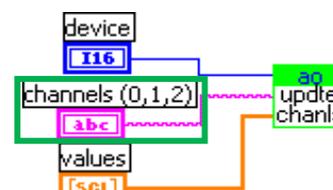


Figure III.32: Bloc (ao) de plusieurs signaux constant

III.7.2 Exemple d'application.

A- Acquisition et commande d'un système thermique:

Dans le but de maîtriser le logiciel (acquisition, commande, sauvegarde sous fichier...etc.), on a effectué plusieurs essais sur différents systèmes tel que le système thermique.

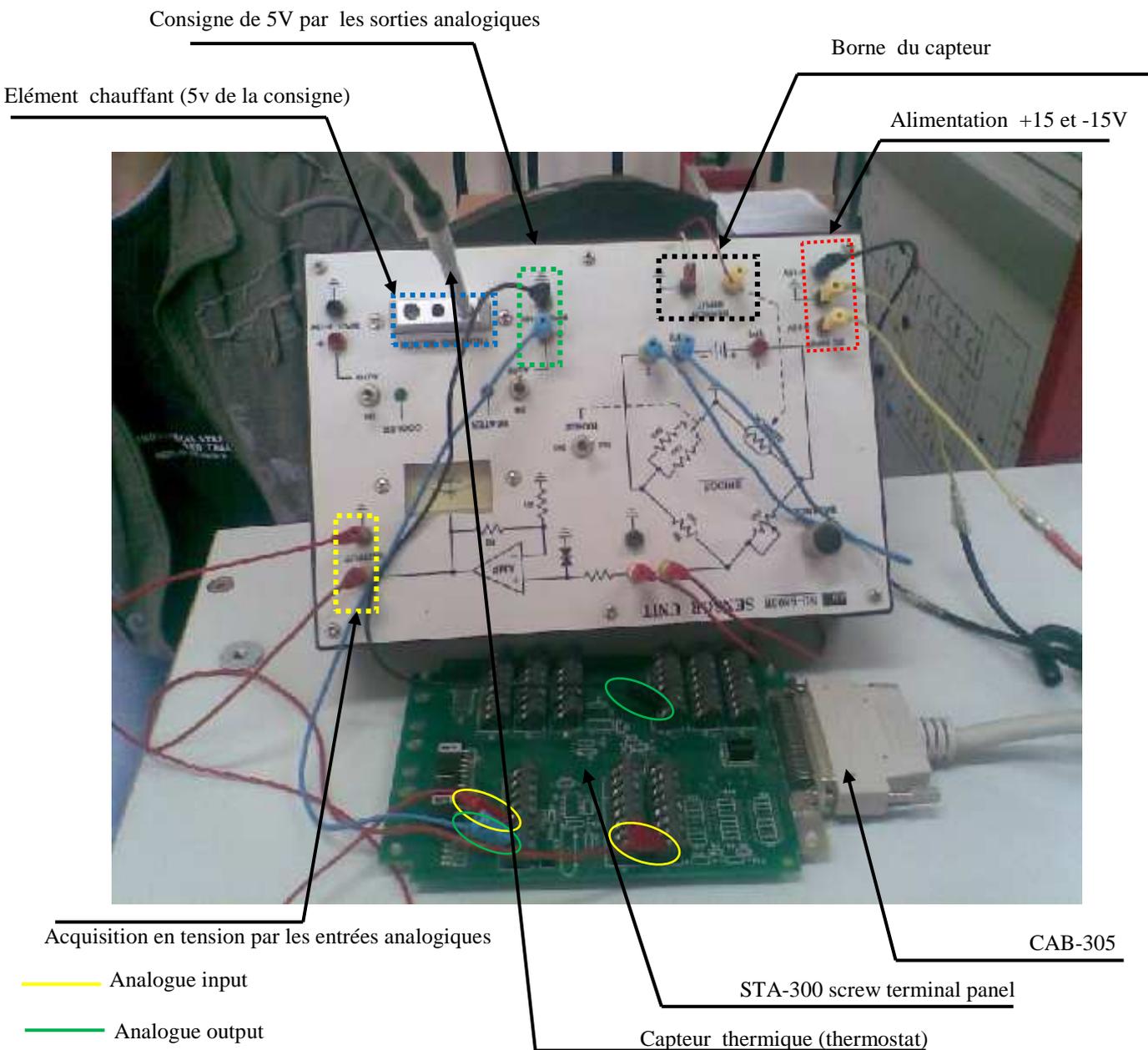
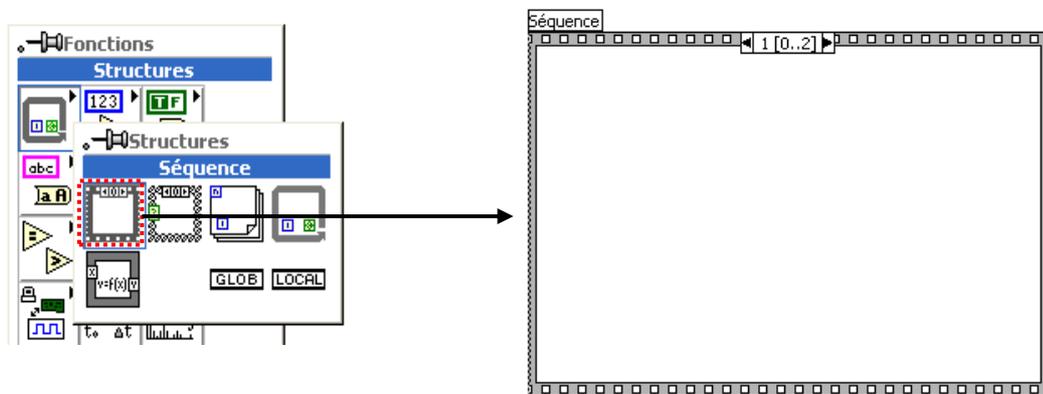


Figure III.33 : Schéma synoptique du système thermique étudié.

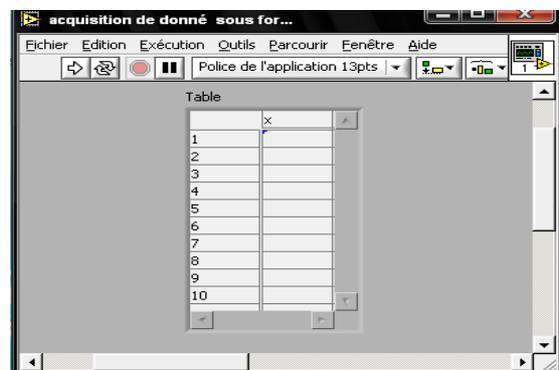
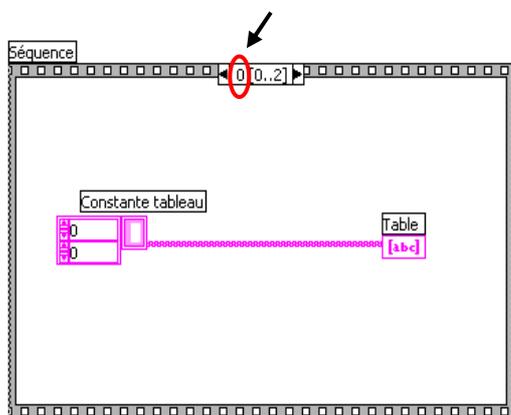
**Problème :** génération d'une commande de 5V, acquérir la variation de la tension en fonction de la température de la résistance chauffante, enfin sauvegarder ces données sous un fichier (.XLS ou .MAT...), pour cela on a effectué un programme de 3 étapes :

- ✓ **Remarque :** la sauvegarde des résultats sous fichier EXL nécessite d'effectuer toutes les étapes sur des pages de séquence qui s'affichent comme suit:



**Etape0 : Initialisation de la table.**

Cette étape permet d'initialiser la table à des valeurs nulles (par défaut).

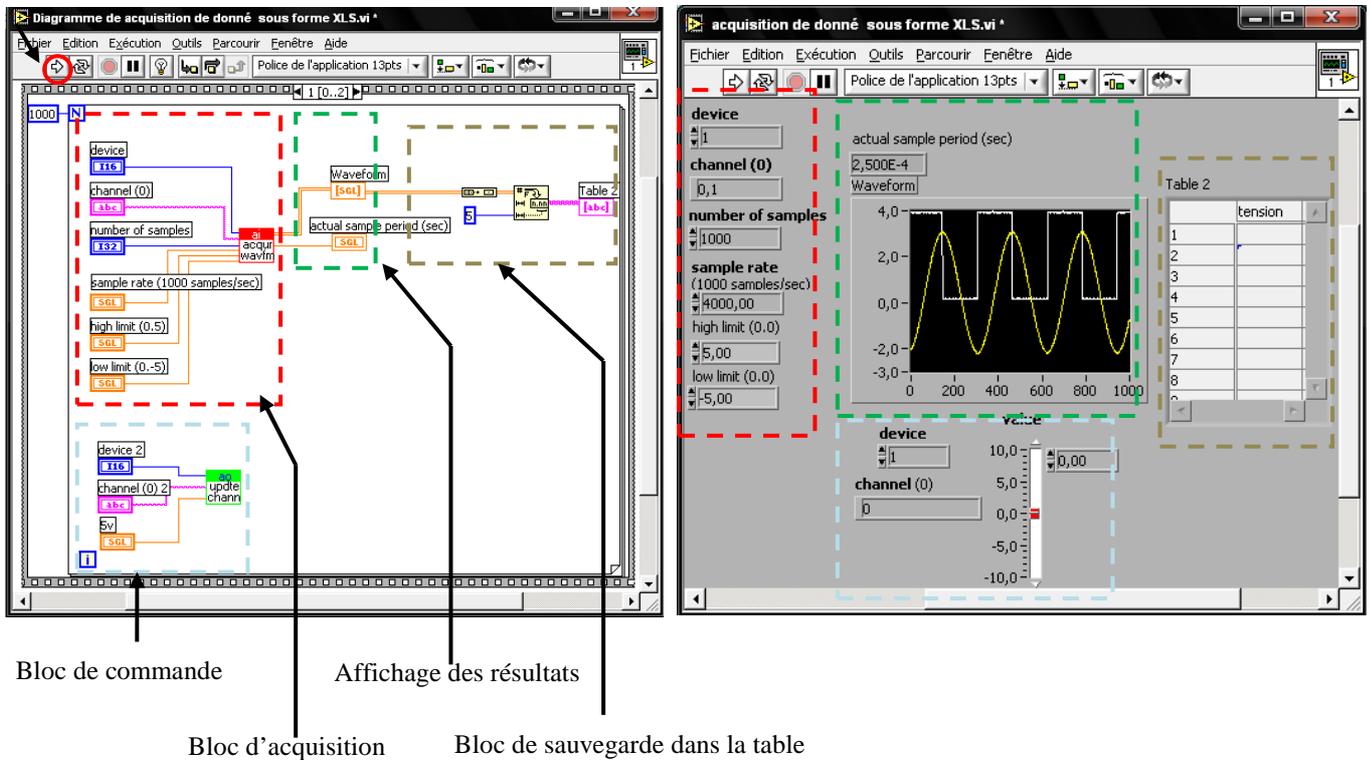


ence de la tab

**Figure III.34 : L'apparition de la table dans la face avant**

✓ **Etape1 : Remplir la table.**

Cette étape correspond à un programme de commande pour le démarrage du processus d'échauffement puis l'acquisition des données qui vont être sauvegardé dans la table.



**Figure III.35 : L'apparence de l'étape 1 dans la face avant et le diagramme.**

**AVEC :**

- 116** **Device** : est un nombre entier entre 1 et 16 lequel identifie le panneau DAQ. Pendant la Configuration DriverLINX, l'utilisateur assigne le nombre de l'appareil à un conducteur DriverLINX. L'entr e par d efaut est 1.
- 132** **Number of samples/ch** : sp cifie le nombre d' chantillons par canal i.
- SGL** **Scan rate (1000 scans/sec)** : sp cifie le taux que ce VI lit chaque canal dans le groupe de canal, L'entr e par d efaut est 1,000 scanners / seconde. DriverLINX pour LabVIEW n'autorise pas une valeur de 0.
- SGL** **Low limit (0.-5)** : est le niveau attendu le plus bas du signal de l'entr e, notre cas a la valeur -5V.
- SGL** **High limit (0.5)** : est le niveau attendu le plus haut du signal de l'entr e, notre cas a la valeur 5V.
- abc** **Channels** : permet de s lectionner un canal ou plusieurs parmi les 6 canaux existant pour l'utiliser   l'acquisition ou   la commande.
- SGL** **Waveform** : est une collection du flotteur   deux dimensions qui contient des donn es unit es, il permet d'afficher les r sultats sous forme de graphe [14].

✓ **Etape2 : Sauvegarde des données sous fichier XLS.**

Après le remplissage de notre table nous sauvegardons les résultats sous fichier .xls son programme est le suivant :

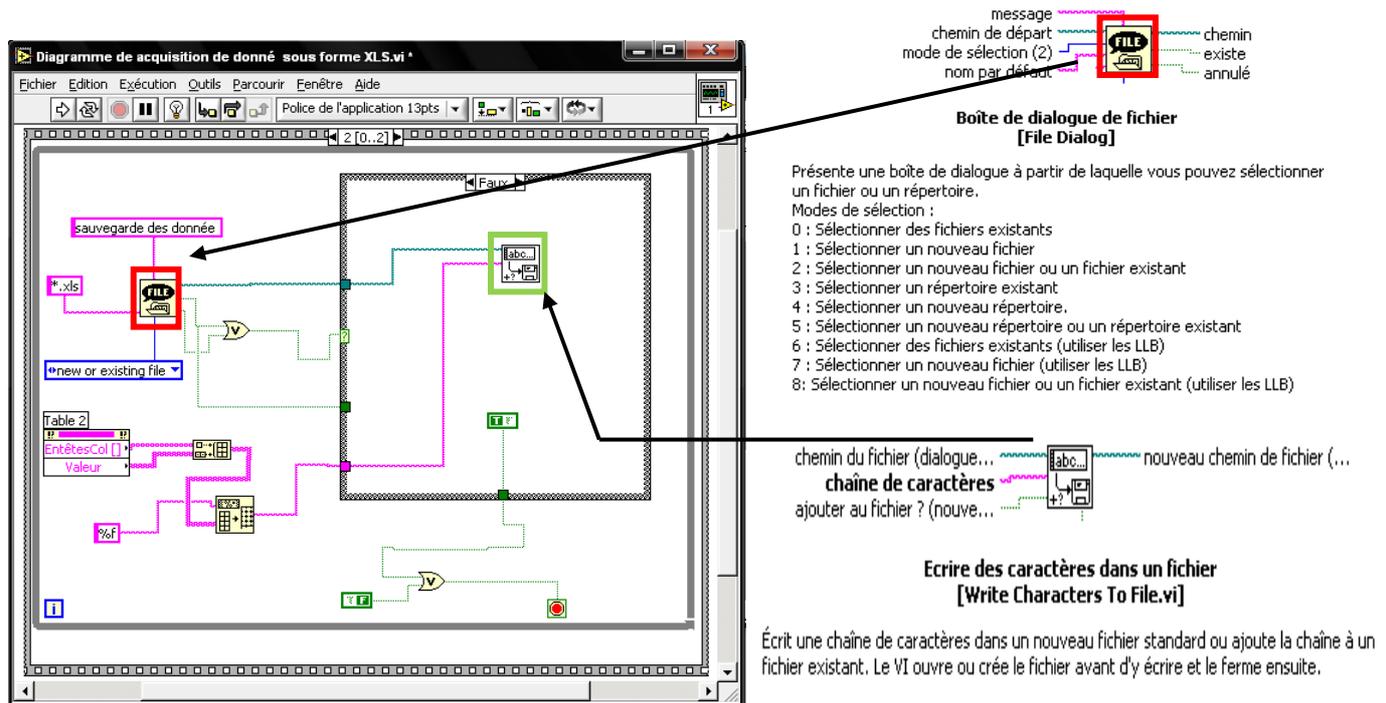
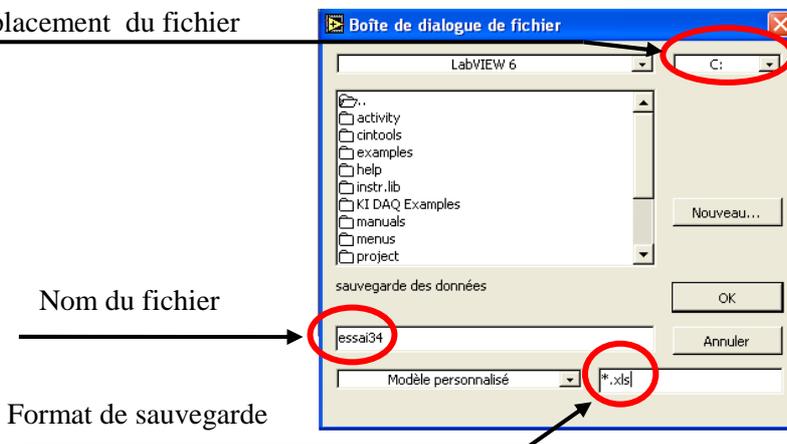


Figure III.36 : Apparence du programme de L'étape 2 dans le diagramme.

**Exécution du programme :**

1. pour exécuter le programme on appuie sur le bouton d'exécution
2. La table 2 de la première étape se remplit jusqu'à la fin des 1000 itérations dans la face avant puis une boîte de dialogue s'apparaît.
3. Après remplissage de la boîte et validation, la deuxième étape déclenche pour la sauvegarde.
4. Pour étudier ces résultats sous l'Excel le fichier se trouvera automatiquement dans l'emplacement prévu dans la boîte de dialogue.

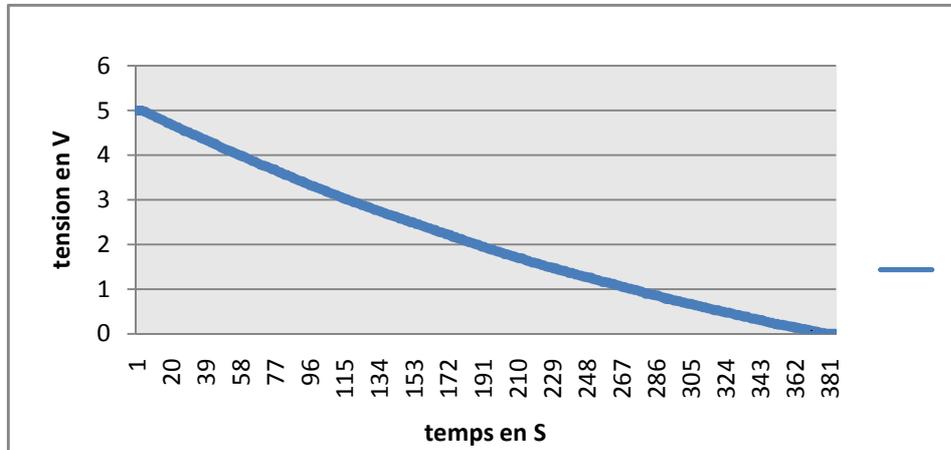
Emplacement du fichier



✓ **Affichage des résultats :**

Après l'enregistrement des données acquises dans le fichier cible (l'Excel) il nous reste qu'à affiché les résultats pour les utiliser dans l'étude des réactions du système aux diverses commandes qu'on peut réaliser et son comportement dynamique.

Les résultats obtenus par cette acquisition sont représentés sur le graphe suivant :



**Figure III.37 : Variation de la tension en fonction du temps (température).**

**Interprétation :**

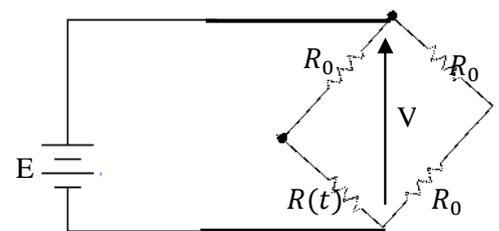
Les résultats obtenus sur cette figure représentent la variation de la tension acquise(V) par rapport à la température de l'élément chauffant du système.

