

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne démocratique et populaire

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

جامعة سعد دحلب البلدية
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculté de Technologie

قسم الإلكترونيك
Département d'Électronique



Mémoire de Projet de Fin d'Études

présenté par

Hasnaoui Samir

&

Kherraz Abdelkader

pour l'obtention du diplôme Master en Électronique option Système de vision et
Robotique

Thème

Etude et Implémentation sur FPGA de la commande floue d'un moteur à courant continu sans balais

Proposé par : Bradai Rafik

Année Universitaire 2011-2012

ملخص:

الهدف من هذه الدراسة هو التحكم في السرعة الخاصة بمحرك ثلاثي الطور بدون فرشاة، ولتبيين ذلك اتبعنا مجموعة من الخطوات حيث تطرقنا الى المفاهيم الأساسية لهذا المحرك، والتي بينا من خلالها مختلف أنواعه وما يتعلق به. ثم قمنا بنمذجته لنصل بعدها الى دراسة المنطق الغامض له و الية التحكم في سرعته لنقوم بعدها بدراسة نظرية شاملة حول FPGA. وبخصوص تصميم هذا التحكم فقد استعملنا أداة تتمثل في System Generators الخاص بشركة XILINX. وفي الأخير قمنا بإنشاء خوارزمية التحكم على نظرية المنطق الغامض.

كلمات المفاتيح: محرك ثلاثي الطور بدون فرشاة، FPGA, System Generator, المنطق

الغامض

Résumé : L'objectif de ce projet est la commande en vitesse d'un moteur BLDC .La première partie concerne des notions fondamentales sur la machine dans laquelle, on cite les différents types de cette machine. La seconde concerne la modélisation de l'ensemble machine – commutateur. La troisième consiste à faire une étude sur la logique floue et la commande en vitesse, ensuite on montre les résultats de simulation de la commande en vitesse. La 4^{ème} partie, on a présenté quelques généralités sur la Technologie des Circuits FPGAs .Finalement on passe à l'implémentation d'un régulateur flou à partir de l'outil System Generator de Xilinx dans une cible FPGA.

Mots clés moteur BLDC, FLC, FPGA, la commande en vitesse.

Abstract : The research purpose is the speed control of the three phase brushless DC motor. The first step, relates to basic concepts on the motor in which we quote the various types of the motor. The second procedure, relates to the modeling of the motor-inverter for operation in the direct sens of rotation. The third instruction, consists in controlling the speed of fuzzy logic, then we show the simulations of the speed, as well the torque control. Before the last stage,in which we have presented generalizations about the technology of circuits FPGAs.Finally we move to the implementation of the regulator flou through the use of Generator System'sXilinx .

Keywords : the three phases brushless DC motor, FLC,FPGA, the speed control.

Dédicaces

Je dédie ce modeste travail à :

*La lumière de ma vie, le soleil de mon ciel, celle qui m'éclaire mère,
que dieu la garde et la protège.*

*Le pilier de ma vie, la lune de mes nuits, mon père, que dieu me le garde et le
protège.*

La fleur de mon cœur, mes chères sœurs.

Mon soutien dans la vie, mes chères frères : Karim, Kamel, Mahmoud.

Aux prunelles de mes yeux, mes chères enfants de mes sœurs.

Mes amis : Farid ,Abdelhakim ,Karim , Salah.

Et mes toutes camarades de 5ème année électronique.

A tous mes oncles et mes tantes, cousines et cousins.

A toute ma belle-famille sans exceptionnelle.

Tous ceux qui me sont chères.

Abdelkader

Dédicaces

Je dédie ce modeste travail à :

Mes très chers parents pour leur encouragement.

Mes frères et mes sœurs surtout sabrina.

Tous les étudiants de PFE promotion 2012.

Tous mes amis.

Samir Hasnaoui.

