

الجمهورية الجزائرية الديمقراطية الشعبية  
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
وزارة التعليم العالي و البحث العلمي  
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE  
SCIENTIFIQUE  
جامعة سعد دحلب البليدة 1  
Université SAAD DAHLEB de BLIDA 1



كلية التكنولوجيا

Faculté de Technologies

قسم الإلكترونيك

Département d'électronique

## Mémoire de projet de

### Fin d'études

Pour l'obtention du diplôme de MASTER en électronique  
Option automatique

---

*Implémentation sur DSP de la commande  
PID optimisée à l'aide de l'algorithme  
TLBO : Application à la commande d'un  
moteur DC*

---

Présenté par :

**BOUKABOUS Fedwa**

**AHMED M'BAREK Manel**

proposé par :

**M. K. KARA**

**M. BENRABAH**

Session: juin 2018



## Remerciements

Nous tenons à remercier fortement Dieu qui nous a donné patience, courage, force et capacité nécessaires pour compléter ce projet. Nous aimerions remercier nos promoteurs **M. K. KARA** et **M. M. BERABAH** pour leurs précieux conseils, leur aide, leur soutien et leur confiance.

Nous exprimons notre gratitude au staff du laboratoire 24 qui nous ont beaucoup aidé et soutenu et pour l'ambiance et le confort qu'ils nous ont apporté pendant toute la durée de ce travail.

Nos remerciements et grâces vont également à M. K. KARA et à tous les enseignants, de nous avoir aidé durant tout le parcours de notre formation. Notre adressons nos sincères remerciements à tous les notables membres du jury pour l'honneur de leur présence qu'ils nous ont fait d'avoir accepté d'évaluer ce travail.

Nous voudrions également exprimer notre gratitude au personnel du département de l'électronique pour leur disponibilité.

Nous n'oublions pas de remercier nos parents pour leur amour et leur soutien moral.

## **Dédicace**

*Je dédie ce travail à mon cher père et à ma mère, qui ne pourront exprimer ma gratitude en quelques mots : sans eux, je ne serai jamais ce que je suis aujourd'hui et je ne pourrai jamais le ramener qui m'a donné amour, affection et soutien. Je vous remercie Du fond de mon cœur.*

*Je le dédie tous les membres de ma famille et surtout mon frère WALID et mes sœurs HALA, CHAHID à qui ont toujours assuré le support, l'aide et le conseil pour réaliser mes ambitions.*

*Je le dédie à mes enseignants (K. KARA, M. BERABAH...) Assurez-vous de trouver dans cette œuvre une expression de moi Une profonde gratitude et un grand respect.*

*A la fin, je le dédie aussi à mes amis et surtout ma meilleure amie MADINA et MANEL.*

***BOUKABOUS FEDWA***

## Dédicace

*Je dédie ce travail :*

*A mon père,*

*Pour son grand amour, son encouragement, son sens du devoir et ses sacrifices pour que je réussisse dans mes études.*

*A ma mère,*

*Pour son affection, sa patience, son encouragement pendant les épreuves difficiles ainsi que ses prières qui m'apportent bonheur et réussite.*

*A ma chère sœur RIMA et mon frère MOHAMED YACINE.*

*A mes chères AFFAF et RADIA, Qui sont toujours à mes côtés, prêts à m'aider.*

*A tous ceux qui m'aiment et à tous ceux que j'aime.*

*A tous ceux que je connais de près ou de loin.*

*AHMED M'BAREK Manel*

## ملخص

ان العمل المنجز يتمثل في التصميم و التثبيت عنصر التحكم PID في معالج الاشارات الرقمي DSP من نوع TMS320F28335 المصنوع من طرف مؤسسة Texas Instruments و ذلك باستخدام خوارزمية التحسين TLBO . ولتقييم اداء عنصر التحكم قمنا بالتحكم في سرعة محرك ذو تيار مستمر. في الشطر الاول من هذا العمل و بعد استخراج النموذج الرياضي للمحرك استخدمنا خوارزمية TLBO من اجل تعيين قانون التحكم الخاص بعنصر تحكم PID باستعمال معاملات ثابتة ,بالإضافة الى قانون تحكم اخر باستعمال خاصية التحكم على التكيف .ولقد قمنا ايضا بعدة عمليات محاكاة من اجل اظهار فعالية كل قانون تحكم. استخدمنا برامج MATLAB للقيام بعمليات المحاكاة. اما في الشطر الثاني فقد قمنا بصياغة برنامج التحكم باستعمال برنامج Code Composer Studio ثم قمنا بتثبيته في DSP. بالاضافة الى انشاء مختلف الدارات الكهربائية اللازمة من اجل القيام بتجارب على النظام المصنع .

الكلمات الرئيسية : معالج الإشارات الرقمي (DSP), التحكم المتكيف, محركات ذات تيار مستمر.

## Résumé

Le travail réalisé consiste à la conception et l'implémentation dans un DSP TMS320F28335 de Texas Instruments, d'un régulateur PID en utilisant l'algorithme d'optimisation TLBO. Pour évaluer les performances de ce régulateur nous avons considéré la commande en vitesse d'un moteur à courant continu. En effet, dans la première partie de ce travail et après avoir procédé à la modélisation du moteur nous avons utilisé l'algorithme TLBO pour déterminer la loi de commande du régulateur PID à paramètres fixes, et mettre au point l'algorithme d'adaptation du régulateur PID auto ajustable. Ensuite, nous avons effectué plusieurs simulations pour mettre en évidence les performances des deux régulateurs obtenus. Nous avons utilisé le logiciel MTLAB pour faire ces simulations.

La deuxième partie de ce travail, consiste à développer, le code qui va être implémenté dans le DSP, mettre au point les différents circuits et effectuer les différents essais expérimentaux permettant d'évaluer les performances de l'algorithme de commande implémenté.

**Mots clés : DSP, Auto ajustable, Moteur courant continu.**

## Abstract

The aim of our work is to implement a DC motor velocity control algorithm on DCP TMS320F2812 of Texas instrument.in fact ,the first part of this work deals with the determination of the control law using a fixed parameter PID controller , then we have set the adaptation algorithm for the self-tuning PID controller ,this done by using TLBO and performed many simulations to highlight the performances achieved by each controller we used MATLAB softuo are to do these simulation .the second part was the conception of a program using the code composer studio to implement it in the DSP and to develop the necessary electrical circuits in order to evaluate each in a real nom simulated environment.

**Key words: DSP, Adaptive control, DC Moteur.**

# Table des matières

Introduction Générale .....	1
-----------------------------	---

## **Chapitre 1 : Notion sur la correction des systèmes**

1.1 Introduction.....	3
1.2 Correction et régulation des systèmes .....	3
1.2.1 Correction en boucle ouverte .....	3
1.2.2 Correction en boucle fermée .....	4
1.2.3 Correcteurs classiques.....	5
1.2.3.a Correction proportionnel (P) .....	5
1.2.3.b Correction proportionnel intégrateur (PI) .....	5
1.2.3.c Correction proportionnel dérivateur (PD) .....	6
1.2.3.d Correction proportionnel intégral dérivé (PID) .....	7
1.2.4 Méthodes de Ziegler et Nichols (ZN) .....	7
1.2.4.a Méthode de la réponse indicielle .....	8
1.2.4.b Méthode du point critique.....	9
1.2.5Le PID numérique.....	9
1.3 Conclusion.....	11

## **Chapitre 02 :**

### **Méthode d'optimisation: Teaching and Learning Based Optimization (TLBO)**

2.1 Introduction.....	12
2.2 Problème d'optimisation.....	12
2.3 Optimisation métaheuristique.....	13
2.3.1 Métaheuristique.....	13
2.3.2 Domaine d'application.....	13

2.4 Teaching Learning based optimization .....	14
2.4.1 Principe d'algorithme TLBO.....	14
2.5 Exemples d'optimisation en utilisant la méthode TLBO.....	18
2.5.1 Comparaison de TLBO avec l'algorithme PSO .....	19
2.6 Conclusion .....	20

### **Chapitre 3 : Modélisation et commande du moteur à courant continu**

3.1 Introduction .....	21
3.2. Moteur à courant continu .....	21
3.3. Modélisation.....	24
3.3.1 Modélisation du MCC.....	24
3.3.2 Modélisation de l'encodeur.....	26
3.4 Hacheur.....	26
3.4.1 Hacheur abaisseur à un quadrant (Série).....	27
3.4.2 Hacheur à quatre quadrants.....	29
3.5 Commande du moteur en utilisant un PID .....	31
3.5.1 Simulation sans perturbations.....	32
3.5.2 simulation avec perturbation additive en sortie.....	33
3.5.3 Simulation avec perturbations additive en entrée.....	34
3.6 Commande du moteur en utilisant un PID auto ajustable .....	35
3.6.1 Simulation sans perturbations.....	35
3.6.2 simulation avec perturbation additive en sortie .....	37
3.6.3 Simulation avec perturbations additive en entrée.....	39
3.7 Conclusion.....	40

### **Chapitre 4 : Implémentation de la commande et essais pratiques**

4.1 Introduction.....	41
4.2 Constituants du banc d'essai.....	41
4.3 Description des éléments du banc d'essai.....	43
4.3.1 Le kit de développement ezdspF28335.....	43



4.4 Simulation du hacheur.....	48
4.4.1 Introduction à PSIM.....	48
4.4.2 Résultats de simulation.....	48
4.5 Circuit de puissance.....	49
4.6 Circuit de commande (carte mère) .....	51
4.7 Circuit d'isolation (isolation galvanique) .....	52
4.8 pupitre de commande et de visualisations.....	53
4.9 Résultats expérimentaux.....	54
4.9.1 Commande en boucle ouverte .....	54
4.9.2 Commande en boucle fermé .....	56
4.10 Conclusion.....	58
Conclusion générale.....	59

## Listes des acronymes et abréviations

---

P	Correcteur proportionnel.
PI	Correcteur proportionnel intégrateur.
PD	Correcteur proportionnel dérivateur.
PID	Correcteur proportionnel, intégrateur et dérivateur.
$k_p$	Gain proportionnel.
$k_i$	Constante d'intégration.
$T_d$	Constante de dérivation.
$r(t)$	La référence.
$y(t)$	Sortie du processus.
$u(t)$	Signale de commande.
$e(t)$	L'erreur.
$K_u$	Le gain critique.
$T_u$	Temps de retard.
$\Delta y$	La différence entre l'amplitude initiale et l'amplitude finale de la sortie du processus
$\Delta u$	La différence entre l'amplitude initiale et l'amplitude finale de l'échelon.
$T_a$	Le temps de montée de la tangente.
$T_0$	Période d'échantillonnage.
FRM	Forward Rectangular Method.
BRM	Backward Rectangular Method.
TRAP	Trapezoid méthode.
TLBO	Teaching and Learning based optimization
N	Taille de population
MCC	Moteur à courant continu.
CC	Courant continu (DC : Direct curent).
$u(t)$	La tension d'alimentation(V).

$K_e$	La constante de la force contre-électromotrice ( $v. s/rad$ )
$w(t)$	La vitesse de rotation angulaire du moteur en ( $rad \setminus s$ ).
$\theta(t)$	La position angulaire du moteur en ( $rad$ ).
$T(t)$	Le couple moteur
$K_c$	La constante du couple moteur en ( $N_m/A$ ).
$b$	Le coefficient de frottement visqueux en ( $K_g m^2/s$ ).
$J_T$	L'inertie de la charge et du moteur en ( $K_g m^2$ ).
$\alpha$	Le rapport cyclique.
<i>RLD</i>	La diode de roue libre.
PWM	Pulse Width Modulation.
DSP	Digital Signal Processor.
GPIO	General purpose input output.
eQEP	enhanced Quadrature Encoder Pulse.
QEP	Quadrature Encoder Pulse.
Interrupt	Interruption.
TI	Texas Instrument.
ALU	Unité arithmétique et logique.
CAP	Unité de capteur du DSP.
ADC	Analog to Digital Converter (Convertisseur analogique/numérique).
DSC	Differential Scanning Calorimetry.
PSIM	Power Sim.
ePWM	Enhanced Pulse Width Modulator.
IRF	International Road Federation.

## Liste des figures

Figures	Chapitre 1	Page
<b>Figure.1.1</b>	Correction en boucle ouverte.	4
<b>Figure.1.2</b>	Correction en boucle fermée	5
<b>Figure.1.3</b>	Méthode de la réponse indicielle.	8
	Chapitre 2	
<b>Figure.2.1</b>	Organigramme d'algorithme TLBO	17
<b>Figure.2.2</b>	Courbe de la fonction Coût $f_1$	18
<b>Figure.2.3</b>	Courbe de la fonction Coût $f_2$	18
	Chapitre 3	
<b>Figure.3.1</b>	Moteur à courant continu	22
<b>Figure.3.2</b>	Moteur excitation séparée	22
<b>Figure.3.3</b>	Moteur excitation série	23
<b>Figure.3.4</b>	Moteur excitation composée	23
<b>Figure.3.5</b>	Schéma électrique simplifié d'un moteur DC en charge	24
<b>Figure.3.6</b>	Schéma simplifié d'une charge commandé par transistor	27
<b>Figure.3.7</b>	Hacheur abaisseur à un quadrant	28
<b>Figure.3.8</b>	Représentation idéale d'un hacheur abaisseur à un quadrant	28
<b>Figure.3.9</b>	Les deux états de fonctionnement du hacheur abaisseur	29
<b>Figure.3.10</b>	Les quatre quadrants	30
<b>Figure.3.11</b>	Commande en tension du hacheur	30
<b>Figure.3.12</b>	Commande PID du MCC	31
<b>Figure.3.13</b>	La réponse indicielle du MCC en boucle ouverte	31
<b>Figure.3.14</b>	Référence, vitesse de rotation du MCC et signale de commande délivrée par le PID	32

<b>Figure.3.15</b>	Ecart entre la sortie réelle du MCC et la référence.	33
<b>Figure.3.16</b>	Référence, vitesse de rotation du MCC et signale de commande délivrée par PID (car avec perturbation 50%)	33
<b>Figure.3.17</b>	Ecart entre la sortie réelle du MCC et la référence	34
<b>Figure.3.18</b>	Référence, vitesse de rotation du MCC et signale de commande délivrée par PID (car avec perturbation 25%)	34
<b>Figure.3.19</b>	Ecart entre la sortie réelle du MCC et la référence	35
<b>Figure.3.20</b>	Réel les valeurs des paramètres $K_p, K_i, K_d$ du PID	36
<b>Figure.3.21</b>	Référence, vitesse de rotation du MCC et signale de commande délivrée par PID auto ajustable	37
<b>Figure.3.22</b>	Ecart entre la sortie réelle du système perturbé et la référence	37
<b>Figure.3.23</b>	Réel les valeurs des paramètres $K_p, K_i, K_d$ du PID	37
<b>Figure.3.24</b>	Référence, vitesse de rotation du MCC et signale de commande délivrée par PID auto ajustable (car avec perturbation 50%)	38
<b>Figure.3.25</b>	Ecart entre la sortie réelle et la référence	38
<b>Figure.3.26</b>	Réel les valeurs des paramètres $K_p, K_d, K_i$ du PID	39
<b>Figure.3.27</b>	Référence, vitesse de rotation du MCC et signale de commande délivrée par PID auto ajustable (car avec perturbation 25%)	39
<b>Figure.3.28</b>	Ecart entre la sortie réelle du système perturbé et la référence	40
<b>Chapitre 04</b>		
<b>Figure.4.1</b>	Photographie du banc d'essais	41
<b>Figure.4.2</b>	Schéma synoptique du dispositif expérimental	42
<b>Figure.4.3</b>	La carte ezdspTM320F335	43
<b>Figure.4.4</b>	Architecture de ezdspTM320F335	44
<b>Figure.4.5</b>	Affectation des lignes d'interruption	46

<b>Figure.4.6</b>	Encodeur de pulsation quadratique	47
<b>Figure.4.7</b>	Schéma du hacheur abaisseur	48
<b>Figure.4.8</b>	La tension de sortie, signale de PWM pour un rapport cyclique de 25%	49
<b>Figure.4.9</b>	La tension de sortie, signale de PWM pour un rapport cyclique de 50%	49
<b>Figure.4.10</b>	Schéma électrique du hacheur (Buck)	50
<b>Figure.4.11</b>	Schéma électrique de l'alimentation de 12 V	50
<b>Figure.4.12</b>	Photographie du circuit de puissance	51
<b>Figure.4.13</b>	Schéma électrique du circuit de commande en boucle ouverte	52
<b>Figure.4.14</b>	Schéma électrique de circuit d'isolation (isolation galvanique)	53
<b>Figure.4.15</b>	Schéma électrique pupitre de commande et de visualisation	54
<b>Figure.4.16</b>	Résultats obtenus pour un rapport de 25%	55
<b>Figure.4.17</b>	Résultats obtenus pour un rapport de 50%	55
<b>Figure.4.18</b>	Résultats obtenus pour un rapport de 75%	56
<b>Figure.4.19</b>	Signale PWM et vitesse du MCC mesurée	57
<b>Figure.4.20</b>	Vitesse du MCC et signale de PWM pour la référence de 530(tr/mint)	57
<b>Figure.4.21</b>	Vitesse du MCC et signale de PWM pour la référence de 1125(tr/mint)	58

## Liste des tableaux

---

<b>Tableau</b>	<b>Chapitre 1</b>	<b>Page</b>
<b>Tableau.1.1</b>	Calcul des paramètres des régulateur P, PI ou PID en utilisant la méthode de la Réponse indicielle ( $K = \frac{\Delta y}{\Delta u}$ )	8
<b>Tableau.1.2</b>	Calcul des paramètres des régulateur P, PI ou PID en utilisant la méthode du point critique.	9
<b>Tableau.1.3</b>	Expression des différents paramètres du PID numérique pour les différentes Méthodes d'approximation de l'intégral.	11
	<b>Chapitre 2</b>	
<b>Tableau.2.1</b>	Résultats d'optimisation en utilisant la méthode TLBO pour plusieurs valeurs de la population	19
<b>Tableau.2.2</b>	Résultats d'optimisation en utilisant la méthode TLBO pour plusieurs valeurs du nombre maximal d'itération	19
<b>Tableau.2.3</b>	Résultats d'optimisation en utilisant la méthode TLBO et la méthode PSO	20

# **Introduction générale**



# Introduction générale

La technologie moderne a permis le développement des sciences tout en imposant l'exploration des domaines théoriques de plus en plus complexes. Parmi ces sciences en pleine expansion et intégrant rapidement l'apport des technologies modernes, on compte l'automatique. Le substantif « automatique » a été utilisé pour la première fois en 1914 dans un article « Essai sur l'Automatique » publié dans une revue scientifique.

L'automatique permet l'automatisation des différentes tâches de fonctionnement des machines et des chaînes industrielles. On parle alors de système asservi ou régulé, d'où est le plus répondu dans notre vie quotidienne, en particulier dans le domaine industriel, car il permet de réaliser plusieurs opérations sans l'intervention de l'être humain, pour de divers besoins.

Dans la plupart des processus industriels, en particulier les moteurs électriques, il est indispensable de maîtriser certains paramètres physiques (vitesse, position, angle...etc.), il est donc nécessaire d'avoir recours à une commande. La commande PID (Proportionnelle-Intégrale-Dérivée) est une méthode qui donne de bons résultats, grâce à ces actions, proportionnelle qui améliore la rapidité, l'intégrale pour la précision, et la dérivée pour la stabilité. Notre travail, qui s'inscrit dans ce contexte, consiste à concevoir et implémenter dans un DSP, *TMS320F335* de Texas Instrument, pour mettre au point la commande PID d'un moteur à courant continu en utilisant l'algorithme d'optimisation Teaching and Learning Based Optimization (TLBO).

Nous pouvons effectuer beaucoup des calculs très complexes en un minimum de temps à l'aide d'un processeur DSP, mais aussi pour accéder facilement à un grand nombre d'entrées / sorties. Les unités de calcul utilisées avec la plupart des DSP sont en virgule fixe. L'absence d'unité arithmétique en virgule flottante rend le prix du composant abordable permettant aussi une grande vitesse de traitement des données. L'objectif de notre travail est d'implémenter sur la Carte DSP une commande PID d'un moteur à courant continu en utilisant l'algorithme de Teaching and Learning Based Optimization (TLBO).

L'utilisation d'une commande auto-ajustable permet de garder un niveau de performances exigé indépendamment des différentes perturbations (erreur, rapidité, stabilité). On peut adapter un régulateur auto-ajustable en temps réel aux différentes perturbations observées dans le système

## **Introduction générale**

---

d'une manière totalement automatique Le régulateur PID reste l'algorithme de commande le plus simple et le plus utilisé en industrie, il permet des performances que nous pouvons offrir aux systèmes en boucle fermée, et satisfait très souvent les cahiers de charge, lorsque les paramètres sont choisis correctement.

Notre travail est organisé en quatre chapitres :

Dans le premier chapitre, nous présentons des notions générales sur la correction des systèmes. Dans le deuxième chapitre, nous présentons l'algorithme d'optimisation Teaching and Learning Based Optimization ainsi que les résultats de simulation de l'optimisation de quelques fonctions mathématiques de test. Le troisième chapitre est consacré au moteur à courant continu, sa modélisation et les résultats de simulation en considérant plusieurs cas (sans perturbations avec perturbations). Dans le dernier chapitre, nous présentons les différents circuits réalisés et les différents résultats obtenus expérimentalement. Enfin, nous terminons par une conclusion générale sur le travail réalisé.

# **Chapitre 1**

## **Notion sur la correction des systèmes**

## 1.1. Introduction

Le régulateur PID (proportionnel intégral dérivé) est sans doute le régulateur le plus employé dans l'industrie. La simplicité de sa structure, le nombre restreint de paramètres à ajuster, et les performances qu'il offre au système en boucle fermée, lui confère la première place du podium des régulateurs.

La complexité et les performances des régulateurs PID sont variables selon la méthode de synthèse choisie. Toutefois, un PID est un outil dont l'ajustement peut être accompli en considérant indépendamment l'effet produit par chacun de ses paramètres. Ce qui a mené à des formules et des règles empiriques d'ajustement avantageuses.

Il faut cependant souligner qu'un modèle de paramètres fixes est souvent loin d'apporter les performances requises dans le cas d'un système dynamique reflétant la réalité industrielle. Ceci mène à l'usage de lois de commandes à caractère adaptable aux variations du système induit par la réalité industrielle.

Le but de ce chapitre est de rappeler brièvement la notion de correction de systèmes et les différents correcteurs classiques.

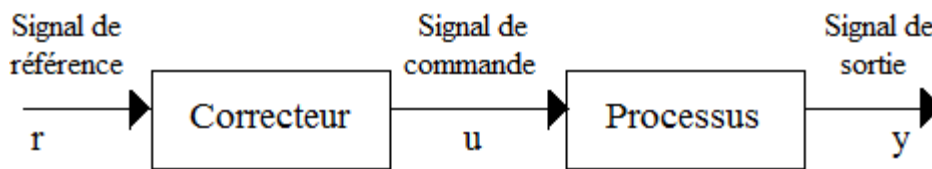
## 1.2 Correction et régulation des systèmes

La marge de stabilité, les dépassements, le temps de réponse et l'erreur en régime permanent, ce sont les critères auxquels doit répondre le correcteur. En sachant que son rôle le plus important est la commande des processus pour avoir une sortie désirée.

La correction des systèmes peut être abordée selon deux approches, la première se fonde uniquement sur la valeur voulue à la sortie du processus (commande en boucle ouverte) et la seconde se base également sur la valeur désirée mais aussi sur la sortie du système réellement obtenue (commande en boucle fermée)

### 1.2.1 Correction en boucle ouverte

Le recours à la correction en boucle ouverte est pertinent dans la mesure où l'on sait qu'une certaine commande appliquée au système va systématiquement donner une réponse identique. En effet, comme on peut le voir dans (la fig.1.1), dans le cas d'une correction en boucle ouverte, la grandeur de commande  $u$  dépend que du signal de référence  $r$ , et le correcteur ne reçoit pas d'information sur la sortie réelle du système  $y$ .



**Figure 1.1** : Correction en boucle ouverte

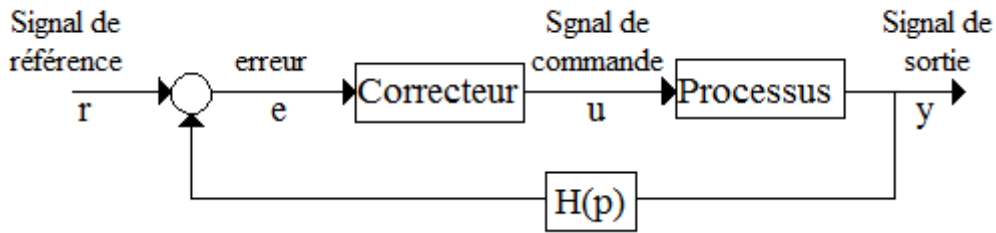
Cette correction présente de nombreux avantages mais également quelques inconvénients majeurs :

- Parmi les avantages :
  - Nous pouvons souligner qu'un asservissement en boucle ouverte permet d'obtenir des temps de réponse très courts.
  - Il n'y a pas d'oscillation à craindre (car il s'agit d'un système dynamiquement stable).
- Parmi les inconvénients :
  - La régulation en boucle ouverte impose de connaître la loi régissant le fonctionnement du processus, et il est très fréquent que l'on ne connaisse pas la loi en question.
  - Il n'y a aucun moyen de contrôler, à plus forte raison de compenser, les erreurs, les dérives, les accidents qui peuvent intervenir à l'intérieur de la boucle.
  - La régulation en boucle ouverte ne compense pas les facteurs perturbateurs.

Quant aux applications de ce genre de correction, elle se trouve généralement dans la commande de moteur pas à pas. Ceci car ce genre de moteur tourne avec un angle précis peu importe sa charge pour chaque impulsion reçue. Ce qui nous permet d'identifier sa position sans avoir recours à un capteur. En effet rien qu'en suivant le nombre d'impulsion il est possible de desseller sa position. Cette correction peut également s'envisager dans le cas des systèmes où les perturbations sont négligeables, ne nécessitant pas de contrôle précis.

### 1.2.2 Correction en boucle fermée

Dans ce cas (figure 1.2), la sortie du procédé est comparée à la référence, et en fonction du résultat obtenu, le correcteur va agir sur le processus. Ce mode de correction est le plus robuste contre les perturbations.



**Figure 1.2 :** Correction en boucle fermée.

La fonction de transfert de ce type de correction est donnée par :

$$C(p) = \frac{y(p)}{r(p)} = \frac{G(p).C(p)}{(1+G(p).C(p)).H(p)} \quad (1.1)$$

Le correcteur peut être placé dans la chaîne directe, en parallèle, ou en série avec le système, dans la chaîne principale de retour ou d'une chaîne de retour secondaire.

### 1.2.3 Correcteurs classiques

#### 1.2.3.a Correction proportionnel (P) :

Dans le cas de ce genre de correction la grandeur de commande est étroitement liée à la dimension de l'erreur, elle est donnée par l'équation suivante [1] :

$$u(t) = K_p e(t) = K_p [r(t) - y(t)] \quad (1.2)$$

Où  $K_p$  représente le gain proportionnel,  $r(t)$  la référence et  $y(t)$  la sortie du processus.