On remarque que la tension diminue exponentiellement avec l'augmentation de la température de cet élément chauffant suivant la relation établie par le capteur de température (thermistance) qui s'effectue comme suit :

La relation entre la température(T) du capteur et sa tension de

sortie (V) est :  $V = \frac{E}{4} \frac{\Delta R(t)}{R_0}$  avec  $R(t) = R_0 e^{\beta(\frac{1}{T})}$

-Si T augmente ; R(t) diminue exponentiellement. Donc la tension V diminue aussi exponentiellement, d'où l'obtention de l'allure de la figure (III.37).



**B- Régulation de la température TOR :**

Dans le but de limiter l'échauffement de l'élément chauffant autour d'une température moyenne (25°C) alors on propose le programme suivant :

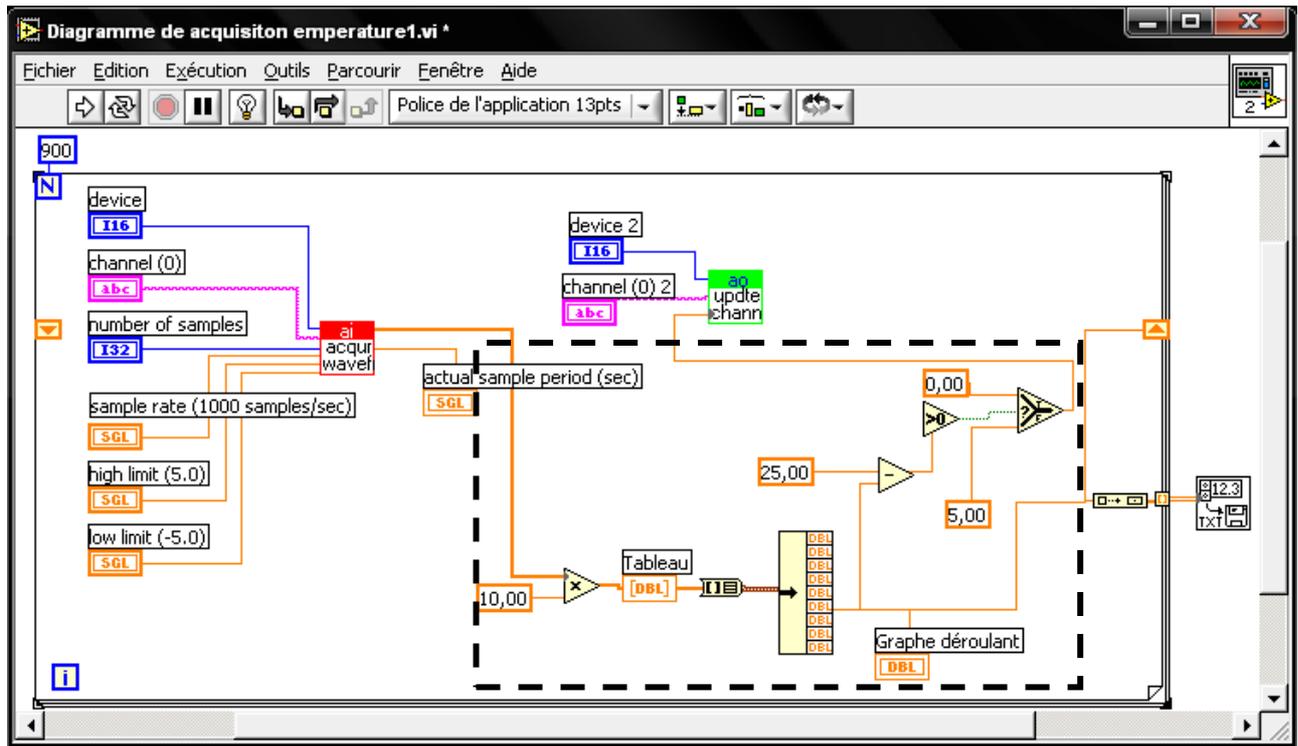


Figure III.38 : Programme utilisé pour la régulation TOR

Cette figure représente le programme pour le TOR, en effet on a effectué seulement un changement au niveau de la commande par rapport au programme précédent (voire l’encadrement en noire).

➤ **Explication du fonctionnement du programme :**

Après la mise en marche avec le bouton (RUN) le bloc d’acquisition effectue un prélèvement des valeurs de température de l’élément chauffant sous forme des tensions, qui seront multipliées par 10 car 1V du capteur donne 10°C, puis ces valeurs seront comparées avec la température voulue (25°C). Si elles sont inférieures le bloc de commande délivre une tension de 5V vers la résistance, si non il délivre un zéro et ainsi de suit jusqu’à avoir une stabilisation au tour de la température désirée. La figure suivante montre ce la :

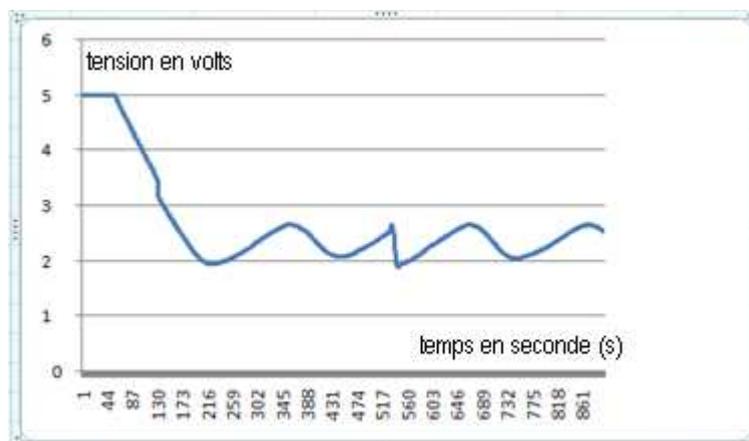


Figure III.39 : Régulation de la température autour

➤ **Interprétation :**

On remarque dans cette figure deux étapes: la première c'est l'étape de transition et l'autre c'est celle permanente.

En effet la première représente l'échauffement de la résistance c'est pour cela qu'on remarque la diminution de la tension.

La deuxième représente l'effet de la régulation TOR, en effet la résistance se chauffe et se refroidisse autour de 2,5V ce qui correspond au 25°C qui est la température désirée.

### **III. 8 Conclusion :**

L'intérêt de logiciel LabVIEW réside dans sa facilité de programmation par rapport aux autres logiciels (Matlab, Pascal ...) ,ce qui a poussé les chercheurs à le développer jusqu'à ce qu'il touche la plupart des domaines scientifiques tel que le traitement d'image, acquisition d'image et vidéos par des web Cam ou des caméras, la météorologie et la commande des systèmes très complexes comme ceux de l'industrie et ça grâce à l'intégration de l'idée langage contact qui à fait naissance de grafcetVIEW. De plus toutes ces applications peuvent être effectuées en temps réel.

# Chapitre IV

*Acquisition, identification  
Et commande du moteur à  
courant continu*

**IV.1 INTRODUCTION :**

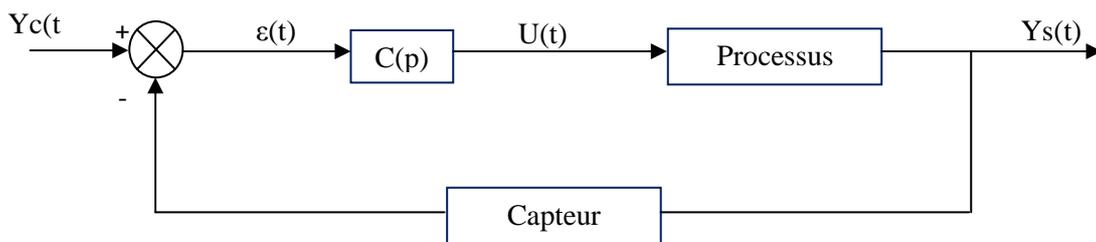
Ce chapitre consiste à faire l'acquisition des données de notre système à étudier et cela grâce au logiciel LabVIEW via la carte d'acquisition KPCI 3102, ces données seront traitées sous MATLAB pour l'identification et la simulation pour la détermination des paramètres de régulation. On terminera par la régulation du système sous LabVIEW.

**PARTIE I :**

**IV.2 Rappel sur la commande et l'identification numériques :**

**IV.2.1 Commande numérique :**

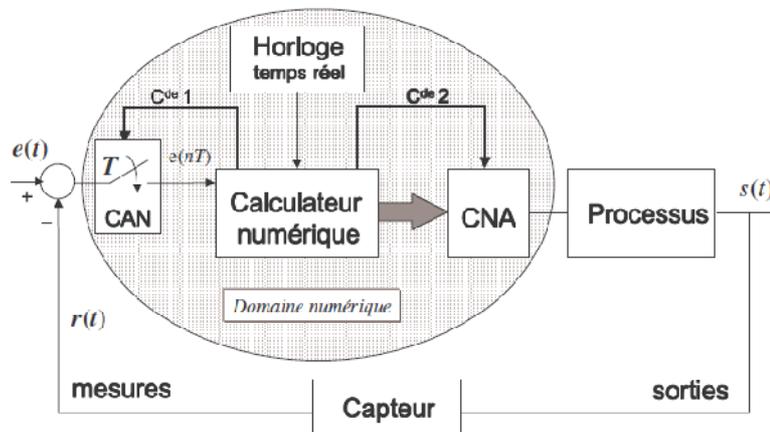
Le principe de la commande numérique est identique à celui de la commande analogique. La structure d'un système bouclé reste la même dans les deux cas, cependant la nature des signaux nécessaires pour le traitement de l'information change [16]. En effet dans la commande numérique on introduit les dispositifs électroniques capable de transformer un signal analogique en un signal numérique et inversement.



**Figure IV.1 : Schéma d'asservissement d'une régulation d'un processus analogique.**

Dans une chaîne de commande un correcteur analogique C(P) produisant la commande U(t) peut être un correcteur classique de type PID, PI, PD. Ou un circuit à avance ou à retard de phase [16].

Cette commande U(t) peut être aussi produite à l'aide d'un ordinateur selon la chaîne d'asservissement suivante :



**Figure IV.2 : Schéma d'asservissement d'une régulation numérique d'un processus.**

A la sortie de l'ordinateur, on doit avoir un CNA, dans ce cas la sortie doit être très proche de  $U(t)$  analogique. Comme le montre cette figure IV.3 :

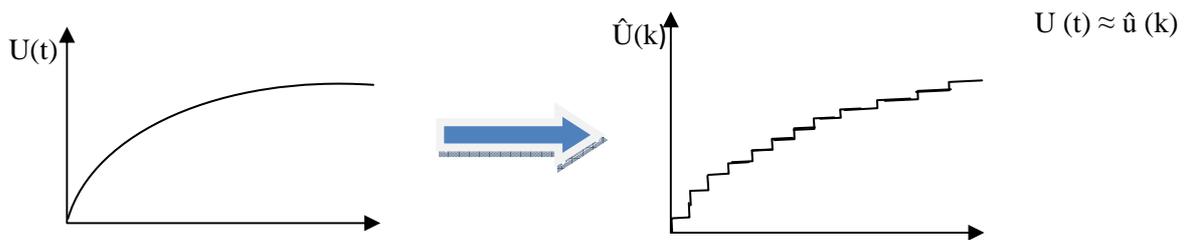


Figure IV.3 : Discretisation d'une consigne continue.

On aura donc :

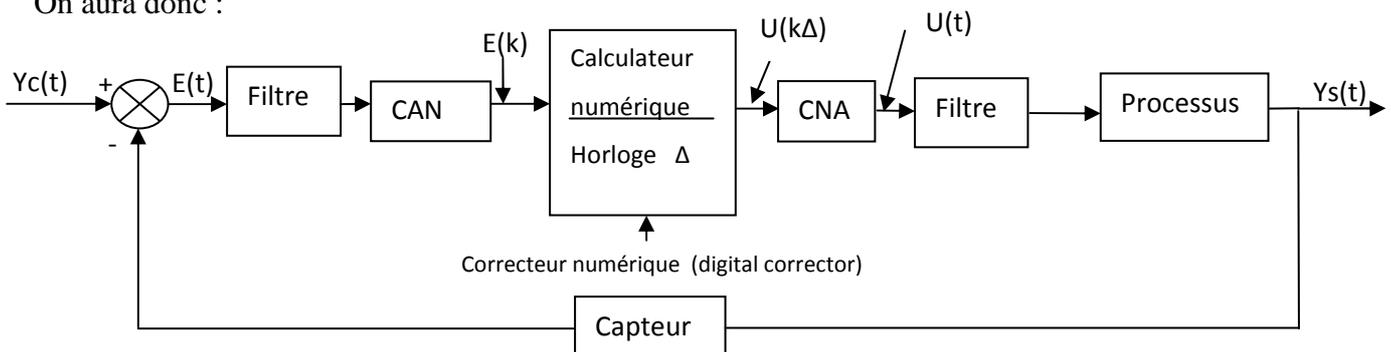


Figure IV.4.a : Schéma d'une correction numérique

Le correcteur dans ce cas est réalisé sous forme d'un programme dans l'ordinateur. Le système est alors vu comme un système numérique de fonction de transfert  $H(z) = \frac{B(z)}{A(z)}$ . Le correcteur est donc représenté sous forme d'équation de récurrence qui lie  $U(k)$  à  $\epsilon(k)$ . Le cas ci-dessus représente une commande analogique à ordinateur [16].

Pour mieux exploiter les possibilités de calculateur on a d'autres dispositifs de commande numérique par ordinateur.

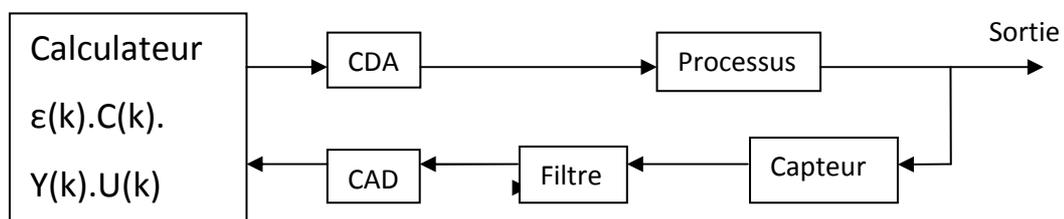


Figure IV.4.b : Schéma d'une commande numérique par ordinateur.

Il n'y a pas de changement sur le plan des dispositifs technologiques par rapport aux cas précédant (analogique par ordinateur). Ici il n'apparaît pas de comparateur, tous les signaux sont traités par programme :  $c(k), y(k), \epsilon(k)$  et  $u(k)$ . Le correcteur particulièrement est déterminé sous forme d'un

programme de l'équation de récurrence liant  $\varepsilon(k), u(k)$ , ce correcteur peut être classique ou de forme quelconque. On notera que ces correcteurs sont toujours précédés d'un BOZ [16].

#### IV.2.2 Formes échantillonnées des correcteurs classiques :

Le régulateur discret élabore une grandeur de commande discrète  $U^*(k)$  en fonction de l'écart de décalage discret  $\mathcal{E}^*(k)$  du système à commander. Selon la complexité du gradateur, la grandeur de commande à l'instant  $t = nT$  est formée en fonction de la valeur de l'écart à cet instant et aux instants précédents  $(n-1)T, (n-2)T, \dots$

Dans le cas des régulateurs continus qui réalisent généralement la relation :

$$U(t) = k_p \varepsilon(t) + \frac{1}{T_i} \int_0^t \varepsilon(t) dt + T_d \frac{d\varepsilon(t)}{dt} \quad (\text{IV.1})$$

Si l'action proportionnelle **P** est des plus simples, les actions intégrales, dérivés méritent une certaine attention.

**A- Action Intégrale I :** sa loi de commande dans le domaine temporel est :

$$u(t) = \frac{1}{T_i} \int_0^t \varepsilon(t) dt \quad (\text{IV.2})$$

Alors si on lui effectue une transformation en **Z** alors cette dernière redevienne :

$$u(z) = \frac{z^{-1}}{z} TZ \left( \frac{1}{T_i p^2} \right) = \frac{1}{T_i} \frac{z^{-1}}{z} TZ \left( \frac{1}{p^2} \right) \quad (\text{IV.3})$$

Et après échantillonnage d'une période  $\Delta$  on obtient définitivement l'action **I** comme suit :

$$I(z) = \frac{\Delta}{T_i} \frac{1}{z-1} \quad (\text{IV.4})$$

Avec :  $T_i$  : constante d'intégration.

#### Equation de récurrence :

D'après l'équation (IV.4) on a  $u(z)(z-1)T_i = \Delta \varepsilon(z) \iff T_i u(z)z = T_i u(z) + \Delta \varepsilon(z)$

$\iff u(z) = u(z)z^{-1} + \frac{\Delta}{T_i} z^{-1} \varepsilon(z)$  et on obtient l'équation de récurrence suivante à l'aide de la  $Tz$

inverse :

$$u(k) = u(k-1) + \frac{\Delta}{T_i} \varepsilon(k-1) \quad (\text{IV.5})$$

**B- Dérivateur :**

Il se traduit par un terme proportionnel à la différence des écarts de réglage aux instants d'échantillonnage  $nT$  et  $(n-1)T$ , et sa loi de commande dans le domaine temporel est :

$$U(t) = T_d \frac{d\varepsilon(t)}{dt} \quad (\text{IV.6})$$

Après échantillonnage d'une période  $\Delta$  on obtient définitivement l'action  $D$  comme suit :

$$D(z) = \frac{z-1}{z} TZ \left[ \frac{TdP}{P(1+\frac{Td}{N})} \right] \implies D(z) = N \times \frac{z-1}{z-z_0} \quad (\text{IV.7})$$

Avec  $z = e^{-\frac{N\Delta}{T_d}}$  et  $Td$  : constante de la dérivée.

Le pôle  $Z_0$  a rendu douce la dérivée d'une entrée numérique.