Listes des acronymes et abréviations

Symboles	Signification	Unité
A	La première phase de la machine.	
B	La deuxième phase de la machine.	
C	La troisième phase de la machine.	
K_h	La constante de Hall.	
V_H	La tension de Hall.	
B	Le champ magnétique d'induction.	
T_1	Le premier transistor de la parie supérieur du commutateur.	
T_2	Le deuxième transistor de la parie supérieur du	
T_3	Le troisième transistor de la parie supérieur du	
T'_1	Le premier transistor de la parie inférieur du commutateur.	
T'_2	Le deuxième transistor de la parie inférieur du	
T'_3	Le troisième transistor de la parie inférieur du commutateur.	
D_1	La première diode de la parie supérieur du commutateur.	
D_2	La deuxième diode de la parie supérieur du commutateur.	
D_3	La troisième diode de la parie supérieur du commutateur.	
D'_1	La première diode de la parie inférieur du commutateur.	
D'_2	La deuxième diode de la parie inférieur du commutateur.	
D'_3	La troisième diode de la parie inférieur du commutateur.	
e_a	La force électromotrice induite dans la phase A.	V
e_b	La force électromotrice induite dans la phase B.	V
e_c	La force électromotrice induite dans la phase C.	V
p_e	La valeur crête de la Fem.	
θ	Angle électrique.	Rd
θ_r	Angle mécanique.	Rd

P	Nombre de paires de pôles du rotor.	
W_r	La vitesse angulaire du rotor.	rd/s
W_m	La vitesse mécanique	rd/s
W_{ref}	La vitesse de référence	rd/s
C_e	Le couple électromagnétique développé	N.m /A
C_r	Le couple de résistant	N.m /A
k_e	Le coefficient de la force électromotrice	V/rad/S
F	Le coefficient de frottement	Nm.s/rad
J	Le moment d'inertie	N/m²
L	L'inductance propre de phase	H
M	La mutuel inductance	H
R	La résistance de phase	Ω
i_a	Le courant dans la phase a	A
i_b	Le courant dans la phase b	
i_c	Le courant dans la phase c.	
v_a	La tension aux bornes de la phase a	V
v_b	La tension aux bornes de la phase b	
v_c	La tension aux bornes de la phase c	
i_d	Le courant continu à l'entrée du commutateur	A
u_d	La tension à l'entrée du commutateur	V
$\vec{\phi}_M$	Le vecteur de flux magnétique créé par l'aimant permanent	
$\vec{\phi}$	Vecteur de flux magnétique créé par les trois phases	
\vec{v}	vecteur des tensions des trois phases	
\vec{i}	Vecteur des courants des trois phases	
T_m	La constante mécanique du moteur	
T_e	La constante électrique du moteur	
f_d	Fréquence de découpage	Hz
τ	Période de découpage du hacheur	
u_t	La porteuse triangulaire	
u_c	Signal de commande comparé avec la porteuse	V

Table des matières

Introduction générale.....	1
Chapitre 1: Notions de base sur Le moteur à courant continu sans balais	
1.1. Introduction	3
1.2. Constitution du moteur BLDC	3
1.3. Principe de fonctionnement du moteur BLDC	5
1.3.1. Principe de base d'un moteur électrique	5
1.3.2. Principe de fonctionnement du BLDC avec l'électronique de pilotage	6
1.3.3. Caractéristiques du BLDC	6
1.4. Technologies du moteur BLDC	7
1.4.1. Le stator	7
1.4.2. Le rotor	8
1.5. Différents types du BLD	9
1.5.1. Moteur à rotor intérieur	9
1.5.2. Moteur à rotor extérieur	10
1.5.3. Moteur à entrefer axial	10
1.6. L'alimentation électronique du BLDC	11
1.6.1. Commutateur de puissance	12
a Alimentation unidirectionnelle	13
b Alimentation bidirectionnelle	13
1.6.2. Capteurs (Capteur à effet Hall)	14
1.6.3. Logique de commande	15
1.6.4. Commande en tension ou en courant	17
a Alimentation en tension	17
b Alimentation en courant	17
1.7 Conclusion	
Chapitre 2: Modélisation du BLDC pour le fonctionnement dans le sens. Direct	
2.1. Introduction	18
2.2. Hypothèses simplificatrices	18
2.3. schéma équivalent d'un moteur BLDC	19