La fonction de transfert du correcteur sera :

$$C(p) = K_p \quad (1.3)$$

Avantage de ce correcteur est provoqué une augmentation du gain, ce qui entraîne une diminution de l'erreur statique (précision améliorée). Mais ce pendant augmente la bande passante du système, ce qui améliore la vitesse du système et augmente l'instabilité du système.

#### 1.2.3.b Correction proportionnel intégrateur (PI)

Ce correcteur inclut l'action Intégrale en plus de l'action Proportionnelle. La loi de la commande est obtenue par :

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau \quad (1.4)$$

Où  $K_i$  représente la constante d'intégration. On peut aussi remarquer que cette action intégrale dépend de toutes les valeurs précédentes de l'erreur. La fonction de transfert sera :

$$C(p) = K_p + \frac{K_i}{p} \quad (1.5)$$

La présence d'un zéro dans l'origine du plan complexe permet d'éliminer l'erreur en régime statique quand le signal de référence est un échelon. [2].

Avec la présence du terme intégrateur, le problème de saturation apparaît. Si la commande de l'actionneur est limitée à une valeur maximale (qui est presque toujours le cas), la sortie du correcteur peut se saturer. Le terme intégral peut devenir de plus en plus grand, et la commande se stabilise autour de la valeur de saturation. Même si la référence a changé, la sortie du correcteur reste saturée. Cette saturation va dégrader les performances du système. Un dépassement important ainsi qu'un temps de réponse très lent vont se manifester. Donc, lors de l'implémentation de cette action, il faut prendre en considération ce problème. La méthode la plus simple pour y remédier, est l'annulation du terme intégrateur de la commande lors de la saturation, et sa réinsertion en dehors de cette zone.

### 1.2.3.c Correction proportionnel dérivateur (PD)

L'action de dérivation est basée sur la prédiction de l'erreur du système dans le futur. Cette action peut être exprimée comme suit :

$$u(t) = T_d \frac{de(t)}{dt} \quad (1.6)$$

Où  $T_d$  est la constante de dérivation.

La fonction de transfert de l'action dérivée est donnée par :

$$C(p) = T_d p \quad (1.7)$$

En ajoutant l'action proportionnelle à l'équation (1.5), on obtient donc un régulateur proportionnel dérivé (PD). Sa loi de commande est donnée par :

$$u(t) = K_p \left( e(t) + T_d \frac{de(t)}{dt} \right) \quad (1.8)$$

L'action dérivation a un grand potentiel pour améliorer les performances du système, elle anticipe la tendance de l'erreur et essaye de corriger. Mais son implémentation pose quelques problèmes, il faut bien filtrer la sortie du processus pour éviter les composantes hautes fréquentes (bruit), qui ont tendance à emmener le terme dérivateur de la commande à l'infini [3].

### 1.2.3.d Correction proportionnel intégral dérivé (PID)

Un régulateur PID est obtenu par l'association de ces trois actions et il remplit essentiellement les fonctions suivantes :

- Fonction proportionnelle donne un système plus précis et plus rapide.
- Fonction intégrale utilisée pour améliorer la précision et éliminer l'erreur statique.
- Fonction dérivée principalement utilisée pour améliorer la stabilité, anticiper les variations de la sortie et accélérer la correction.

Sa loi de commande est donnée par :

$$u(t) = K_p \left[ e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right] \quad (1.9)$$

Avec l'écart défini comme suit :

$$e(t) = r(t) - y(t) \quad (1.10)$$

Sa fonction de transfert s'écrit :

$$G(p) = \frac{u(p)}{e(p)} = K_p \left[ 1 + \frac{1}{T_i p} + T_d p \right] \quad (1.11)$$

Le PID classique a des limites malgré ses bonnes performances [4]. Les performances du système seront dégradées lorsque les perturbations varient d'une manière importante.

Il existe plusieurs méthodes pour déterminer les paramètres des régulateurs P, PI, et PID. nous présentons, dans le paragraphe suivant la méthode de Ziegler et Nichols.

### 1.2.4 Méthodes de Ziegler et Nichols (ZN)

Pour l'ajustement des paramètres du contrôleur PID la méthode la plus utilisée dans plusieurs applications est celle de Ziegler-Nichols. Cette méthode a deux approches qui diffèrent en fonction de la nature du système [5].

En 1942, Ziegler et Nichols [6] ont proposé deux approches heuristiques basées sur leur expérience et quelques simulations pour ajuster rapidement les paramètres des régulateurs P, PI et PID. La première méthode nécessite l'enregistrement de la réponse indicielle en boucle ouverte, alors que la deuxième demande d'amener le système bouclé à sa limite de stabilité.



1.2.4.a Méthode de la réponse indicielle

Dans cette méthode il suffit d’injecter un signal échelon dans l’entrée du processus en boucle ouverte et de récupérer sa réponse indicielle. Ensuite, on trace la tangente au point d’inflexion de la courbe et on mesure le temps de retard  $T_u$ , le temps de montée de la tangente  $T_a$ , la différence entre l’amplitude initiale et l’amplitude finale de la sortie du processus  $\Delta_y$ , et la différence entre l’amplitude initiale et l’amplitude finale de l’échelon  $\Delta_u$  (Fig. 1.4)

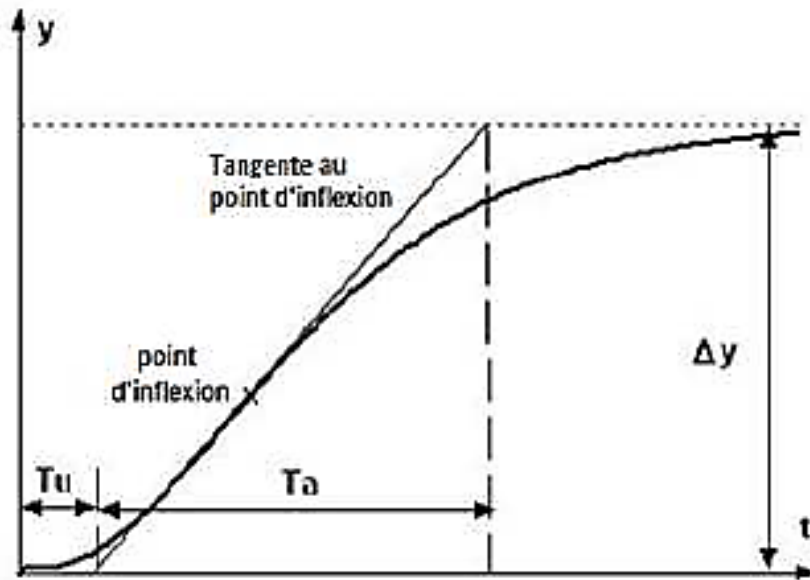


Figure 1.3 : Méthode de la réponse indicielle.

Ziegler et Nichols ont établi le tableau suivant pour calculer les paramètres des régulateur P, PI ou PID à partir de ces mesures ( $T_u, T_a, \Delta_y, \Delta_u$ )

Régulateur	$K_p$	$T_i$	$T_d$
P	$\frac{T_a}{T_u} \cdot \frac{1}{K}$	/	/
PI	$\frac{T_a}{T_u} \cdot \frac{0.9}{K}$	$3.33T_u$	/
PID	$\frac{T_a}{T_u} \cdot \frac{1.2}{K}$	$2T_u$	$0.5T_u$

Tableau 1.1: Calcul des paramètres des régulateur P, PI ou PID en utilisant la méthode de la

Réponse indicielle ( $K = \frac{\Delta_y}{\Delta_u}$ )

### 1.2.4.b Méthode du point critique :

Cette méthode est basée sur la connaissance du point critique du processus. Expérimentalement, on boucle le processus sur un simple régulateur proportionnel dont on augmente le gain jusqu'à amener le système à osciller de manière permanente ; on se trouve ainsi à la limite de stabilité [7]. Après avoir relevé le gain critique  $K_u$  du régulateur et la période d'oscillation  $T_u$  de la réponse, on peut calculer les paramètres du régulateur choisi à l'aide du (tableau 1. 2). Ici également, les valeurs proposées conduisent à un temps de montée relativement court malheureusement assorti d'un dépassement élevé. Cette situation n'étant pas toujours satisfaisante, on peut être amené à corriger les coefficients proposés et, en particulier, à diminuer le gain  $K_p$ .

Régulateur	$K_p$	$T_i$	$T_d$
<b>P</b>	$0.5K_u$	/	/
<b>PI</b>	$0.45K_u$	$0.85T_u$	/
<b>PID</b>	$0.6K_u$	$0.5T_u$	$0.12T_u$

**Tableau 1.2 :** Calcul des paramètres des régulateur P, PI ou PID en utilisant la méthode du point critique.

### 1.2.5 PID numérique

Pour implémenter le correcteur dans un calculateur numérique, il faut discrétiser les résultats obtenus en continu pour avoir la version numérique de la fonction du transféré [8]. Donc, on doit discrétiser le terme dérivateur présent dans l'équation (1.11).

Si la période d'échantillonnage  $T_0$  est petite et le bruit dans le signal de sortie est bien filtré, on peut remplacer la dérivée par une différence du premier ordre :

$$\frac{de}{dt} \approx \frac{e(k) - e(k-1)}{T_0} = \frac{\Delta e(K)}{T_0} \quad (1.12)$$

Pour le terme intégral, il y a trois approximations. La première consiste à remplacer ce terme par une somme des rectangles inférieurs (FRM ; Forward Rectangular Method) cela va donner :

$$\int_0^t e(\tau) d\tau \approx T_0 \sum_{i=0}^K e(i-1) \quad (1.13)$$

L'équation du PID discrétisée est alors donnée par :

$$u(k) = K_p \left\{ e(k) + \frac{T_0}{T_i} \sum_{i=1}^k e(i-1) + \frac{T_d}{T_0} [e(k) - e(k-1)] \right\} \quad (1.14)$$

Si le terme intégral est remplacé par une somme des rectangles supérieurs (BRM : Back Ward Rectangular Method) :

$$\int_0^t e(\tau) d\tau \approx T_0 \sum_{i=1}^K e(i) \quad (1.15)$$

L'équation du PID discrétisée sera donnée par :

$$u(k) = K_p \left\{ e(k) + \frac{T_0}{T_i} \sum_{i=1}^K e(i) + \frac{T_d}{T_0} [e(k) - e(k-1)] \right\} \quad (1.16)$$

La dernière approximation utilise la méthode des trapézoïdes (TRAP : Trapezoid Method), elle est la plus précise des trois. Le terme intégral est donné par :

$$\int_0^t e(\tau) \cdot d\tau \approx T_0 \sum_{i=1}^k \frac{e(i)+e(i-1)}{2} \quad (1.17)$$

Ce qui donne l'équation suivante pour le PID discrétisé :

$$u(k) = K_p \left\{ e(k) + \frac{T_0}{T_i} \left[ \frac{e(0)+e(k)}{2} + \sum_{i=1}^{k-1} e(i) \right] + \frac{T_d}{T_0} [e(k) - e(k-1)] \right\} \quad (1.18)$$

Si la période d'échantillonnage est suffisamment petite, on n'observe pas de différences (apparentes) entre les trois équations (1.14, 1.16 et 1.18). L'équation (1.18) est la plus utilisée en pratique. Les trois équations (1.14, 1.16 et 1.18) sont de nature non-récurrente, il faut connaître toutes les valeurs de l'erreur depuis le début de l'opération de contrôle, ce qui n'est pas pratique. C'est pour cette raison qu'on utilise la version récursive de ces trois équations.

Pour l'équation (1.16), sa version récursive est donnée par :

$$u(k) = \Delta u(k) + u(k-1) \quad (1.19)$$

$$\Delta u(k) = K_p \left\{ e(k) - e(k-1) + \frac{T_0}{T_i} e(k) + \frac{T_d}{T_0} [e(k) - 2e(k-1) + e(k-2)] \right\} \quad (1.20)$$

Pour avoir une seule expression du PID discret regroupant les trois méthodes, on utilise l'équation suivante :

$$u(k) = q_0 e(k) + q_1 e(k-1) + q_2 e(k-2) + u(k-1) \quad (2.21)$$

Les paramètres  $q_0$ ,  $q_1$  et  $q_2$  sont donnés dans le tableau suivant [9].

Paramètres du correcteur	FRM	BRM	TRAP
$q_0$	$K_p(1 + \frac{T_d}{T_0})$	$K_p(1 + \frac{T_d}{T_l} + \frac{T_d}{T_0})$	$K_p(1 + \frac{T_d}{2T_l} + \frac{T_d}{T_0})$
$q_1$	$-K_p(1 + \frac{T_d}{T_l} + \frac{2T_d}{T_0})$	$-K_p(1 + \frac{2T_d}{T_0})$	$K_p(1 + \frac{T_0}{2T_l} + \frac{2T_d}{T_0})$
$q_2$	$K_p \frac{T_d}{T_0}$	$K_p \frac{T_d}{T_0}$	$K_p \frac{T_d}{T_0}$

**Tableau. 1.3:** Expression des différents paramètres du PID numérique pour les différentes Méthodes d'approximation de l'intégral.

### 1.3 Conclusion :

Dans ce chapitre, nous avons donné un aperçu sur les différents correcteurs classiques. Nous avons aussi présenté la version numérique du régulateur PID. Cette dernière va servir dans la simulation est l'implémentation de la commande PID.

## **Chapitre 02**

# **Méthode d'optimisation: Teaching and Learning Based Optimization (TLBO)**

## 2.1 Introduction :

L'ajustage des paramètres d'un régulateur PID compte parmi les problèmes d'optimisations les plus étudiés. Il consiste à déterminer ces paramètres, tout en minimisant une fonction coût. Pour faire face à ces problèmes d'optimisation, des approximations et des algorithmes heuristiques ont été utilisés comme un compromis entre la qualité de la solution et le temps de calcul [10].

Dans ce mémoire nous avons opté d'utiliser l'algorithme d'optimisation Teaching and Learning Based Optimization (TLBO) pour déterminer les valeurs des paramètres du correcteur PID. L'algorithme TLBO nécessite seulement un nombre réduit de paramètres, telles que le nombre des élèves dans la classe et le nombre des matières enseignées aux élèves, pour son fonctionnement.

Le présent chapitre est consacré à la présentation de cet algorithme et les développements qui en découlent.

## 2.2 Problème d'optimisation :

Face à la résolution d'un problème d'optimisation, il est important de bien identifier à quelle catégorie ce problème appartient. En effet, les algorithmes développés sont conçus pour résoudre un type de problème donné et sont peu efficaces pour un autre type. La classification des problèmes d'optimisation change d'un auteur à l'autre. Par exemple, on distingue :

- **Les problèmes d'optimisation avec et sans contraintes**

Il est important de bien distinguer les problèmes où des contraintes existent sur les variables de décision. Ces contraintes peuvent être simplement des bornes et aller jusqu'à un ensemble d'équations de type égalité et de type inégalité. Il est parfois possible d'éliminer une contrainte égalité par substitution dans la fonction objective. Naturellement, les problèmes avec contraintes sont plus compliqués à résoudre et utilisent des algorithmes dédiés.

- **Les problèmes d'optimisation mono-objectif ou multi-objectif**

Les problèmes mono-objectifs sont définis par une unique fonction objective. Les problèmes multi-objectifs existent quand un compromis est à rechercher entre plusieurs objectifs contradictoires. Il est éventuellement possible (mais pas nécessairement efficace)

de reformuler un problème multi-objectif avec une seule fonction objectif sous forme d'une combinaison des différents objectifs ou en transformant des objectifs sous forme de contraintes.

## 2.3 Optimisation métaheuristique

### 2.3.1 Métaheuristique

Le terme méta-heuristique a été inventé par Fred Glover en 1986[11], lors de la conception de la recherche tabou. Les méta-heuristiques peuvent être considérées comme des algorithmes stochastiques itératifs, où ils manipulent une ou plusieurs solutions à la recherche de l'optimum. Les itérations successives doivent permettre de passer d'une solution de mauvaise qualité à la solution optimale. L'algorithme s'arrête après avoir atteint un critère d'arrêt, consistant généralement à atteindre le temps d'exécution ou une précision demandée. Ces méthodes tirent leur intérêt de leur capacité à éviter les optimums locaux, soit en acceptant des dégradations de la fonction objectif lors du traitement, soit en utilisant une population de points comme méthode de recherche. Les méta-heuristiques forment une famille d'algorithmes d'optimisation pour résoudre des problèmes d'optimisation difficiles, pour lesquels nous ne connaissons pas de méthodes classiques plus efficaces. Ils sont généralement utilisés comme des méthodes génériques qui peuvent optimiser un large éventail de problèmes différents, d'où le qualificatif de méta. Leur capacité à optimiser un problème à partir d'une quantité minimale d'informations est contrebalancée par le fait qu'ils n'offrent aucune garantie quant à l'optimalité de la meilleure solution trouvée. Cependant, du point de vue de la recherche opérationnelle, cette constatation n'est pas nécessairement un inconvénient, puisque l'on préfère toujours une approximation de l'optimum global trouvée rapidement à une valeur exacte trouvée dans un temps inacceptable [12].

### 2.3.2 Domaine d'application :

Ils sont extrêmement variés : optimisation d'un trajet, de la forme d'un objet, d'un prix de vente, d'une réaction chimique, du contrôle aérien, du rendement d'un appareil, du fonctionnement d'un moteur, de la gestion des lignes ferroviaires, du choix des investissements.

## 2.4 Teaching-Learning Based Optimization

TLBO est une méthode d'optimisation globale à l'origine développée par Rao et al. (Rao et al. 2011a ; Rao et al. 2012 ; Rao & Savsani 2012). C'est un algorithme inspiré du processus d'enseignement-apprentissage et est basé sur l'effet de l'influence d'un professeur sur la production des élèves dans une classe. L'algorithme décrit deux modes de base de l'apprentissage : (i) par l'intermédiaire de professeur, connu sous le nom de phase de professeur et (ii) par l'interaction avec les autres élèves, phase dite des élèves. Dans cet algorithme d'optimisation, un groupe des élèves est considéré comme population et les différentes matières proposés aux élèves sont considérés comme étant les variables du problème d'optimisation (les paramètres impliqués dans la fonction objectif) et le résultat d'un élève est considérée comme étant la valeur de la fonction fitness (fonction d'évaluation) [13]. La meilleure solution dans l'ensemble de la population, qui correspond à la meilleure valeur de la fonction objectif, est affectée au professeur.

### 2.4.1 Principe de l'algorithme TLBO :

Dans tous les problèmes d'optimisation il y a un certain nombre de paramètres à ajuster. Ces paramètres à ajuster dans la méthode TLBO sont constituées par les matières enseignées aux élèves et le résultat des élèves est la valeur de la fitness du problème d'optimisation. Cet algorithme est basé sur un groupe d'élèves constituant la population dans la méthode TLBO. On distingue deux phases dans la méthode TLBO : phase professeur (apprentissage par le professeur) et phase élèves (apprentissage par interaction d'élèves entre eux). Les sous-sections ci-dessous expliquent le principe de la méthode TLBO [14].

#### **Initialisation :**

$m$  : Nombre des matières enseignées aux élèves.

$n$  : Nombre des élèves dans la classe.

$i$  : Nombre des itérations.

$k$  : Variation de la population. ( $k = 1, 2, \dots, n$ ).



$j$  : Variation des variables de la conception. ( $j = 1, 2, \dots, m$ ).

$M_{j,i}$  : Résultat moyen de la classe dans le module particulier  $j$ .

$X_{total\_kbest,i}$  : Résultat du meilleur élève (le majorant).

La population est initialisée aléatoirement dans son espace de recherche en utilisant l'équation suivante :

$$X_{(j,i)} = X^{min}_j + rand \times (X^{min}_j - X^{max}_j) \quad (2.1)$$

Où  $X^{min}_j$  et  $X^{max}_j$  sont les valeurs minimales et maximales des variables de la conception.

### Phase du professeur :

C'est la première partie de l'algorithme d'optimisation. Pendant cette phase les élèves apprennent à travers le professeur, donc le professeur essaie d'améliorer la moyenne de la classe dans les matières enseignées par lui. Le meilleur apprenant identifié est considéré par l'algorithme comme le professeur. La différence entre le résultat de professeur pour chaque matière et le résultat moyen existant de chaque matière est donné par :

$$Difference\_Mean_{j,k,i} = r_i (X_{j,kbest,i} - T_F M_{j,i}) \quad (2.2)$$

Où,  $X_{j,kbest,i}$  est le résultat du meilleur élève dans la matière  $j$

$r_i$  : Nombre aléatoire entre 0 et 1

$T_F$  : Est le facteur d'enseignement qui décide la valeur de la moyenne à échangé, sa valeur est choisie aléatoirement, elle peut prendre les valeurs 1 ou 2. L'équation suivante est utilisée pour générer  $T_F$  :

$$T_F = round[1 + rand(0,1)\{2 - 1\}] \quad (2.3)$$

Sur la base de  $Difference\_Mean_{j,k,i}$ , la solution existante est mise à jour dans la phase de professeur selon l'expression suivante :

$$X'_{j,k,i} = X_{j,k,i} + Difference\_Mean_{j,k,i} \quad (2.4)$$

Où,  $X'_{j,k,i}$  est la valeur mise à jour de  $X_{j,k,i}$ .

$X'_{j,k,i}$  est accepté s'il nous donne la meilleure valeur de la fonction à optimiser. Ces valeurs deviennent l'entrée de la phase d'élèves. La phase d'élèves dépend de la phase de professeur.

### Phase d'élèves :

C'est la deuxième partie de l'algorithme où les élèves augmentent leurs connaissances par interaction entre eux. Un élève interagit de manière aléatoire avec d'autres élèves pour améliorer ses connaissances. Considérant une taille de population de 'N', le phénomène d'apprentissage de cette phase est expliqué ci-dessous :

Choisis aléatoirement deux élèves  $p$  et  $Q$  tels que  $X'_{total-p,i} \neq X'_{total-Q,i}$  (où,  $X''_{total-p,i}$  et  $X''_{total-Q,i}$  sont les valeurs mises à jour de  $X'_{total-p,i}$  et  $X'_{total-Q,i}$  de  $p$  et  $Q$  respectivement à la fin de la phase professeur)

$$X''_{j,p,i} = X'_{j,p,i} + r_i (X'_{j,p,i} - X'_{j,Q,i}), \text{ if } X'_{total-p,i} < X'_{total-Q,i} \quad (2.5)$$

$$X''_{j,p,i} = X'_{j,p,i} + r_i (X'_{j,Q,i} - X'_{j,p,i}), \text{ if } X'_{total-Q,i} < X'_{total-p,i} \quad (2.6)$$

$X''_{j,p,i}$  est accepté s'il nous donne la meilleure valeur de la fonction à optimiser.

Les équations (2.5) et (2.6) sont utilisées pour des problèmes de minimisation. Dans le cas de problèmes de maximisation, les équations. (2.7) et (2.8) sont utilisées.

$$X''_{j,p,i} = X'_{j,p,i} + r_i (X'_{j,Q,i} - X'_{j,p,i}), \text{ if } X'_{total-Q,i} < X'_{total-p,i} \quad (2.7)$$

$$X''_{j,p,i} = X'_{j,p,i} + r_i (X'_{j,p,i} - X'_{j,Q,i}), \text{ if } X'_{total-p,i} < X'_{total-Q,i} \quad (2.8)$$

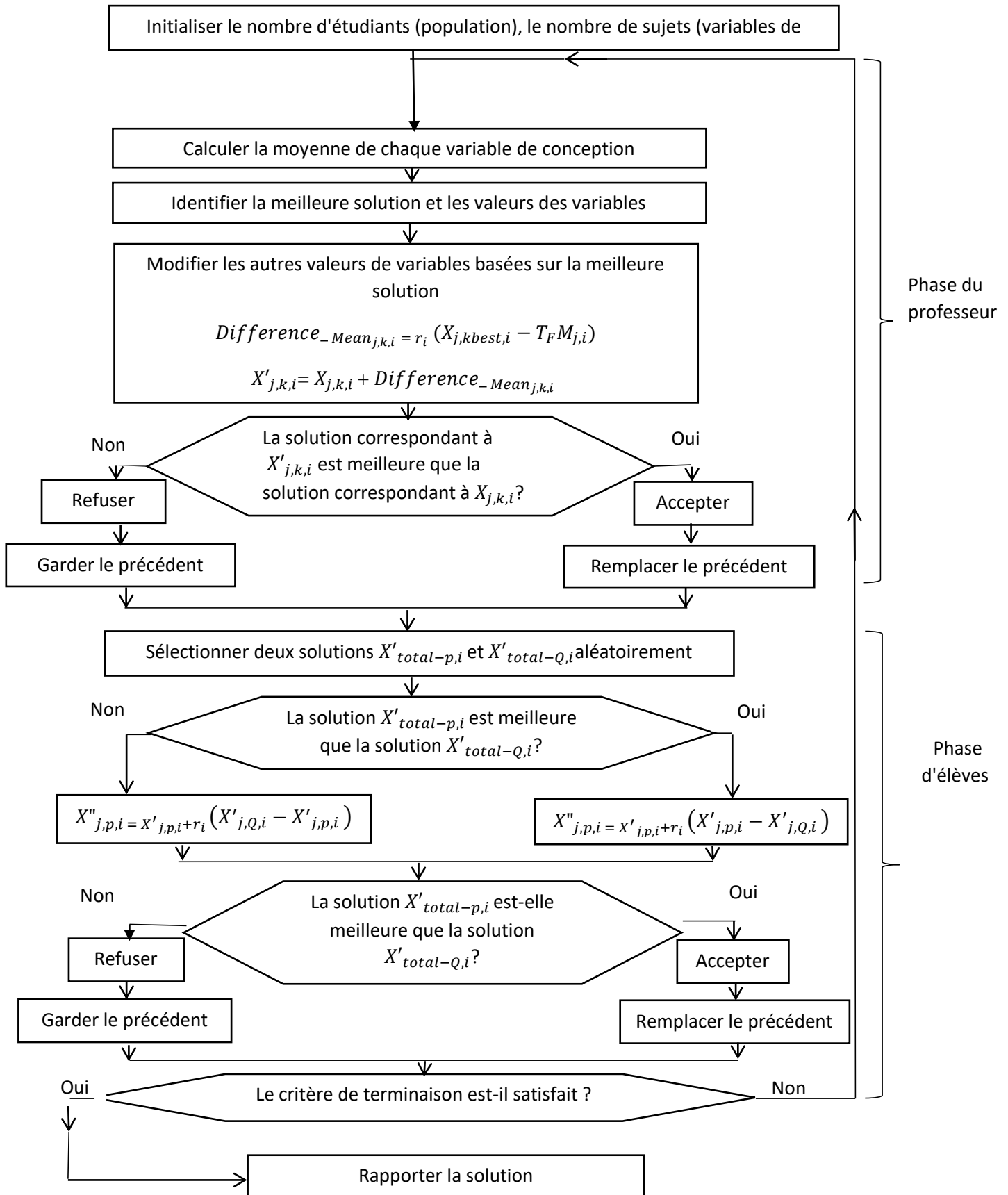


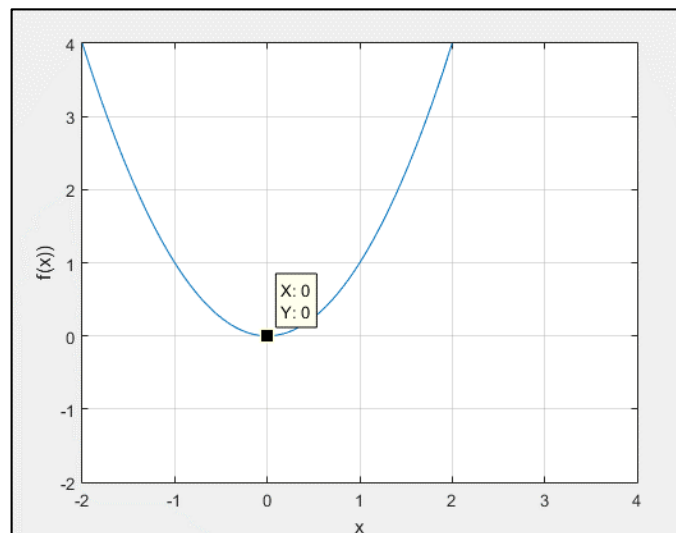
Figure 2.1 : Organigramme d'algorithme TLBO

## 2.5 Exemples d'optimisation en utilisant la méthode TLBO

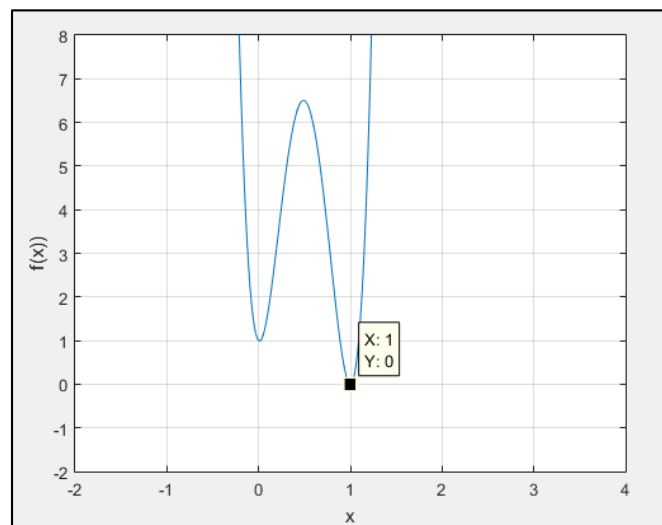
Dans cette partie, pour montrer l'efficacité de l'algorithme TLBO, nous avons utilisé le logiciel Matlab pour coder utilisons l'algorithme, ensuite nous avons considéré l'optimisation de plusieurs fonctions mathématiques. Les fonctions que nous avons considéré sont les suivantes :

$$f_1(x) = x^2 \quad \text{Min}(f_1) = 0 \quad \text{pour } x = 0 \quad (\text{figure 2.2})$$

$$f_2(x) = [100(x^2 - x)^2 + (1 - x)^2] \quad \text{Min}(f_2) = 0 \quad \text{pour } x = 1 \quad (\text{figure 2.3})$$



**Figure 2.2** courbe de la fonction coût  $f_1$



**Figure 2.3** courbe de la fonction coût  $f_2$

Les tableaux suivants illustrent les résultats de l'application de la méthode TLBO Pour trouver le minimum global des deux fonctions ( $f_1$ ) et ( $f_2$ ). Dans le premier cas nous avons

fixé le nombre d'itérations à 20 itérations et fait varier la taille de la population (tableau 2.1). Dans le deuxième cas, nous avons fixé la taille de la population à 5 étudiants et fait varier le nombre d'itérations (tableau 2.2).