**Equation de récurrence :**

Si  $\varepsilon(p) = \frac{1}{p}$  donc  $\varepsilon(z) = \frac{z}{z-1}$  on aura :

$$u(z) = D(z) \times \varepsilon(z) = N \frac{z-1}{z-z_0} - \frac{z}{z-1} = \frac{z}{z-z_0}$$

$$u(k) = \sum \text{Res} \left[ \frac{Nz^k}{z-z_0} \right] = Nz_0^k \implies \boxed{u(k) = Ne^{-\frac{N\Delta}{T_d}k}} \quad (\text{IV.8})$$

**C- Correcteur PID :**

Le correcteur PID est très largement utilisé dans les procédés industriels. Sa popularité est due à sa simplicité, il n'y a que trois paramètres à régler. Le régulateur PID se base sur le régulateur PI, auquel on ajoute une composante dérivée.

$$c(p) = K \left[ 1 + \frac{1}{T_i p} + T_d p \right] \quad (\text{IV.9})$$

Et après échantillonnage d'une période  $\Delta$  on obtient définitivement l'action  $PID$  comme suit :

$$c(z) = K \left[ 1 + \frac{1}{T_i} \frac{1}{z-1} + N \frac{z-1}{z-z_0} \right] \quad (\text{IV.10})$$

$z_0$  : De l'ordre de 0.2 à 0.4 (amortie)

$N$  : règle l'effet dérivé

$T_i$  : règle l'effet intégral

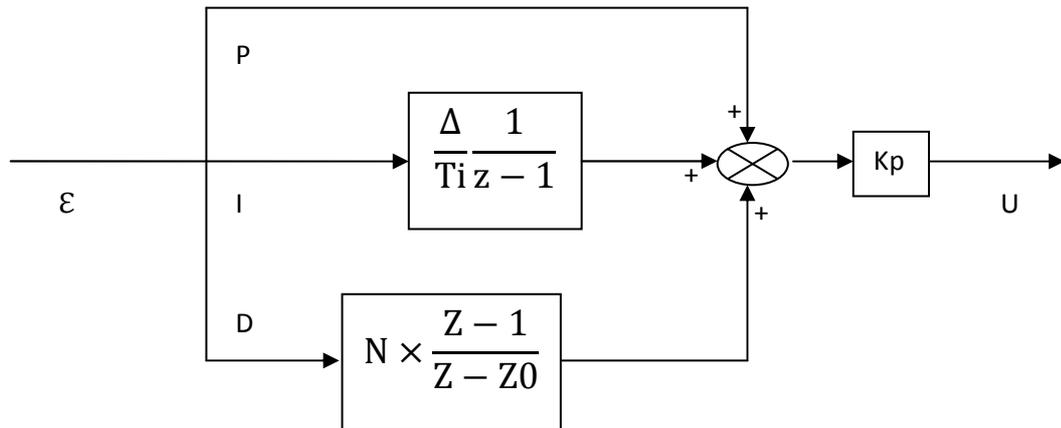
$K$ : Amplitude de correcteur

**Loi de commande :**

La loi de la commande du régulateur PID numérique peut être obtenue par discrétisation de celle d'un régulateur PID analogique :

$$u(t) = K[\mathcal{E}(t) + \frac{1}{T_i} \int_0^t \mathcal{E}(t) dt + T_d \frac{d\mathcal{E}}{dt}] \quad (IV.11)$$

**Remarque :** bien souvent, dans les régulateurs standards, aussi bien continus que numériques, l'action P est commune aux autres actions I et D. La structure d'un PID se présente alors selon le schéma ci-dessous :

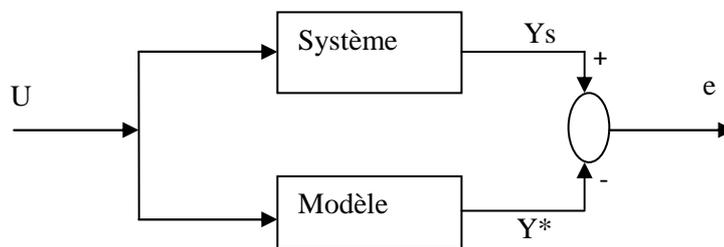


**Figure IV.5 : Structure numérique d'un correcteur PID**

**IV.3 Généralité sur l'identification :**

L'identification consiste à calculer les paramètres d'un modèle du système à partir des données expérimentales de façon à ce que le comportement du système et celui du modèle soient identiques et ceci pour toutes les séquences des variables d'entrée habituellement utilisées [17].

Le principe de l'indentification est donné par la figure suivante :



**Figure IV.6 : Comparaison du système au modèle**

Avec : Ys est la sortie du système.

Y\* est la sortie du modèle.

e=Ys-Ym.

### IV.3.1 Types de modèle :

#### A- Modèle de connaissance :

Il est élaboré à partir des lois de la physique ou de la chimie. Son objectif principal est d'expliquer un phénomène par une relation mathématique comportant de nombreux paramètres ayant un sens physique [17] (résistance, inductance, masse, coefficient de frottement, section, longueur,...)  $\Rightarrow$  **boite blanche**.

#### B- Modèle de comportement :

C'est un modèle généralement linéaire dont la validité reste limitée à de petites variations autour du point de fonctionnement.

On les appelle aussi fonction de transfert [17], les coefficients du modèle sont les paramètres du modèle qui ont rarement un sens physique  $\Rightarrow$  **Boite noire**.

#### C- Modèle intermédiaire :

Il constitue un hybride entre les deux modèles précédents. On peut les considérer comme des modèles de connaissances simplifiés [17]. La structure est déterminée par des lois physiques (modèle de connaissance), et les paramètres à partir des mesures  $\Rightarrow$  **Boite grise**.

### IV.4 Etapes de l'identification :

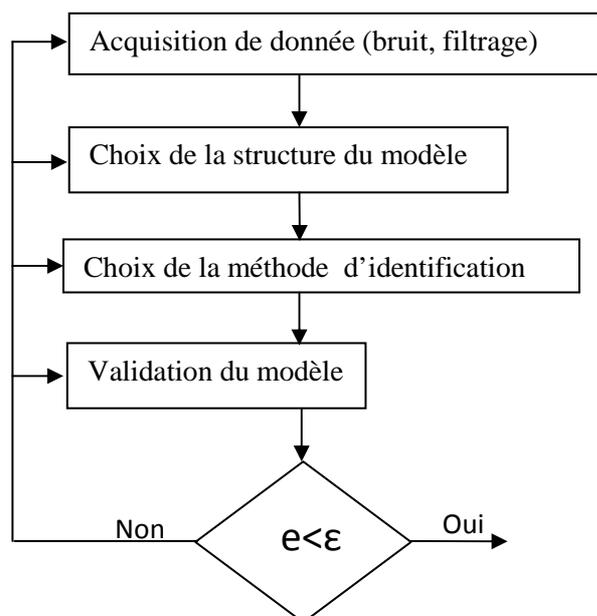


Figure IV.7 : Résumé des étapes d'identification.

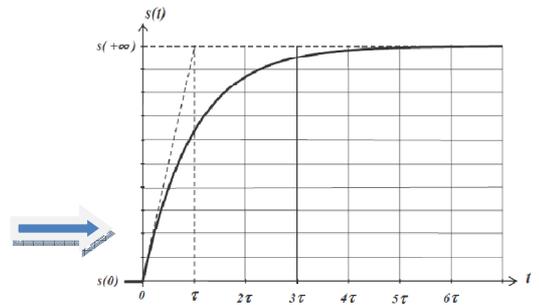
**IV.5 Choix de la structure des modèles :**

Le problème rencontré dans le cas des modèles de comportement (fonction de transfert) est le choix de l'ordre des polynômes (numérateur et dénominateur) de la fonction de transfert.

**Quelques modèles :** les figures suivantes représentent les différentes allures qu'on peut avoir et leurs fonctions de transfert :

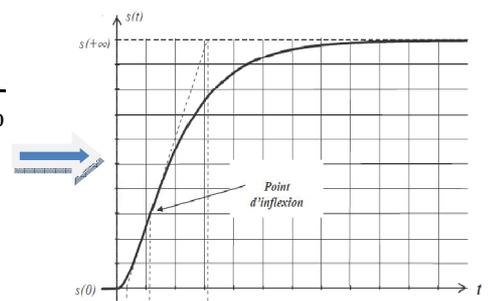
**A- Un system de 1<sup>er</sup> ordre :**

Sa fonction de transfert est sous la forme de  $G(s) = \frac{b_0}{a_n s + a_0}$   
 Et sa réponse à un échelon est représentée par la figure si contre.



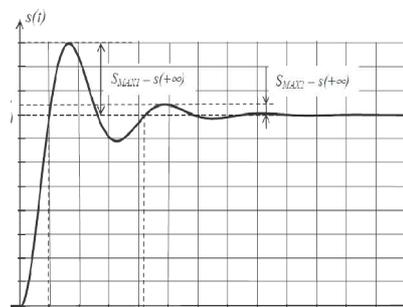
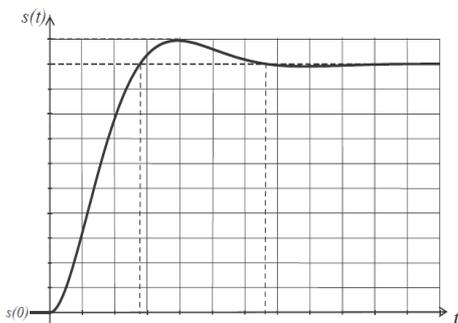
**B- Un système de 2<sup>e</sup> ordre sur amortie :**

Sa fonction de transfert est sous la forme de  $G(s) = \frac{b_0}{a_2 s^2 + a_1 s + a_0}$   
 Et sa réponse à un échelon est représentée par la figure si contre.



**C- Un système de 2eme ordre sous amortie :**

Sa fonction de transfert est sous la forme de  $G(s) = \frac{b_0}{a_2 s^2 + a_1 s + a_0}$   
 Et sa réponse à un échelon est représentée par les 2 figures suivantes :



**IV.6 Choix de la méthode d'identification :**

Il existe différentes méthodes pour l'identification, et l'utilisation de ces dernières diffère d'un système à un autre et cela dépend de nombre et de types de paramètres recherchés, en effet on peut trouver :

**IV.6.1 Méthodes graphiques :**

Elles sont basées sur la réponse indicielle de système. Ces méthodes permettent d'avoir des résultats précis mais peuvent déduire à un modèle suffisant.

**IV.6.2 Méthodes numériques :****A- Méthode des moindres carrées :**

Le principe des moindres carrés, selon Gauss, est de choisir les paramètres d'un modèle de telle sorte que la somme des carrés de la différence entre les valeurs prédites par le modèle et celles mesurées, éventuellement multipliées par un coefficient caractérisant le degré de précision, soit minimale [17].

**B- Application des moindres carrés à l'identification des systèmes**

La méthode des moindres carrés peut être utilisée pour l'identification des paramètres d'un modèle dynamique d'un système. A la différence des méthodes graphiques [17], la méthode utilisée donne la fonction de transfert sous la forme :

$$G(z^{-1}) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_m z^{-m}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n}} = \frac{y(z)}{u(z)} \quad (\text{VI.12})$$

Posant :  $a_0 = 1$  l'équation aux différences donne :

$$y(z) + a_1 y(z)z^{-1} + a_2 y(z)z^{-2} + \dots + a_n y(z)z^{-n} = b_0 u(z) + b_1 u(z)z^{-1} + b_2 u(z)z^{-2} + \dots + b_m u(z)z^{-m}$$

$$y(k) = -a_1 y(k-1) - a_2 y(k-2) - \dots - a_n y(k-n) + b_0 u(k) + b_1 u(k-1) + b_2 u(k-2) + \dots + b_m u(k-m)$$

Pour appliquer la méthode des moindres carrés, on suppose qu'une séquence d'entrée  $\{u(1), u(2), u(3), \dots, u(n)\}$  a été appliquée au système, et que la séquence de sortie correspondante  $\{y(1), y(2), y(3), \dots, y(n)\}$  a été mesurée.

Les paramètres sont regroupés dans le vecteur suivant :  $\Theta = [a_1, a_2, a_3, \dots, a_n, b_0, b_1, b_2, \dots, b_m]$

Et le vecteur de régression  $\phi(k+1)$  est représenté sous la forme :

$$\phi(k+1) = [-y(k), y(k-1), y(k-2), \dots, y(k-n+1), u(k), u(k-1), u(k-2), \dots, u(k-m+1)]$$

Le modèle dynamique du système s'écrit:  $y(k+1) = \phi(k+1) \Theta$ . (Iv.13)

$$y(n+1) = -a_1 y(n) - a_2 y(n-1) - \dots - a_n y(1) + b_0 u(n+1) + b_1 u(n) + b_2 u(n-1) + \dots + b_m u(n-m+1)$$

$$y(n+2) = -a_1 y(n+1) - a_2 y(n) - \dots - a_n y(2) + b_0 u(n+2) + b_1 u(n+1) + b_2 u(n) + \dots + b_m u(n-m+2)$$

$$y(N) = -a_1 y(N-1) - a_2 y(N-2) - \dots - a_n y(N-1-n) + b_0 u(N+1) + b_1 u(N-1) + b_2 u(N-2) \dots + b_m u(N-m)$$

On peut écrire ces vecteurs sous la forme matricielle suivante :

$$\underbrace{\begin{bmatrix} y(n+1) \\ y(n+2) \\ \vdots \\ y(N) \end{bmatrix}}_{\mathbf{y}_N} = \underbrace{\begin{bmatrix} -y(n) & -y(n-1) & \dots & -y(1) & u(n+1) & u(n) \dots & u(n-m+1) \\ -y(n+1) & -y(n) & \dots & -y(2) & u(n+2) & u(n+1) \dots & u(n-m+2) \\ \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\ -y(N-1) & -y(N-2) & & & \dots & U(N) & \dots & U(N-n) \end{bmatrix}}_{\Phi_N} \underbrace{\begin{bmatrix} a1 \\ a2 \\ \vdots \\ am \\ b0 \\ b1 \\ \vdots \\ bn \end{bmatrix}}_{\Theta_N}$$

$$\mathbf{y}_N = \begin{bmatrix} \phi(n+1) \\ \phi(n+2) \\ \vdots \\ \phi(n) \end{bmatrix} \Theta_N \text{ DONC} \quad \mathbf{y}_N = \Phi_N \Theta_N \tag{IV.14}$$

la dernière itération est représentée sous la forme suivante :  $\theta_N = (\phi_N^t \phi_N)^t \phi_N^t y_N$  (IV.15)

**C- Méthode des moindres carrés récursives :**

Il est possible de calculer la nouvelle valeur de vecteur des paramètres lors d'un nouvel instant de mesure t=N+1 est cela on ajoutant une autre mesure obtenue à ce moment [17].

On a  $\mathbf{y}_N = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(N) \end{bmatrix}$  et  $\Phi_N = \begin{bmatrix} \phi(1) \\ \phi(2) \\ \vdots \\ \phi(N) \end{bmatrix}$   $\Theta_N = [\Theta_N^t \quad \Theta_N^t] y$  alors si on ajoute une mesure pour k=N+1 alors

on aura :

$$\mathbf{y}_{N+1} = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(N) \\ y(N+1) \end{bmatrix}, \text{ et } \Phi_{N+1} = \begin{bmatrix} \phi(1) \\ \phi(2) \\ \vdots \\ \phi(N) \\ \phi(N+1) \end{bmatrix}$$

D'après la méthode des moindres carrés non récursifs on a :  $\Theta_{N+1} = \underbrace{[\Theta_{N+1}^t \quad \Theta_{N+1}^t]^{-1}}_{P_{N+1}} \Theta_{N+1}^t y_{N+1}$

Grâce à cette étape on peut chercher la solution de la (N+1) ième de n'importe quelle système de données, y (N+1) valeur et on peut le calculer comme suit :

$$P_{N+1} = P_N - \frac{P_N \phi(N+1)^T \phi(N+1) P_N}{1 + \phi(N+1) P_N \phi^T(N+1)} \tag{VI.16}$$

Puis on calcule la nouvelle estimation des paramètres à l'aide de la formule suivante :

$$\Theta_{N+1} = \Theta_N + P_{N+1} \phi^T(N+1) [y_{N+1} - \phi(N+1) \Theta_N] \tag{IV.17}$$

Avec :

$\Theta_{N+1}$  : Nouvelle estimation des paramètres

$\Theta_N$  : Estimation précédente des paramètres.

$P_{N+1} \Phi^T(N+1)$  : Gain d'adaptation.

$y_{N+1}$  : Vecteur des dérivations à N+1.

$\Phi(N+1) \Theta_N$  : Vecteur des dérivations précédentes.

### IV.6.3 System Identification Toolbox™:

On peut aussi utiliser une autre méthode d'identification qui est intégrée dans le logiciel MATLAB, et cette méthode est la plus simple, il suffit de l'exécuter selon les étapes suivantes :

#### A- Chargement des Données dans l'Espace de travail MATLAB :

Pour charger les données dans proc\_data.mat on utilise la commande suivante dans la fenêtre de commande de MATLAB : « load proc\_data »

#### B - Ouvrir l'outil d'identification des systèmes :

Pour ouvrir l'Outil de l'Identification du Système, en tapant la commande suivante dans la fenêtre commande de MATLAB: « ident » puis entrée.

Une fenêtre avec Le nom de la session par défaut, Untitled, s'apparaît dans la barre du titre comme suit :

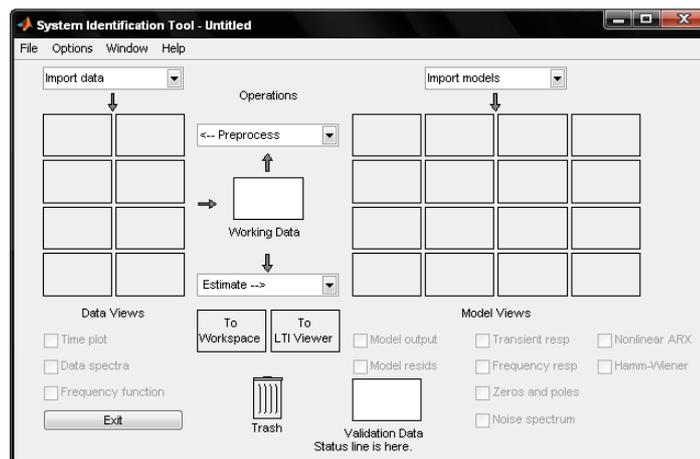
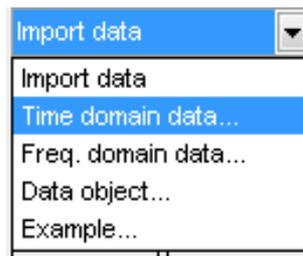


Figure IV.8 : Fenêtre d'identification des systèmes sur MATLAB.

#### C- Importer les données de WORKSPACE vers l'outil d'identification :

Pour effectuer cette opération on passe par les étapes suivantes :

- dans l'Outil de l'Identification du Système : select Import data > Time domain data, Cette action ouvre la boîte de dialogue suivante :



- Spécifier les options suivantes:

-Les noms des variables chargées dans Workspace : Les entrées (Input) et les sorties (output) du système cible.

-Nommer le dossier et tout cela on peut l'effectuer sur cette fenêtre.

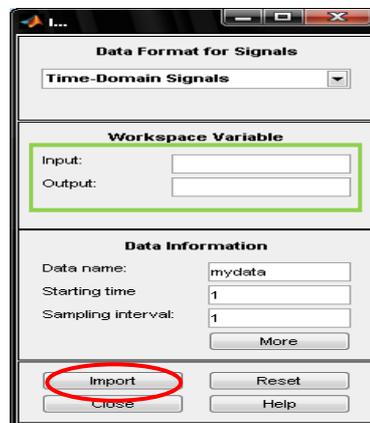


Figure IV.9 : Spécifier l'entrée et la sortie

Le clic sur Import va permettre de placer les données dans l'outil d'identification des systèmes et pour la confirmation on peut remarquer l'apparition d'une icône nommée mydata comme le montre la figure IV.10 :

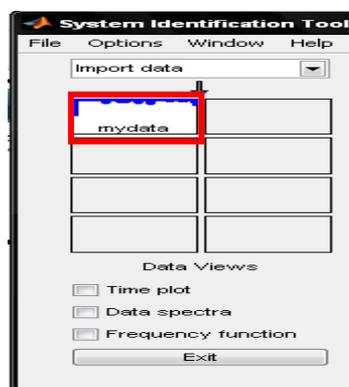


Figure IV.10 : Importation effectuée

➤ **Opération d'identification :**

Pour effectuer cette opération il suffit de cliquer sur « estimate » de la fenêtre d'identification puis choisir le mode avec lequel nous voulons identifier et cette fenêtre parait comme suit :

Par exemple on prend 'process model' (méthode des modèles), puis on choisi notre modèle 1<sup>er</sup> ordre ou bien 2<sup>ème</sup> ordre dans la place encadrée en rouge, en suite on apuis sur estimate la zone encadrée en vert,on aura l'istimation des parametres dans la zone encadrée en bleu et l'identification est terminée.