2.4. La force électromotrice induite	21
2.5. L'équation mécanique du mouvement	22
2.5.1. Vitesse du moteur	22
2.5.2. Couple électromagnétique	23
2.6. Modèle du BLDC alimenté par commutateur	23
2.6.1. Signaux des transistors et les séquences des phases alimentées	24
2.6.2. Expressions des forces électromotrices	25
2.6.3. Expressions des courants et de tension phase	26
2.7. Simulation en boucle ouverte	27
2.7.1. Démarrage du moteur	27
a Analyses et commentaires	29
2.7.2. Influence des paramètres de la machine sur la forme du courant	29
a Influence de resistance	29
b Influence de l'inductance	30
c Analyses et	30
2.8. Conclusion	30
Chapitre 3: La Commande en vitesse par La logique floue	
3.1. Commande en	31
3.1.1. Introduction	31
3.1.2. Modèle du BLDC lorsque deux phases sont	31
3.1.3. Stratégie d'asservissement du courant	33
a Principe du hacheur	34
b Commande en modulation de largeur d'impulsion	35
3.2 Réglage de la vitesse par la logique	37
3.2.1 Introduction	37
3.2.2 Historique.....	37
3.2.3 Généralité sur la logique floue.....	38
a Variables linguistique et ensemble floue.....	38
b Les fonctions d'appartenances.....	38
c. Univers de discours et classe d'appartenance	39
3.2.4. Les opérateurs de la logique floue	40
3.2.5. les systemes flous	40
a Fuzzification	41
b Les regles floues	41
c Inférences	42
d Défuzification	42
3.2.6 Avantages et désavantages du réglages par logique floue	43
3.2.7 Réglages de la vitesse d'un moteur BLDC par un régulateur flou à 49	44
regles	
a Description du contrôleur flou	44

b	Loi de commande	45
c	Synthèse du régulateur flou à 49 règles	46
3.2.8.	Simulation et résultats obtenus	48
a	Simulation pour le sens direct de la vitesse	48
b	Simulation pour les deux sens de rotation	50
c	analyse et commentaire	52
d	Test de la robustesse du régulateur de la vitesse	52
3.2.9	Conclusion	53

Chapitre 4: Technologie des circuits FPGA

4.1.	Introduction	54
4.2.	Les circuits logiques programmables	55
4.2.1.	Les ASICs (Application Specific Integrated Circuit)	55
4.2.2.	Les PLDs (Programmable Logic Device)	56
4.2.3	Les EPLDs "Erasable PLD"	57
4.2.4.	Les PALs "Programmable Array Logic"	57
4.2.5	Les EEPLDs "Electrically Erasable PLD"	57
4.2.6	Les FPGA "Field Programmable Gate Arrays" ou "réseaux des blocs	58
4.3.	Les FPGAs "Field Programmable Gate Arrays"	58
4.3.1 .	Description des composants FPGA	58
4.3.2.	Architecture des circuits FPGA	59
a	IOB (Input Output Bloc)	61
b	CLB (Configurable Logic Bloc)	63
4.3.3.	PROGRAMMATION ET CONFIGURATION DES CIRCUITS FPGAS	64
a	Circuit configurable	65
b	Circuit reconfigurable	65
c	Circuit partiellement reconfigurable	65
d	Circuit dynamiquement reconfigurable	65
4.3.4 .	Les avantages et les inconvénients des FPGA	66
a	Les avantages	66
b	Les Inconvénients	66
4.4.	System Generator	66
4.4.1.	Principales fonctions disponibles	67
a	Basic Element	67
b	Math	69
c	Matlab IO	69
d	Memory	69
e	DSP	69
4.5.	Conclusion	69