Population	3		5		7	
Fonction	$f1$	$f2$	$f1$	$f2$	$f1$	$f2$
<b>F min</b>	8.7185e -36	0.09899	4.8824e -36	9.7835e-7	5.8062e -39	9.8914e-10
<b>Valeur de x</b>	9.3373e -17	0.102	2.2096e-18	0.88925	7.6198e-20	0.90183

**Tableau 2.1** Résultats d'optimisation en utilisant la méthode TLBO pour plusieurs valeurs de la population

Itérations	5		15		25	
Fonction	$f1$	$f2$	$f1$	$f2$	$f1$	$f2$
<b>F min</b>	5.8834e -36	9.8891 e-8	5.8062e -39	9.9183e -11	2.3821e -43	9.99e -14
<b>Valeur de x</b>	3.2098e-18	0.90512	7.6198e -20	0.99985	4.8807e -22	1.000

**Tableau 2.2** Résultats d'optimisation en utilisant la méthode TLBO pour plusieurs valeurs du nombre maximal d'itérations

Les résultats obtenus, montrent qu'il y a une amélioration de la précision lorsque on augmente le nombre d'itérations et la taille de la population.

### 2.5.1 Comparaison de TLBO avec l'algorithme PSO

Dans le tableau (2.3) nous avons donné les résultats de comparaison en considérant quatre fonctions de test (benchmark). Les résultats obtenus en utilisant l'algorithme PSO sont prises de la référence [15].

Fonction	L'algorithme TLBO	L'algorithme PSO
Sphere: $f_1 = x^2$	Min $f_1 = 0.00$ Pour $x = 0.00$	Min $f_1 = 7.96 \text{ e-}51$ Pour $x = 3.56 \text{ e-}50$
Rastrigin $f_2 = x^2 - 10\cos(2\pi x) + 10$	Min $f_2 = 6.77 \text{ e-}08$ Pour $x = 3.68 \text{ e-}07$	Min $f_2 = 5.82 \text{ e+}00$ Pour $x = 2.96 \text{ e+}00$
Ackley: $f_3 = -20e^{(-0.2\sqrt{x^2})} - e^{1/10(\cos 2\pi x)} + 20$	Min $f_3 = 3.55 \text{ e-}15$ Pour $x = 8.32 \text{ e-}31$	Min $f_3 = 1.58 \text{ e-}14$ Pour $x = 1.60 \text{ e-}02$
Griewank: $f_4 = \frac{1}{4000} x^2 - \cos\left(\frac{x}{\sqrt{x^2}}\right) + 1$	Min $f_4 = 0.00$ Pour $x = 0.00$	Min $f_4 = 9.69 \text{ e-}02$ Pour $x = 5.01 \text{ e-}02$

**Tableau 2.3** Résultats d'optimisation en utilisant la méthode TLBO et la méthode PSO

## 2.6 Conclusion :

Dans ce chapitre, nous avons présenté et étudié l'algorithme d'optimisation TLBO. Les résultats d'optimisation des fonctions de test considérées ont montrés clairement l'intérêt d'utiliser l'algorithme TLBO dans la recherche d'un optimum global. La comparaison effectuée montre que l'algorithme TLBO donne une meilleure précision que l'algorithme PSO.

## **Chapitre 03**

# **Modélisation et commande du moteur à courant continu**

### 3.1 Introduction

La première étape de la conception du régulateur consiste à obtenir un modèle du processus physique pouvant être considéré comme résultant d'une modélisation utilisant les lois de la physique ou obtenu par identification. Ce modèle, est utilisé pour déterminer les paramètres du régulateur et simulation du comportement d'un système en boucle fermée.

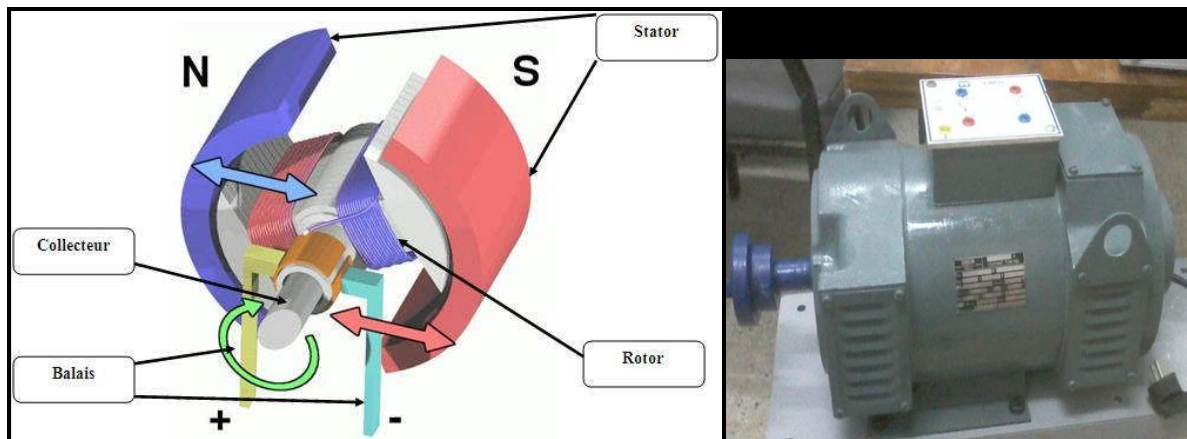
La première partie de ce chapitre est consacrée à la modélisation du moteur à courant continu, et de l'encodeur. Le calcul et l'ajustement des paramètres du correcteur PID sont considérés dans la deuxième partie. Pour mettre en évidence les performances de l'algorithme de commande, de nombreux résultats de simulation sont présentés à la fin de ce chapitre.

### 3.2 Moteur à courant continu

Le moteur à courant continu (MCC) est l'une des premières machines utilisées pour convertir l'énergie électrique en énergie mécanique. Il est constitué de trois parties principales (Figure. 3.1) :

- L'inducteur (stator) : Le stator est situé dans la partie fixe du moteur, son rôle sert à créer le champ magnétique dans le moteur. Le stator est un aimant ou électroaimant (bobinage parcouru par un courant continu).
- L'induit (rotor) : Le rotor est situé dans la partie tournante du moteur, il est constitué de plusieurs bobines identiques réparties uniformément autour d'un noyau cylindrique et disposées de telle façon que leurs deux cotés coupent respectivement le flux provenant d'un pôle nord et d'un pôle sud de l'inducteur.
- Le dispositif collecteur / balais : Le collecteur est situé dans la partie tournante du MCC, c'est un ensemble de lames de cuivre son rôle est de relier les extrémités du bobinage du rotor, les balais (charbons) sont situés dans la partie fixe du MCC leur rôle est d'alimenter les bobines du rotor en utilisant le frottement entre les lames du collecteur et les charbons.





**Figure 3.1 :** Moteur à courant continu

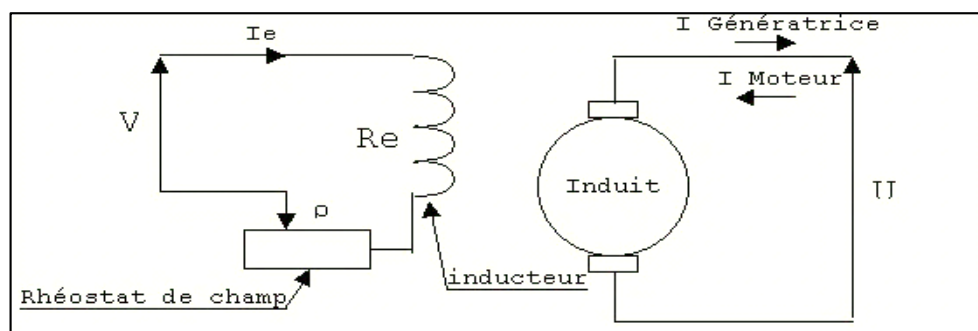
Le moteur à courant continu comporte deux parties électriques distinctes, le bobinage du rotor et son homologue dans le stator. Il existe plusieurs modes de connexions électriques :

- **Moteur à excitation séparée ou à aimant permanent :**

Dans un moteur à excitation séparée (figure 3.2), le stator et le rotor sont alimentés par deux sources distinctes. Les cas fréquents où la tension d'excitation est constante sont équivalents à ceux des moteurs à aimant permanent, dont le flux est constant.

Ce type de moteur peut fonctionner dans les quatre quadrants. Le passage d'un quadrant à un autre sans changement du signe de la vitesse sera imposé par la charge.

Tandis que les transitions horizontales entre quadrants seront obtenues par une inversion de la polarité de l'excitation [16].



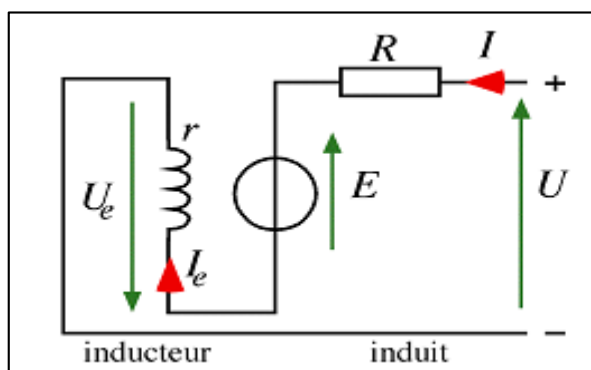
**Figure 3.2 :** Moteur à excitation séparée.

- **Moteur à excitation série :**

Un moteur à excitation série a son stator connecté en série avec le rotor (figure 3.3). Le bobinage inducteur comporte, dans ce cas, peu de spires, il est réalisé avec du fil de gros

diamètre. Cette conception lui procure une très bonne robustesse face aux vibrations et lui a valu un succès inégalé en traction ferroviaire.

Avec cette excitation, le moteur ne peut pas fonctionner comme génératrice [16].



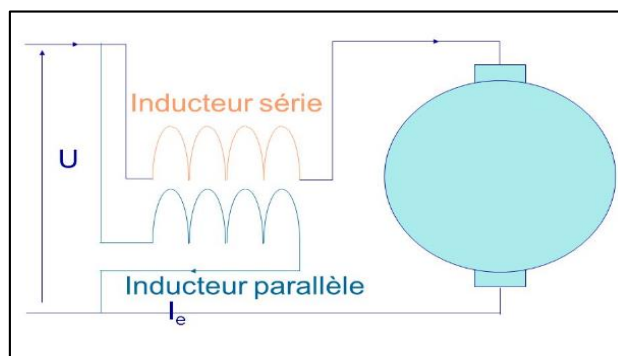
**Figure 3.3 :** Moteur à excitation série.

**- Moteur à excitation composée :**

Le bobinage du stator est divisé en deux parties, l'une connectée en série et l'autre en parallèle.

L'inducteur série est connecté de façon à diminuer le flux lorsque le courant d'induit augmente. Cet effet démagnétisant accélère l'arbre, en opposition avec l'effet de la résistance d'induit.

De nos jours, les moteurs à excitation composée (figure 3.4) ne sont presque plus utilisés, car la compensation qu'ils offrent dépend de leur état de saturation, de plus, l'électronique de puissance permet depuis longtemps de réguler très précisément la vitesse sans recourir à ces artifices [16].



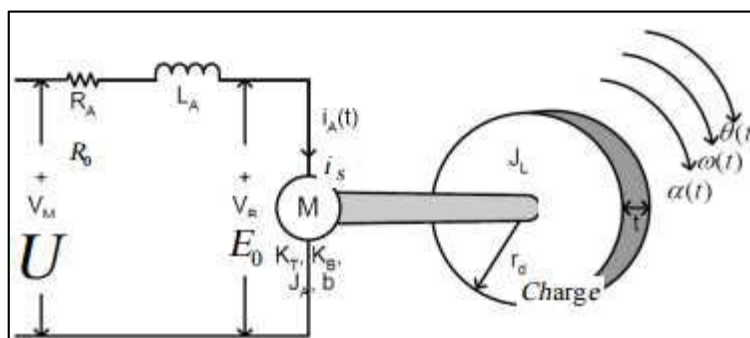
**Figure 3.4 :** Moteur à excitation composée.

Dans notre expérience nous avons utilisé un moteur à excitation séparée qui délivre une tension de 220 V et un courant continu de 6.8 A

### 3.3 Modélisation

#### 3.3.1 Modélisation du MCC

La fonction de transfert d'un moteur à courant continu englobe ses caractéristiques électriques ainsi que ses caractéristiques mécaniques. Cette fonction permet de simuler la réaction du moteur pour les différents stimulants existants dans son environnement (charge, tension...etc.). Le schéma électrique simplifié d'un MCC est donné par la figure 3.5.



**Figure 3.5 :** Schéma électrique simplifié d'un moteur DC en charge.

D'après la figure 3.5 en utilisant la loi de Kirchoff on aura :

$$u(t) = i_A(t)R_0 + L_A \frac{di_A}{dt} + E_0(t) \quad (3.1)$$

D'autre part, on a :

$$E_0 = K_e w(t) = K_e \frac{d\theta(t)}{dt} \quad (3.2)$$

Où :

$u(t)$ : la tension d'entrée.

$K_e$ : La constante de la force contre-électromotrice ( $v. s/rad$ )

$w(t)$ : la vitesse de rotation angulaire du moteur en ( $rad \setminus s$ ).

$\theta(t)$  : la position angulaire du moteur en ( $rad$ ).

Pour la partie mécanique, nous avons les équations suivantes :

$$T(t) = K_c i_A(t) \quad (3.3)$$

$$T(t) = J_T \alpha(t) + b w(t) \quad (3.4)$$

$$T(t) = J_T \frac{dw(t)}{dt} + b \frac{d\theta(t)}{dt} \quad (3.5)$$

$$T(t) = J_T \frac{d^2\theta(t)}{dt^2} + b \frac{d\theta(t)}{dt} \quad (3.6)$$

Avec :

$T(t)$  : est le couple moteur

$K_c$  : est la constante du couple moteur en  $(N_m/A)$ .

$b$  : est le coefficient de frottement visqueux en  $(K_g m^2/s)$ .

$J_T$  : est l'inertie de la charge et du moteur en  $(K_g m^2)$ .

En utilisant la transformée de Laplace, nous pouvons déduire la fonction de transfert qui relie la position du MCC à la tension appliquée :

$$\frac{\theta(s)}{U(s)} = \frac{K_e}{s^3 J_T L_A + s^2 (J_T R_0 + b L_A) + s (b R_0 + K_e K_c)} \quad (3.7)$$

Pour avoir la relation entre la vitesse angulaire et la tension d'entrée il suffit de multiplier la fonction de transfert (3.7) par la variable  $s$ . Nous obtenons :

$$G(s) = \frac{\Omega(s)}{U(s)} = \frac{K_e}{s^2 J_T L_A + s J_T R_0 + (b R_0 + K_e K_c)} \quad (3.8)$$

En utilisant l'approximation par un bloqueur d'ordre zéro  $(\frac{z^{-1}}{z \times T_e})$ , on obtient une version discrète de la fonction de transfert (3.8) de la forme :

$$G(z) = \frac{\alpha z + \beta}{z^2 + \gamma z + \delta}$$

Les valeurs des coefficients  $(\alpha, \beta, \gamma, \delta)$  ont été calculées directement en utilisant les valeurs des paramètres du moteur sur lequel nous avons travaillé. La fonction de transfert obtenue est donnée par :

$$G(z) = \frac{4.4z + 0.2909}{z^2 - 0.3111z + 0.00001984} \quad (3.9)$$

A partir de (3.9) nous pouvons obtenir l'équation aux différences décrivant le moteur, elle est donnée par :

$$\Omega(k) = 0.992\Omega(k-1) - 0.00067183\Omega(k-2) + 0.7434U(k-1) + 0.01155U(k-2)$$

### 3.3.2 Modélisation de l'encodeur

L'encodeur a pour rôle de convertir la vitesse angulaire (entrée)  $rad/s$  en un nombre par seconde (sortie). Chaque rotation de  $2\pi$  correspond à un nombre bien déterminé d'impulsions.

$$N_{imp/s} = \frac{\omega(t)}{2\pi} N_{resolution} \quad (3.10)$$

$N_{imp/s}$ : Nombre d'impulsions enregistrées par seconde.

$N_{resolution}$ : Nombre d'impulsion résultant pour chaque rotation complète du rotor.

La fonction de transfert sera :

$$G_{znc}(s) = \frac{N_{imp/s}}{\omega(t)} = \frac{N_{resolution}}{2\pi} \quad (3.11)$$

Notre encodeur a une résolution de 32. La fonction de transfert de l'encodeur sera :

$$G_{enc}(s) = 50.24(impulsion \setminus Rad) \quad (3.12)$$

### 3.4 Hacheur

Un hacheur est un convertisseur statique qui permet de régler la tension ou le courant dans la charge (récepteur) à partir d'une source de tension continue. Il s'agit donc d'un convertisseur continu-continu qui sert dans les alimentations à découpage et pour alimenter les moteurs à courant continu dont on veut faire varier la vitesse.

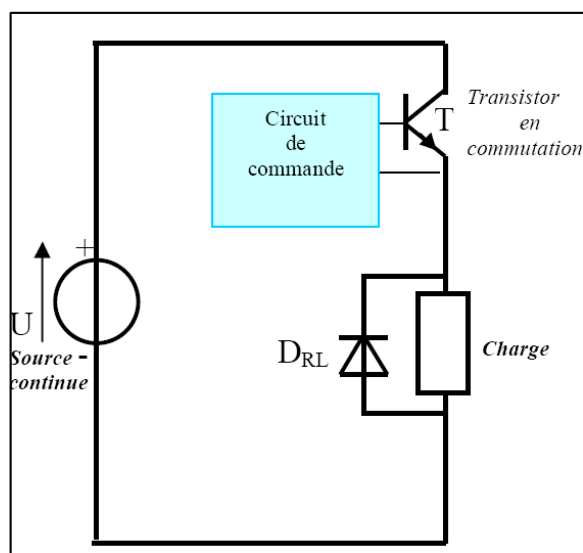
Le principe du hacheur consiste à établir puis interrompre périodiquement la liaison source charge à l'aide d'un interrupteur électronique. Celui-ci doit pouvoir être fermé ou ouvert à volonté, ce sera un thyristor ou un transistor de puissance fonctionnant en régime de commutation. Il sera commandé par un signal PWM (Pulse Width Modulation).

Le transistor (thyristor) fonctionne en commutation (tout ou rien), il est donc soit bloqué soit saturé. La tension de commande du transistor (reliée à la base) est une tension créneau (PWM), de fréquence et rapport cyclique variables indépendamment l'une de l'autre. Lorsque cette tension de commande est positive (niveau haut), elle rend le transistor passant et saturé, ainsi le courant va passer, la charge va sentir la tension  $U$ . Lorsque cette tension est nulle (niveau bas), le transistor sera bloqué, la charge ne reçoit plus de courant et la tension aux bornes de la charge sera nulle. Donc suivant le rapport cyclique du signal de commande, la charge va recevoir le courant (de tension  $U$ ) pendant un moment de temps, et sera bloquée pendant un autre. Mais comme la tension appliquée à la charge sera filtrée (soit par la charge elle-même ou par un circuit de filtrage monté en parallèle avec la charge), elle ne va pas sentir ces

interruptions, mais elle va sentir une tension moyenne tout le temps. Ainsi, pour changer cette tension (ressentie par la charge), il suffit de changer la valeur du rapport cyclique  $\alpha$  défini par :

$$\alpha = \frac{\text{durée de fermeture}}{\text{période}} = \frac{t_f}{T} \quad (3.13)$$

La figure (3.6) montre un schéma simple d'un montage où la charge est commandée (sa tension varie) par un transistor en commutation. La diode de roue libre  $RL D$  assure la continuité du courant dans la charge si celle-ci est inductive (bobine ou moteur à courant continu) quand le transistor est bloqué.

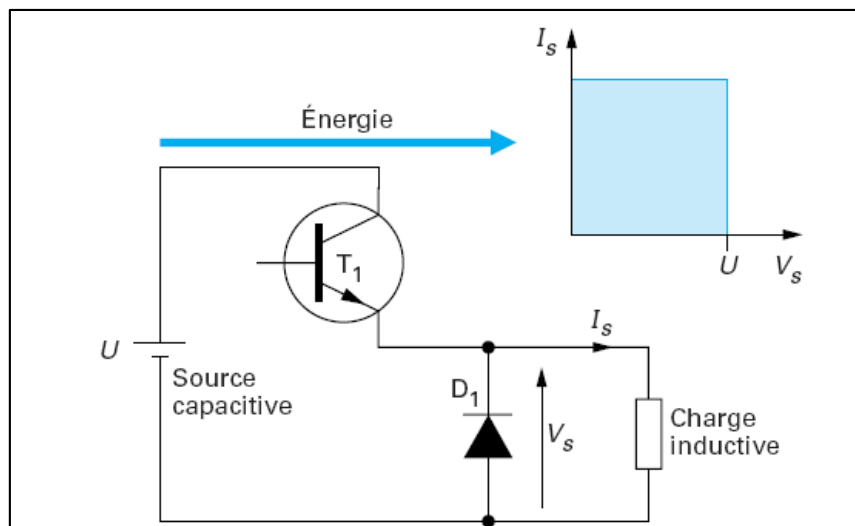


**Figure 3.6 :** Schéma simplifié d'une charge commandée par transistor [17].

Parmi les types de hacheurs

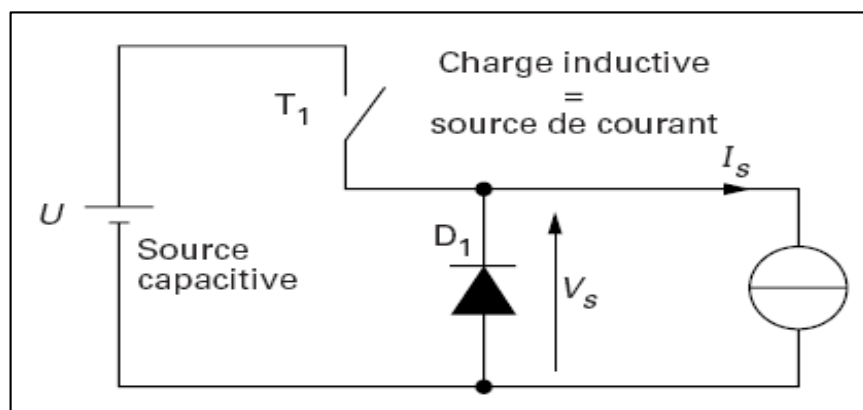
### 3.4.1 Hacheur abaisseur à un quadrant (Série)

Le schéma structurel du hacheur abaisseur à un quadrant est donné sur la figure (3.7) Cet hacheur utilise un couple transistor-diode (T1-D1), qui connecte périodiquement la source de tension vers la charge. L'énergie transite de façon unidirectionnelle depuis la source de tension vers la charge, comme l'indique le quadrant de fonctionnement représenté dans le plan  $(V_s, I_s)$  de la figure (3.7).