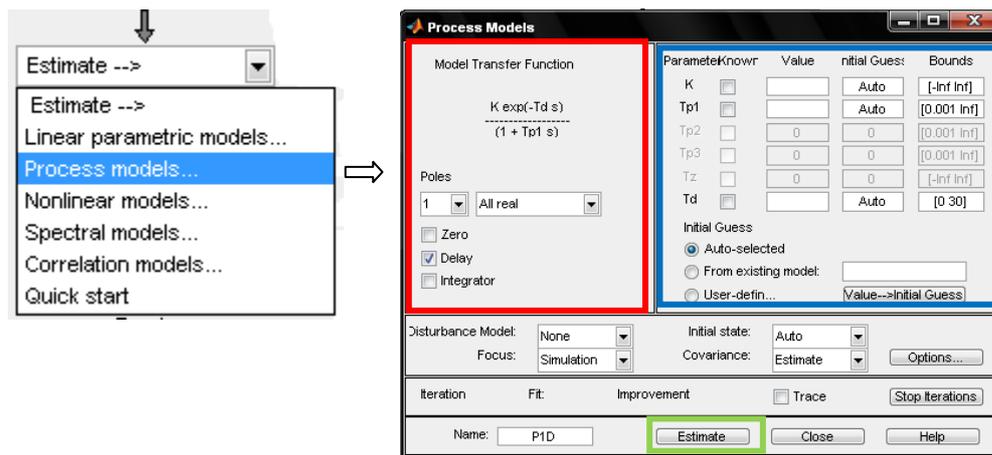


Figure IV.11 : La face d'identification avec la méthode « process Model »

## PARTIE II Partie pratique

Dans cette partie on va appliquer les notions théoriques étudiées, telle que sur le plan d'identification, la commande numérique et aussi la programmation sur LabVIEW et MATLAB.

### IV.7 Présentation du matériel utilisé :

Le matériel utilisé durant notre manipulation est celui de notre laboratoire ou sein du département d'Automatique.

Notre banc d'essai se compose de :

- **Source :** on a utilisé deux types de sources, l'une est alternative qui vaut 220V et de fréquence de 50HZ utilisée pour alimenter le moteur à courant continu et l'autre est continue qui vaut (-15V et +15V) utilisée pour alimenter le générateur d'impulsion.
- **Moteur à courant continu à excitation séparée MCC :**

Il a deux entrées E1 et E2 pour l'excitation de l'inducteur avec une tension de 120V et d'un courant de 0,19A, deux autres A1 et A2 pour l'excitation de l'induit avec une tension variable, qui va nous permettre d'agir sur la vitesse du moteur.

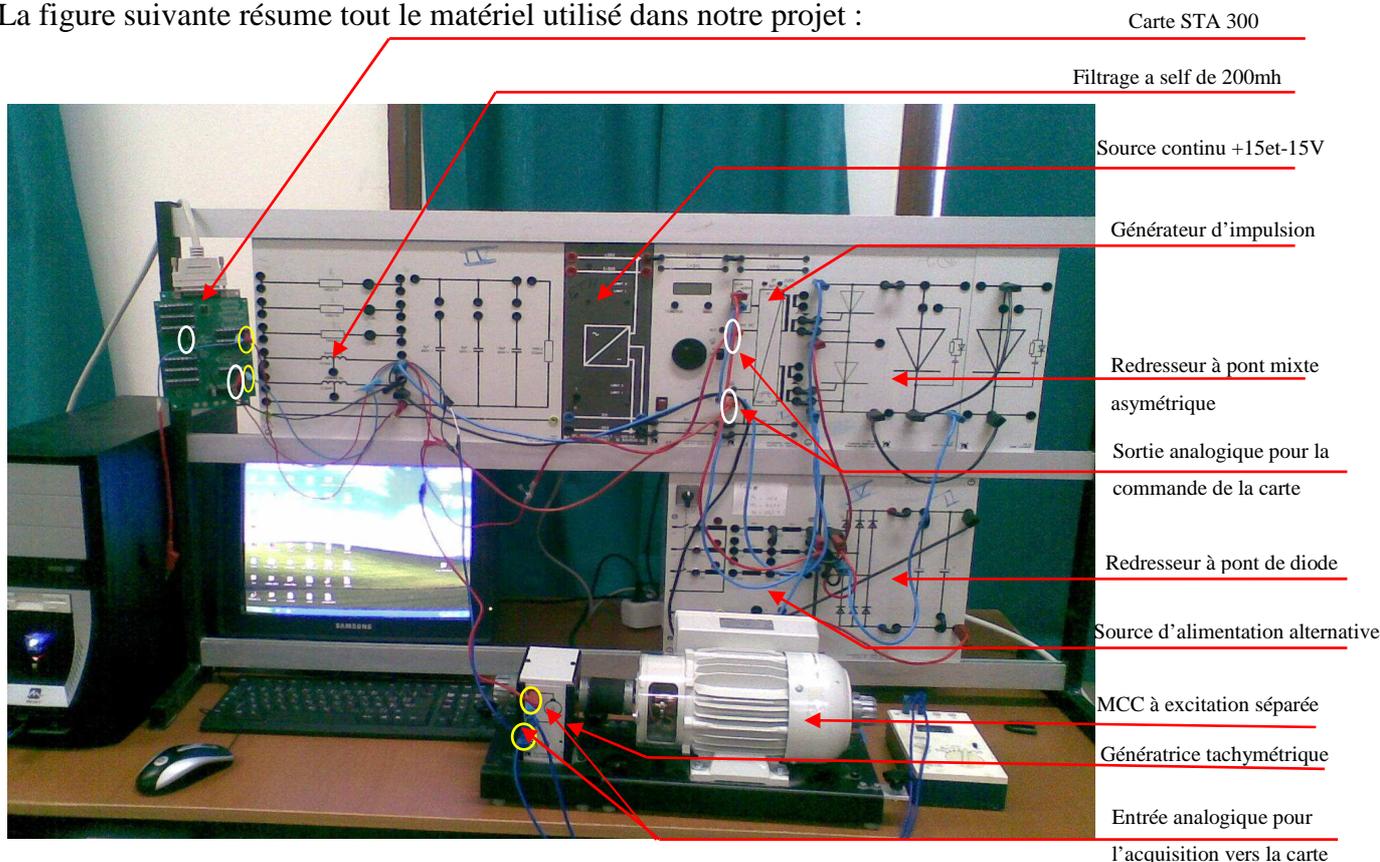
- Pour alimenter l'induit on utilise un redresseur à pont mixte asymétrique. Pour assurer le lissage du courant on utilise une self de 200mh en série avec le moteur.
- Pour l'alimentation de l'inducteur (E1 et E2), elle est effectuée avec la même source alternative redressée à l'aide d'un pont complet de diodes et filtrée par un condensateur de capacité  $C=2000\mu\text{f}$ .
- Le générateur d'impulsion : il est utilisé pour commander les thyristors du pont mixte en agissant sur les angles à retard d'amorçage.

La variation de la tension du générateur de 0 à 10v, fait varier les angles à retard d'amorçage de 0 à  $\pi$

En effet le changement sur la valeur de ces angles, va provoquer une variation de la tension moyenne continue de l'induit (A1 et A2) qui va agir sur la vitesse de rotation du moteur.

- Capteur de vitesse : C'est la génératrice tachymétrique, qui convertit l'énergie mécanique de rotation en énergie électrique (vitesse de rotation en tension) avec une sensibilité,  $1\text{v}/1000\text{tr}/\text{min}$ .
- Pour le contrôle et la communication avec le système on utilise un micro-ordinateur avec la carte KPCI 3102 intégrée dans l'unité centrale du PC et le logiciel LabVIEW qui va avec.

La figure suivante résume tout le matériel utilisé dans notre projet :



**Figure IV.12 : Banc d'essai de notre système**

**Remarque :**

Le système qu'on veut identifier dans ce projet ne contient pas seulement le MCC, mais la totalité de (redresseur +pont mixte+MCC).

**IV.8 Etude statique du système :**

Cette étude sert à déterminer la zone linéaire du système et aussi pour la commande en boucle ouverte.

En effet elle consiste à injecter une tension d'entrée variable de [0V ... 6V] à l'entrée du générateur d'impulsions puis noter les vitesses en sortie du moteur à l'aide du tachymètre.

Le tableau suivant représente la variation des vitesses  $V_s$  en fonction des tensions d'entrées  $U_e$  :

| <b>U<sub>e</sub> (volt)</b> | <b>U<sub>s</sub> (volt)</b> | <b>V<sub>s</sub> (tr/min)</b> | <b>4.3</b> | <b>0.443</b> | <b>443</b> |
|-----------------------------|-----------------------------|-------------------------------|------------|--------------|------------|
| <b>0</b>                    | 0                           | 0                             | <b>4.4</b> | 0.475        | 475        |
| <b>2.6</b>                  | 0                           | 0                             | <b>4.5</b> | 0.505        | 505        |
| <b>2.7</b>                  | 0.01                        | 0                             | <b>4.6</b> | 0.538        | 538        |
| <b>2.8</b>                  | 0.02                        | 20                            | <b>4.7</b> | 0.566        | 566        |
| <b>2.9</b>                  | 0.04                        | 40                            | <b>4.8</b> | 0.598        | 598        |
| <b>3</b>                    | 0.065                       | 65                            | <b>4.9</b> | 0.630        | 630        |
| <b>3.1</b>                  | 0.9                         | 90                            | <b>5</b>   | 0.660        | 660        |
| <b>3.2</b>                  | 0.117                       | 117                           | <b>5.1</b> | 0.691        | 691        |
| <b>3.3</b>                  | 0.143                       | 143                           | <b>5.2</b> | 0.720        | 720        |
| <b>3.4</b>                  | 0.172                       | 172                           | <b>5.3</b> | 0.748        | 748        |
| <b>3.5</b>                  | 0.200                       | 200                           | <b>5.4</b> | 0.772        | 772        |
| <b>3.6</b>                  | 0.229                       | 229                           | <b>5.5</b> | 0.797        | 797        |
| <b>3.7</b>                  | 0.257                       | 257                           | <b>5.6</b> | 0.817        | 817        |
| <b>3.8</b>                  | 0.287                       | 287                           | <b>5.7</b> | 0.842        | 842        |
| <b>3.9</b>                  | 0.314                       | 314                           | <b>5.8</b> | 0.860        | 860        |
| <b>4</b>                    | 0.347                       | 347                           | <b>5.9</b> | 0.880        | 880        |
| <b>4.1</b>                  | 0.378                       | 378                           | <b>6</b>   | 0.900        | 900        |
| <b>4.2</b>                  | 0.410                       | 410                           |            |              |            |

### IV.8.1 Courbe de caractéristique statique :

La figure suivante représente les caractéristiques statiques du moteur :

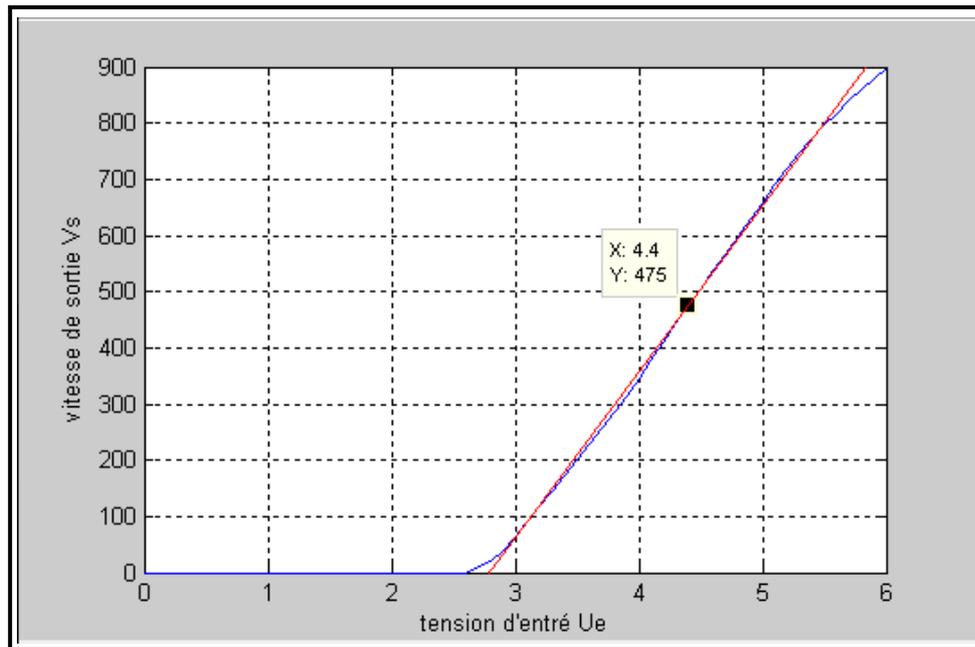


Figure IV.13: La variation de la vitesse en fonction de la tension.

#### IV.8.1.1 Analyse et interprétation de la courbe :

A travers cette courbe on peut constater que le moteur ne réagit pas aux tensions inférieures à 2.7V, par contre il agit linéairement à celle qui sont supérieures.

Explication :

- Les tensions inférieures à 2.7V, ne sont pas suffisantes pour créer le couple de force utile pour faire tourner le rotor, qu'on appelle zone morte du moteur.
- Par contre pour les tensions supérieures à 2.7V le moteur répond normalement c'est-à-dire, l'augmentation de la tension traduit l'augmentation du courant injecté à l'induit d'où le couple C du rotor augmente et il réagit selon la relation suivante :  $C = K \Phi I$ .

#### Conclusion :

✓ On peut représenter le comportement du système dans cette étude par une équation linéaire qui ne passe pas par l'origine et elle est donnée sous la forme suivante :

$$y = p1*x + p2 \quad (IV.18)$$

Avec les Coefficients :  $p1 = 288.4$        $p2 = -798.32$

✓ La zone linéaire du processus se situe alors entre 3v et 5v, dont on peut prendre le point 4.4V comme un point de fonctionnement.

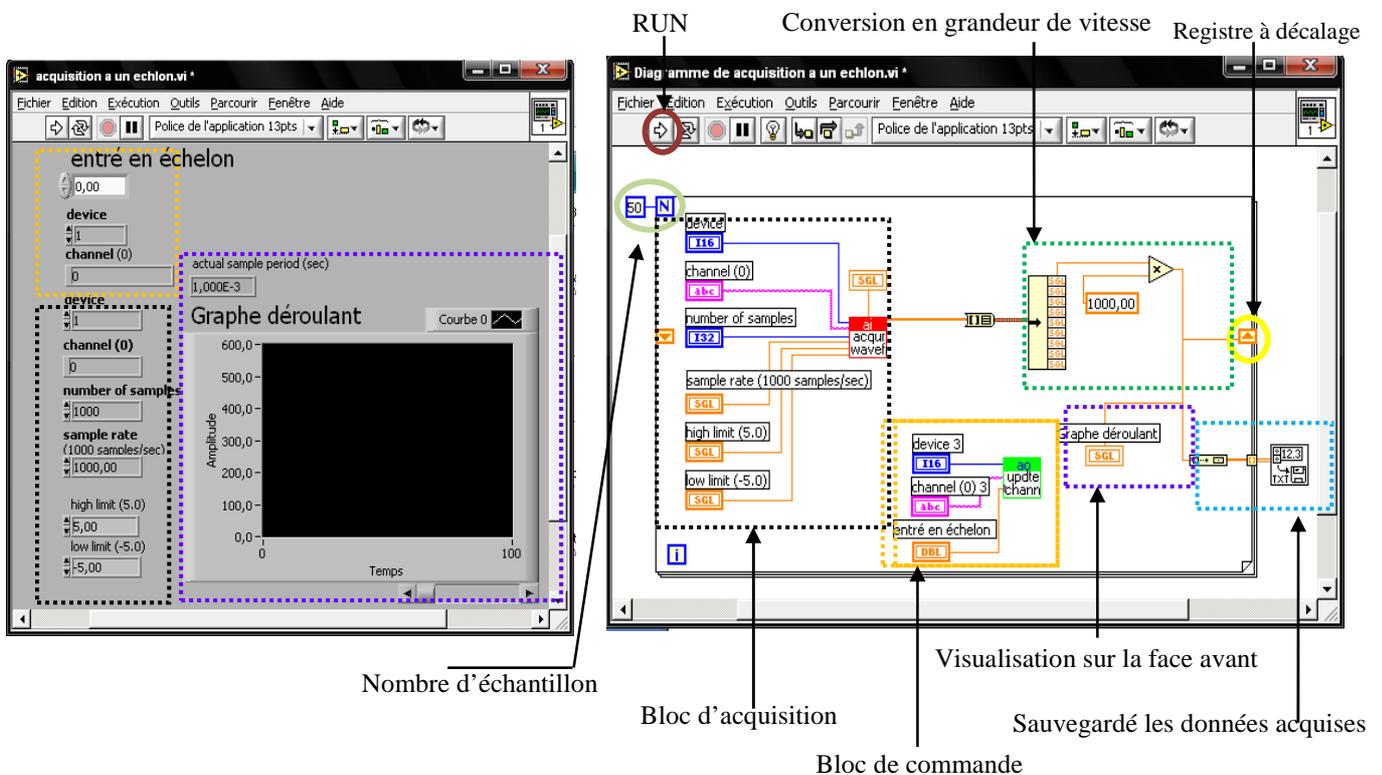
**IV.9 Etude dynamique :**

Dans cette partie l'étude a comme objectif d'identifier notre système suivant sa réponse indicielle à une entrée échelon de 4.4v comme étant un point de fonctionnement pris dans la zone linéaire, suivant un programme sur LabVIEW qui permettra l'acquisition, la commande et l'enregistrement des données.

**IV.9.1 Programmation avec logiciel LabVIEW :**

Comme on l'a vu précédemment dans la chapitre III, notre programme comporte trois blocs, un pour la commande (coloré en vert), l'autre pour l'acquisition (coloré en rouge), plus un bloc pour l'enregistrement.

Cette figure représente le diagramme (le programme utilisé) avec la face avant :



**Figure IV.14: Programme d'acquisition et de la commande**

**Tableau en cluster :**

Il convertit un tableau d'une dimension (1D) en un cluster d'éléments dans lequel ces derniers sont du même type que ceux du tableau. On utilise cette fonction lorsque nous voulons afficher des éléments du même type dans un indicateur de cluster de la face-avant.

- **Exécution du programme** : pour cela il suffit de cliquer sur le bouton RUN et on obtient les résultats sur la face avant comme suit :

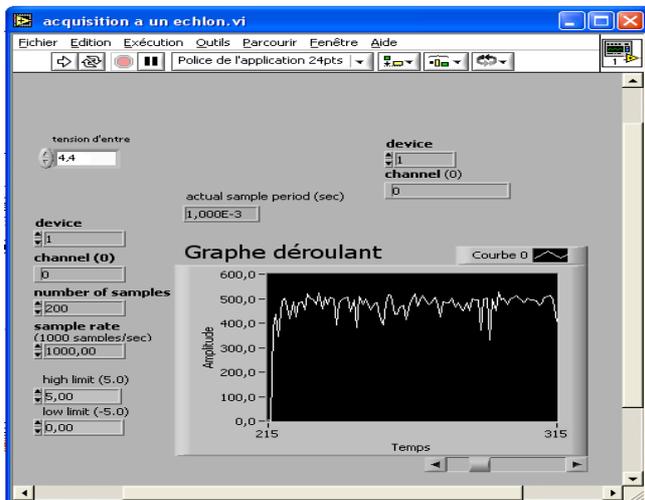


Figure IV.15: Réponse à un échelon de 4,4v.

Figure IV.16: Boite de dialogue d'enregistrement

Une boite de dialogue d'enregistrement s'apparaisse indiquant la fin d'exécution voir la figure IV.16.