Chapitre 5: Implémentation sur FPGA

5.1	Introduction	69
5.2.	Implémentation d'un régulateur flou sur	69

5.2.1.	Module de calcul de fonctions d'appartenances	71
a	Fuzzification des variables	71
b	degré d'appartenance	71
c	Ensemble d'appartenances	71
5.2.2.	Module de calcul de l'inférences	72
a	Les règles d'inférences	72
b	Réalisation de la fonction OU	73
5.2.3.	Réalisation de la defuzzification	73
5.3.	Réalisation d'un intégrateur	75
5.4	Conclusion	76
	Conclusion générale	77

Liste des figures

Figure1-1	Structure élémentaire d'un moteur a courant continu sans balais.....	4
Figure1-2	Courbe des forces électromotrices, le couple et des courant dans une machine triphasée.....	7
Figure1-3	Différents types des stators.	8
Figure1-4	Différents types des rotors.	8
Figure1-5	Moteur à rotor externe.	10
Figure1-6	Moteur à entrefer axial.	11
Figure1-7	Structure générale de l'alimentation d'un moteur BLDC.....	12
Figure1-8	Alimentation unidirectionnelle.....	13
Figure1-9	Alimentation bidirectionnelle d'un moteur en étoile.	13
Figure1-10	Position des capteurs pour un moteur à trois phases.....	14
Figure1-11	Signaux logiques issus des capteurs pour une machine triphasée.....	14
Figure1-12	Signaux de commande pour une alimentation bidirectionnelle triphasée.....	16
Figure1-13	Génération des impulsions pour une alimentation bidirectionnelle triphasée.....	16
Figure2-1	Circuit de base.	19
Figure2-2	Modèle électrique simplifié du BLDC.	20
Figure2-3	Les formes typiques des courants et des	22
Figure2-4	Méthode d'élaboration des signaux de commande des transistors.....	24
Figure2-5	Signaux des transistors pour le sens direct.....	24
Figure2-6	Simulation pour le démarrage du moteur.....	28
Figure2-7	Courant de phase b(pour $R=12\text{ Ohm}$ et $L=0.0014\text{H}$).....	29
Figure2-8	Courant de phase b(pour $R=4\text{ Ohm}$ et $L=0.008\text{H}$).....	30
Figure3-1	Schéma synoptique de commande.....	33
Figure3-2	Hacheur réversible en courant.....	34

Figure3-3	Principe d'un modulateur de largeur impulsion	35
Figure3-4	Impulsions d'allumage de T_{c1} , T_{c2} obtenues par une MLI.....	36
Figure3-5	Fonction d'appartenance.....	39
Figure3-6	Structure d'un contrôleur flou	40
Figure3-7	Schéma d'un contrôleur flou typique.....	44
Figure3-8	Figure5-5 l'erreur.....	46
Figure3-9	La dérive de l'erreur.	46
Figure3-10	La sortie (u).....	46
Figure3-11	Résultats pour le fonctionnement du moreur pour le sens positive.....	49
Figure3-12	Résultats pour le fonctionnement dans les deux de sens rotation.....	51
Figure3-13	Forme de vitesse (cr=0.02).....	52
Figure3-14	Forme de vitesse (cr=0.04).....	53
Figure3-15	Forme de vitesse (cr=0.06).....	53
Figure4-1	Classification des circuits numériques.	55
Figure4-2	Les connexions dans un PLD.	57
Figure4-3	Cnnexions dans un PAL.	57
Figure4-4	Architecteur générique d'un circuit FPGA.....	58
Figure4-5	Architecteur interne du FPGA.....	59
Figure4-6	Blocs et Interconnexions programmables.....	60
Figure4-7	Situation du réseau SRAM.....	60
Figure4-8	Structure d'une cellule SRAM.....	61
Figure4-9	Exemple d'IOB.....	62
Figure4-10	Schéma d'un bloc d'entrée/sortie (IOB).....	62
Figure4-11	Bloc CLB.....	64
Figure4-12	Classification des circuits FPGAs selon leurs configurations	65
Figure5-1	Algorithme général pour l implémentation du régulateur flou.	70
Figure5-2	Implémentation du module fonction d' appartenances de mfe (NG).....	71
Figure5-3	S comportementale de (mfde NG).....	72
Figure5-4	Implémentation de ``si e est mfe(PG) et de est mfde N	72
Figure5-5	Implémentation de module de fonction ``OU``, variable(NP)..	73