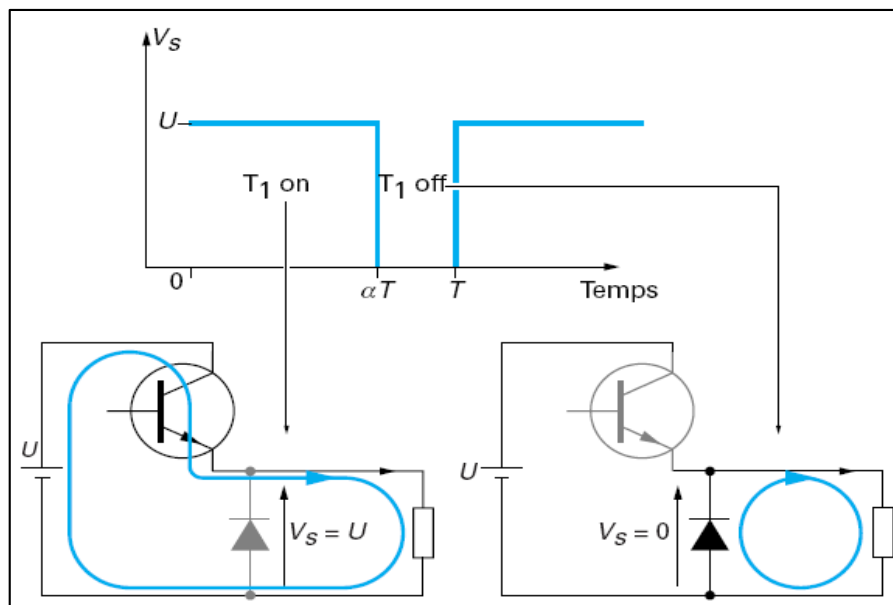
**Figure. (3.7) :** Hacheur abaisseur à un quadrant [17].

Remplaçons chacun des éléments du convertisseur par son homologue idéal. Le transistor et la diode sont des interrupteurs idéaux (chute de tension et temps de commutation nuls) et la charge est remplacée par une source de courant réceptrice car les variations de courant y sont négligeables. Nous obtenons alors le schéma de la figure (3.8).



**Figure. (3.8) :** Représentation idéale d'un hacheur abaisseur à un quadrant

On peut remarquer la présence de deux états distincts dans le fonctionnement. Premièrement, si  $T_1$  est fermé et  $D_1$  est ouverte, et deuxièmement si  $T_1$  est ouvert et  $D_1$  est fermée. Dans le premier cas, la source de tension est connectée directement sur la charge, tandis que dans le deuxième cas, la source de tension d'entrée est au repos et la charge ne reçoit pas d'énergie. Nous retrouvons sur la figure (3.9) ces deux états stables du convertisseur. La diode  $D_1$  est dite diode de roue libre car elle écoule le courant de sortie lorsque l'interrupteur est ouvert.



**Figure. (3.9) :** Les deux états de fonctionnement du hacheur abaisseur

Les tensions obtenues sont discontinues lors de la permutation des états. En réalité la présence d'un circuit de filtrage va éliminer ces discontinués, et on aura un signal plus au moins constant. En fait, c'est ce signal qui est ressenti par la charge. Nous pouvons maintenant calculer la valeur moyenne de la tension de sortie  $V_s$ , que nous noterons  $V_{so}$ . En négligeant la durée des commutations, nous pouvons écrire :

$$\begin{aligned}
 V_{so} &= \frac{1}{T} \int_0^T V_s(t) dt \\
 V_{so} &= \frac{1}{T} \int_0^{\alpha T} U dt \\
 V_{so} &= \alpha U
 \end{aligned} \tag{3.14}$$

Cette valeur est indépendante du courant de charge  $i_s$ . Nous en concluons que le convertisseur étudié se comporte vis-à-vis de sa charge comme une source de tension variable, de valeur  $V_{so} = \alpha U$  [19].

### 3.4.2 Hacheurs à quatre quadrants

Pour commander un hacheur à quatre quadrants Il suffit de commander les interrupteurs en utilisant un signal *PWM* (Figure (3.10)) pour chaque bras. Quand le premier signal *PWM* attaque le premier bras, son inverse va attaquer le deuxième. Ainsi, Le bras de gauche voit la tension  $(1 - \alpha)U$ , tandis que celui de droite voit  $\alpha U$ . La charge voit alors une source réversible d'amplitude :

$$V_{so} = (1 - 2\alpha)U \tag{3.15}$$



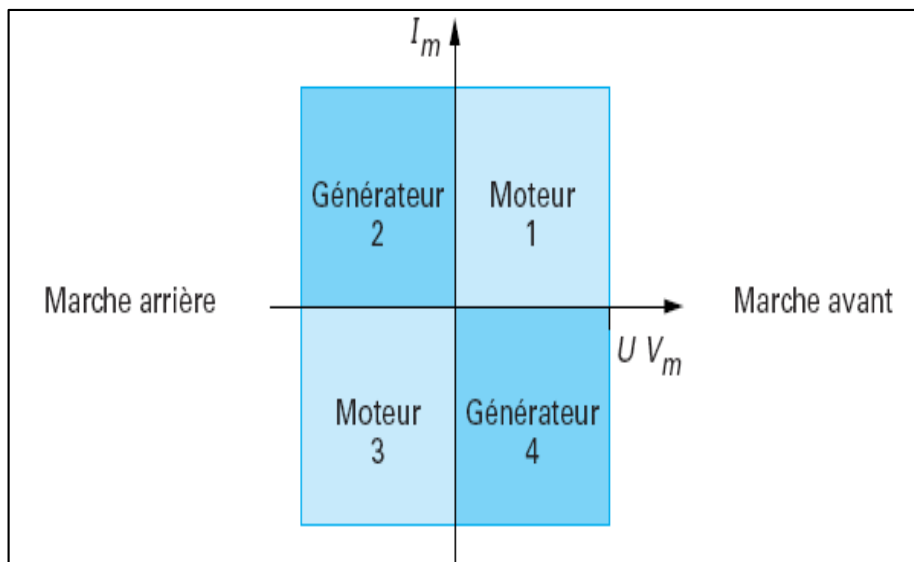


Figure 3.10 : Les quatre quadrants [18]

D'après l'équation (3.11), un rapport cyclique inférieur à 0.5 va résulter en une tension moyenne  $V_{so}$  positive, et un rapport cyclique supérieur à 0.5 va donner une tension moyenne  $V_{so}$  négative. Donc pour contrôler le sens de rotation d'un moteur, il suffit de varier le rapport cyclique selon le besoin. On peut toujours observer, qu'un rapport cyclique de 0.5, va générer une tension moyenne nulle, donc pour arrêter un moteur à quatre quadrants, il suffit de donner un rapport cyclique de 0.5 au signal de commande (PWM).

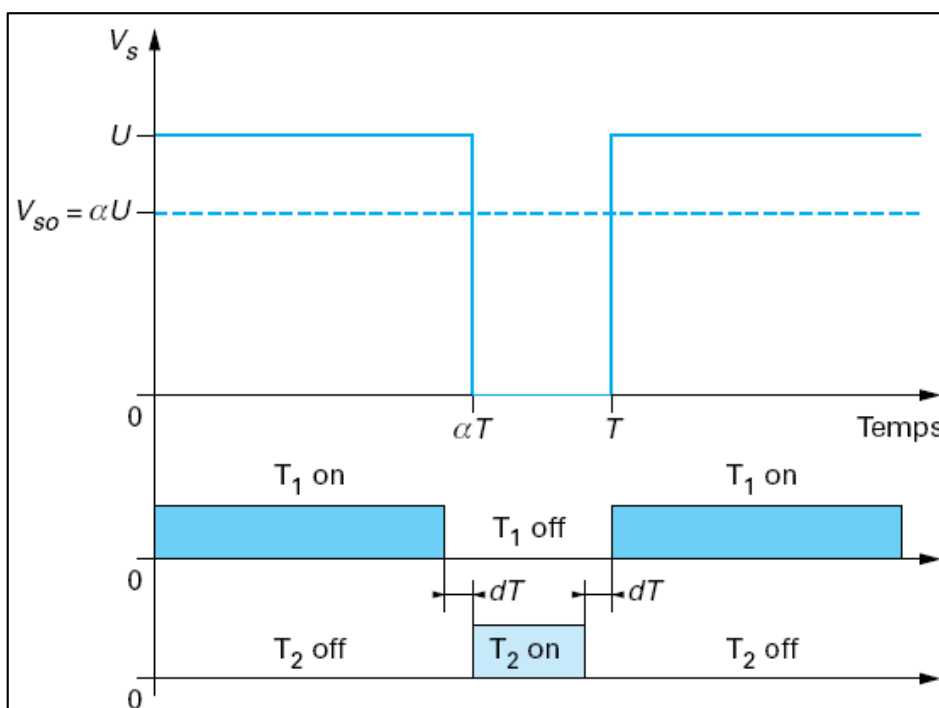


Figure 3.11: Commande en tension du hacheur [18]

### 3.5 Commande du moteur en utilisant un PID classique

Le schéma de commande est illustré par la figure 3.12. Nous utilisons pour le calcul des paramètres du PID la méthode TLBO. La réponse indicielle du MCC en boucle ouverte, simulée par Matlab, est donnée par la figure 3.13.

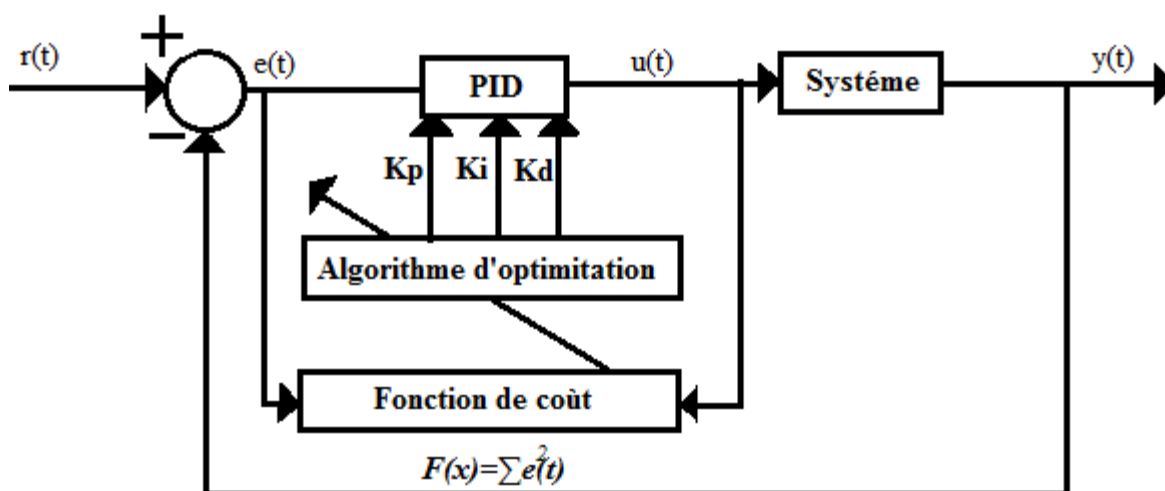


Figure 3.12 : Commande PID du MCC.

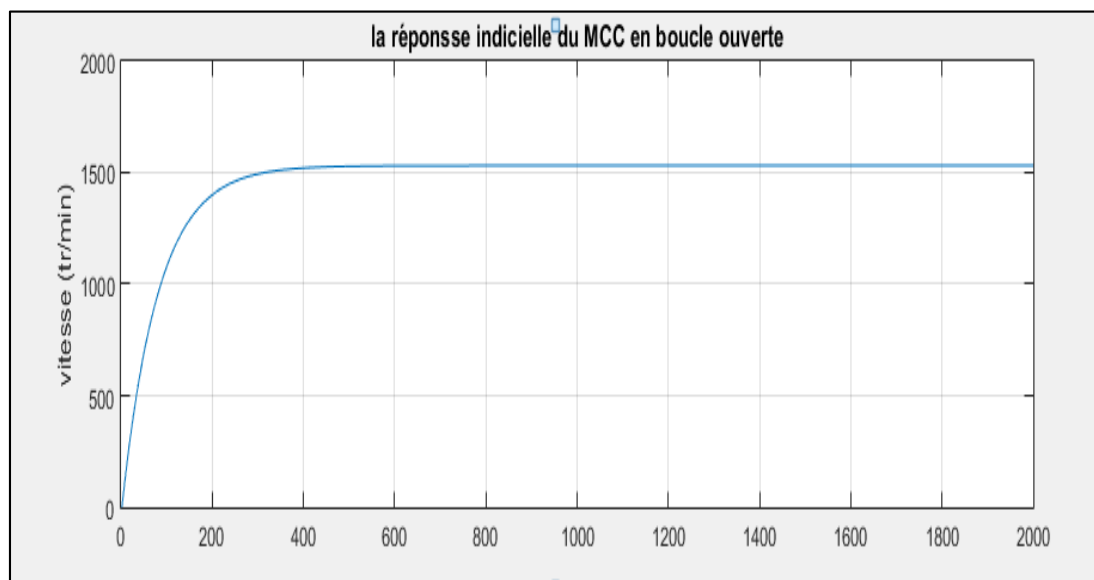


Figure 3.13 : la réponse indicielle du MCC en boucle ouverte

La fonction de coût à optimiser est donnée par :

$$F(t) = \sum_{1}^{1000} e^2(t) \quad (3.16)$$

Où :

$$e(t) = \text{réf}(t) - y(t)$$

réf :référence.

y : la sortie du système.

Les valeurs des paramètres de l'algorithme TLBO utilisées sont :

Taille de population :5

Nombre d'itérations :100

Nombre des variables de conception : 3

Nous avons obtenu pour les paramètres du PID les valeurs suivantes :

$$K_p = 0.0157$$

$$K_i = 0.0335$$

$$K_d = 0.0271$$

### 3.5.1 Simulation sans perturbation

Les résultats de commande en utilisant le PID calculé pour la référence choisie sont donnés par la figure 3.14. L'écart entre la vitesse du moteur et le signal de référence est donnée par la figure 3.15.

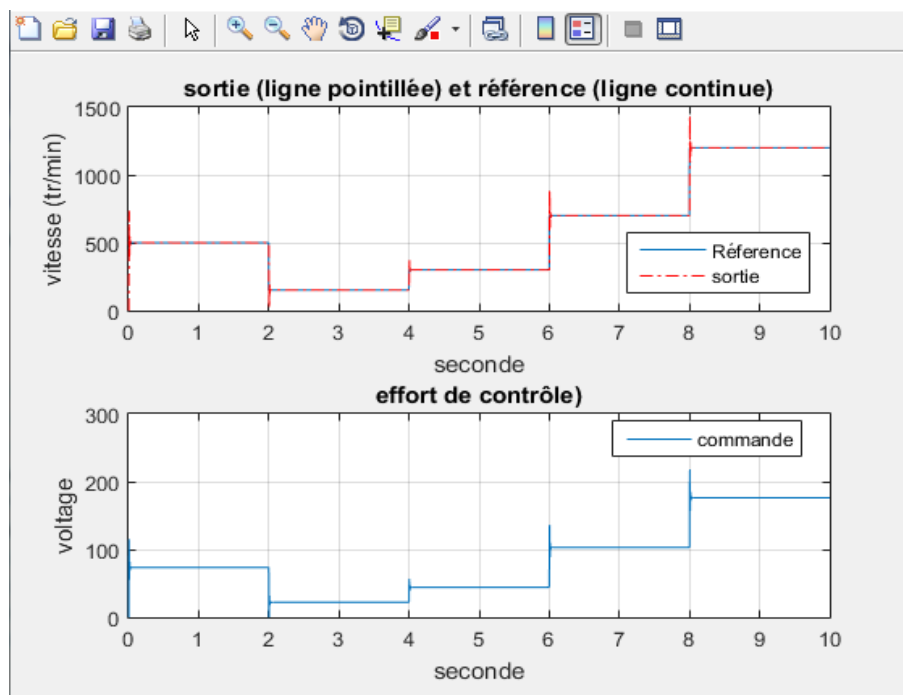


Figure 3.14 : Référence, vitesse de rotation du MCC et signal de commande délivré par le PID.

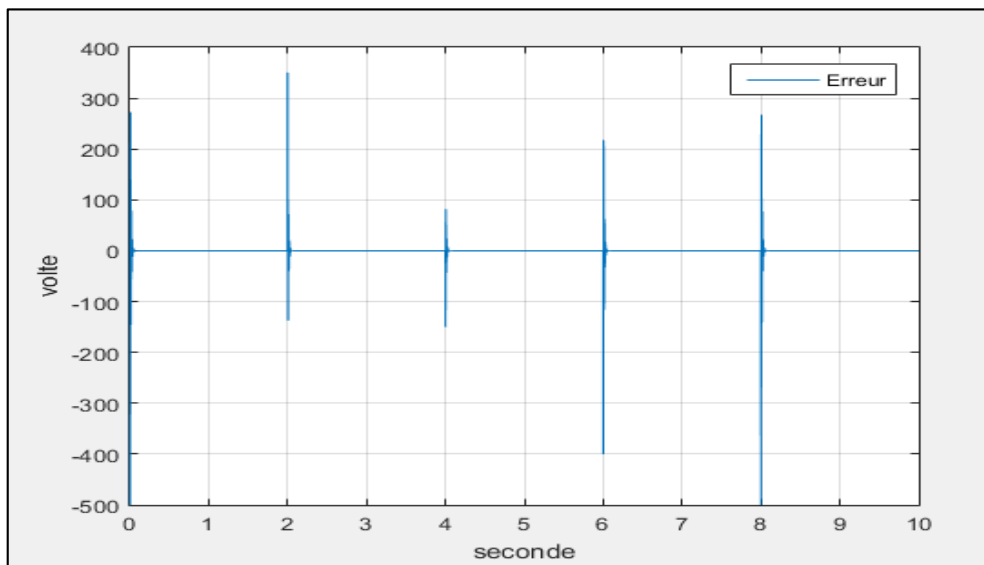


Figure 3.15 : Ecart entre la sortie réelle du MCC et la référence.

A partir des figures 3.14 et 3.15 nous remarquons que la vitesse de rotation du moteur suit la référence jusqu'à atteint 1200 tr /min et qu'il y a des dépassements de faible d'amplitude.

### 3.5.2 Simulation avec perturbation additive en sortie

Nous introduisons une perturbation, additive en sortie, de 50 % durant l'intervalle [4 s 8s], pour compenser l'effet des perturbations nous avons utilisé l'algorithme d'optimisation les résultats obtenus sont donnés par les figures 3.16 et 3.17.

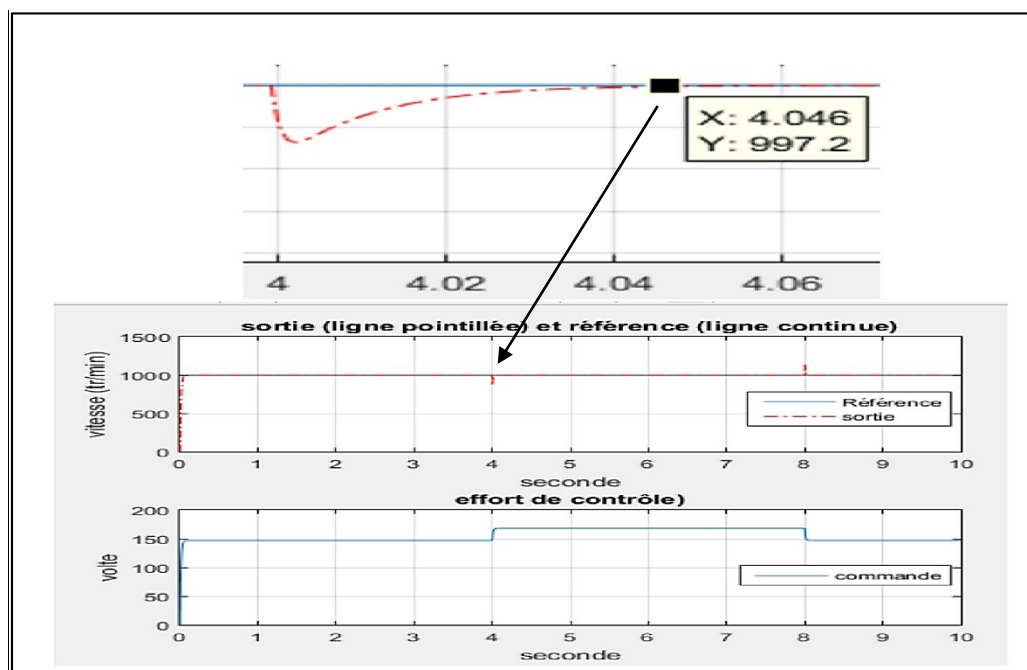


Figure 3.16 : Référence, vitesse de rotation du MCC et signal de commande délivré par le PID (cas avec perturbation 50%).

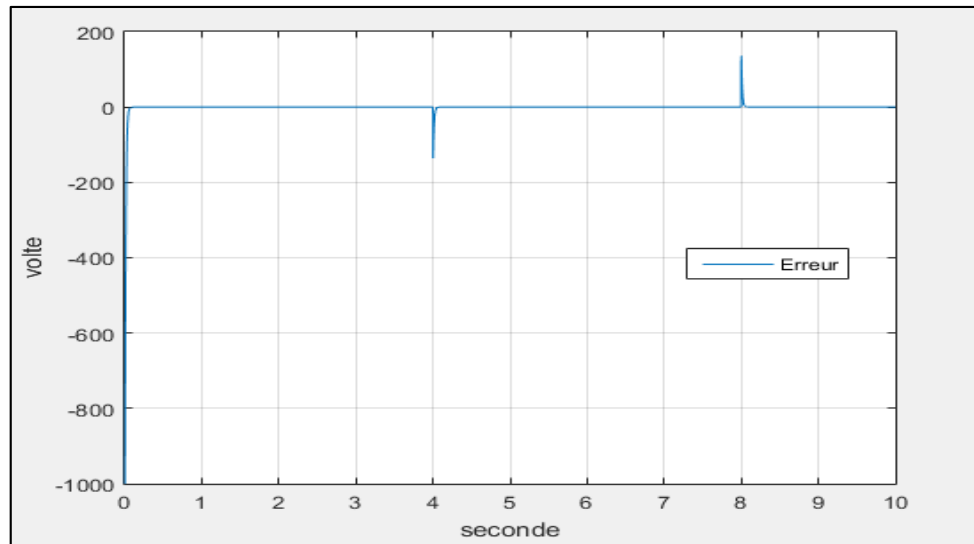


Figure 3.17 : Ecart entre la sortie réelle du MCC et la référence.

A partir des figures 3.16 et 3.17 nous remarquons que le PID met un temps de l'ordre de 46ms pour compenser l'effet de la perturbation en sortie du système.

### 3.5.3 Simulation avec perturbation additive en entrée

Une perturbation de 25% additive en entrée du système est introduite durant l'intervalle [4s 8s]. A partir des résultats obtenus (figures 3.18 et 3.19), nous remarquons que l'effet de la perturbation a été compensée par le régulateur.

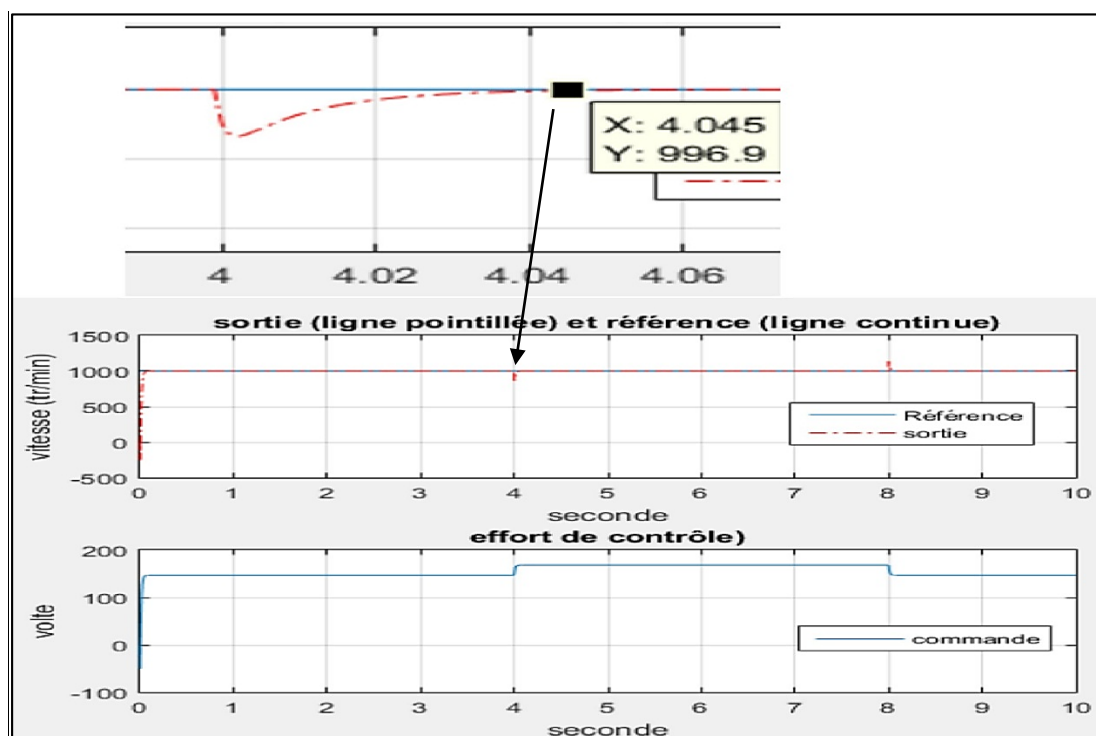
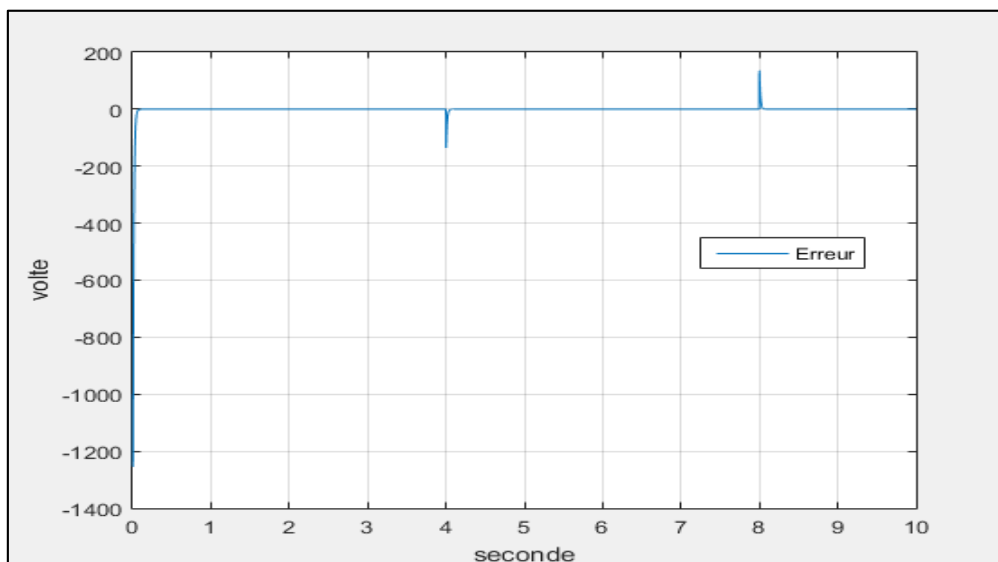


Figure 3.18: Référence, vitesse de rotation du MCC et signal de commande délivré par le PID (cas avec perturbation 25%).



**Figure 3.19 :** Ecart entre la sortie réelle du système perturbé et la référence.

A partir des figures 3.18 et 3.19 nous remarquons que le PID met un temps de l'ordre de 45ms pour compenser l'effet de la perturbation en entrée du système.

### 3.6 Commande du moteur en utilisant un PID auto ajustable

Dans ce cas les paramètres du PID sont adaptés en ligne en utilisant l'algorithme TLBO.