Les résultats de la réponse du système à une entrée de 4.4V sont représentés dans le tableau suivant :

| Tension Ue | Vitesse Vs | 4,4 | 478,516 | 4,4 | 490,723 |
|------------|------------|-----|---------|-----|---------|
| 4,4        | 26,856     | 4,4 | 429,688 | 4,4 | 478,516 |
| 4,4        | 451,66     | 4,4 | 458,985 | 4,4 | 471,192 |
| 4,4        | 490,723    | 4,4 | 488,282 | 4,4 | 476,074 |
| 4,4        | 517,578    | 4,4 | 483,399 | 4,4 | 441,895 |
| 4,4        | 507,813    | 4,4 | 507,813 | 4,4 | 478,516 |
| 4,4        | 468,75     | 4,4 | 410,157 | 4,4 | 471,192 |
| 4,4        | 473,633    | 4,4 | 488,282 | 4,4 | 478,516 |
| 4,4        | 510,254    | 4,4 | 495,606 | 4,4 | 468,75  |
| 4,4        | 488,282    | 4,4 | 476,074 | 4,4 | 327,149 |
| 4,4        | 483,399    | 4,4 | 478,516 |     |         |
| 4,4        | 510,254    | 4,4 | 388,184 |     |         |
| 4,4        | 471,192    | 4,4 | 480,957 |     |         |
| 4,4        | 498,047    | 4,4 | 476,074 |     |         |
| 4,4        | 397,949    | 4,4 | 505,371 |     |         |
| 4,4        | 488,282    | 4,4 | 483,399 |     |         |
| 4,4        | 476,074    | 4,4 | 466,309 |     |         |
| 4,4        | 493,164    | 4,4 | 490,723 |     |         |
| 4,4        | 466,309    | 4,4 | 478,516 |     |         |
| 4,4        | 495,606    | 4,4 | 458,985 |     |         |
| 4,4        | 417,481    | 4,4 | 478,516 |     |         |

Pour bien comprendre le type et l'ordre du système on va visualiser l'allure de sa réponse à l'échelon de 4.4V, utilisant les données du tableau, qui donnent la figure suivante :

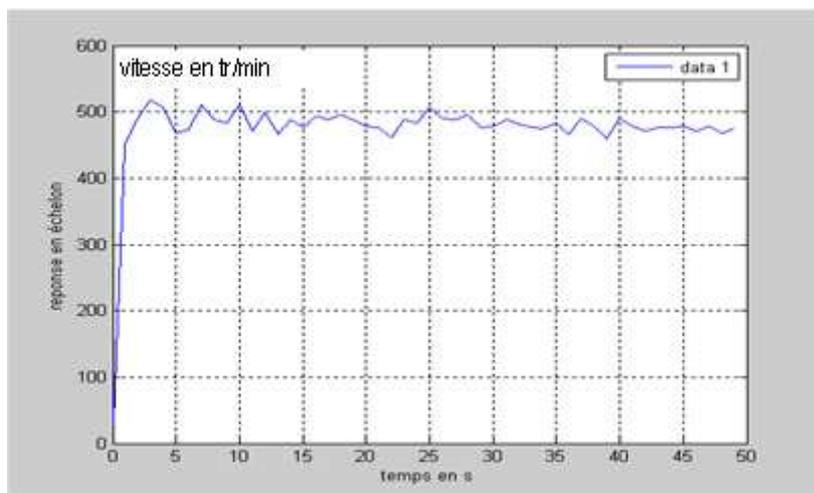


Figure IV.17: La réponse indicielle du système.

A travers cette figure on remarque que la réponse est tellement rapide, qu'elle a causé un dépassement. Et si on compare cette allure à celle étudiée dans la première partie, on peut comprendre que le système (boite noire) recherché correspond à celui du 2ème ordre de fonction de transfert donnée sous la forme :

$$G(s) = \frac{K}{(1 + \tau_1 s)(1 + \tau_2 s)} \tag{IV.19}$$

**IV.10 Identification des paramètres (K, τ1, τ2) du système:**

Pour l'identification on propose deux méthodes, la première c'est celle des moindres carrés récursives et la deuxième c'est celle de « système identification Toolbox».

**IV.10.1 Méthode des moindres carrés :**

**IV.10 .1.1 Calculer les équations aux différences :**

Comme on a pu le constater précédemment que notre système est de 2ème ordre sous fonction de transfert (VI.19) qui peut être écrite sous la forme :

$$G(s) = \frac{K}{1 + (\tau_1 + \tau_2)s + (\tau_1 \tau_2)s^2} \Rightarrow \frac{y(s)}{u(s)} = G(s) = \frac{K}{1 + (\tau_1 + \tau_2)s + (\tau_1 \tau_2)s^2}$$

On pose  $\alpha = \tau_1 + \tau_2$  et  $\beta = \tau_1 \tau_2$  (IV.20)

$$\Leftrightarrow y(s)[1 + \alpha s + \beta s^2] = K u(s) \Leftrightarrow y(s) + y(s)\alpha s + y(s)\beta s^2 = K u(s)$$

Passage dans le domaine temporel:

$$y(s) + y(s)\alpha s + y(s)\beta s^2 = K u(s) \Leftrightarrow y(t) + \alpha \frac{dy(t)}{dt} + \beta \frac{d^2y(t)}{dt^2} = K u(t) \tag{IV.21}$$

Passage dans le domaine discret:

Pour effectuer ce passage on peut appliquer la transformée en z, ou bien le théorème des développements limités (DL) qui est sous la forme d'équation suivante :

$$f(x + nx) = f(x) + nxf'(x) + \frac{nx^2}{2!} f''(x) \quad (\text{IV.22})$$

$$f(x - nx) = f(x) - nxf'(x) + \frac{nx^2}{2!} f''(x) \quad (\text{IV.23})$$

Calculer la  $f''(x)$  :

On fait la somme de l'équation (VI.21+VI.22), on obtient :

$$f''(x) = \frac{f(x + nx) - 2f(x) + f(x - nx)}{nx^2} \quad (\text{IV.24})$$

**Application sur l'équation des DL (VI.20) :**

$$\frac{dy(t)}{dt} = y'(k) = \frac{y(k+1) - 2y(k) + y(k-1)}{2T_e} \quad (\text{IV.25})$$

**Et**

$$\frac{d^2y(t)}{dt^2} = y''(k) = \frac{y(k+1) - 2y(k) + y(k-1)}{T_e^2} \quad (\text{IV.26})$$

On remplace dans l'équation (VI.20) :

$$y(k) + \alpha \left[ \frac{y(k+1) - 2y(k) + y(k-1)}{2T_e} \right] + \beta \left[ \frac{y(k+1) - 2y(k) + y(k-1)}{T_e^2} \right] = Ku(k) \text{ avec } T_e = 1s$$

Après simplification on obtient l'équation aux différences suivante :

$$y(k+1) = Ay(k) + By(k+1) + Cu(k) \quad (\text{IV.27})$$

$$\text{Avec : } A = \frac{4\beta-2}{2\beta+\alpha} \quad \text{et} \quad B = \frac{\alpha-2\beta}{\alpha+2\beta} \quad \text{et} \quad C = \frac{2K}{\alpha+2\beta} \quad (\text{IV.28})$$

Déclaration des 6 premières valeurs :

Pour :

$$k = 1 \implies y(2) = Ay(1) + By(0) + Cu(1)$$

$$k = 2 \implies y(3) = Ay(2) + By(1) + Cu(2)$$

$$k = 3 \implies y(4) = Ay(3) + By(2) + Cu(3)$$

$$k = 4 \implies y(5) = Ay(4) + By(3) + Cu(4)$$

$$k = 5 \implies y(6) = Ay(5) + By(4) + Cu(5)$$

$$k = 6 \implies y(7) = Ay(6) + By(5) + Cu(6)$$

On peut écrire ces dernières équations sous forme matricielle  $y_6 = \Phi_6 \theta_6$  comme suit :

$$\underbrace{\begin{bmatrix} y(2) \\ y(3) \\ y(4) \\ y(5) \\ y(6) \\ y(7) \end{bmatrix}}_{\mathbf{y}_6} = \underbrace{\begin{bmatrix} y(0) & y(1) & u(1) \\ y(1) & y(2) & u(2) \\ y(2) & y(3) & u(3) \\ y(3) & y(4) & u(4) \\ y(4) & y(5) & u(5) \\ y(5) & y(6) & u(6) \end{bmatrix}}_{\Phi_6} \underbrace{\begin{bmatrix} B \\ A \\ C \end{bmatrix}}_{\theta_6}$$

La  $\theta_{50}^{i\text{ème}}$  valeur est la solution de cette méthode, alors on fait appel à MATLAB pour le calcul de cette valeur selon le programme script suivant :

```

clear all
clc
%déclaration des valeurs acquises.
clc
y=[26.856
451.66
490.723
517.578
507.813
468.75
473.633
510.254
488.282
483.399
510.254
471.192
498.047
397.949
488.282
476.074
493.164
466.309
495.606
417.481
478.516
429.688
458.985
488.282
483.399
507.813
410.157
488.282
495.606
476.074
478.516
388.184
480.957
476.074
505.371
483.399
466.309
490.723
478.516
458.985
490.723
478.516
471.192

476.074
441.895
478.516
471.192
478.516
468.75
327.149]

%déclaration de l'échelon.
u=ones(1,length(y))*4.4
%déclaration des 6 premières valeurs.
y6= y(2:7);

y6 = [y(3) y(4) y(5) y(6) y(7) y(8)]';
phi6 = [y(2) y(1) u(1);
        y(3) y(2) u(2);
        y(4) y(3) u(3);
        y(5) y(4) u(4);
        y(6) y(5) u(5);
        y(7) y(6) u(6)]

%chercher la 50 eme valeurs.
theta6 = inv(phi6'*phi6)*phi6'*y6;

p6 = inv(phi6'*phi6)

for i=9:50
    phi = [y(i) 4.4 4.4];
    P = p6-
    (p6*phi'*phi*p6)/(1+phi*p6*phi');

    theta = theta6 + P*phi'*(y(i)-
    phi*theta6);
    theta6= theta
    p6 = P;
end
T=1;
w=theta(1)*y'+theta(2)*y'+theta(3)*u;
t=1:50;
subplot(3,1,1);plot(t,y,'r')
title('systeme')
grid
subplot(3,1,2);plot(t,w,'g')
title('systeme identifie')
grid
subplot(3,1,3);plot(t,y,'r',t,w,'g')
title('comparaison')
grid

```

L'exécution du programme nous permet d'avoir les résultats, graphique et numérique suivants :

Graphique

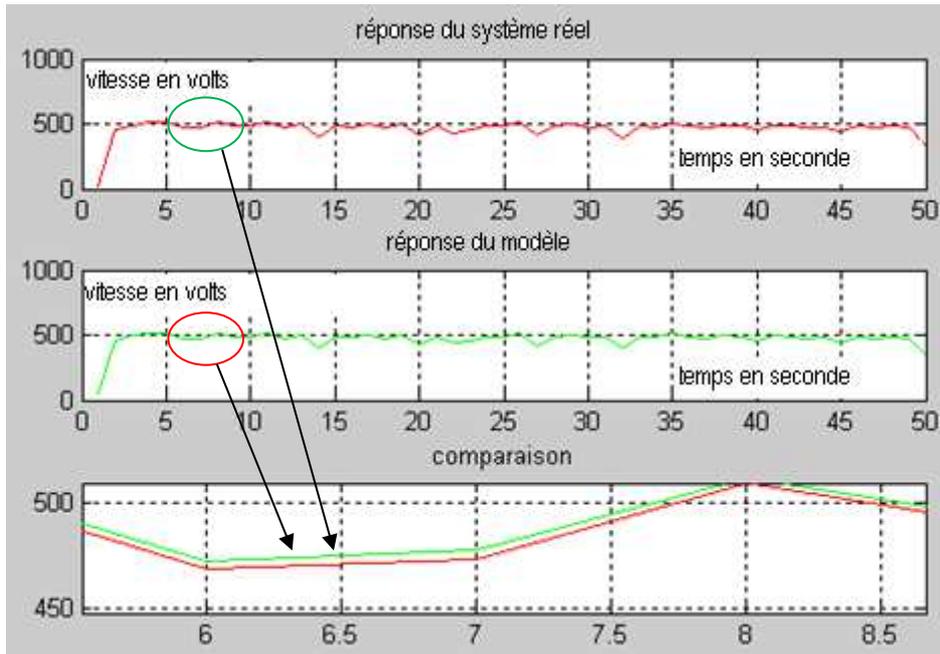


Figure IV.18: Comparaison de la réponse du modèle à celle du système réel.

Numérique :  $A = 0,0772 = \frac{4\beta - 2}{2\beta + \alpha}$  ,  $B = 0,2145 = \frac{\alpha - 2\beta}{\alpha + 2\beta}$  ,  $C = 83,6661 = \frac{2K}{\alpha + 2\beta}$

Après calcul, on a trouvé que:

$\beta=0,55$   $\alpha=1,66$  et  $K=115,45$  avec la relation (IV.28)

$\tau_1 = 1,2s$  et  $\tau_2 = 0,5s$  avec la relation (IV.20)

**Interprétation du graphe :**

Dans le graphe de la figure IV.18 on peut distinguer 3 courbes :

- La première (en rouge) : représente la réponse du système réel à l'échelon en tension de 4.4v.
- La deuxième(en vert) : représente la réponse pour la même entrée (4.4v) du modèle trouvé par la méthode des moindres carrés récursives suivant l'équation aux différences :

$$y(k+1) = 0,0772 y(k) + 0,2145 y(k-1) + 83,66 u(k). \tag{IV.29}$$

-la troisième présente les deux réponses au même temps dans le but de faire une comparaison entre le comportement du système réel et celui du modèle trouvé.

En effet dans cette dernière courbe on peut remarquer un petit écart entre les deux courbes ce qui signifie que le modèle estimé se rapproche de notre système réel d'où la fonction de transfert de ce modèle sera sous la forme suivante :

$$G_1(s) = \frac{115,45}{1 + 1,75s + 0,6s^2} \tag{IV.30}$$

**IV.10.2 Méthode « System Identification Toolbox™ » :**

C'est un utile dans le quelle des programmes de haut niveaux son introduits, et parmi eux on trouve les moindres carrés qui sont plus sophistiqués que celle déjà vue précédemment.

Le but d'utilisation de cette méthode est d'effectuer une comparaison et de voire la quelle des deux méthodes donne de bon résultat et aussi de montrer la facilité d'identification.

**IV.10.2.1 Etapes d'identification :**

Les étapes d'identification de cette méthode sont déjà présentées dans la partie I théorique de ce chapitre, et le résumé de ces étapes est comme suit :

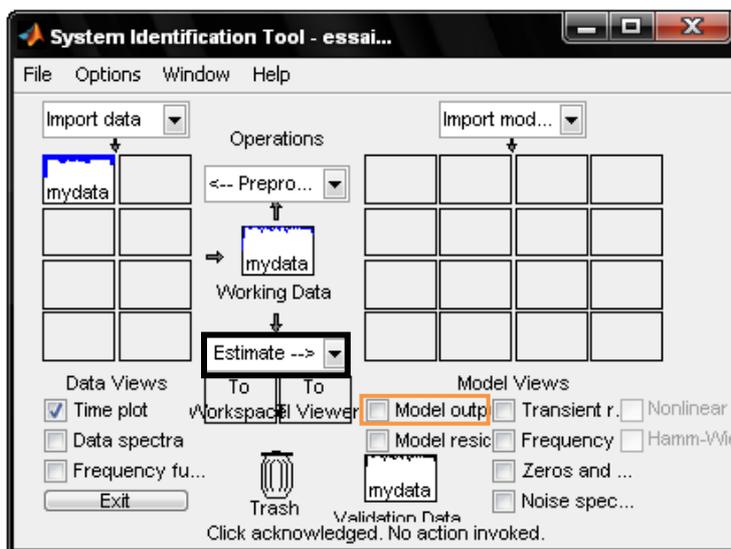
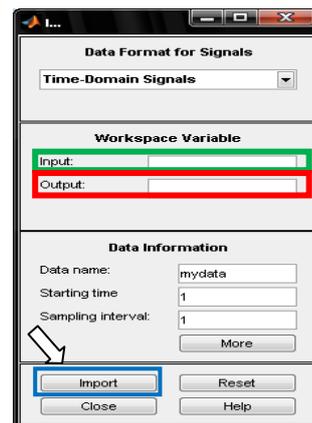
- Chargement des données dans l'espace de travail MATLAB.
- Ouvrir l'outil d'identification des systèmes.
- Importer les données de WORKSPACE vers l'outil d'identification,

Comme suit:

La case input : Ue (en vert qui correspond, à l'échelon d'entrée de 4.4v)

La case output : Vs (en rouge qui correspond, à la réponse en vitesse)

Puis, appuyer sur « import » la fenêtre suivante s'apparait :



**Figure IV.19: Fenêtre de système identification Toolbox.**

La dernière étape est d'appuyer sur « Estilmate » (cadre en noire), qui va nous permettre de choisir le type de model avec le quel on veut identifier. Par exemple dans notre cas on a choisi « process models » comme le montre la figure suivante :

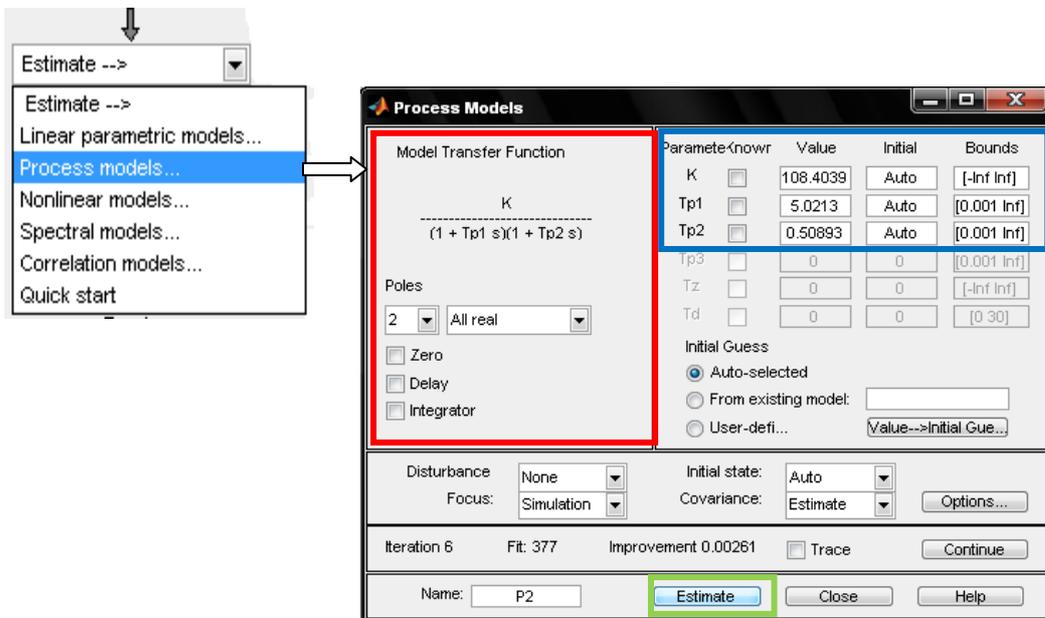


Figure IV.20: L'espace d'identification avec « process model »

**Remarque :**

- 1/ il faut effectuer le choix du modèle de la fonction du transfert recherchée : dans notre cas c'est le modèle du deuxième ordre qui figure dans le cadre rouge de la figure IV.20.
- 2/ appuyer sur le bouton « Estimate » qui figure dans le cadre vert de la figure IV.20.
- 3/les résultats des paramètres identifiés sont figurés dans le cadre bleu qui sont :

$$K=108,4039 \text{ et } \tau_1 = 5,0213s \text{ et } \tau_2 = 0,50893s.$$

Donc la fonction de transfert sera :

$$G_2(s) = \frac{108,4039}{1 + 5,53S + 2,51S^2} \tag{IV.31}$$

- 4/pour voir notre modèle de cette fonction il suffit d'appuyer sur « model output » de la figure IV.19 :

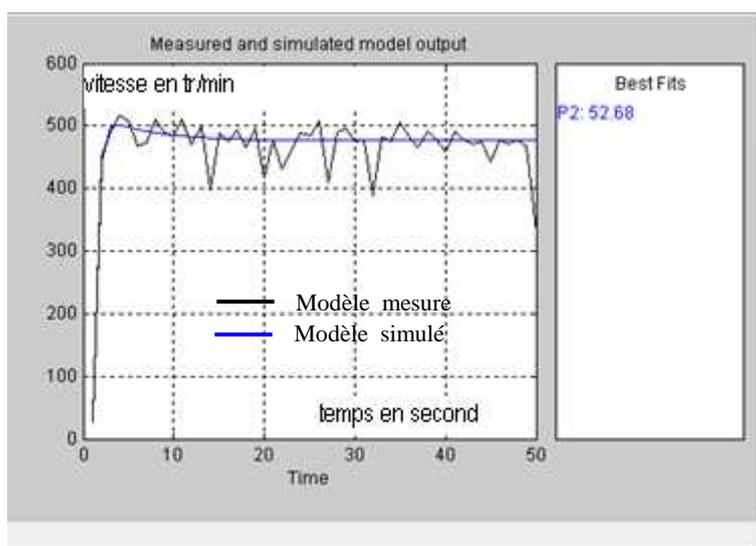


Figure IV.21: Le modèle mesuré et le modèle simulé.