Figure5-6	Implémentation de module de la defuzzification.....	74
Figure5-7	Réalisation du numérateur.....	74
Figure5-8	Réalisation du dénominateur	75
Figure5-9	Implémentation de l'intégral.....	75
Figure5-10	Résultat de simulation de l'intégral.....	75

Liste des tableaux

Tableau 2-1 Séquence des phases allumées et les états des transistors sur une période.

Tableau 2-2 Expressions des *FEM* des 3 phases (e_a, e_b, e_c) sur une période.

Tableau 3.1 : Tableau des règles (49) de décision floue (inférences)

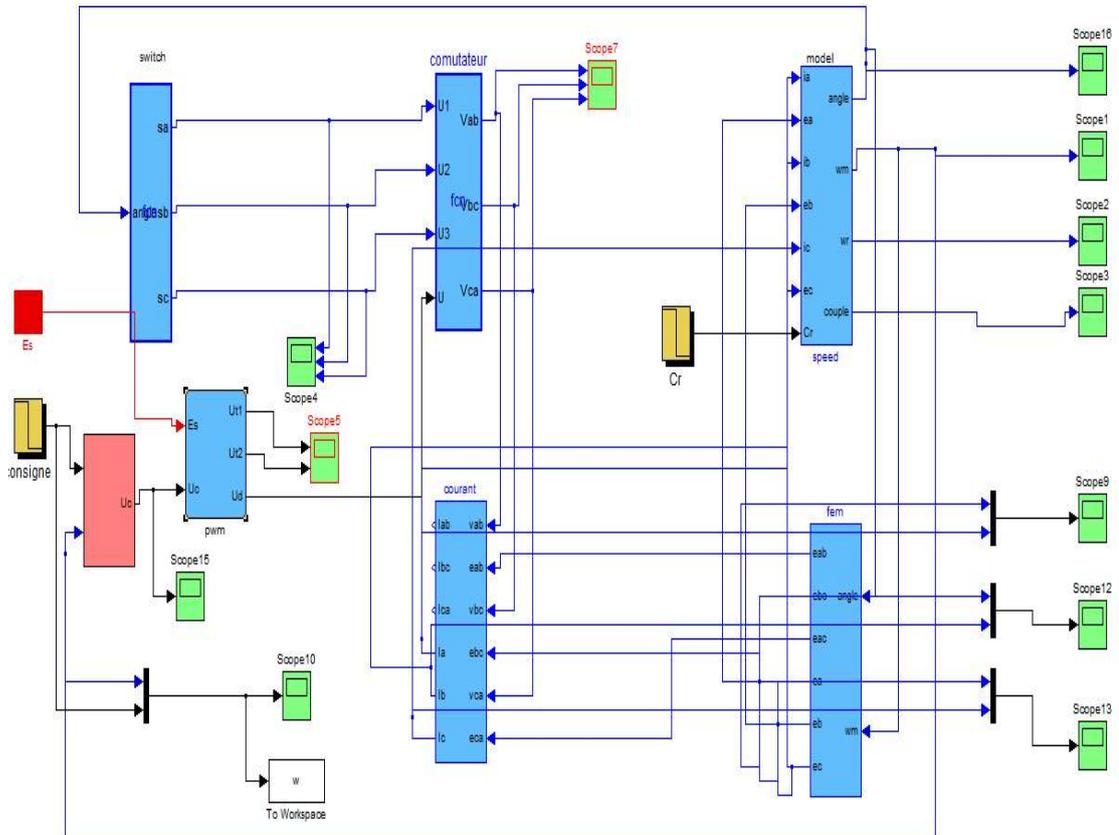
Tableau 5.1 : caractéristique du régulateur flou.