Les valeurs des paramètres de l'algorithme d'optimisation sont :

Taille de population : 6

Nombre d'itérations : 10

Nombre des variables de conception : 3

Les valeurs initiales des variables de conception ( $K_p$   $K_i$   $K_d$ ) sont choisies aléatoirement.

La fonction de coût à optimiser est donnée par :

$$F(t) = \sum_1^{1000} e^2(t) \quad (3.17)$$

#### 3.6.1 Simulation sans perturbations

Pour compenser l'effet des perturbations nous avons utilisé l'algorithme d'optimisation pour adapter en temps réel les valeurs des paramètres  $K_p$   $K_d$   $K_i$  du PID. Les variations des ces paramètres sont représentés sur la figure 3.20.

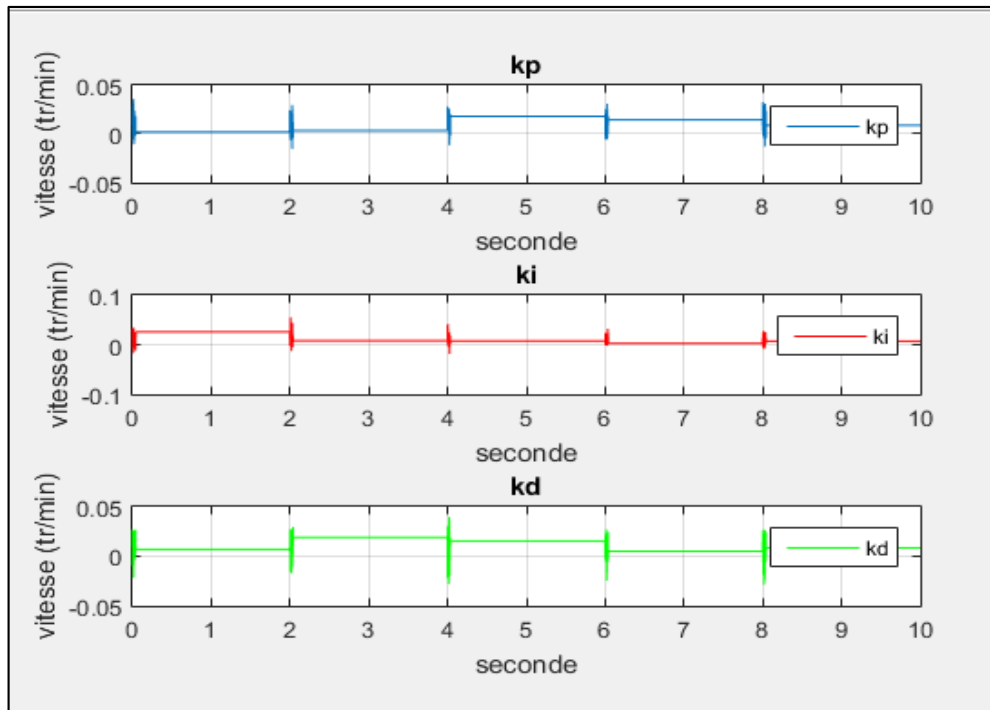


Figure 3.20 : réel les valeurs des paramètres  $K_p$   $K_d$   $K_i$  du PID.

La figure 3.21 représente la référence utilisée, la vitesse du MCC obtenue et la commande délivrée par le régulateur PID auto ajustable. L'écart de poursuite est donné par la figure 3.22.

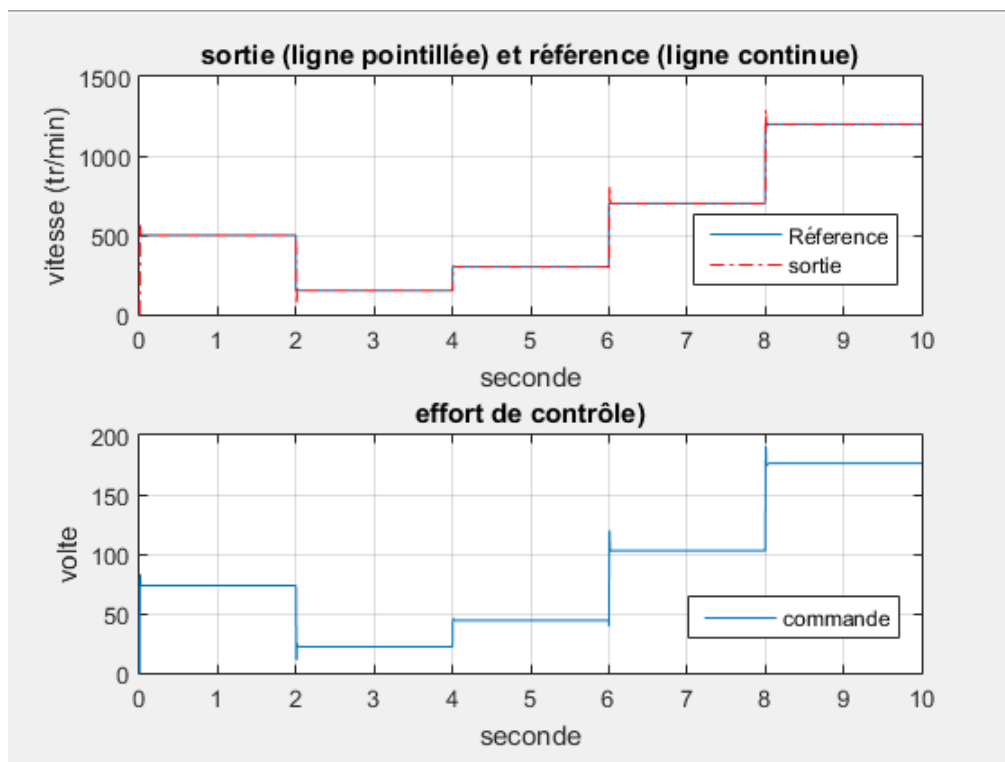
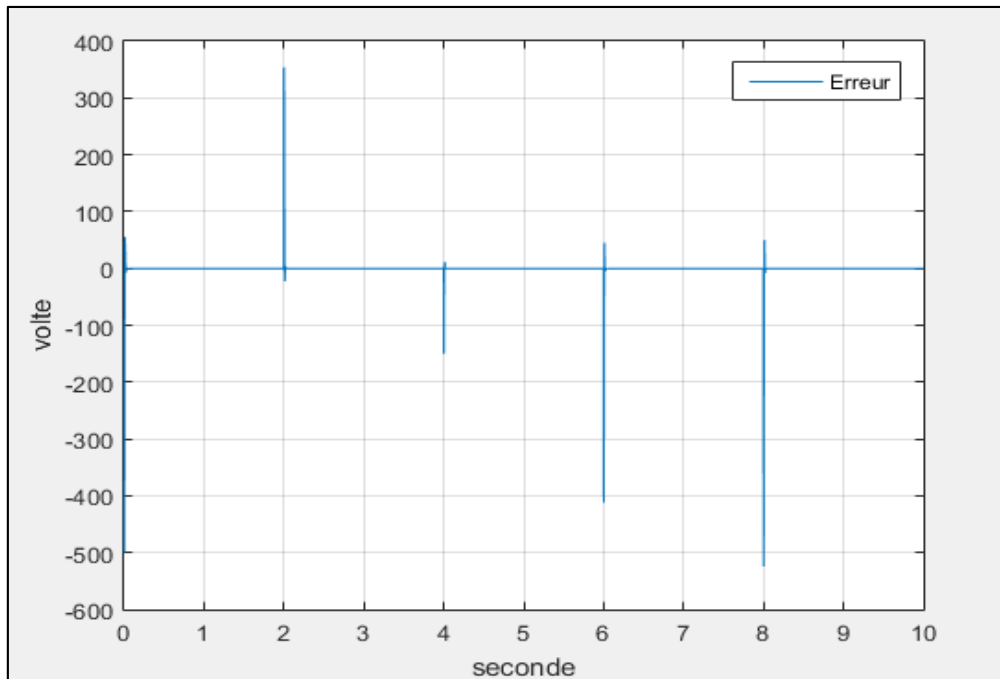


Figure 3.21 Référence, vitesse de rotation du MCC et signal de commande délivré par le PID auto ajustable.

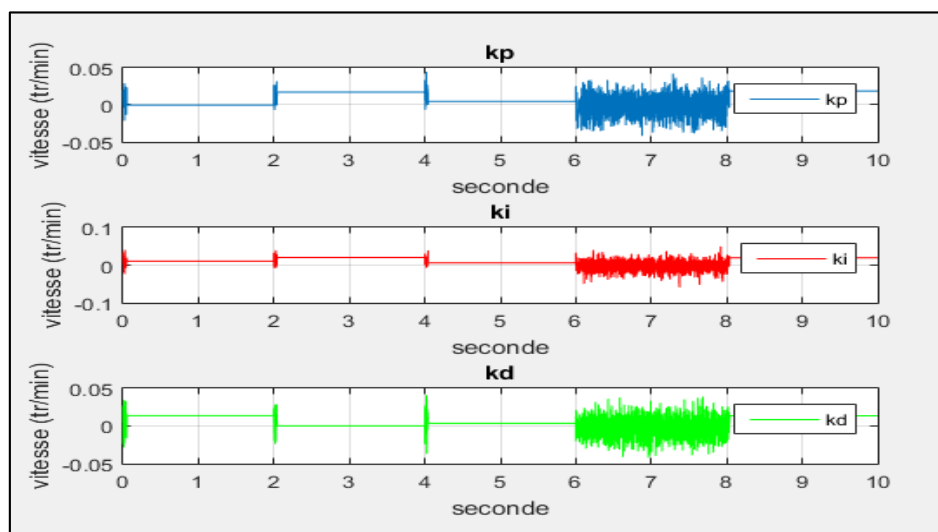


**Figure 3.22:** Ecart entre la sortie réelle et la référence.

A partir des figures 3.21 et 3.22 nous pouvons conclure qu'il y a une bonne poursuite de la trajectoire de référence. Nous remarquons aussi qu'il n'y a pas des oscillations rapides sur le signal de commande

### 3.6.2 Simulation avec une perturbation additive en sortie

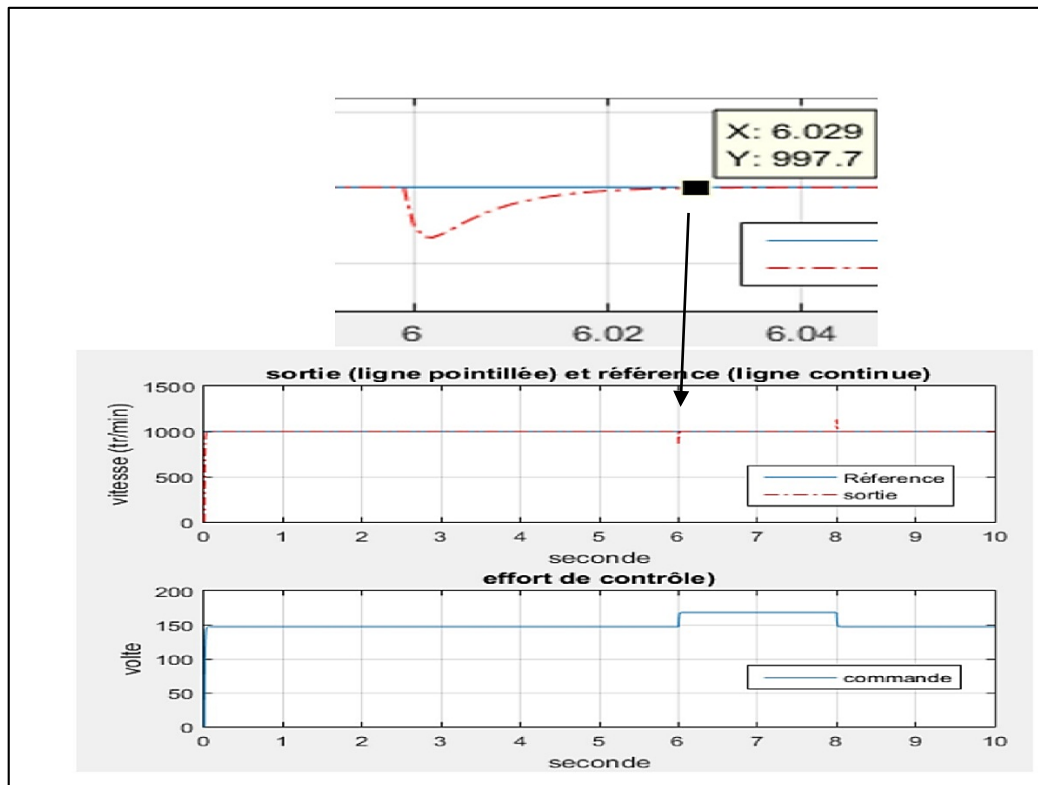
Pour compenser l'effet des perturbations nous avons utilisé l'algorithme d'optimisation pour adapter en temps réel les valeurs des paramètres  $K_p$   $K_d$   $K_i$  du PID. Les variations de ces paramètres sont représentées sur la figure 3.23.



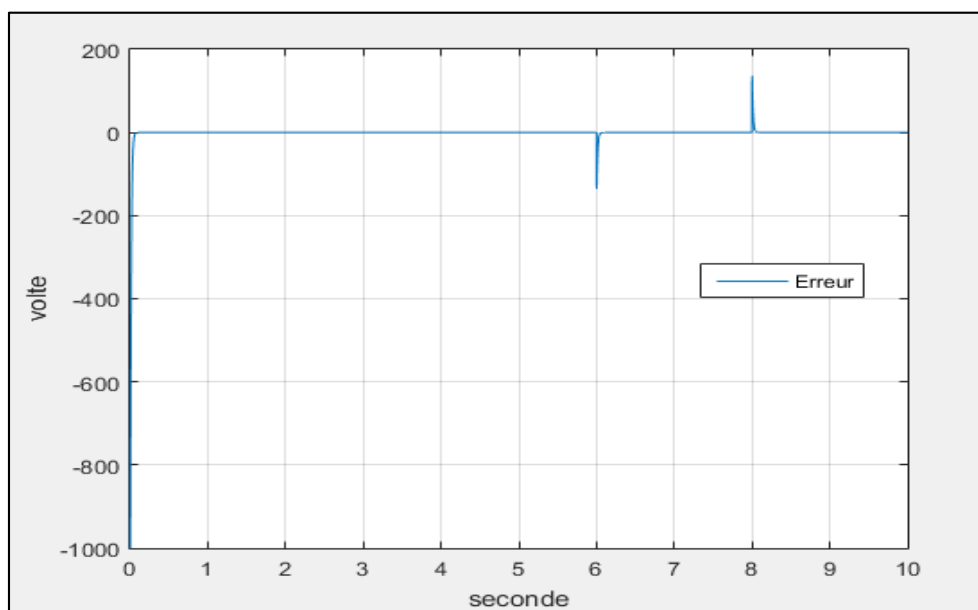
**Figure 3.23 :** réel les valeurs des paramètres  $K_p$   $K_d$   $K_i$  du PID.



Une perturbation de 50% additive en sortie du système est introduite durant l'intervalle [6s, 8s]. Les résultats obtenus sont donnés par les figures 3.24 et 3.25, nous remarquons que le régulateur agit rapidement pour compenser l'effet de la perturbation.



**Figure 3.24 :** Référence, vitesse de rotation du MCC et signal de commande délivré par le PID auto ajustable (cas avec perturbation 50%).



**Figure 3.25 :** Ecart entre la sortie réelle du système perturbé et la référence.

A partir des figures 3.24 et 3.25 nous remarquons que le PID auto ajustable met un temps de l'ordre de 29ms pour compenser l'effet de la perturbation en sortie du système.

### 3.6.3 Simulation avec une perturbation additive en entrée

Une perturbation de 25% additive en entrée du système est introduite durant l'intervalle [6s 8s].

Les variations des paramètres du PID sont représentés sur la figure 2.26.

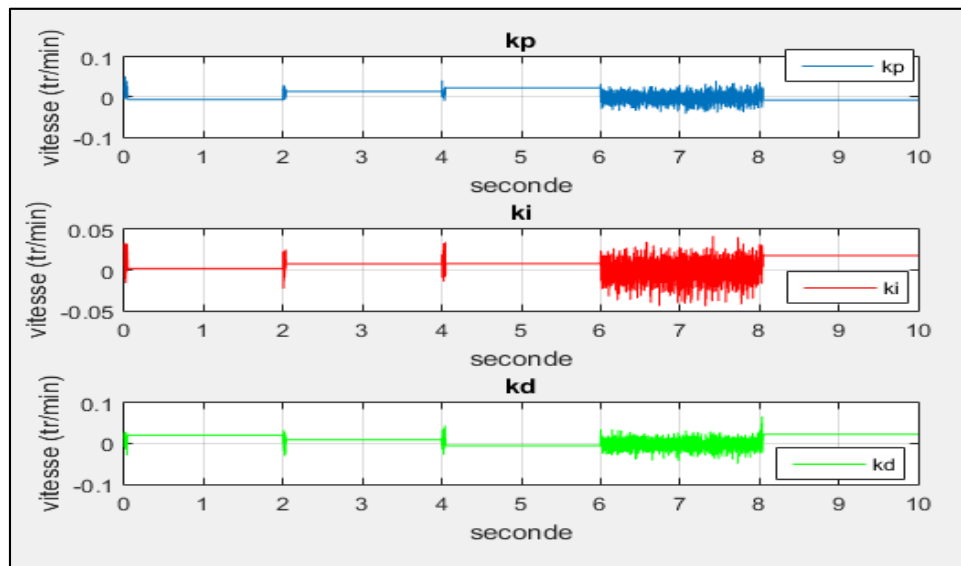


Figure 3.26 : réel les valeurs des paramètres  $K_p$   $K_d$   $K_i$  du PID.

Les résultats obtenus sont donnés par les figures 3.27 et 3.28, nous remarquons que l'effet de la perturbation a été compensé par le régulateur.

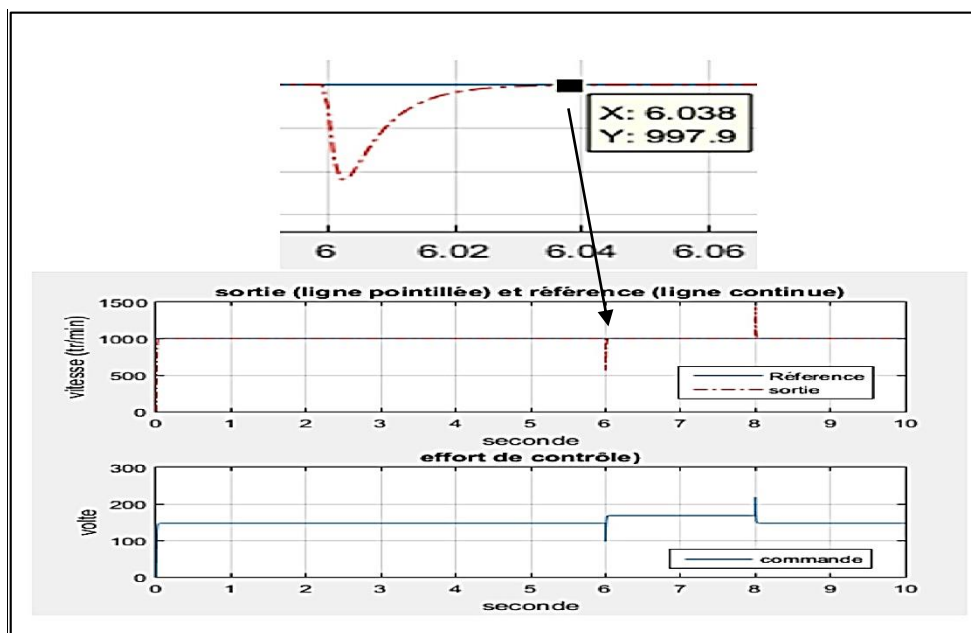
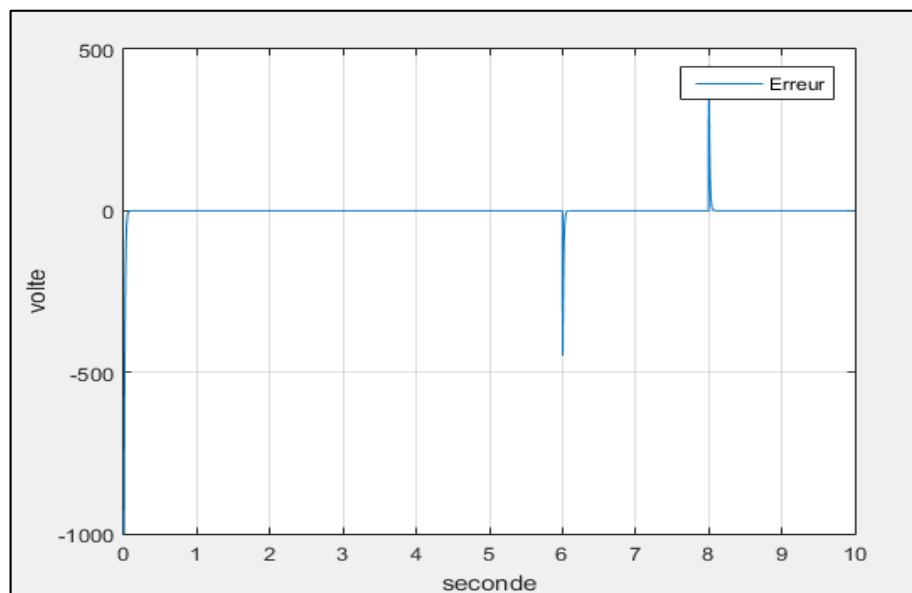


Figure 3.27 : Référence, vitesse de rotation du MCC et signal de commande délivré par le PID auto ajustable (cas avec perturbation 25%).



**Figure 3.28 :** Ecart entre la sortie réelle du système perturbé et la référence.

A partir des figures 3.27 et 3.28 nous remarquons que le PID auto ajustable met un temps de l'ordre de 38 ms pour compenser l'effet de la perturbation en entrée du système.

A partir des résultats obtenus, nous concluons que le PID auto ajustable ne présente, en aucun cas, de dépassement au contraire du PID sans adaptation où des dépassements ont été observés. D'après les résultats obtenus sur le temps de réponse, nous remarquons que le régulateur agit rapidement pour compenser l'effet de la perturbation.

### 3.7 conclusion

Après avoir modélisé le moteur, et l'encodeur, nous avons mis en évidence, par plusieurs simulations, les performances de la commande PID. En particulier, nous avons montré que le régulateur auto ajustable est très robuste que le régulateur sans adaptation dans la commande du système en temps réel. Dans le chapitre suivant, nous reprendrons tout ce que nous avons vu dans les chapitres précédant d'une manière pratique en envisageant l'implémentation sur DSP de La commande PID.

## **Chapitre 04**

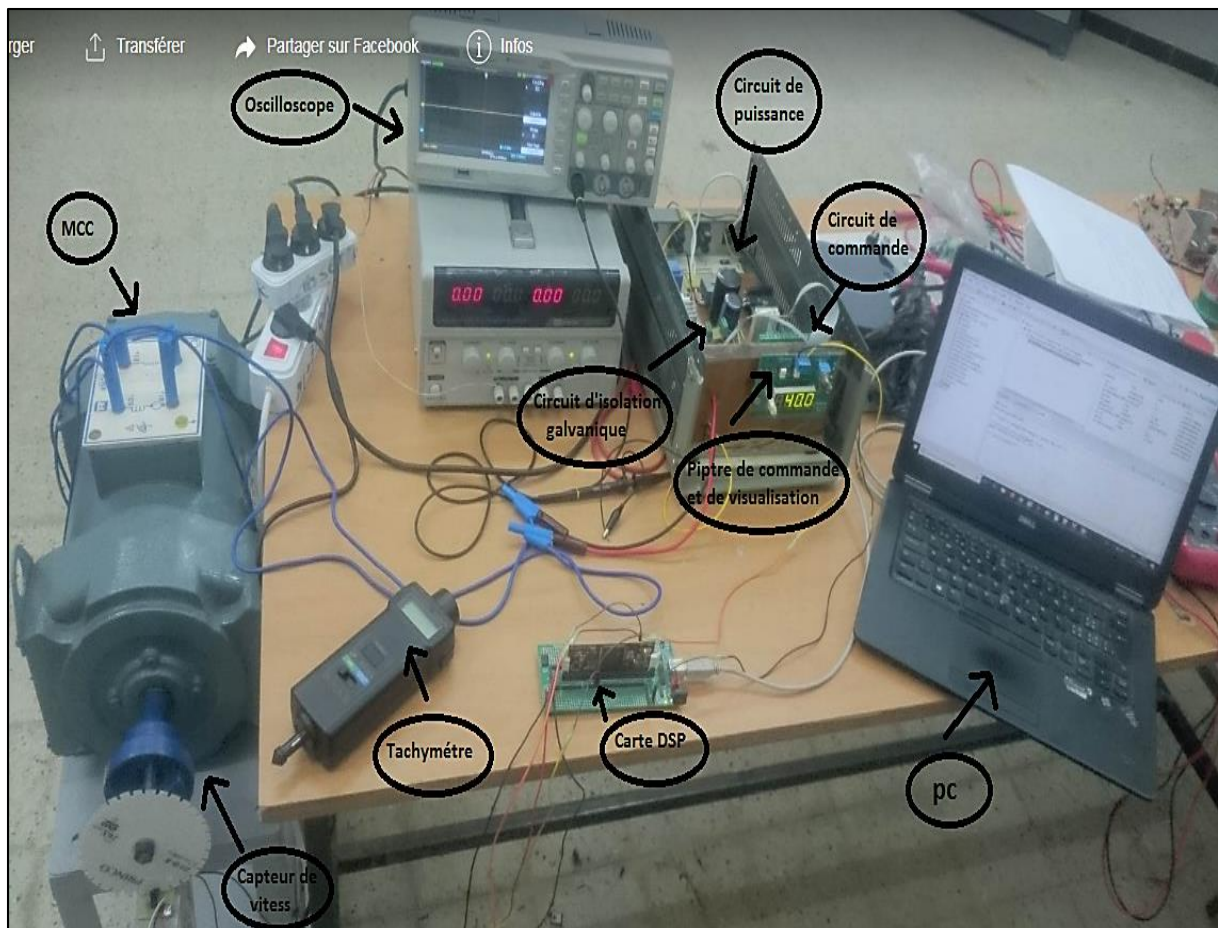
### **Implémentation de la commande et essais pratiques**

## 4.1 Introduction

Pour évaluer pratiquement les performances de la commande PID nous considérons la réalisation de la commande d'un moteur à courant continu. L'algorithme de commande est implémenté numériquement en utilisant la carte de développement *eZdspTMS320F28335*. Dans ce chapitre, nous présentons une description succincte de la carte *eZdspTMS320F28335* et de l'outil de développement Code Composer Studio, nous considérons la réalisation des différents circuits nécessaires à la mise en marche de l'application et les résultats des essais pratiques effectués. Cette étude a été effectuée au laboratoire LABSET (**L**aboratoire des **S**ystèmes **E**lectrique et **T**élécommande) à L'USDB.