**A- Comparaison des deux méthodes :**

Pour le choix du meilleur modèle parmi les deux obtenus, on va effectuer une comparaison pour voir lequel reproduit la meilleure réponse (vitesse) à une tension d'entrée de 4.4V, par deux méthodes :

**A-1 La simulation des FT des deux modèles sous simulink :**

D'après les caractéristiques statiques du système, une entrée de 4.4V donne 475tr/min .

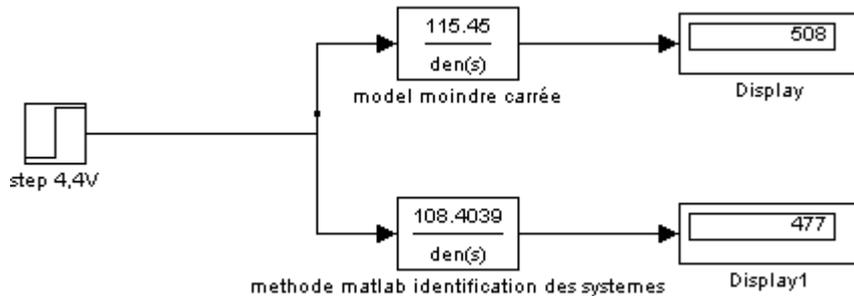


Figure IV.22 : Comparaison sur simulink des deux modèles trouvés.

**A-2 Avec le programme du Système identification Toolbox™ :**

Cette fois on fait le chemin inverse c'est-à-dire, on fixe la forme du modèle, puis les paramètres correspondants à la FT  $G1(s)$ (voir le cadre bleu de la figure IV.20) et on appuie sur « Estimate ». Pour  $G2(s)$  déjà estimée reste-t-elle qu'elle est. On visualise les deux courbes au même temps et la meilleure réponse est indiquée dans la case « best fits », par ordre de classement où la première correspond à la meilleure réponse.

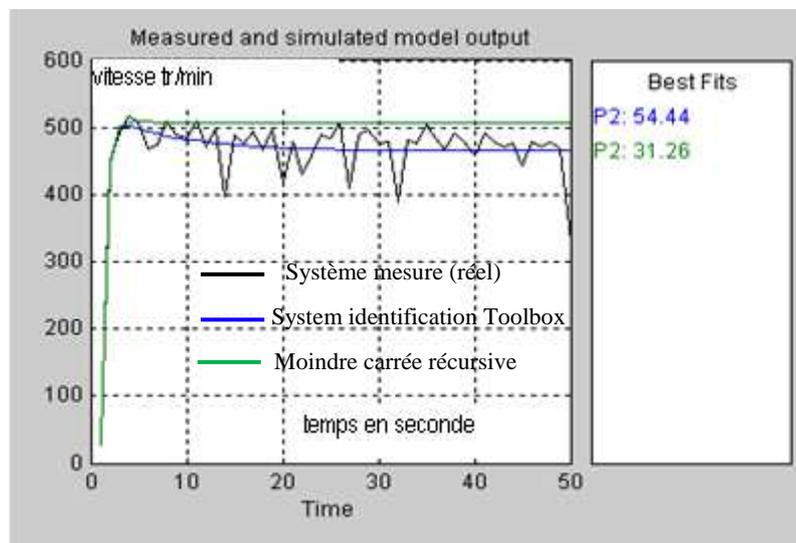


Figure IV.23 : Comparaison sur 'System Identification Toolbox™'.

**Conclusion :**

- Les résultats obtenus par les deux méthodes de comparaison confirme, en premier lieu la précision des résultats obtenus par System Identification Toolbox™, cela signifie en deuxième lieu que le modèle qui reproduit en mieux la réponse réelle du système est celui de  $G_2(s)$ .

$$G_2(s) = \frac{108,4039}{1 + 5,53s + 2,51s^2} \quad (IV.32)$$

- Notre modèle identifié  $G_2(s)$  donne des résultats rapprochés des réels au voisinage de la consigne 4,4V, mais dès qu'on s'éloigne de ce point les vitesses de la sortie s'éloignent des valeurs réelles.

Cela parce que notre système en réalité est non linéaire, à cause de la présence de la zone morte c'est-à-dire notre moteur ne démarre qu'à partir de 2,7V.

**B- Estimation d'un modèle plus proche du réel pour toutes les entrées :**

On a effectué plusieurs essais d'identifications pour différentes entrées échelon en tension, on a pu constater que parmi les trois paramètres (K,  $\tau_1$ ,  $\tau_2$ ), il y'a que le gain qui change à chacune des entrées.

La caractéristique statique du système est donnée par :  $V_s = 288.4 U_e - 798.32$ .

Dans le but d'aboutir à des  $V_s$  proches de la réalité pour toutes tensions d'entrée  $U_e$  le gain statique du système doit être égale à 288.4 faisant référence à la pente de la partie linéaire, mais cela engendre un changement au niveau de l'entrée qui va se transformer en  $U'_e = U_e - 2,7$ , Pour que toutes entrées inférieures à 2.7 donnent une valeur nulle. Ce changement va bien décrire notre système réel et répondra pour n'importe quelle valeur d'entrée située dans la zone linéaire.

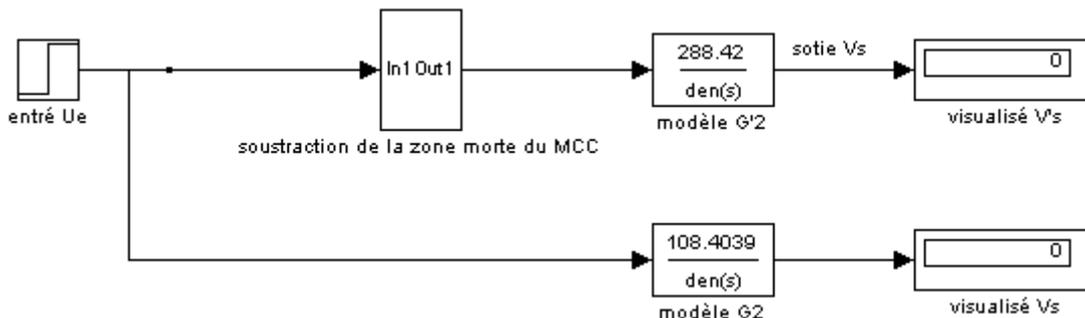
Notre nouveau model sera sous la forme suivante :

$$V_s = \frac{288,4}{1+5,53s+2,51s^2} U'_e \quad (IV.33)$$

**C Vérification du modèle sur simulink :**

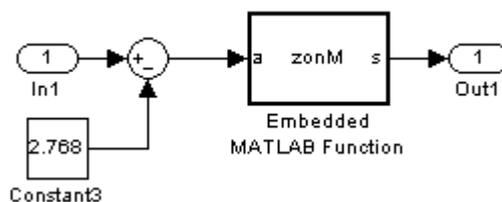
Dans cette partie on a effectué une simulation sous simulink pour comparer le model  $G_2$  et celui de  $G'_2$  au niveau des vitesses  $V_s$  de sortie.

Le schéma de simulation utilisé sous simulink est le suivant :



**Figure IV.24: Simulation et comparaison de  $G_2$  et  $G'_2$ .**

**Avec :** le bloc « soustraction de la zone morte du MCC » est constitué de :



Tandis que le bloc « MATLAB Function » est un programme utilisé pour que « s=0 » si « a<=0 » ce cas correspond à des tensions inférieures à 2,7V.

« s=a » si « a>0 » ce cas correspond à des tensions supérieur à 2.7V et le programme est le suivant :

```
function s=zonM(a)
if a<=0
    s=0;
else
    s=a;
end;
```

➤ Les résultats suivants sont obtenus par simulation de quelques entrées en échelon :

| Tension Ue (volt) | Système réel | Vs du G2(s) tr/min | V's du dG'2(s) tr/min |
|-------------------|--------------|--------------------|-----------------------|
| 2                 | 0            | 216.8              | 0                     |
| 3.4               | 172          | 368.6              | 182.3                 |
| 4                 | 347          | 433.6              | 355.3                 |
| 4.4               | 475          | 477                | 470.7                 |
| 5.3               | 748          | 574.6              | 730.3                 |
| 5.8               | 860          | 628.8              | 874.5                 |
| 6                 | 900          | 650.4              | 932.2                 |

**Remarque :**

A travers ce tableau on comprend que le model  $G_2$  est bon seulement au point 4.4V, mais ailleurs il ne correspond absolument pas au système réel, par contre les réponses V's du modèle  $G'_2$  son les plus proche à la réalité par rapport au modèle  $G_2$ .

**Conclusion :**

Pour qu'on puisse faire une régulation en vitesse en boucle fermé on prend le modèle qui a la fonction du transfert suivante :

$$G'_2(s) = \frac{288,4}{1+5,53s+2,51s^2} \quad (\text{IV.34})$$

**IV.11 PARTIE REGULATION :**

La régulation de notre procédé consiste à atteindre la consigne désirée on agissant sur l'écart par la commande à action proportionnelle en suite par la commande à action proportionnelle et intégrale.

**IV.11.1 Action proportionnelle :**

Elle consiste à donner une commande proportionnelle à l'écart  $\mathcal{E} = y_c - y$ .

Sa fonction de transfert est  $c(p) = \frac{U(p)}{\mathcal{E}(p)} = k_c$

**Son rôle:**

-elle permet de réduire les écarts statiques par augmentation  $k_c$  sans toutes fois les annuler (amélioration de la précision statique).

-elle permet d'augmenter aussi la rapidité en diminuant la marge de phase. Cependant l'augmentation excessive de  $k_c$  risque de produire une marge de phase trop faible avec apparition des oscillations de fortes amplitudes et risque de déstabiliser l'asservissement (dilemme précision statique-stabilité).

**L'application1:** cherchant  $k_c$  pour une réduction de l'écart statique en premier lieu à  $\mathcal{E}(t) = 0,012V$  pour une consigne de 400tr/min correspondante à une tension de 4,14V.

Notre fonction de transfert en boucle ouverte est :  $G(p) = \frac{288,4}{1+5,52s+2,51s^2}$

Donc  $T_c(p) = C(p)G(p) = \frac{K_c \cdot 0,2884}{1+5,52s+2,51s^2}$  fonction de transfert en BO de l'asservissement corrigé.

Or  $\mathcal{E}(p) = \frac{1}{1+T_c(0)} = \frac{1}{1+k_c \cdot 0,2884} = 0,012$  donc  $k_c = 277,39$

IV.11.2 Simulation sous MATLAB :

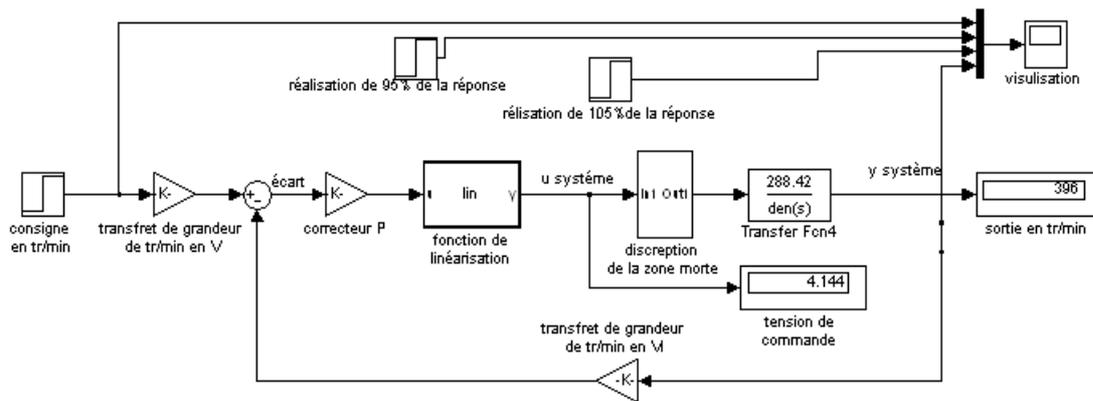


Figure IV.25 : Schéma de simulation de la régulation à correcteur P.

L'introduction du bloc « fonction de linéarisation » a permis le décalage du point de départ 2,7 de la zone linéaire ([2,7V ...5V]) vers l'origine (0V) dans le but de réduire la non linéarité du système réel et une sécurité ( $y=8v$  si  $y>8v$  et  $y=0$  si  $y \leq 0$ ) pour ne pas détériorer la carte qui délivre au maximum 10v et pour ne pas délivrer des tensions négatives qui engendre une rotation inverse du moteur et le programme de cette fonction est le suivant :

```
function y = lin(u)
%#eml
y = u+2.768;
if y > 8
    y=8;
else
    if y <= 0
        y=0
    else
        y=y
    end
end
```

L'exécution sous simulink de ce programme a aboutie à la courbe de la figure suivante, qui est une réponse à la consigne en vitesse de 400 tr/min.

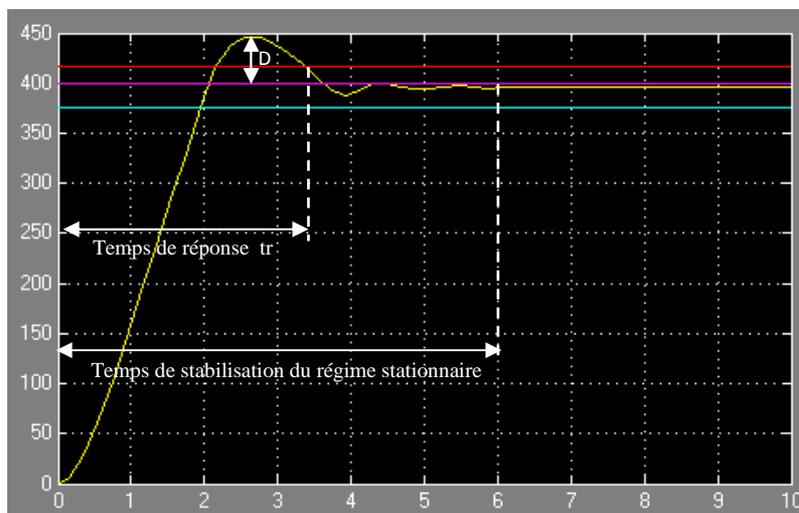


Figure IV.26 : Réponse du a correction P pour un échelon de 400tr/min.

➤ **Observation**

A partir de cette figure on remarque trois comportements, le premier c'est le dépassement qui vaut 47,25tr/min qui correspond à 11,7%

Le deuxième c'est celui de la valeur final de la réponse qui vaut 396tr/min au niveau du régime permanent et un temps de réponse 3.3s et un temps de montée 1.3s au niveau du régime transitoire.

Le troisième c'est au niveau de temps de stabilisation du régime stationnaire qui vaut 5s.

➤ **Explication :**

Le dépassement fait référence à une réponse pseudopériodique et ce la revient au coefficient d'amortissement  $h$  qui est entre  $0,45 < h < 0,7$ , en effet après calcul on s'appuyant sur la formule :

$$D\% = 100 \cdot e^{\frac{-\pi h}{\sqrt{1-h^2}}} \quad (IV.35)$$

On a obtenu un  $h=0,5$ .

Le dépassement  $D$  est causé par le gain de régulateur, en effet l'erreur imposée est tellement petite que ce gain est grand, d'où ce dépassement de 11,7%.

**Vérification de la stabilité :**

La stabilité d'un système selon le critère de Nyquist dit: dans le lieu de Nyquist, si on parcourt le sens croissant des pulsations ( $\omega$ ) d'un système. S'il dépasse le point critique  $(-1,0)$  vers les valeurs négatives on parle d'un système instable, s'il passe juste par ce point il est à la limite de stabilité sinon ailleurs le système est stable.

On effectuera cette vérification par une étude harmonique du système à correction proportionnelle, on utilisant un programme sous script MATLAB qui est le suivant:

```
%déclaration de la fonction du transfert en BF a correction proportionnelle.
num=79.99
den=[2.51 5.52 80.99]
G= tf(num,den)
figure, nyquist(G)
grid
```

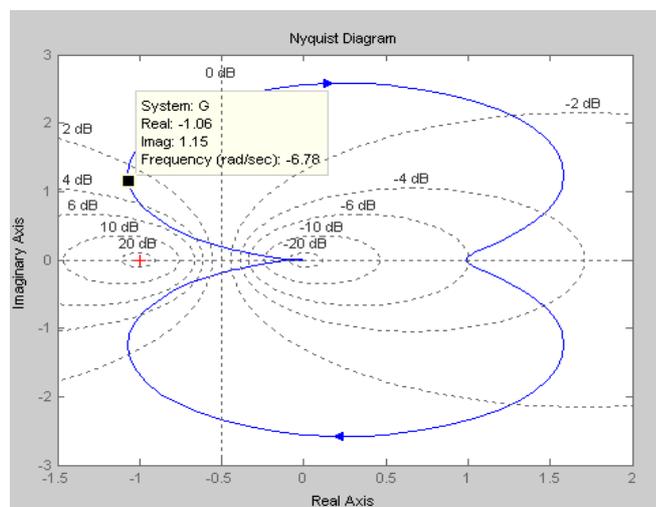


Figure IV.27 : Lieu de Nyquist pour le système a correction P à gain  $K_c=277,39$ .

A travers cette figure on remarque que le lieu de Nyquist dépasse légèrement le point critique (dépassement négligeable de 0,06) cela signifie que le système est juste à la limite de la stabilité avec  $k_c = 277,39$ .