Annexes

Les paramètres du moteur utilisé.

Les paramètres du moteur étudié		Valeurs
La résistance de phase	R [Ω]	4
L'inductance propre de phase	L [H]	0.0014
Le moment d'inertie	J [N/m^2]	$4.65 * 10^{-6}$
Le constant du couple électromagnétique	k_e [$N.m/A$]	$26.1 * 10^{-3}$
Le constant du couple résistant	k_r [$N.m/A$]	$1.6667 * 10^{-4}$
Le coefficient de frottement	f [$Nm.s/rad$]	$1.5 * 10^{-6}$
L'inductance mutuelle	M [H]	- 0.000466
La tension nominale	U_d [V]	24
Le courant max	I_d [A]	2

Model du moteur utilisé et Programme des interrupteurs .



Programme des interrupteurs

```

if ((angle>=30)&& (angle<=90) )
    sa = 1; sb = 0; sc = 0;

else if ((angle>=90)&& (angle<=150))
    sa = 1 ; sb = 1; sc = 0;
else if ((angle>=150) &&(angle<=210))
    sa = 0 ;sb = 1; sc = 0;
else if ((angle>=210)&&(angle<=270))
    sa = 0; sb = 1; sc = 1;
else if ((angle>=270)&&(angle<=330))
    sa = 0 ; sb = 0; sc = 1;
else
    sa = 1 ; sb = 0; sc = 1;

end
end
end
    
```


Bibliographie

[1] : DJEMAI Mourad , TELLAA Mawloud .'commande de la vitesse et de la position d'un moteur à courant continu sans balais pour les application en robotique', Ecole nationale polytechnique EENP', El Harrach-Alger, 2006-2007.

[2] : Pierre Mayé, « Moteurs électriques pour la robotique ». Editions Dunod, Paris, 2000.

[3] Moussa Bouhrik, ' Impact des Nouveaux Matériaux ((NdFeB et SmCo)) sur les Moteurs(BLDCM) Destinés aux Véhicules Electriques ' , Thèse magister université de Batna, Mars2007.

[4] : Frédéric Gillon, « Modélisation et optimisation par plans d'expériences d'un moteur à commutation électronique ». Thèse doctoral université de Lille.

[5] :B.kazed, 'cour intelligence robotique',,,semestre3',option Système de Vision et Robotique,faculté d' Electronique , USD Blida, 2011-2012.

[6] : M. V. Ramesh, J. Amarnath, S. Kamakshaiah and G. S. Rao .'SPEED CONTROL OF BRUSHLESS DC MOTOR BY USING FUZZY LOGIC PI CONTROLLER '

Department of Electrical and Electronics Engineering, P.V.P. Siddhartha Institute of Technology, Vijayawada, A.P, India .

Department of Electrical and Electronics Engineering, J N T U H College of Engineering, Hyderabad, A.P., India .

Department of Electrical and Electronics Engineering, Vignan Institute of Technology and Science, India , September 2009 .

[7] : Atef Saleh Othman Al-Mashakbeh, ' Proportional Integral and Derivative Control of Brushless DC Motor', *Department of Electrical Engineering Tafila Technical University P.O. Box 179, Tafila – Jordan,2009.*

[8]:C.Gencer and M.Gedikpınar 'Modiling and simulation of BLDCM using Matlab/Simulink' ,¹Departement of electrical Education ,faculty of technical Education ,Firat universty Elazig 23100 ,Turkey. ²Departement of Electronic and Computer Education ,Faculty of technical Education, Firat universty Elazig 23100 ,Turkey,2006.