## 4.2 Constituants du banc d'essai

Le banc d'essai que nous avons réalisé autour de la carte de développement *eZdspTMS320F28335* ainsi que la disposition des différents circuits nécessaires à son mise en marche sont illustrés par la photo de la figure 4.1.

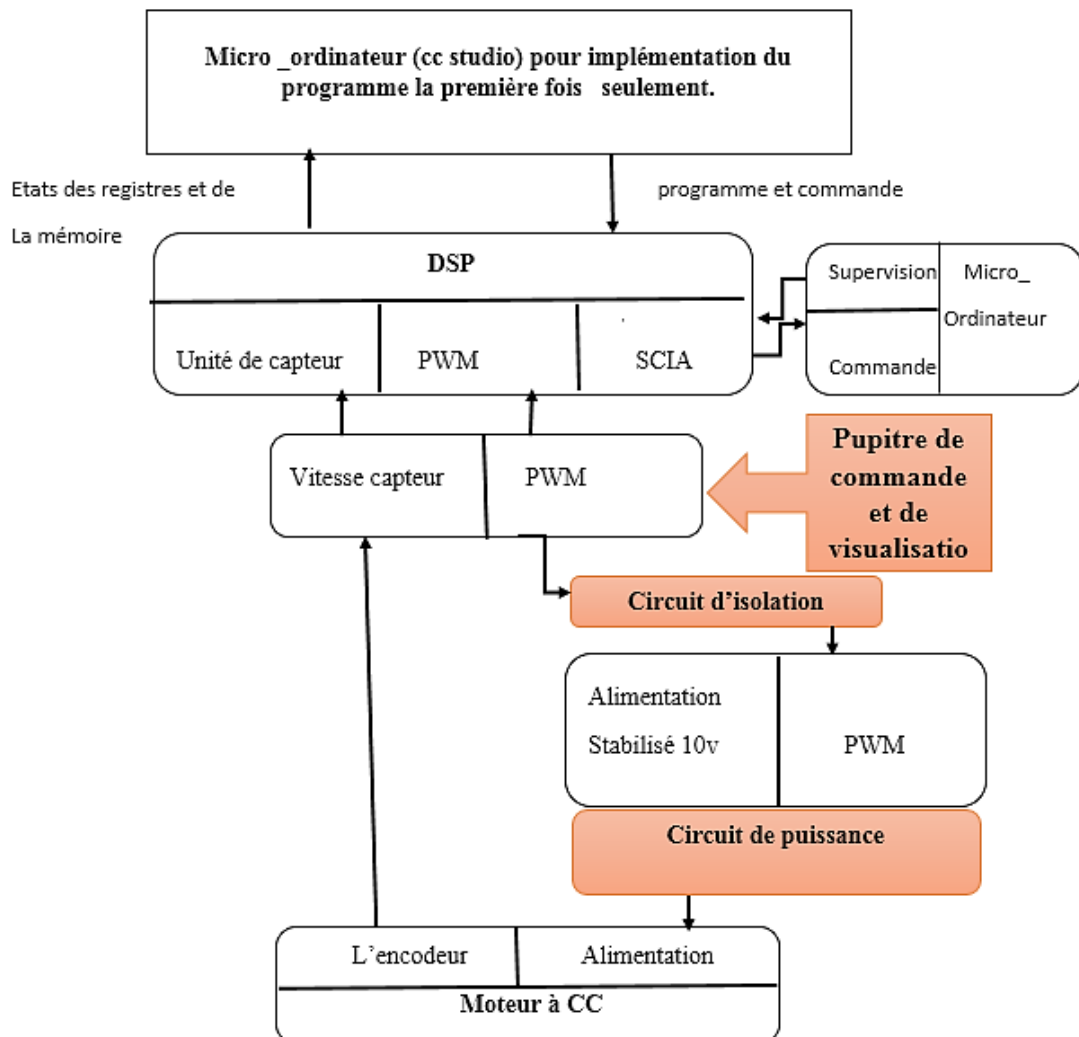


**Figure 4.1:** Photographie du banc d'essais.

Notre banc d'essai se compose des éléments suivants :

- Un circuit de puissance.
- Un circuit de commande (la carte mère).
- Un circuit d'isolation galvanique.
- Pupitre de commande et de visualisation.
- Un encodeur.
- Un kit de développement *eZdsp TMS320F28335*.
- Un moteur à courant continu.
- Un oscilloscope numérique pour la mesure.
- Tachymètre pour mesurer la vitesse du MCC.

La figure 4.2, illustre le schéma synoptique du dispositif expérimental réalisé.

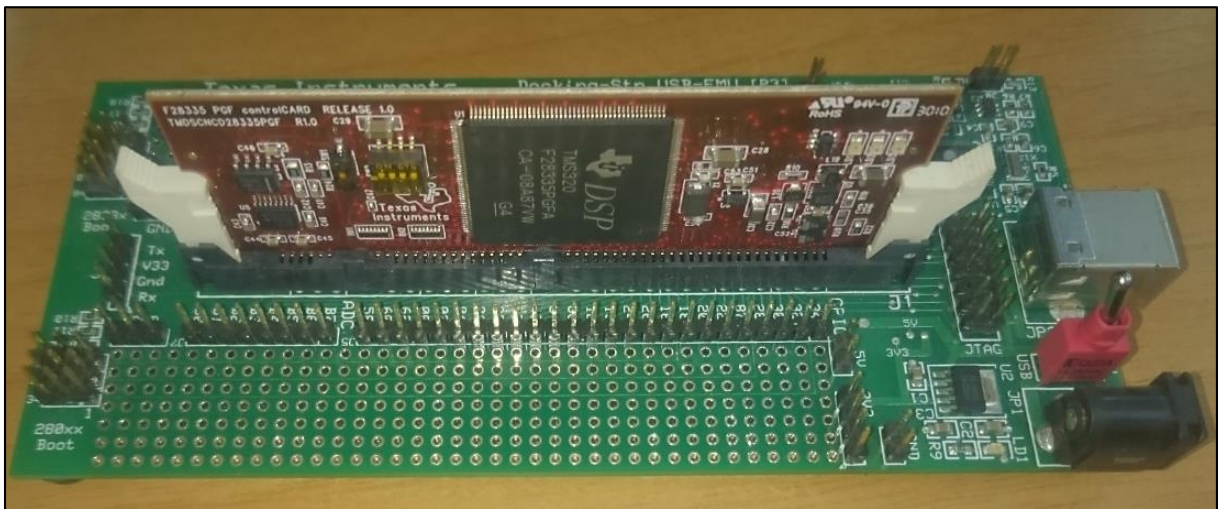


**Figure 4.2 :** Schéma synoptique du dispositif expérimental.

## 4.3 Description des éléments du banc d'essai

### 4.3.1 Le kit de développement *eZdsp F28335*

Le kit de développement *eZdsp F28335* est une solution qui nous permet d'évaluer nos programmes mis en œuvre pour accomplir une tâche spécifique (commande d'un moteur par exemple). Elle incorpore comme processeur le DSP TMS320F28335, et contient toutes les extensions qui permettent le dialogue entre le processeur et son entourage (moteur, capteur, cartes électroniques, ...etc.). Toutes les opérations se font à travers d'un micro-ordinateur en utilisant l'outil de développement code composer studio fourni avec le kit. La figure (4.3) montre la carte *eZdsp F28335*.



**Figure 4.3** : la carte *eZdsp F28335*.

Les caractéristiques principales de la carte *eZdsp TMS320F28335* sont :

- Vitesse d'utilisation 150 MHz.
- Unité à virgule flottante 32 bits sur puce.
- RAM 68 Ko.
- 512 Ko de mémoire flash.
- 256 Ko de mémoire SRAM.
- Convertisseur analogique-numérique (ADC) 12 bits avec 16 canaux d'entrée.
- Horloge d'entrée 30 MHz.
- Connecteur RS232 embarqué avec conducteur de ligne.
- Interface CAN 2.0 embarquée avec le driver de ligne et connecteur.
- Connecteurs d'extension multiples (analogiques, E / S).
- Contrôleur USB JTAG embarqué.
- Fonctionnement à 5 volts uniquement avec l'adaptateur secteur fourni.



- Connecteur d'émulation JTAG IEEE 1149.1 embarqué.

### a. Architecture du processeur *TMS320F28335*

Le *TMS320F28335* est un processeur de traitement des signaux qui fonctionne en virgule fixe et en virgule flottante [19]. Son architecture est conçue spécifiquement pour le traitement de signal en temps réel. Il se caractérise par ses performances élevées et son coût réduit. Les signaux traités par le *TMS320F28335* sont à 32bits. L'horloge de ce DSP est générée par un oscillateur extérieur de fréquence de 30MHz qui est petite par rapport à la fréquence interne du DSP qui peut atteindre « 150MHz ». L'espace mémoire du *TMS320F28335* est divisée en 1 K 16-bit \* 4.DSP OTP de mémoire de données, et 4M mot en mémoire de programmation. Les bus de système qui assurent le contact entre les blocs fonctionnels du système se basent sur l'architecture « HARVARD » qui offre une meilleure flexibilité de fonctionnement [19]. Le microcontrôleur est géré par un code appelé : « Code Composer Studio ». Il permet d'effectuer des opérations à l'intérieur du microcontrôleur, comme saisir le programme en langage « C », le compiler, le charger sur le DSP et l'exécuter. La figure 4.4 illustre l'architecture du processeur *TMS320F28335*.

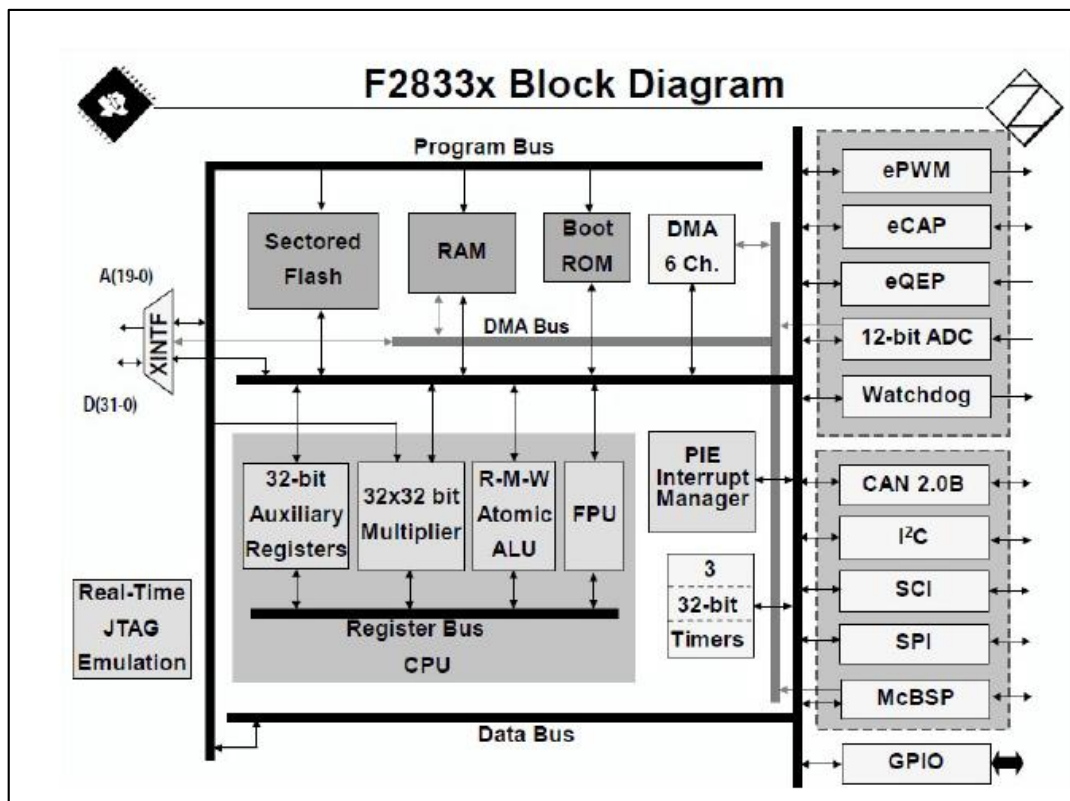
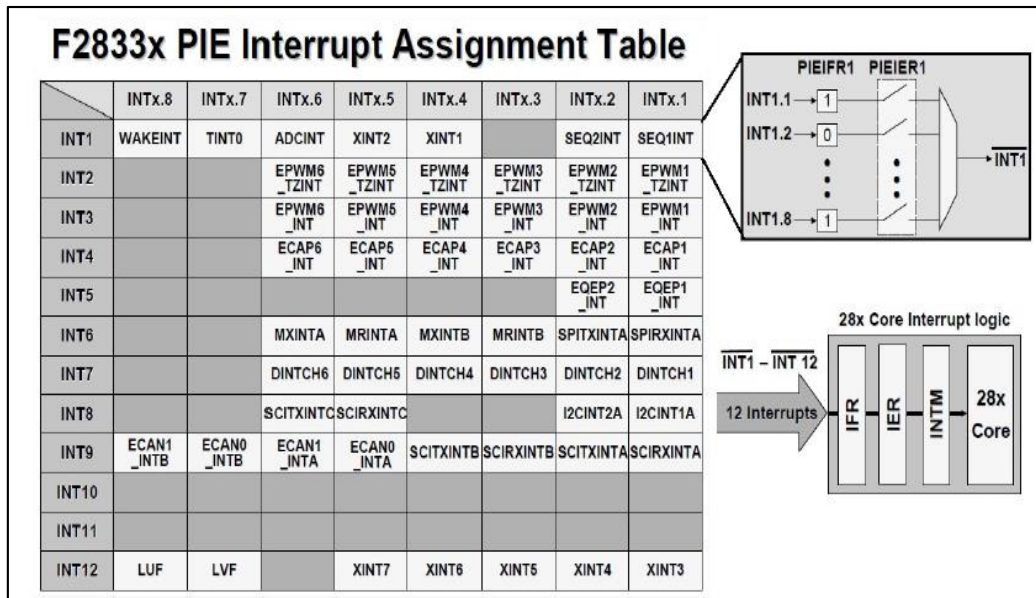


Figure 4.4: Architecture du *TMS320F28335* [19].

### b. Circuit principale



- **Flotting Point Unit** : Sa fonction est d'effectuer des opérations en virgule flottante dix fois plus rapidement que le module ALU standard. Ce dernier est réservé pour les opérations en virgule fixe.
- **Multiplieur 32×32 bits** : Les multiplications 32×32 bits ne peuvent se faire nativement sur l'ALU standard, car le résultat est sur 64 bits. Ce module aide le processeur dans cette tâche.
- **Emulation "JTAG"** Ce module permet de relier le microcontrôleur à l'ordinateur via le câble "USB". Il confère la possibilité à l'utilisateur de programmer sur le DSC à partir du PC, et aussi de lire les registres internes de façon transparente.
- **Gestionnaire d'interruptions "PIE"** : Il y a deux moyens de communication entre le processeur et les autres périphériques qui seront discutés ultérieurement : Les registres de contrôle des périphériques qui peuvent être lus par le processeur, et l'interruption qui est envoyée par un des modules pour notifier le processeur d'un événement. Ce dernier va arrêter l'exécution du programme principal et traiter le contenu d'une fonction spéciale. Celle-ci est de type **interrupt** que l'on doit configurer pour chacun des modules spécifiquement. Le module PIE sert donc à gérer toutes ces interruptions, tout en donnant un ordre de priorité, dans le cas où plusieurs interruptions arrivent en même temps. L'ordre de priorité d'utilisation des modules est déterminé préalablement par le constructeur selon l'importance des périphériques. L'ordre des modules utilisé est le suivant : Timer et ADC, ePWM, eCAP, eQEP. Le PIE dispose de 12 groupes d'interruption, et chaque groupe contient 8 interruptions, pour un total de 96 lignes d'interruptions (voir la figure 4.5). Pour activer une interruption d'un périphérique donné, il faut configurer les registres de contrôle du PIE qui sont : IFR, IER et INTM. "Texas Instrument" a fourni une carte présentée sur la figure 4.5, qui fait correspondre chaque module à sa ligne d'interruption.

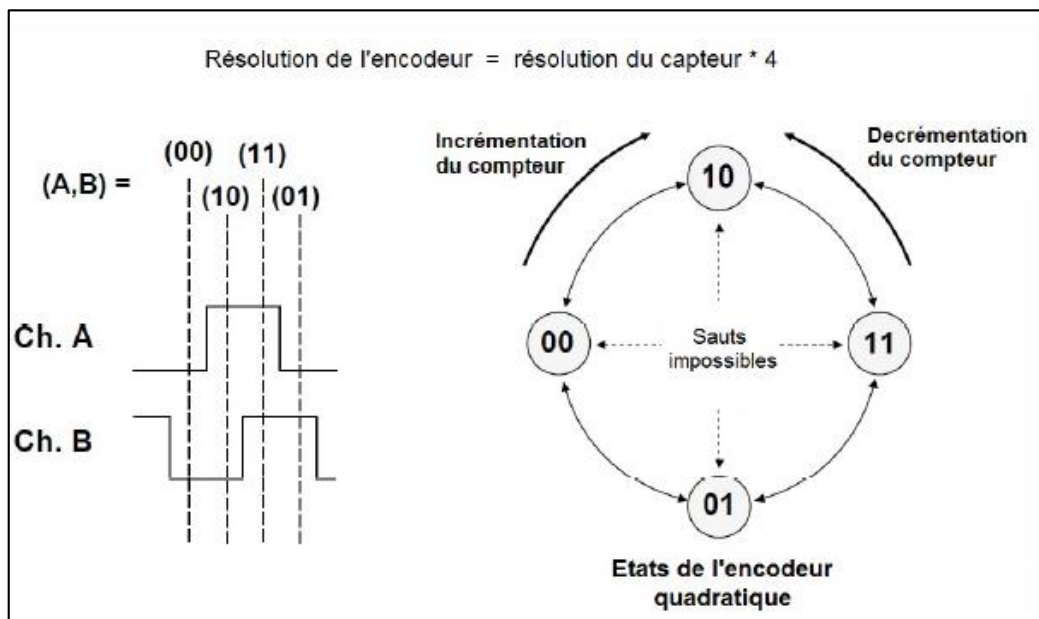


**Figure 4.5:** Affectation des lignes d'interruptions [19]

- **Timers** : Le rôle du Timer est de générer une interruption. Sa fonction principale est de fixer la fréquence d'exécution du programme principal. En effet, il s'exécute chaque fois qu'il reçoit l'interruption du compteur. La fréquence d'interruption du Timer est réglée dans ce cas à 20 kHz.
- **C. Périphériques (E/S)**

Le *TMS320F28335* dispose de 88 pins E/S multiplexés, répartis sur 3 multiplexeurs. Ceci veut dire que chaque pin peut être configuré pour 4 fonctions différentes à l'aide des registres de contrôle appelés GPAMUX, GPBMUX et GPCMUX [22].
- **GPIO (General Purpose Input Output)** : La fonction entrée/sortie GPIO peut être configurée sur les 88 pins disponibles sur le microcontrôleur. Pour ce faire il faut paramétrer le registre GPxMUX sur la valeur 0, pour le pin associé. Une fois la fonction GPIO sélectionnée, il faut choisir la direction entrée ou sortie déterminée par le registre GPxDIR. Dans nos programmes, la fonction GPIO est choisie pour la génération des signaux de commande des transistors.
- **Convertisseur analogique numérique (ADC)** : Le module ADC dispose de deux séquenceurs qui fonctionnent de manière simultanée. Chaque séquenceur peut effectuer 8 conversions séquentielles automatiquement, pour un maximum de 16 canaux de conversions. La durée d'une conversion sur un séquenceur donné est de 80ns. Les résultats des 16 canaux sont mis sur 16 registres adressables individuellement.

- **Modulation à largeur d'impulsion, ( Enhanced Pulse Width Modulator: ePWM) :**  
Une des sorties est configurée sur la fonction ePWM. La fréquence des impulsions est réglée grâce au registre TBPRD du registre de contrôle EPwm2Regs.
- **Encodeur de pulsations en quadrature (enhanced Quadrature Encoder Pulse : eQEP) :** Ce module fonctionne avec le capteur incrémental. Les deux signaux du capteur décalés entre eux de  $\frac{\pi}{4}$  sont reliés aux deux entrées correspondantes du eQEP. Le module intègre un registre qui s'incrémente suivant les différents états de l'encodeur, comme présenté sur la figure 4.6. En effet, les états consécutifs doivent être respectés pour que l'incrémement se fasse, afin d'éviter les erreurs. L'utilisation du eQEP présente les avantages suivants :
  - Une résolution de position multipliée par quatre.
  - Minimisation des erreurs dues à la compatibilité électromagnétique



**Figure 4.6:** Encodeur de pulsations quadratique [19]

- **Dispositif de sécurité (Watchdog) :** Le but du Watchdog est de surveiller tout plantage qui peut survenir tout le long du programme pour une raison quelconque. En effet, un bug du programme peut avoir de lourdes conséquences sur le circuit externe. L'utilisation du Watchdog exige l'exécution d'une certaine commande spécifique chaque 4.37 ms. Dans le cas contraire, il initie un Reset du microcontrôleur remettant toutes les sorties à zéro.

## 4.4 Simulation du hacheur

### 4.4.1 Introduction à PSIM

PSIM (Power Sim) est un logiciel de simulation principalement destiné à l'électronique de puissance.

Le logiciel permet de dessiner le schéma du montage, à partir des éléments de la bibliothèque (machine, éléments de commande et contrôle...). Les appareils de mesure disposés sur le schéma de montage définissent les courbes représentatives des grandeurs électriques et mécaniques que l'on peut obtenir après simulation.

Le logiciel est constitué de 3 programmes :

- SIMCAD : dessin du schéma.
- PSIM : simulation (calcul des variables).
- SIMVIEW : tracé des courbes.

La simulation d'un montage se décompose en 4 étapes :

- dessin du schéma de montage.
- entrée des valeurs des éléments du montage et des paramètres de la simulation.
- calcul.
- visualisation des courbes.

### 4.4.2 Résultats de simulation

La figure (4.7) représente le schéma électrique du hacheur abaisseur et les figures (4.8) et (4.9) représentent la tension de sortie et le signal de PWM pour un rapport cyclique de 25% et 50% respectivement.

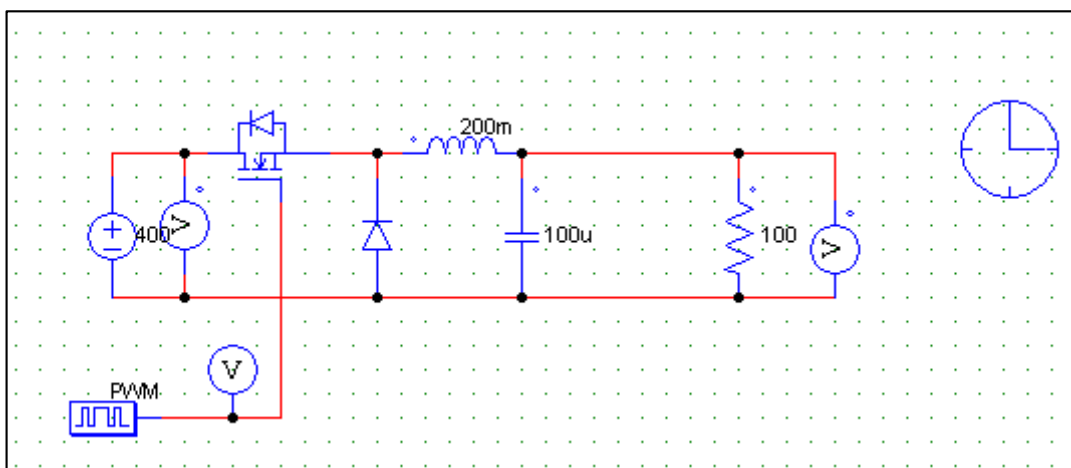
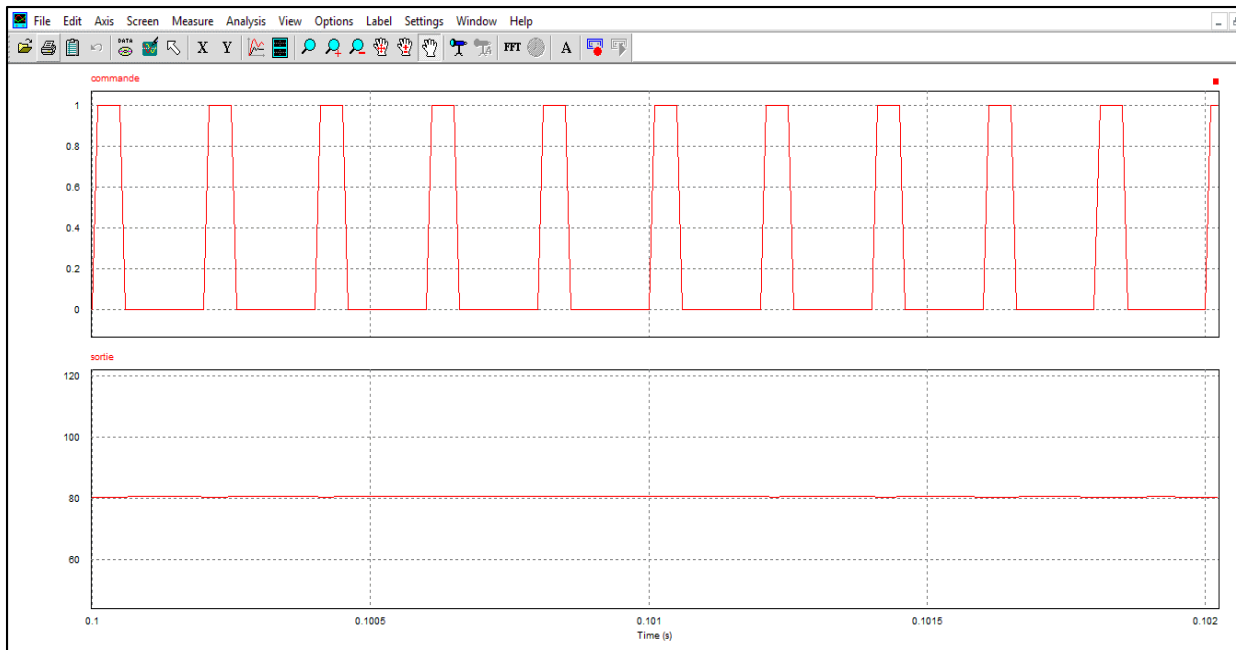
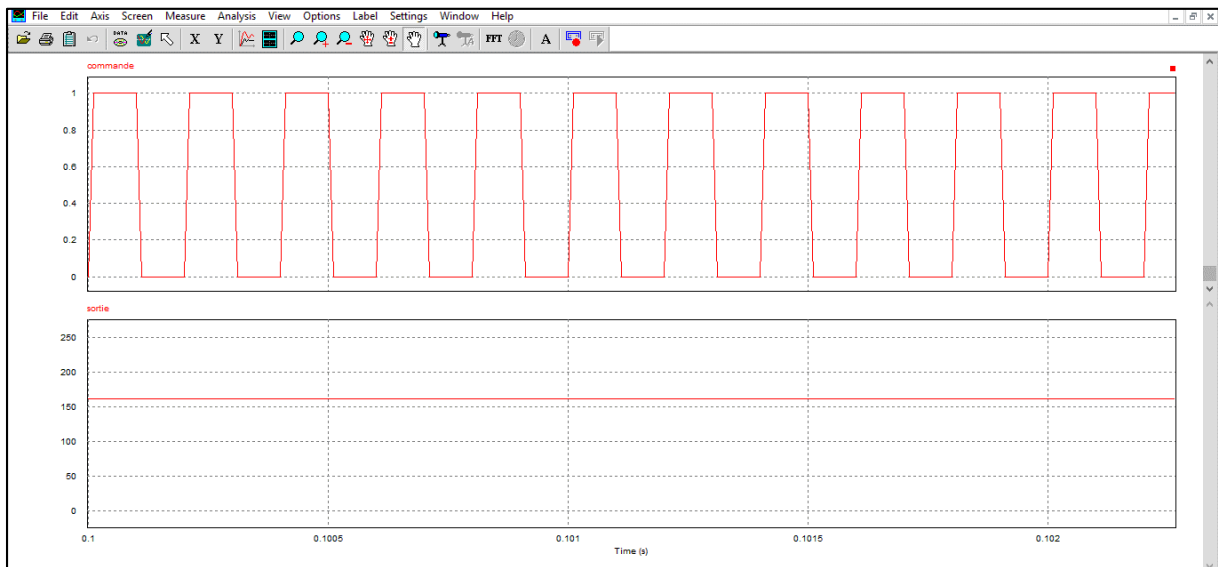


Figure 4.7 Schéma électrique du hacheur abaisseur



**Figure 4.8 :** la tension de sortie, signal de PWM pour un rapport cyclique de 25%.



**Figure 4.9 :** la tension de sortie, signal de PWM pour un rapport cyclique de 50%.

Nous remarquons que la tension de sortie augmente si le rapport cyclique augmente. Donc la valeur de la tension de sortie peut être ajustée en jouant sur la valeur du rapport cyclique.