- **L'application2** : pour la même consigne de 400tr/min, on impose une erreur statique de 10% ce qui va influencer sur le gain qui aura une valeur  $k_c = 31,2$ .
- **Observation** :

Un dépassement nul avec une vitesse de sortie loin de la consigne qui vaut 360tr/min

**Vérification de la stabilité** : la stabilité est vérifiée par un programme comme le cas précédent:

```
%declaration de la fonction du transfert a correction proportionnelle.
```

```
num=31.2
```

```
den=[2.51 5.52 32.2]
```

```
G= tf(num,den)
```

```
figure, nyquist(G)
```

```
grid
```

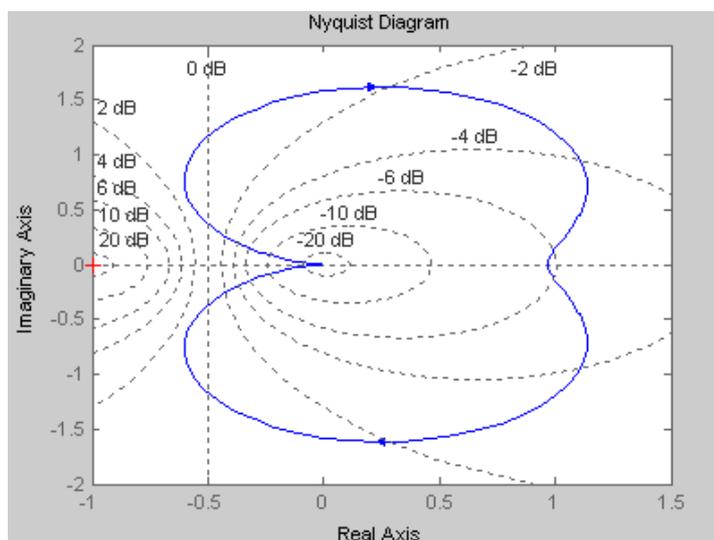


Figure IV.28 : Lieu de Nyquist pour le système a correction P à gain  $K_c=31.2$ .

- A travers cette figure on remarque que le lieu de Nyquist ne dépasse pas le point critique (vers les valeurs négatives) ce la signifie que le système avec  $k_c = 31.2$  est stable mais il est de précision acceptable.

#### Conclusion1 :

Le correcteur proportionnel est insuffisant pour avoir un système stable, précis et rapide au même moment en effet si on gagne la précision et la rapidité on perd la stabilité et si on gagne la stabilité on perd la précision et la rapidité.

#### IV.11.4 Action proportionnelle intégrale :

Elle combine l'action proportionnelle et action integrale.ci-dessous on adopte la structure dans laquelle les deux actions sont mises en parallèles avec l'action proportionnelle qui agit sur l'action intégrale. Toutefois, on peut envisager d'autres structures.

Sa fonction de transfert est :  $c(p) = k_c [1 + \frac{1}{T_i p}]$

Avec  $k_c$  : gain statique et  $T_i$  : est la constante d'intégration

**Son rôle:**

-du fait qu'il introduit un zéro à l'origine, il annule l'écart statique de position pour un système de classe zéro, d'annuler l'écart statique de vitesse pour un système de classe une ainsi de suite...

➤ **L'application :**

Le calcul du correcteur revient à trouver les valeurs des paramètres  $k_c$  et  $T_i$  afin de satisfaire les performances requises. Considérons l'asservissement précédent dont la fonction de transfert en boucle ouverte est donnée par :

$$G(p) = \frac{288,4}{1+5,52s+2,51s^2}$$

On voudrait déterminer un correcteur PI afin de minimiser l'erreur statique et une marge de phase de  $45^\circ$  pour assurer la stabilité.

La fonction de transfert en BO du système asservi corrigé s'écrit :  $T_c(s) = \frac{k_c(1+T_i s)}{T_i s} \frac{0,2884}{(1+5,02s)(1+0,5s)}$

➤ **Solution par compensation de pôles pour le calcul de la constante d'intégration  $T_i$  :**

La fonction  $G(p)$  possède deux constantes de temps  $\tau_1 = 5,02s$  et  $\tau_2 = 0,5s$  correspondantes aux pôles  $-50,2$  et  $-5$  respectivement.  $\tau_1$  est la constante de temps dominante au correspondant au pôle  $(-50,2)$  le plus proche de l'axe imaginaire.

On choisit  $T_i = 5,02s$  pour que le pôle dominant se simplifie, et avec ce choix nous aurons :

$$T_c(s) = \frac{k_c}{5,02s} \frac{0,2884}{(1+0,5s)}$$

Reste le calcul de  $k_c$  pour avoir une marge de phase  $\Delta\varphi=45^\circ$  dans le but d'assurer la stabilité.

La phase de la réponse en fréquence  $T_c(j\omega)$  est égale à  $\varphi_c(\omega) = -\frac{\pi}{2} - \arctg(0,5\omega)$ . la marge de phase du système corrigé est (en degré)  $\Delta\varphi_c = 180^\circ + \varphi_c(\omega_0)$  où  $\omega_0$  est la pulsation de passage à 0dB, c'est-à-dire vérifiant la relation  $\|T_c(j\omega_0)\| = 1$ .

Pour avoir une marge de phase de  $45^\circ$ , il faut que  $\varphi_c(\omega_0) = -\frac{\pi}{2} - \arctg(0,5\omega_0) = -135^\circ$ , on aura donc

$\text{Arctg}(0,5\omega_0)=45^\circ$  d'où  $\omega_0 = 2\text{rad/s}$ . De la relation  $\|T_c(j\omega_0)\| = \frac{k_c}{\omega_0\sqrt{1+0,25\omega_0^2}} = 1$ , on tire la valeur de

correcteur proportionnel permettant d'obtenir une marge de phase de  $45^\circ$  qui est  $k_c = 49,23$

#### IV.11.5 Simulation sous MATLAB :

Le schéma de simulation de la régulation PI dans simulink est représenté comme suit :

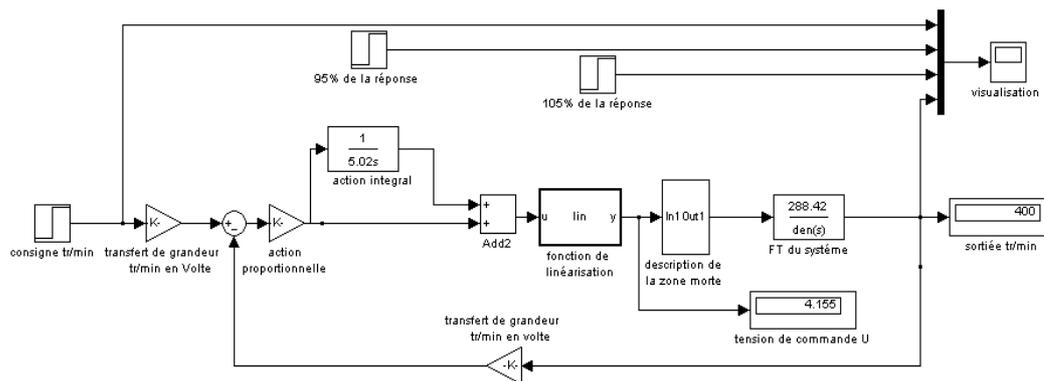


Figure IV.29 : Schéma de simulation d'une réguler a correcteur PI.

L'exécution sous simulink de ce programme a aboutie à la courbe de la figure suivante, qui est une réponse d'une consigne vitesse de 400 tr/min.

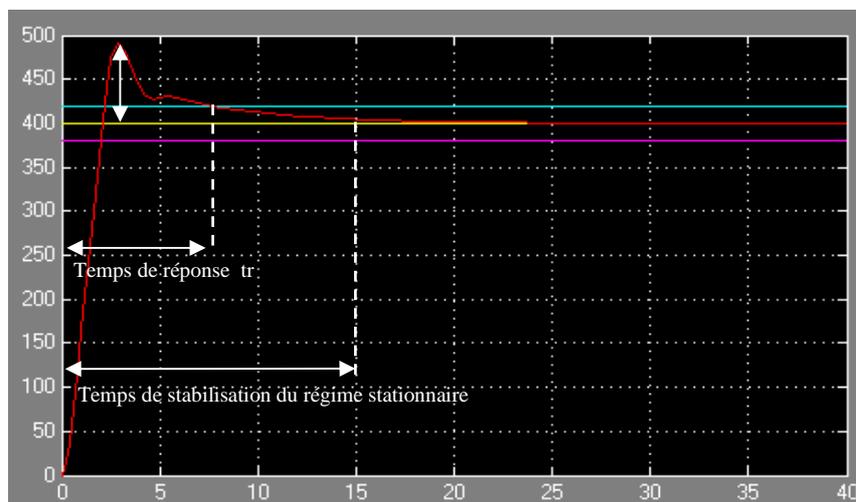


Figure IV.30 : Réponse du système à correcteur PI pour un échelon de 400tr/min.

➤ **Observation**

A partir de cette figure on remarque trois comportements, le premier c'est le dépassement qui vaut 85tr/min qui correspond à 21,25%.

Le deuxième c'est celui de la valeur final de la réponse qui vaut 400tr/min au niveau du régime permanent et un temps de réponse de 7.5s et un temps de montée  $t_m$  de 1.4s au niveau du régime transitoire.

Un temps d'établissement du régime stationnaire à la vitesse désirée de 15s.

➤ **Explication :**

L'ajout de l'action intégrale a éliminé complètement l'écart statique, d'où l'obtention d'une vitesse de sortie égale à celle de la consigne.

Le dépassement obtenu fait référence à une réponse pseudopériodique et ce la revient au coefficient d'amortissement  $h$  qui est entre  $0.45 < h < 0.7$ , en effet après calcul en s'appuyant sur la formule IV.30 On obtiendra un  $h=0,44$ .

**IV.11.6 Vérification de la stabilité :** de la même manière que précédemment on va s'appuyer sur le lieu de Nyquist à l'aide du programme qui suit :

```
%déclaration de la fonction du transfert 'BF' à correction proportionnelle intégrale.
num=[71.27 14.19]
den=[12.6 27.71 76.29 14.19]
G= tf(num,den)
figure, nyquist(G)
grid
```

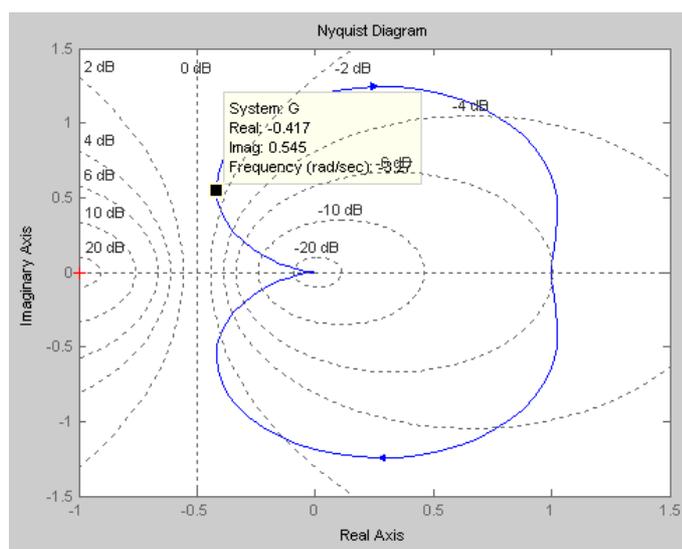


Figure IV.31 : Lieu de Nyquist pour le système à correction PI.

➤ A travers cette figure on remarque que le lieu de Nyquist ne dépasse pas le point critique ce la signifie que le système est stable et que le dépassement de 21,25% n'a pas d'influence sur sa stabilité.

### Conclusion 2:

Les paramètres du régulateur PI influent sur la réponse du système de la manière suivante :

-Si le gain statique  $k_c$  est grand, le temps de montée est plus court et cela crée un dépassement important et là aussi le temps de l'établissement du régime stationnaire varie peu et l'erreur statique s'améliore mais sans qu'elle soit totalement nulle.

$-T_i$  : lorsque  $\frac{1}{T_i}$  augmente, le temps de montée est plus court mais il y a un dépassement plus important. Le temps d'établissement au régime stationnaire s'allonge mais dans ce cas on assure une erreur statique nulle, donc plus qu'on augmente ce paramètre, plus la réponse du système est lente.

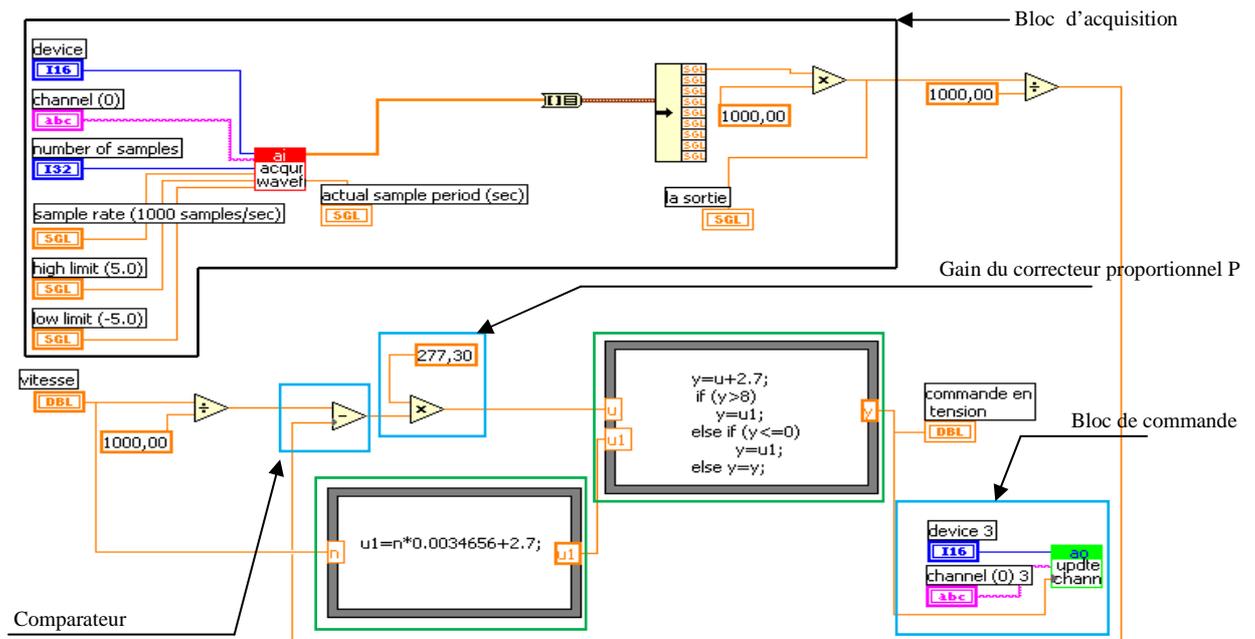
Donc il faut prendre en compte d'un bon ajustement de ces paramètres en même temps pour assurer la rapidité, la stabilité et la précision du système régulé.

L'association des deux actions Pet I nous permet d'avoir un système stable et précis.

**IV.12 Application du correcteur P et PI pour la régulation réelle :**

Après avoir calculé et choisi les bons paramètres des régulateurs qui s’adaptent mieux pour la régulation en vitesse sur le plan réel, on va mettre en œuvre ces résultats pour la commande numérique du procédé à l’aide d’un programme conçu sur LabVIEW.

**IV.12 .1 Application du correcteur P pour la régulation réelle :** dans la fenêtre de la programmation on réalise le programme suivant :



**Figure IV.32 : Action proportionnelle sous la face « diagramme » de LabVIEW**

**Explication :**

Ce programme permis de faire la correction comme suit :

Les boites de calcul de LabVIEW (entourée en vert) nous permettent d’écrire la même fonction de linéarisation vue sous MATLAB.

Si le système ne subit pas de perturbation la commande (y) applique directement la fonction (u1) qui suit la consigne (vitesse), dans le cas contraire, la perturbation détectée par le comparateur (écart≠0) déclenche la correction proportionnelle (u) qui sera injectée par la commande (y) dans le bloc de sortie (a.o), en effet l’exécution du programme pour une consigne de 300 tr/min en présence des perturbation (freinage manuelle du moteur) donne la figure suivante :

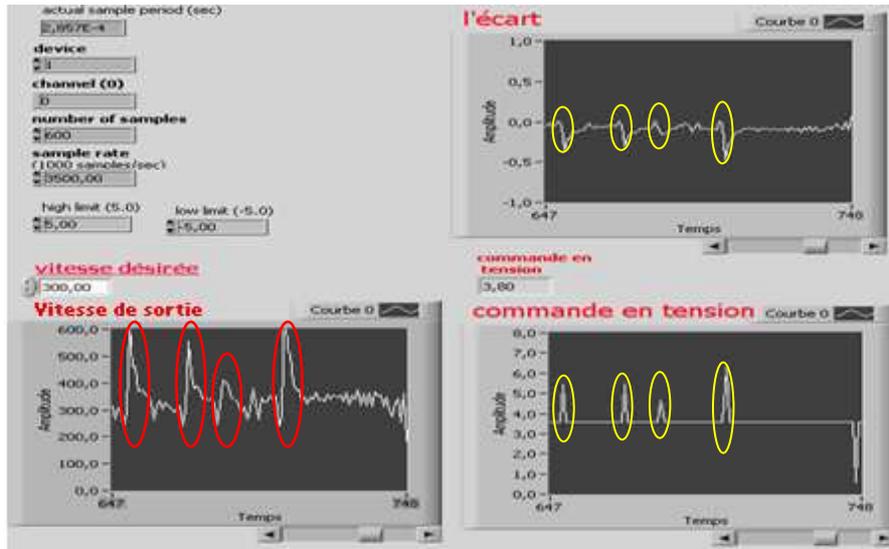


Figure IV.33 : Face avant de programme après exécution (au temps réel).

**Interprétation :**

Cette figure représente la réponse du système bouclé à action proportionnel pour la consigne de 300tr/min. En effet à travers le graphe « vitesse de sortie » de cette figure, on remarque que le système donne en sortie une vitesse qui est proche de la consigne, malgré la présence des perturbations (entourées en rouge), cela revient à l'intervention du correcteur P, qui fait en sorte de ramener à chaque fois la sortie du système à la consigne en augmentant ou en diminuant la tension de la commande, qui est présentée sur le graphe « commande en tension » pour éliminer l'écart entre la consigne et la sortie comme le montre le graphe « écart » de la figure IV.33.

**IV.12.2 Application du correcteur PI pour la régulation réelle**

**A- Calcule en z du correcteur PI :**

Le calcul d'un PI en z peut être effectué par la discrétisation du correcteur du domaine de LAPLACE vers le domaine discret (Z), et pour cela on procède par une programmation sous matlab comme suit :

```
% la fonction de transfert du régulateur I à structure parallèle
num=1
den=[5.02 0]
g=tf(num,den)
%discrétisation de g avec une période d'échantillonnage 1s
gd=c2d(g,1)
```

Après exécution on aura :

Transfer Function:  $\frac{0.1992}{z-1}$  (IV.31)

La fonction du transfert du régulateur PI en z est obtenue comme

$$U(p) = k_c \left(1 + \frac{1}{T_{ip}}\right) = 49.23\left(1 + \frac{1}{5.02p}\right)$$

La régulation est faite en structure parallèle donc la TZ du régulateur sera :

$$TZ(kc)=TZ(49.23)=49.23 \quad \text{et la} \quad TZ\left[\left(\frac{1}{5.02p}\right)\right]=TZ\left(\frac{1}{T_i p}\right)=\frac{\Delta}{T_i} \frac{1}{z-1} = \frac{0,1992}{z-1}$$

Et pour être sur d'exactitude des calculs on a procédé par une simulation sous « simulink » pour une consigne de 700tr/min et le résultat est présenté comme suit :

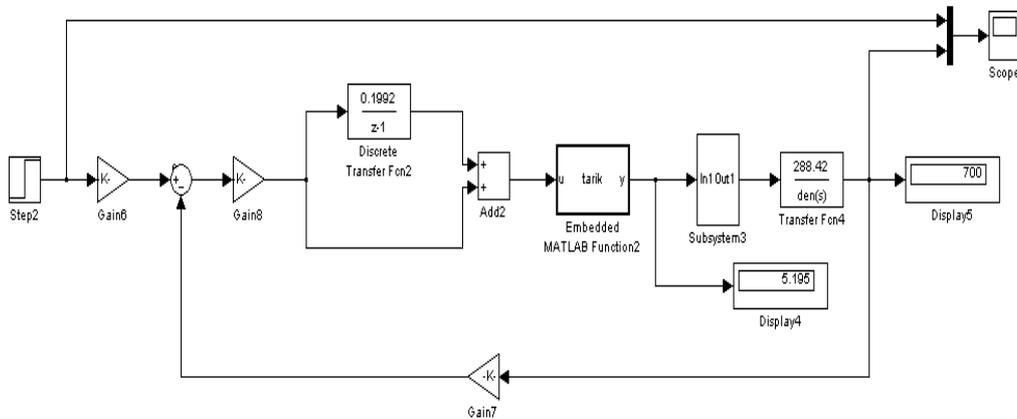


Figure IV.34 : Simulation de la régulation discrète

**B- Application de l'action PI sous LabVIEW :**

Dans LabVIEW, il existe des blocs de régulation (PID,PI...) mais leurs exécutions nécessitent des modules externes et des cartes à microprocesseur, ce qui n'est pas le cas dans notre application.