## 4.5 Circuit de puissance

Le circuit puissance est constitué d'un redresseur en ponts diodes (35A, 1000V) qui permet de redresser une tension de 220V alternative à une tension de 400V continue, et d'un filtre passe bas qui permet d'abaisser la tension de 400V à une tension de 200V et d'un hacheur en série (Buck). Le schéma électrique de cet hacheur et de sa commande est donné par la figure

(4.10). Une alimentation continue de 12 V est nécessaire pour la polarisation du transistor de commande. Le schéma électrique de cette alimentation est donné par la figure (4.11)

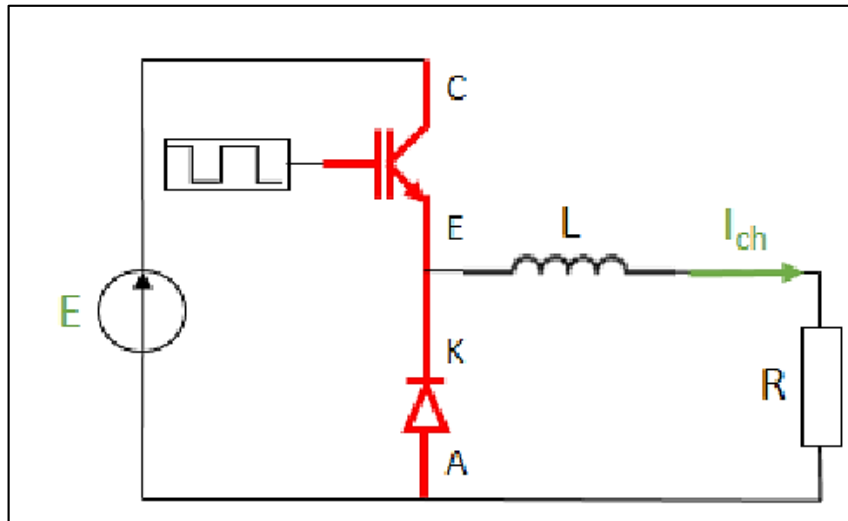


Figure 4.10 : schéma électrique du hacheur (Buck).

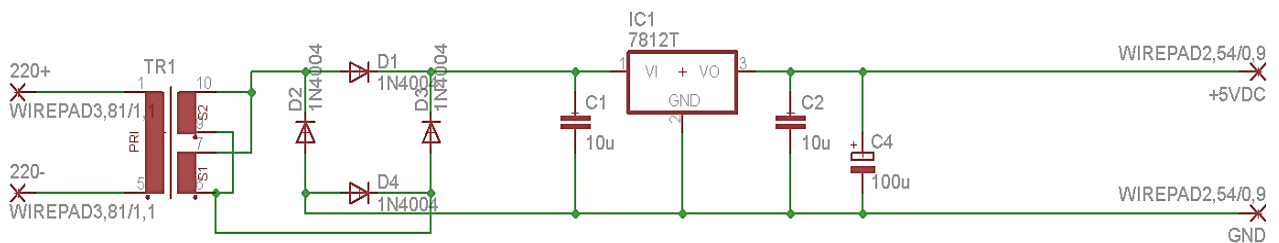


Figure 4.11 : Schéma électrique de l'alimentation stabilisé 12V.

Nous avons réalisé une bobine à noyau de ferrite de 2mH pour le lissage du courant.

La valeur de la bobine est donnée par :

$$L = u_r \times \frac{N^2 \times S}{l} \times u_0 \quad (4.1)$$

Où :

- $u_r$  est la perméabilité relative effective du matériau magnétique. Dans notre cas  $u_r=100$ .

- $S$  représente la section du noyau magnétique en mètres carrés ( $m^2$ ). Sa valeur est  $S = 1.1304 \times 10^{-4} m^2$
- $N$  représente le nombre de spires.
- $L$  représente l'inductance en milli henry(mH).
- $l$  est la longueur effective du noyau magnétique en mètres(m). Sa valeur est  $l = 0.037 m$ .
- $\mu_0$  est la constante magnétique, elle est égale à  $1.26 \cdot 10^{-8}$

Le nombre de spires de la bobine est donné par :

$$N = \sqrt{\frac{L \times l}{\mu_r \times S \times \mu_0}} = \sqrt{\frac{0.074}{1.4243 \times 10^{-8}}} = 227 \text{ spires} \quad (4.2)$$

La figure 4.12 montre une photo du circuit réalisé.

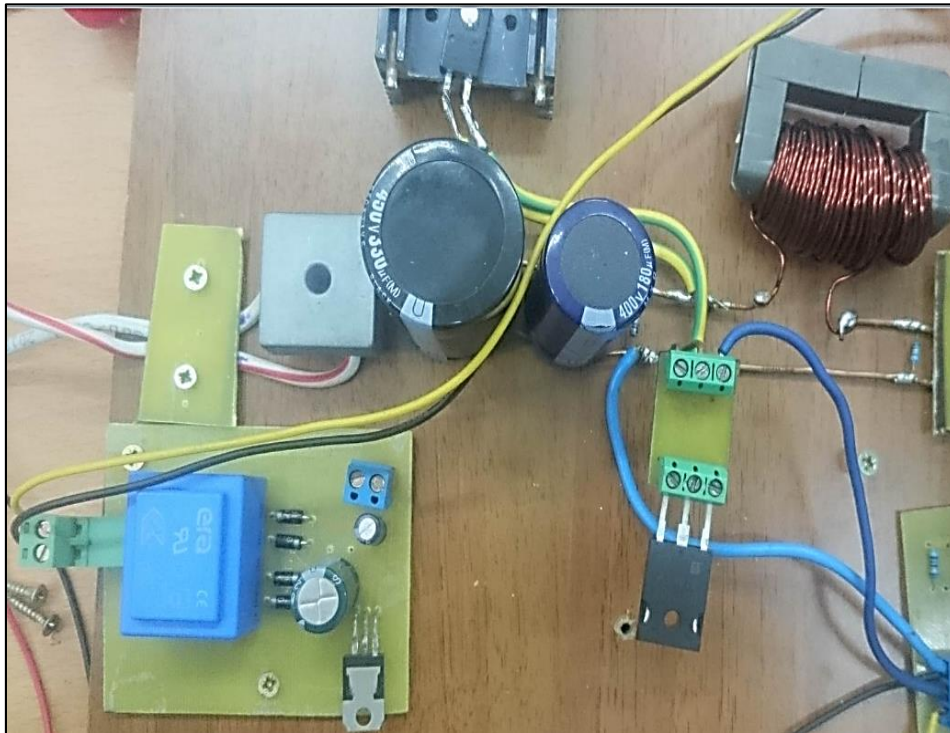


Figure 4.12 : Photographie du circuit de puissance.

## 4.6 Circuit de commande (carte mère)

Pour commander le moteur manuellement (en boucle ouverte) nous avons réalisé un circuit de commande à base d'un microcontrôleur (annexe B3) de la famille micro chip (pic 18F4550). Ce microcontrôleur permet de générer un signal PWM selon le rapport cyclique fixé par un potentiomètre. Le schéma électrique de ce circuit ainsi que celui de son alimentation sont



illustrés par la figure 4.13. Nous notons que dans le cas de la commande en boucle fermée le signal PWM est généré par le DSP.

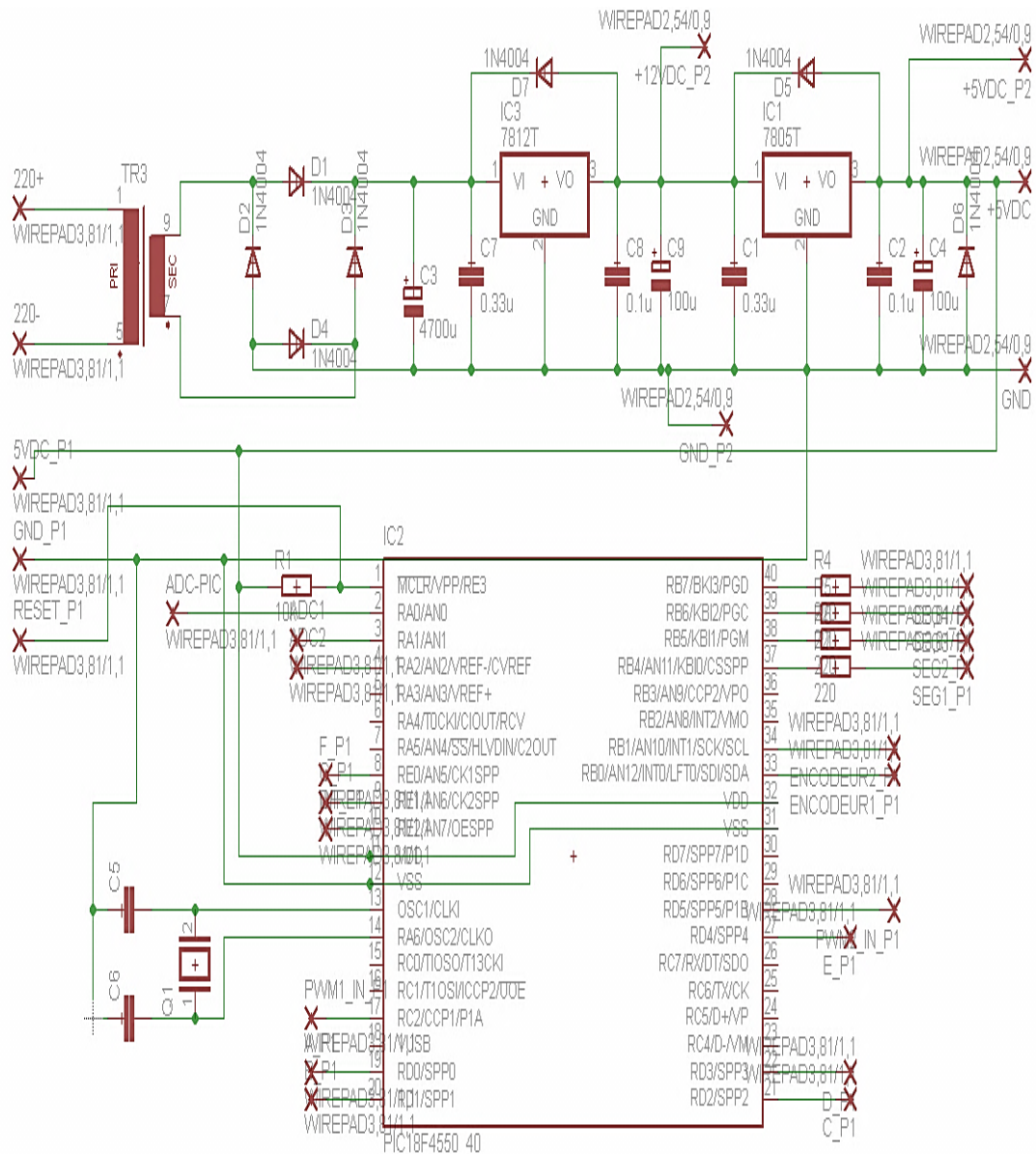
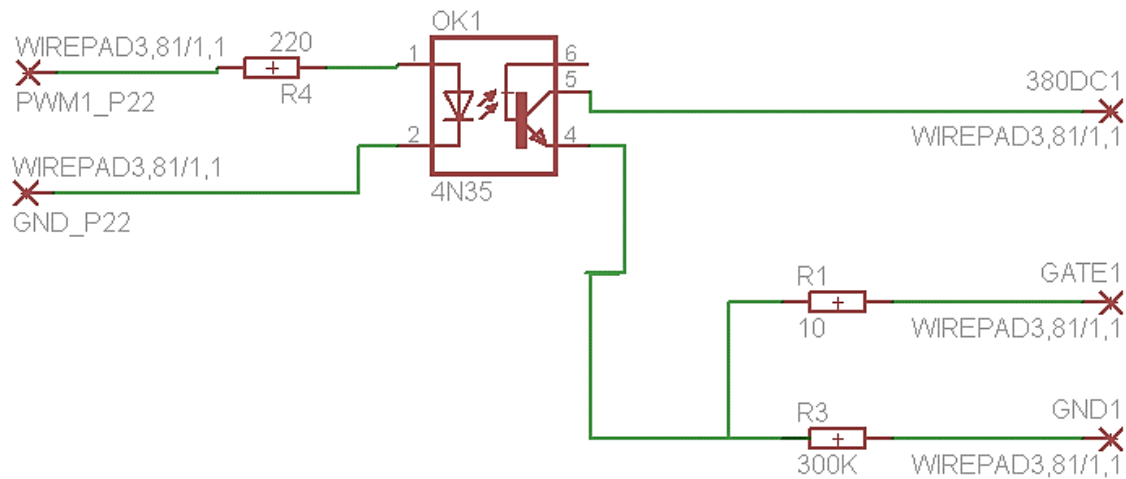


Figure 4.13 : Schéma électrique du circuit de commande en boucle ouverte.

## 4.7 Circuit d'isolation (isolation galvanique)

Nous avons utilisé ce circuit pour protéger la partie de commande avec un optocoupleur de la série (4N35). Un optocoupleur est un composant électronique capable de transmettre un signal d'un circuit électrique à un autre. La figure (4.14) illustre le circuit d'isolation galvanique.





**Figure 4.14 :** Schéma électrique du circuit d'isolation (isolation galvanique).

## 4.8 Pupitre de commande et de visualisation :

Ce circuit contient deux parties, une partie de commande pour faire varier la tension de command à travers le changement du rapport cyclique, et une autre de visualisation pour afficher la valeur de la tension correspondante. Dans ce circuit il y a un switch qui nous permet de basculer en le mode commande en boucle ouverte et le mode commande en boucle fermée. Pour protéger notre carte de DSP nous avons utilisé une isolation galvanique doublé. La figure (4.15) illustre le circuit de ce pupitre ainsi que le circuit d'isolation galvanique que nous avons réalisé pour protéger la carte de DSP.

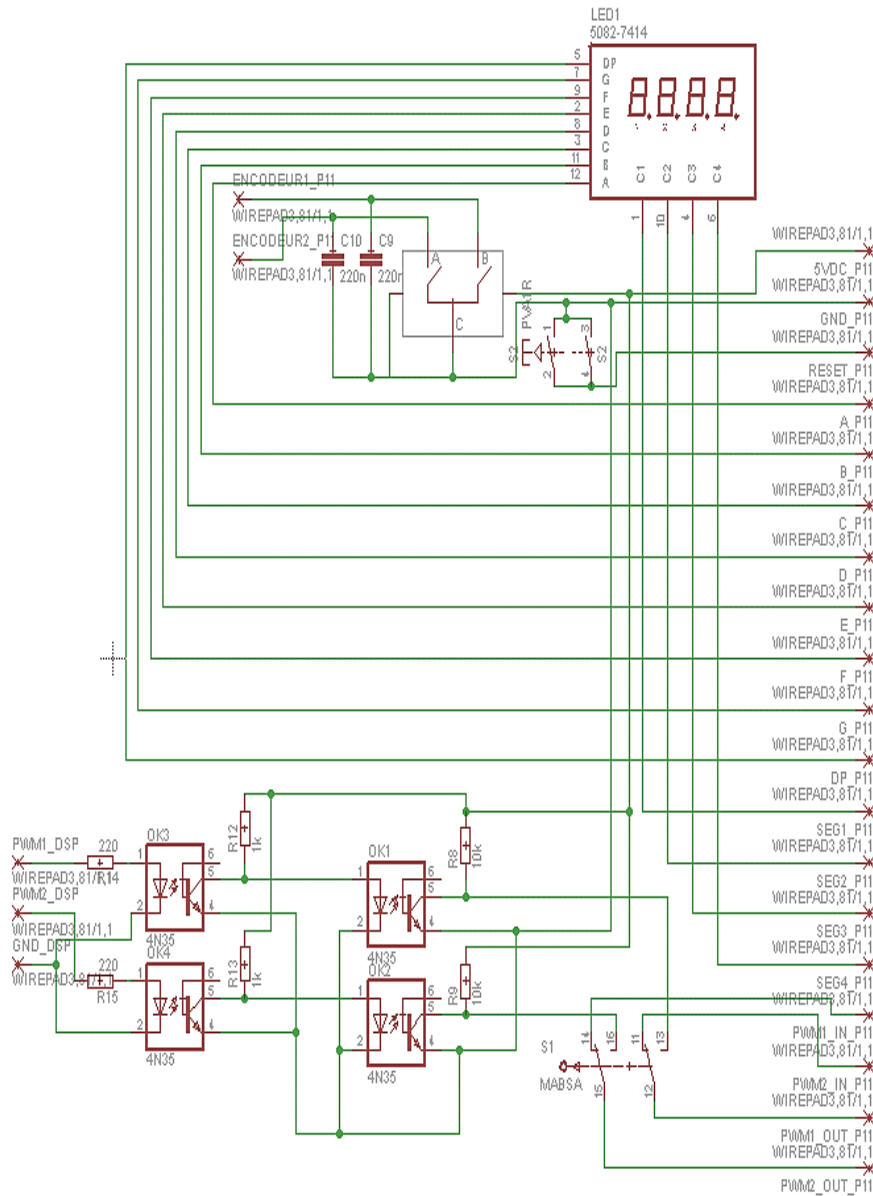


Figure 4.15 : Schéma électrique du pupitre de commande et de visualisation.

## 4.9 Résultats expérimentaux

Nous avons asservi la vitesse du moteur à courant continu par deux commandes différentes. La première (en boucle ouverte) par le microcontrôleur (PIC) et la deuxième en utilisant le régulateur PID implémenté sur la carte DSP. Nous avons effectué les essais suivants :

### 4.9.1 Commande en boucle ouverte :

Dans le premier essai nous avons commandé la vitesse du MCC par le pupitre de commande qui envoie un rapport cyclique de 25% ,50% et 75% au microcontrôleur, respectivement. Les figures (4.16), (4.17) et (4.18) montrent les signaux PWM obtenus ainsi que les vitesses et les

tensions (en V) de commande correspondante. La valeur de la tension de commande est indiquée sur l'afficheur et celle de la vitesse (en tr/min) est mesurée en utilisant tachymètre.

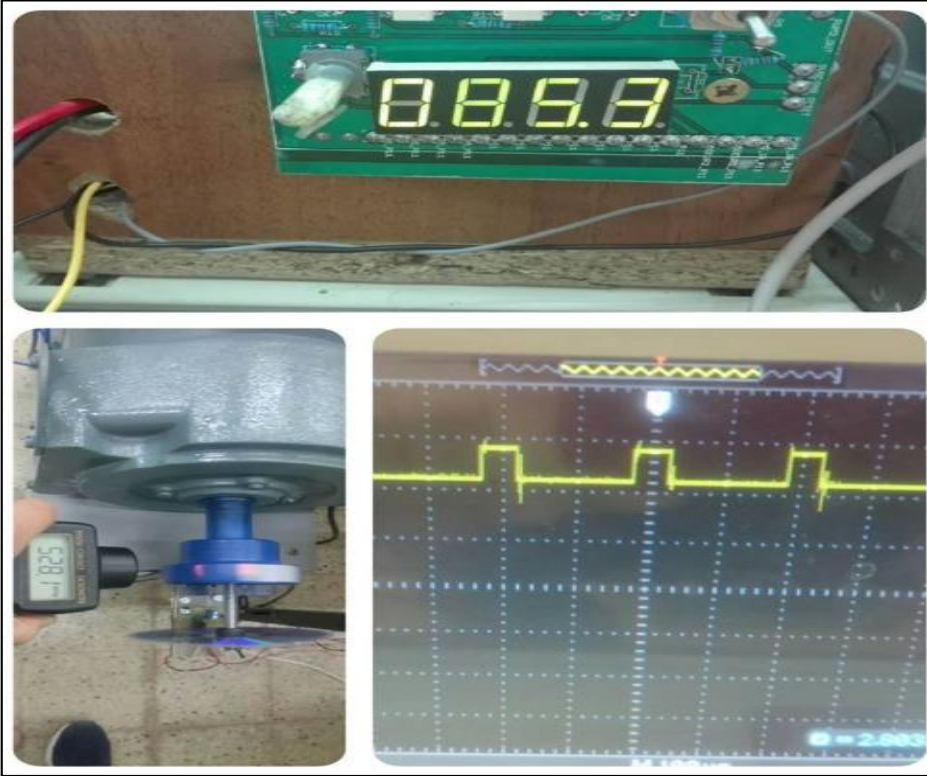


Figure 4.16 : Résultats obtenus pour un rapport cyclique de 25%.

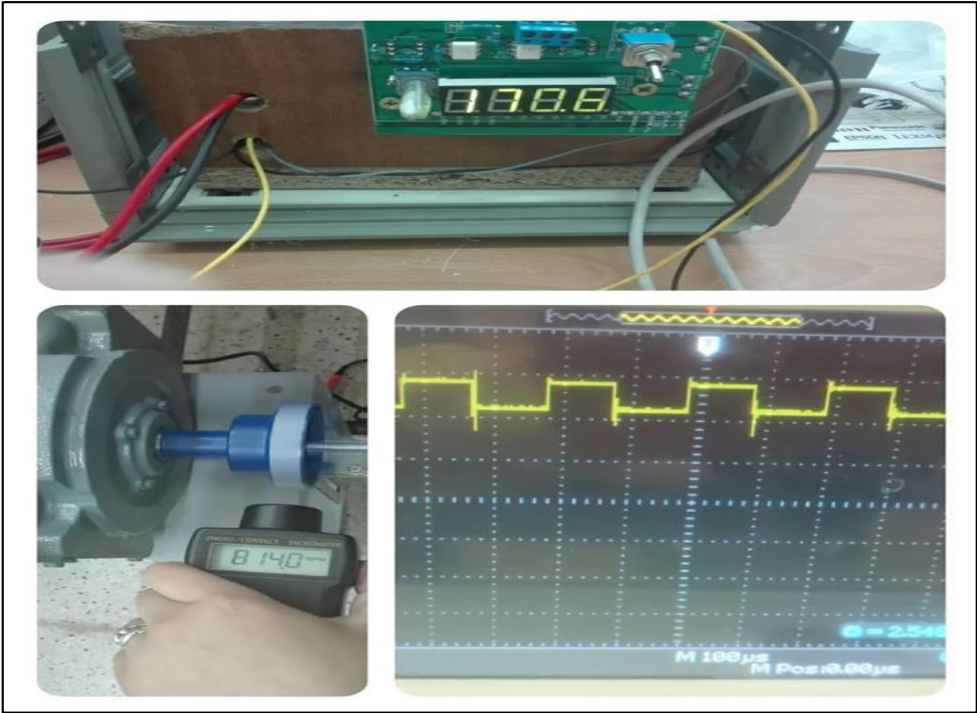
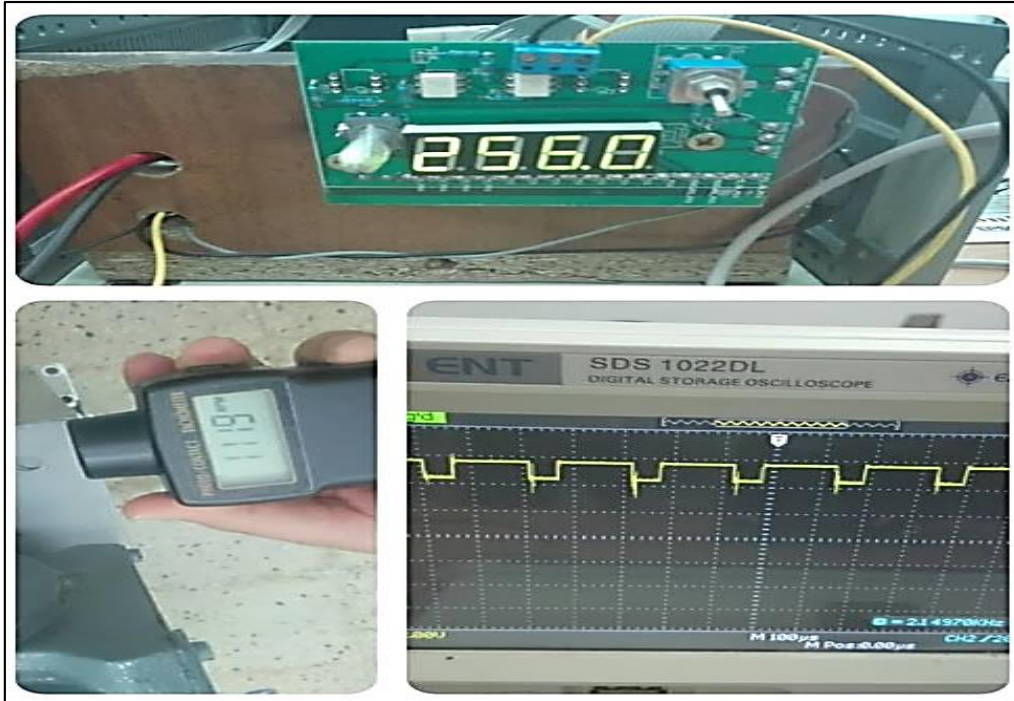


Figure 4.17 : Résultats obtenus pour un rapport cyclique de 50%.



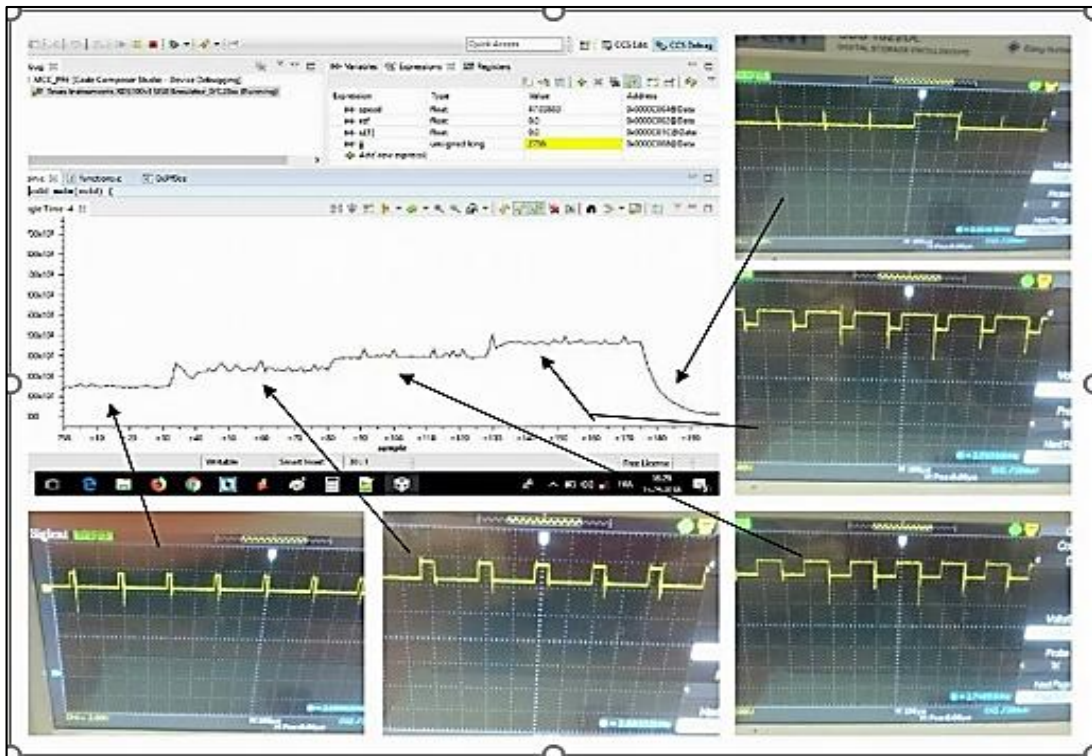
**Figure 4.18** : Résultats obtenus pour un rapport cyclique de 75%.

Nous remarquons à partir des figures précédentes que pour chaque rapport cyclique choisi correspond une valeur de la tension de commande et la valeur de la vitesse de rotation du moteur.

#### **4.9.2 Commande en boucle fermée :**

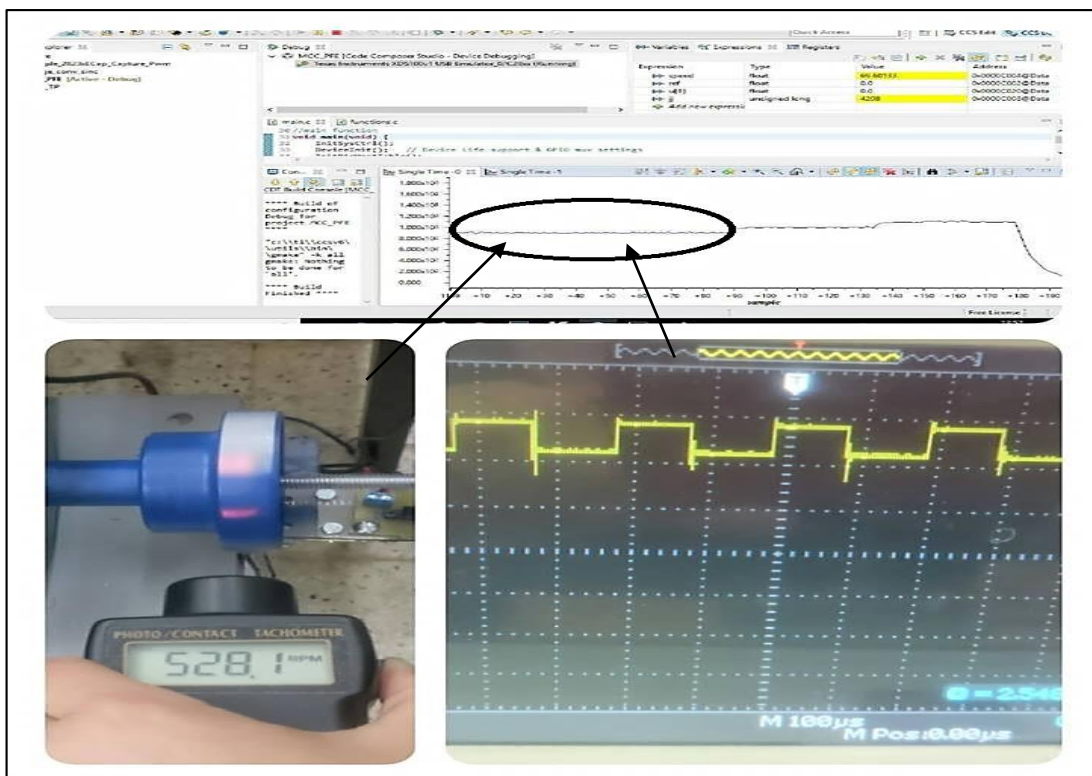
Dans le deuxième essai nous avons commandé la vitesse du MCC par un régulateur PID que nous avons conçu et implémenté sur la carte DSP. Pour mettre en évidence les performances du PID implémenté, nous avons utilisé une référence qui possède plusieurs niveaux (250 tr/min, 550 tr/min, 900 tr/min, 1200 tr/min, 0 tr/min). La figure (4.19) illustre la vitesse mesurée du moteur pour la référence choisie :



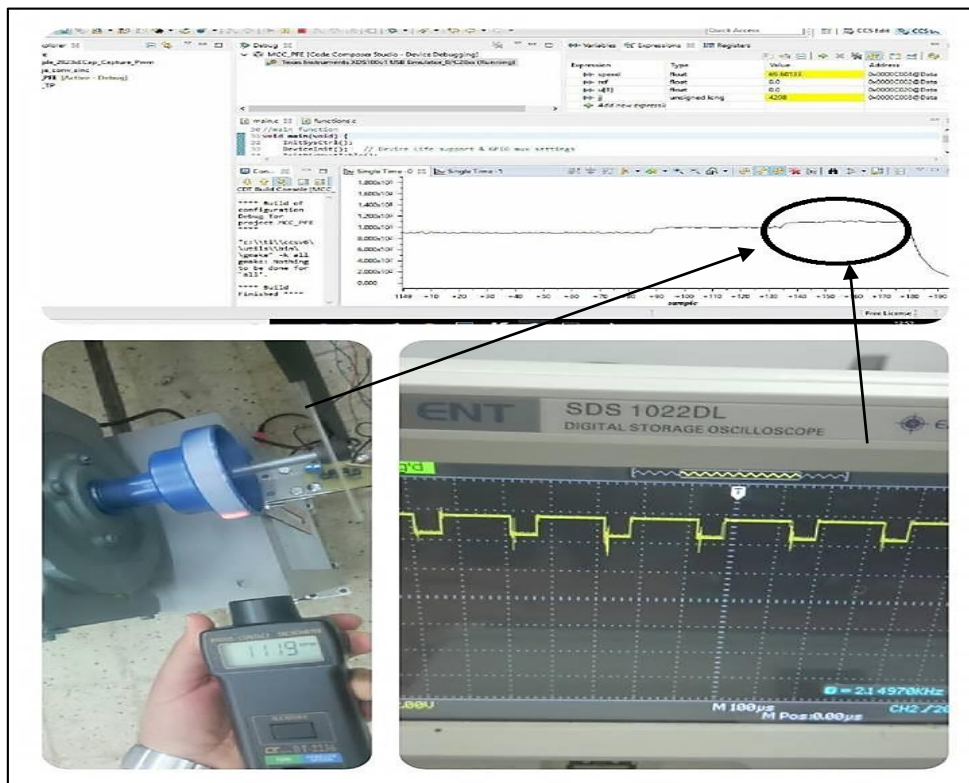


**Figure 4.19 :** Signal PWM et vitesse du MCC mesurée.

Nous avons utilisé un tachymètre pour mesurer la vitesse du MCC pour deux références différentes : 530 et 1125 (tour/min) (figures (4.20), (4.21)).



**Figure 4.20 :** Vitesse du MCC et signal de PWM pour la référence de 530(tour/min).



**Figure 4. 21 :** Vitesse du MCC et signal de PWM pour la référence de 1125(tour/min).

D'après les figures précédentes nous observons qu'il y a un petit écart entre la vitesse mesurée et la référence. Cet écart est dû à la mauvaise précision de l'encodeur utilisé.

## 4.10 conclusion

Nous avons introduit, dans ce chapitre, les différents circuits que nous avons réalisés et testés. Ensuite, nous avons effectué plusieurs essais pour montrer le bon fonctionnement des circuits réalisés et du régulateur PID conçu et implémenté sur DSP. Les résultats des essais effectués montrent que tous les circuits réalisés fonctionnent correctement. Nous devons noter que les périphériques dont la carte de développement *eZdspF28335* dispose nous ont beaucoup facilité la tâche d'implémentation et de réalisation des algorithmes de commande développés. Nous notons aussi que le banc d'essai réalisé peut être utilisé pour tester l'efficacité d'autres algorithmes de commande.

## **Conclusion générale**

### Conclusion générale

Dans de nombreux domaines de l'industrie, l'utilisation de moteurs à courant continu est préférée en raison de leur facilité de commande. Cette commande peut être assurée mécaniquement ou électroniquement

La commande électronique des moteurs à courant continu fourni par l'utilisation des convertisseurs statiques contrôlés par des processeurs, se caractérise par sa flexibilité due au fait qu'il peut être utilisé de diverses manières pour plusieurs applications. Il suffit d'implémenter l'algorithme qui répond aux exigences requises. Le choix du type de régulateur peut se faire après avoir étudié différents types de régulateurs tout en visant celui qui répondra aux spécifications du cahier des charges.

L'objectif qui a été donné pour notre projet est l'utilisation du DSP *TMS320F28335* de Texas Instruments pour mettre au point la commande PID d'un moteur à courant continu en utilisant l'algorithme d'optimisation Teaching and Learning Based Optimization (TLBO).

Le système de commande que nous avons réalisé est composé principalement du kit de développement *ezdsp F28335* dans lequel nous avons implémenté l'algorithme de commande, d'un circuit de puissance, d'un circuit d'isolation galvanique, d'un pupitre de commande et de visualisation, d'un circuit de commande et d'un moteur à courant continu.

Le choix du processus DSP *TMS320F335* est suite à sa richesse en périphériques et sa facilité d'utilisation, surtout pour la commande des moteurs. En fait, ce processeur intègre plusieurs fonctionnalités conçues pour simplifier et rendre facile la tâche de commande des systèmes.

Le projet réalisé est conçu de telle sorte qu'il puisse être adapté facilement pour qu'il fasse partie d'un ensemble plus complexe. Il pourrait servir comme un banc de travaux pratiques sur lequel il est possible de tester et de mettre en évidence les performances de différents algorithmes de commande.

A l'issue de cette étude, nous avons découvert des nouvelles algorithmes d'optimisation efficaces, maîtrisé l'implémentation des algorithmes de commande sur DSP et appris à mettre au point des réalisations électrique.



## **Références bibliographiques**

- [1] A Besançon – voda, S Gentil, Régulation PID analogique et numérique, technique de l'ingénieur.
- [2] Mohammad H, Moradi & Michael A. Johnson. PID Control New Identification and Design Methods. United Kingdom, 2005, 558 p.
- [3] CLAUDE Bergmann, Electronique et communication BTS industriels. Paris : Dunod, 2005, 238 p.
- [4] Daniel GARCIA, Auto-ajustement de régulateurs PID robustes dans le domaine fréquentiel, THÈSE NO 3520 (2006), 5 mai 2006,12p.
- [5] F. MUDRY, eivd \_ Ajustage des paramètres des Régulateur PID, description des régulateur, mars 2006, 6p.
- [6] Y.X. Su, Dong Sun, B.Y. Duan. Design of an enhanced nonlinear PID controller [en ligne]. Hong Kong, 2005, 20p. Format PDF. Disponible sur : « <http://www.sciencedirect.com/science/article/pii/S0957415805000590> » (Consulté le 10/06/2016).
- [7] Rahmani Oussama. La méthode de Ziegler nichols pour la détermination des paramètres d'un contrôleur PID pour un système 2 ordre.2013-2014,32p.
- [8] Dingyu Xue, yang Quan chen, and derk p. Atherton, liner feedback control ,2007.
- [9] M. berabah, A. dehebi, Commande des systèmes,2016.5p Format FDF.
- [10] Page S. Buckley, William L. Luyben, Joseph P. Shunta, 'Distillation Column Control Systems', Edward Arnold, 1985.
- [11] Fred Glover, Future Paths for Integer Programming and Links to Artificial Intelligence, Comput. & Ops. Res.Vol. 13, No.5, pp. 533-549, 1986.
- [12] ABBAS El Dor. Perfectionnement des algorithmes d'optimisation par essaim particulaire : applications en segmentation d'images et en électronique. Autre [Cs. OH]. Université Paris-Est, 2012.
- [13] RESEARCH: A teaching learning-based optimization based on orthogonal design for solving global optimization problems by: Suresh Chandra Satapathy Anima Naik and K Parvathi.
- [14] Teaching-Learning-Based Optimization (TLBO) algorithm and its codes developed by Dr. R. Venkata Rao and his team at S.V. National Institute of Technology

- [15] R. Venkata Rao, Vivek Patel, an improved teaching- learning- based optimization algorithm for solving unconstrained optimization problems [**en ligne**], 740p. Format PDF. Disponible : «<https://www.sciencedirect.com/science/article/pii/S0010448510002484> » (Consulté le 9 octobre 2012).
- [16] F. BERNOT, Machines à courant continu, constitution en fonctionnement, Techniques de l'ingénieur, D3355 ,1999.
- [17] M. marbah, Généralités sur les moteurs à courant continue, 49p.format PDF.
- [18] A. Arcole, Commande des hacheurs à quatre quadrants, 16p.format PDF,2001.
- [19] TEXAS INSTRUMENT, DSPs the TMS320F2833x, architecture & peripherique, technical reference 2011 (41 43) p.

# **ANNEXE**

## **A1 (liste de Composants)**

### **Circuits de puissance**

C1, C2: 500  $\mu F$  .

B1: IRFP460

R1: 300K $\Omega$ .

D1 : BYR29R

L1 : 2  $mH$ .

### **Circuit de commande (carte mère)**

Transformateur 220VAC-18VDC.

IC1 : 7805.

IC3 : 7812.

IC2: pic16F4550.

D1, D2, D3, D4, D5, D6, D7: 1N4004

C2, C8 :0,1 $\mu F$ .

C1, C7 :0.33 $\mu F$ .

C9, C4 :100  $\mu F$ .

C5 :220  $\mu F$ .

C6 :1000  $\mu F$ .

C3 :4700 $\mu F$ .

R1 :10 K $\Omega$ .

R7, R6, R5, R4 :220  $\Omega$ .

## Circuit d'isolation (isolation galvanique)

R4 :220  $\Omega$ .

R2: 10  $\Omega$ .

R17: 300 K $\Omega$ .

OK1 :4N35.

## Pupitre de commande et de visualisation

DIS1: 7 segments 4 digits.

OK1, OK2, OK3, OK4: 4N35.

R9:40 K $\Omega$ .

R9:10 K $\Omega$ .

R12, R13:1 K $\Omega$ .

R14, R15: 220  $\Omega$ .

C9, C10 :220nF.

S2: PVA1R.

## A2 (programme de PIC 16F4550)

```
void main() {
    trisa=0xff;
    trisb=0x0f;    portb=0x00;
    trisc=0x00;    portc=0x00;
    trisd=0x00;    portd=0xff;
    trise=0x0;    porte=0x7;

    ADCON1.pcfg0=0;    ADCON1.pcfg1=1; ADCON1.pcfg2=1; ADCON1.pcfg3=1;    //
    configurer AN0 comme analog ( les autres digitales)

    intcon2.intedg0=0;    // 1 for rising edge , 0 for falling edge

    //option_reg.t0cs=0;

    intcon=0b10010000; // enable RB0 interrupt

    ADC_Init();
}
```

```

TMR2=0;
CCP1CON=0b10001100; // bit 0 et bit 1 si active high 00 si active low 11
ECCP1DEL=0b0000111; // dead band
PR2=255; // la periode du PWM
CCPR1L=0; // rapport cyclique
//TMR2 frequence le registre T2CON
T2CKPS1_bit=0 ;
T2CKPS0_bit=0 ;
TMR2ON_bit=1 ;
while (1){
    CCPR1L=rapport_cyclique ;
    ang=ADC_Read(0);
    tension1=ang/204.6;
    //tension2=tension1*(RA+RB)/RB;
    tension2= rapport_cyclique*34/25.5;
    C3= tension2/100;
    C2= (tension2-C3*100)/10;
    C1= tension2-C3*100-C2*10;
    C0= tension2*10- C3*1000-C2*100-C1*10;
    /*portd.f1=0; porte.f1=1; portb.f6=1; portd.f4=1; portb.f4=1; portb.f5=1; porte.f2=1;
    portd.f2=0;
    portb.b7=1; portd.b3=1; porte.b0=1; portd.b0=1; delay_ms(1);*/
    sept_segments(C0, 0); delay_ms(1);
    sept_segments(C1, 1); portd.b2=0; portd.b3=1; delay_ms(1);
    sept_segments(C2, 2); delay_ms(1);
    sept_segments(C3, 3); delay_ms(1);
}
}

```

# B1( \_ Datasheet IRFP460)

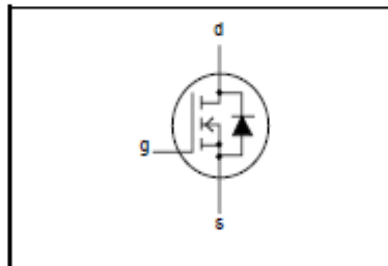
## PowerMOS transistors Avalanche energy rated

IRFP460

### FEATURES

- Repetitive Avalanche Rated
- Fast switching
- Stable off-state characteristics
- High thermal cycling performance
- Low thermal resistance

### SYMBOL



### QUICK REFERENCE DATA

$$V_{DSS} = 500 \text{ V}$$

$$I_D = 20 \text{ A}$$

$$R_{DS(ON)} \leq 0.27 \Omega$$

### GENERAL DESCRIPTION

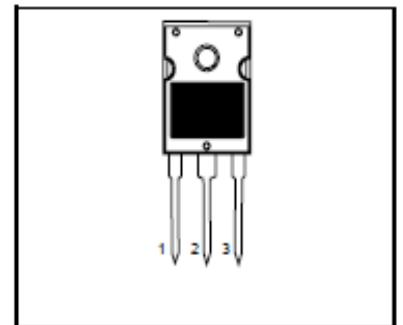
N-channel, enhancement mode field-effect power transistor, intended for use in off-line switched mode power supplies, T.V. and computer monitor power supplies, d.c. to d.c. converters, motor control circuits and general purpose switching applications.

The IRFP460 is supplied in the SOT429 (TO247) conventional leaded package.

### PINNING

PIN	DESCRIPTION
1	gate
2	drain
3	source
tab	drain

### SOT429 (TO247)



### LIMITING VALUES

Limiting values in accordance with the Absolute Maximum System (IEC 134)

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
$V_{DSS}$	Drain-source voltage	$T_j = 25 \text{ }^\circ\text{C}$ to $150 \text{ }^\circ\text{C}$	-	500	V
$V_{DGR}$	Drain-gate voltage	$T_j = 25 \text{ }^\circ\text{C}$ to $150 \text{ }^\circ\text{C}$ ; $R_{GS} = 20 \text{ k}\Omega$	-	500	V
$V_{GS}$	Gate-source voltage		-	$\pm 30$	V
$I_D$	Continuous drain current	$T_{mb} = 25 \text{ }^\circ\text{C}$ ; $V_{GS} = 10 \text{ V}$	-	20	A
		$T_{mb} = 100 \text{ }^\circ\text{C}$ ; $V_{GS} = 10 \text{ V}$	-	12.4	A
$I_{DM}$	Pulsed drain current	$T_{mb} = 25 \text{ }^\circ\text{C}$	-	80	A
$P_D$	Total dissipation	$T_{mb} = 25 \text{ }^\circ\text{C}$	-	250	W
$T_j, T_{stg}$	Operating junction and storage temperature range		- 55	150	$^\circ\text{C}$

### AVALANCHE ENERGY LIMITING VALUES

Limiting values in accordance with the Absolute Maximum System (IEC 134)

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
$E_{AS}$	Non-repetitive avalanche energy	Unclamped inductive load, $I_{AS} = 20 \text{ A}$ ; $t_p = 0.2 \text{ ms}$ ; $T_j$ prior to avalanche = $25 \text{ }^\circ\text{C}$ ; $V_{DS} \leq 50 \text{ V}$ ; $R_{GS} = 50 \Omega$ ; $V_{GS} = 10 \text{ V}$	-	1300	mJ
$E_{AR}$	Repetitive avalanche energy <sup>1</sup>	$I_{AR} = 20 \text{ A}$ ; $t_p = 2.5 \mu\text{s}$ ; $T_j$ prior to avalanche = $25 \text{ }^\circ\text{C}$ ; $R_{GS} = 50 \Omega$ ; $V_{GS} = 10 \text{ V}$	-	32	mJ
$I_{AS}, I_{AR}$	Repetitive and non-repetitive avalanche current		-	20	A



## B2( \_ Datasheet BYR29R)

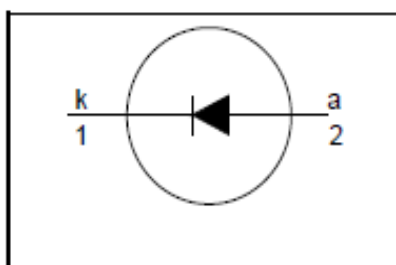
### Rectifier diodes ultrafast

### BYR29 series

#### FEATURES

- Low forward volt drop
- Fast switching
- Soft recovery characteristic
- Reverse surge capability
- High thermal cycling performance
- Low thermal resistance

#### SYMBOL



#### QUICK REFERENCE DATA

$$V_R = 500 \text{ V} / 600 \text{ V} / 700 \text{ V} / 800 \text{ V}$$

$$V_F \leq 1.5 \text{ V}$$

$$I_{F(AV)} = 8 \text{ A}$$

$$t_{rr} \leq 75 \text{ ns}$$

#### GENERAL DESCRIPTION

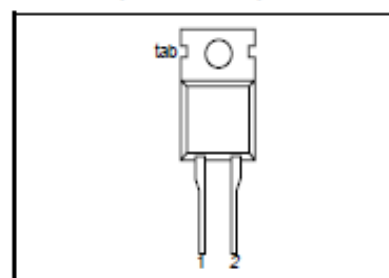
Ultra-fast, epitaxial rectifier diodes intended for use as output rectifiers in high frequency switched mode power supplies.

The BYR29 series is supplied in the conventional leaded SOD59 (TO220AC) package.

#### PINNING

PIN	DESCRIPTION
1	cathode
2	anode
tab	cathode

#### SOD59 (TO220AC)



#### LIMITING VALUES

Limiting values in accordance with the Absolute Maximum System (IEC 134).

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.				UNIT
				-500	-600	-700	-800	
$V_{RRM}$	Peak repetitive reverse voltage	BYR29	-	500	600	700	800	V
$V_{RWM}$	Crest working reverse voltage		-	500	600	700	800	V
$V_R$	Continuous reverse voltage		-	500	600	700	800	V
$I_{F(AV)}$	Average forward current <sup>1</sup>	square wave; $\delta = 0.5$ ; $T_{mb} \leq 115 \text{ }^\circ\text{C}$	-	8				A
$I_{FRM}$	Repetitive peak forward current	$t = 25 \text{ } \mu\text{s}$ ; $\delta = 0.5$ ; $T_{mb} \leq 115 \text{ }^\circ\text{C}$	-	16				A
$I_{FSM}$	Non-repetitive peak forward current	$t = 10 \text{ ms}$	-	60				A
		$t = 8.3 \text{ ms}$	-	66				A
$T_{stg}$	Storage temperature		-40	150				$^\circ\text{C}$
$T_J$	Operating junction temperature		-	150				$^\circ\text{C}$

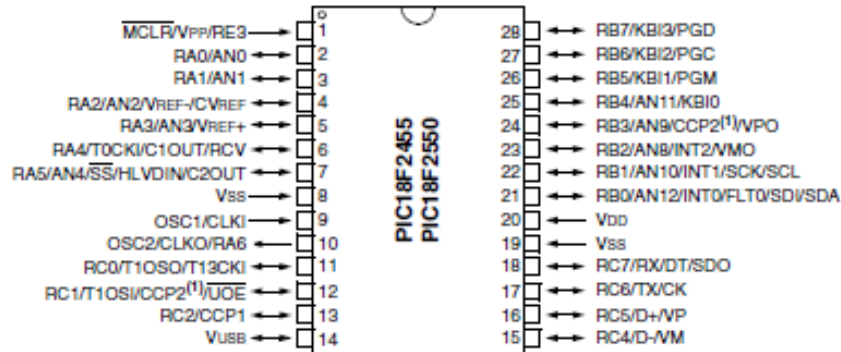
#### THERMAL RESISTANCES

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
$R_{th(j-mb)}$	Thermal resistance junction to mounting base		-	-	2.5	K/W
$R_{th(j-a)}$	Thermal resistance junction to ambient	in free air.	-	60	-	K/W

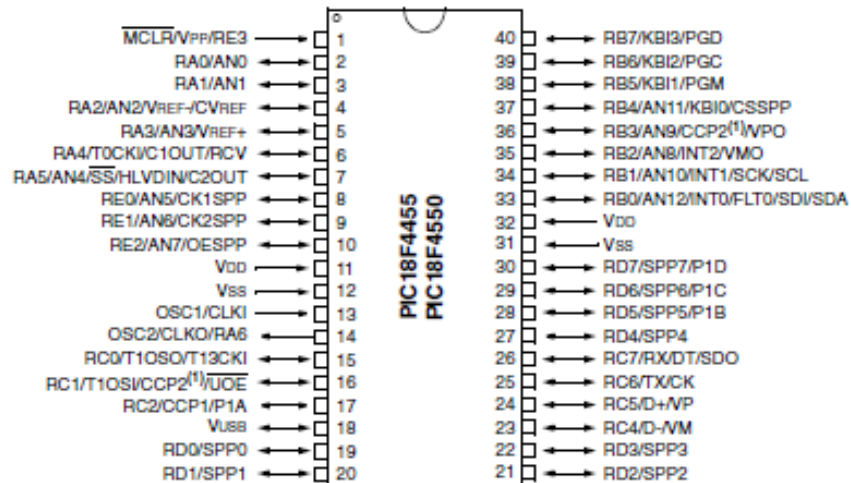
## B3( \_ Datasheet PIC18F4550)

### Pin Diagrams

#### 28-Pin PDIP, SOIC



#### 40-Pin PDIP



**Note 1:** RB3 is the alternate pin for CCP2 multiplexing.