Mais en à fait recours à d'autres méthodes qui sont :

**a-Script MATLAB :**

La boîte de MATLAB script (encadrée en vert, figure IV.28) permet d'implanter dans LabVIEW des programmes réalisés sous simulink suivant les étapes:

- appeler la boîte de MATLAB script suivant le chemin « Palette des fonctions, Mathématiques, Formules, script MATLAB ». Comme le montre la figure suivante :

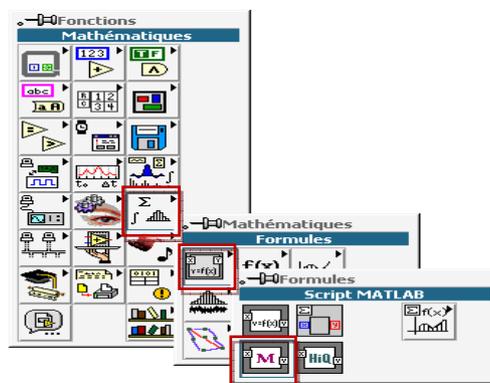


Figure IV.26. Chemin d'appel de script MATLAB

- Après avoir l'appeler, cliquer sur elle avec le bouton droit, sélectionner « importation » comme ça on ramène un fichier MATLAB en script ou bien directement un fichier programme sous simulink. Comme le montre cette figure :

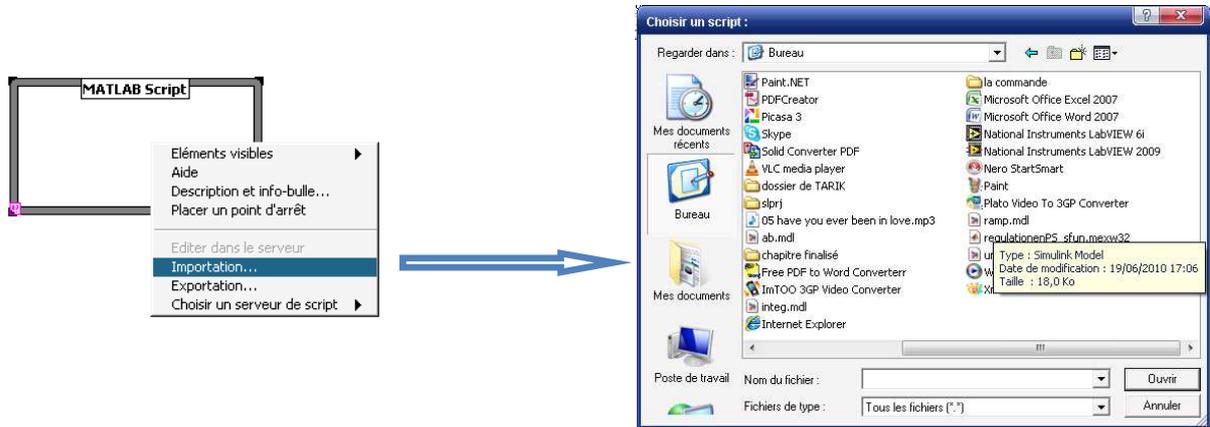


Figure IV.27. Chemin d'importation de programme de MATLAB vers LabVIEW.

- En fin, le programme complet de la régulation PI est représentée sous la figure suivante :

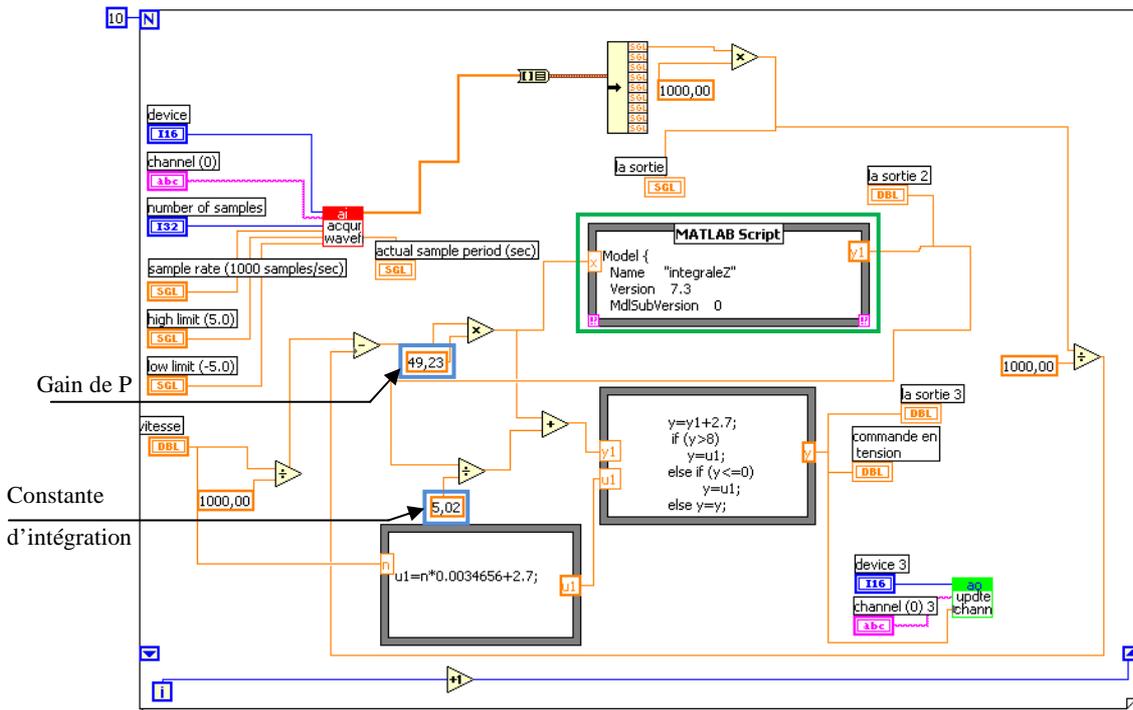


Figure IV.28 Programme de la régulation PI dans la face « diagramme » de LabVIEW.

- Après l'exécution de ce programme, on a enregistré les résultats présentés sur sa face avant sous la figure suivante :

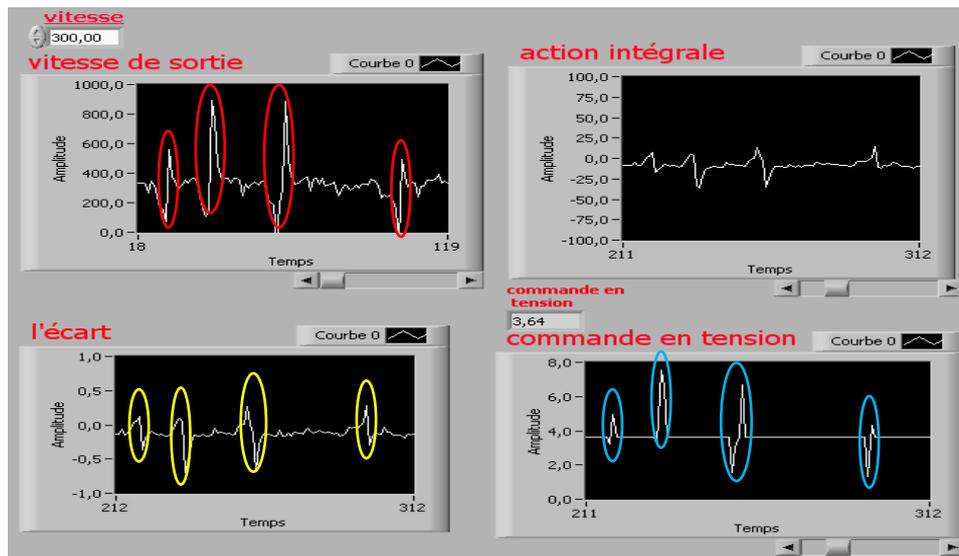


Figure IV.29 : La réponse du système à régulateur PI à une consigne de 300tr/min

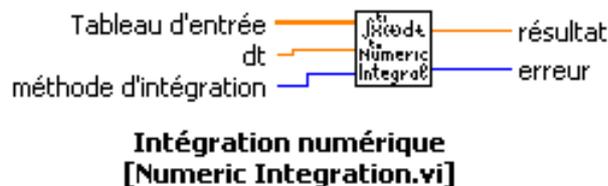
**Interprétation :**

Cette figure représente la réponse du système bouclé à action proportionnel intégrale pour la consigne de 300tr/min.

En effet à travers le graphe « vitesse de sortie » de cette figure, on remarque que le système donne en sortie une vitesse de 300tr/min, malgré la présence des perturbations (entourées en rouge), cela revient à l'intervention des deux actions proportionnelle et intégrale , qui font en sorte de ramener à chaque fois la sortie du système à la consigne en augmentant ou en diminuant la tension de la commande, qui est présentée sur le graphe « commande en tension » (entourées en bleu) pour éliminer l'écart entre la consigne et la sortie comme le montre le graphe « L'écart » (entourées en jaune) de la figure IV.29.

**b-Bloc d'intégration numérique:**

Ce bloc effectue une intégration numérique sur le tableau d'entrée de données en utilisant une des quatre méthodes d'intégration numérique couramment utilisées.



Le programme PI réalisé à l'aide de ce VI et le suivant :

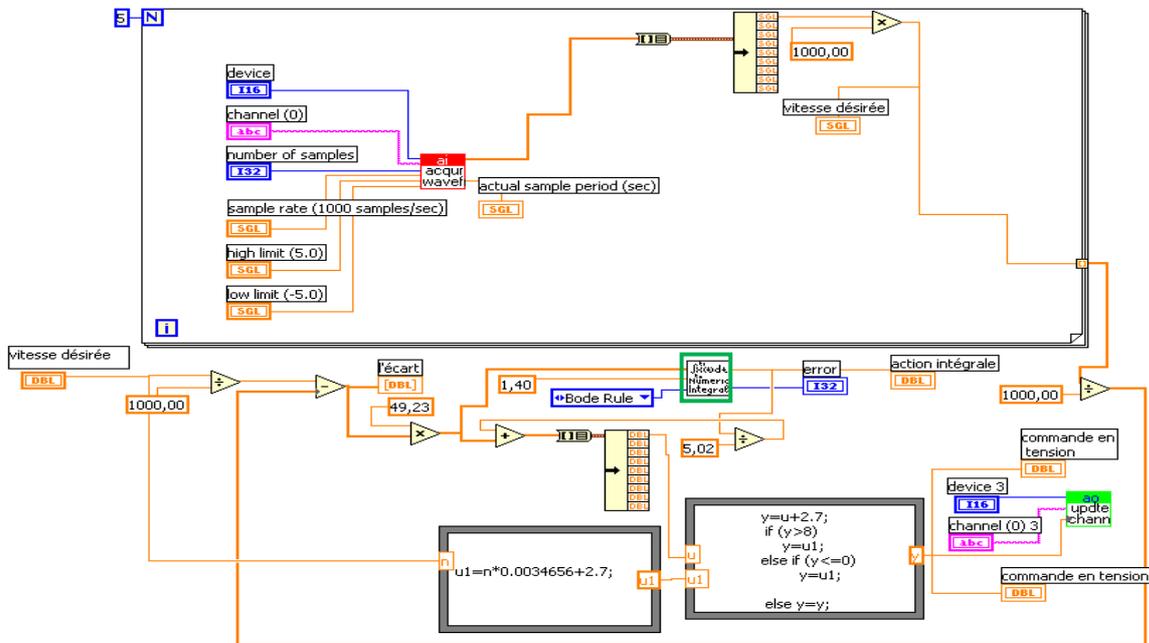


Figure IV.30 Programme de la régulation PI dans la face « diagramme » de LabVIEW.

- Après l'exécution de ce programme, on a enregistré les résultats présentés sur sa face avant sous la figure suivante :

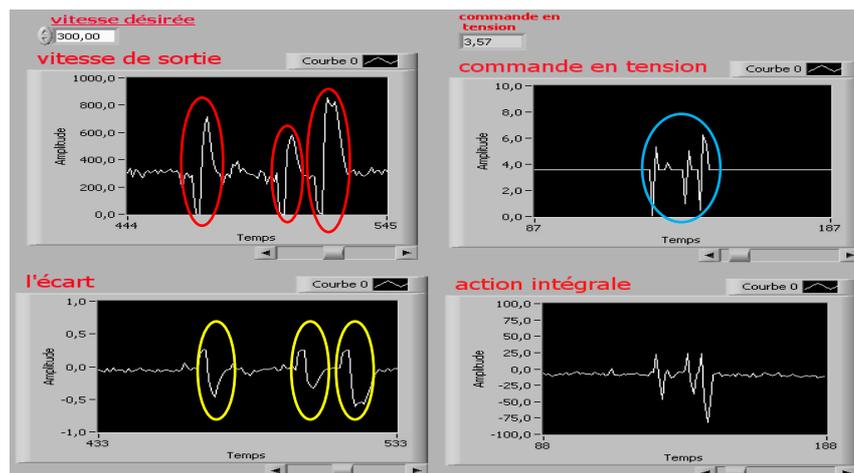


Figure IV.31 : La réponse du système à régulateur PI à une consigne de 300tr/min

**Interprétation :**

Une perturbation effectuée sur le moteur (une charge) qui est présentée par le graphe « vitesse de sortie » (entourée en rouge) influence sur la commande (diminution), comme l'indique l'écart (entourée en jaune), à ce moment l'action PI s'intervient pour la régulation en augmentant la commande .le graphe déroulant (commande en tension) montre la variation de la commande.

**IV.13 Conclusion :** on a pu effectuer une régulation sous LabVIEW malgré le manque de fiabilité du matériel, il est toujours possible d'apporter des améliorations avec un perfectionnement, tel que l'ajout des modules externes de *National Instrument*, l'utilisation des PC de grande vitesse d'exécution et des cartes d'acquisitions dédiées à la commande en temps réel.

## *Conclusion générale*

---

L'étude qu'on nous avons menée nous a permis d'approfondir nos connaissances sur l'acquisition et la commande numérique au temps réel.

Nous avons réalisé la commande en vitesse d'un moteur à courant continu à excitation séparé par ordinateur en faisant varier la tension de l'induit. La mesure de la vitesse du moteur est assurée par une génératrice tachymétrique, l'acquisition de l'information concernant la vitesse et la commande du système sont assurées via la carte KPCI et la programmation par le logiciel LabVIEW.

Cette commande est vérifiée d'abord par le logiciel MATLAB, où on a aussi effectué une identification du système.

Ce travail nous a permis d'apprendre un nouveau langage de programmation, de commander un moteur à courant continu et de voir et étudier les constituants de la chaîne d'acquisition.

Cet exemple concret de la commande numérique nous a permis de mettre en évidence avantages d'acquisition de données et d'améliorer nos connaissances sur la notion du temps réel.

Dans ce sens on espère que ce travail servira comme de support pour les promotions à venir pour développer et implémenter de nouvelles commandes de régulation.

# Références bibliographiques

---

- [1] **THEODORE, WILID.** And **GILBERT, SYBILLE.**, “*électrotechnique*”. Ingénieur, institut de recherche d’Hydro-Québec, 2000.
- [2] **GEORGES, ASCH.**, et **COLLABORATEURS.**, “*Acquisition de données du capteur à l’ordinateur*”. Dunod, paris, 2003.
- [3] **FRANCIS, COTTET.**, “*LabVIEW Programmation et applications*”, Dunod ,paris ,2001.
- [4] **ETIENNE, KIEIN.** “*Machine à courant continu : fonctionnement et structure interne*”. *Unité de la physique*, vol.2-133 ,2002.
- [5] **ALAIN, CHARBONNEL.** “*le moteur à courant continu aimants permanent*”. *Science de l’ingénieur*, 2007, [www.discip.crdp.ac-caen.fr](http://www.discip.crdp.ac-caen.fr).
- [6] **BERNARD, BAYLE.** “*technologie des asservissements variateurs de vitesse*”. *Ecole nationale supérieur de physique de Strasbourg*. 2009.
- [7] **DECOURT, R.** et **DORDOR, P.** “*introduction a LabVIEW*”; édition 2003/2004.
- [8] **DEFAIX, OLIVIER.** “*asservissement de la température sous MATLAB et LabVIEW*”; *master SEA*, 2004/2005.
- [9] **NICOLAS, POUSSET** “*cour d’initiation au logiciel LabVIEW*”; de *NATIONAL INSTRUMENTS*, 2008.
- [10] **PHILIPPE, BERGER.** « Cour capteurs d’automatique », [http://philippe.berger2.free.fr/automatique/cours/cpt/les\\_capteurs.htm](http://philippe.berger2.free.fr/automatique/cours/cpt/les_capteurs.htm)
- [11] **PATRICK, ABATI.** « Capteur de vitesse et de position », <http://stielec.ac-aix-marseille.fr/cours/abati/captvit.htm>
- [12] **OLIVIER, FRANÇAIS.** “*théorie de l’échantillonnage et de quantification*”, *ESIEE*, 2002, PP.V.3.
- [13] **CHRITIAN, BISSIERS.** « *échantillonnage et conversion du signal* », *TS IRIS, physique appliqué*, [cbissprof.free.fr/telechargements/tsiris/.../acquisitionsignal.pdf](http://cbissprof.free.fr/telechargements/tsiris/.../acquisitionsignal.pdf).
- [14] **AURORA, ROAD.** “*DriverLINX for LabVIEW Manual*”, *Keithley Instruments*, 2001, <http://www.keithley.com>
- [15] **AURORA, ROAD.** “*KPCI-3101-4 User's Manual*”, *Keithley Instruments*, 2001, <http://www.keithley.com>
- [16] **DIAF, MOUSSA.** “ Cours commande numérique”, université de Mouloud Mammeri Tizi ousou, 2008/2009.
- [17] **MAIDI, AHMED.** “ Cours identification des système”, université de Mouloud Mammeri Tizi ousou, 2008/2009.
- [18] Mémoire de fin d’études, « *Modélisation et simulation de a machine à courant continu sous logiciel Matlab* », proposé et dirigé par **Mr H.DJOUDI**, U.M.M.T.O

# *Résumé*

Notre projet a comme objectif de faire une conception et une réalisation d'une commande au temps réel du procédé constitué d'un moteur à courant continu couplé à une génératrice tachymétrique précédé d'un étage de puissance.

Afin de mener à bien notre travail nous avons réparti ce mémoire en quatre chapitres :

- Le premier, était des généralités sur le moteur à courant continu.
- Le second est consacré pour l'étude générale d'une chaîne d'acquisition.
- Le troisième nous a permis de découvrir le logiciel LabVIEW.
- Et dans le quatrième nous avons donné quelques rappels sur la commande, la régulation et les méthodes d'identification numériques que nous avons terminé par la présentation des résultats obtenus en pratique.

Et nous avons terminé par une conclusion générale.