[09] : Mehmet Çunkaşa, Omer Aydoğdu, 'realisation of fuzzy logic controlled brushless DC motor drivers using matlab/similink', Department of Electronics and Computer, Technical Education Faculty, *Selcuk University,42003, Konya, Turkey. mcunkas@selcuk.edu.tr*,2010.

[10] : Stefán Baldursson ,' BLDC Motor Modelling and Control – A Matlab/Simulink Implementation', thèse Master , Institutionen för Energi och Miljö, May 2005.

[11] : P. Borne J. Rozinoer J.Y. Dieulot et L. Dubois ' Introduction à la Commande Floue', Technip, Paris, France, 1998.

[12]: H. Buhler ' Réglage par logique floue ', presses polytechniques etUniversitaires romandes, 1ère édition, Lausanne, Suisse, 1994.

[13] : J. Godjevac ' Idées nettes sur la logique Floue ', presses polytechniques et universitaires romandes, 1ère édition, Lausanne, Suisse, 1999.

[14]: A. Tlemçani ' Contribution à l'application des commandes adaptatives par les systèmes flous à une machine synchrone à aimants permanents alimentée par un convertisseur multicellulaire série ', thèse doctorat, Ecole nationale polytechnique, Alger, Algérie, 2007.

[15]: K.M. Passino et S. Yurkovich ' Fuzzy Control ', Addison Wesley Longman, Californie, 1998.

[16] Livre Gonzalo Cabodevila 'Commande et Identification Floue' , École Nationale Supérieure de Mécanique et des Microtechniques26, chemin de l'Épitaphe 25030 Besançon cedex – France,1993.

[17] CHEKROUN Soufyane 'COMMANDE NEURO-FLOUE SANS CAPTEUR DE VITESSE D'UNEMACHINE ASYNCHRONE TRIPHASEE' , thèse magister , ENSET d'Oran ,Algerie , Soutenu le 25 Octobre 2009

[18] : BEN YAHIA `` Commande floue et neurofloue d'une machine asynchrone alimentée par un onduleur à trois niveaux (PFE)`` . E.N.P, génie électrique 1999.

[19] : Zahir Ait Ouali 'Application des FPGA à la commande D'un moteur asynchrone ',Mémoire de Fin d'Etudes, MAGISTER en Automatique , Université MOULOUD MAMMERI de TIZI-OUZOU .

[20] : M.Walid, M.I.Moctar . ' Implémentation d'un modulateur OFDM sur un circuit FPGA. Mémoire de Fin d'Etudes, Electronique. Alger : Ecole Nationale Polytechnique (ENP), 2007.

[21] B.Farida, D.Sabrina ' Système temps réel embarqué pour commander un robot Mobile' . Mémoire de Fin d'Etudes, Automatique, Ecole Nationale Polytechnique (ENP), Alger, juin 2006 .

[22] : M. AIB Abdelghani 'Conception et implémentation d'un algorithme de vision artificielle sur FPGA' , Mémoire de Fin d'Etudes, Electronique. Alger : Ecole Nationale Polytechnique (ENP), 2008.

[23] : Mohamed Wissem NAOUAR, Mémoire de Fin d'Etudes 'Commande numérique à base de composants FPGA d' une machine synchrone Algorithmes de contrôle du courant ' ,Génie Electrique (ENIT) Mastère Systèmes Electriques (ENIT) , Tunis le 6 décembre 2007.

[24] : M. DAOUD Housseyn, Mémoire de Fin d'Etudes.' IMPLEMENTATION D'UNE COMMANDE MPPT FLOUE SUR FPGA', Ecole Nationale Polytechnique (ENP), 2005, Alger.

[25] :M' BRADAI Rafik , 'Elaboration d'éléments optimises de commandes vectorielles d'une machine asynchrone et leur implémentation sur un circuit a architecture reconfigurable de type FPGA' ,thèse Doctorat en Sciences en Génie Electrique ,option Electrification et Automatisation des Procédés Industriels, Faculté des Hydrocarbures et de la Chimie, Boumerdès ,2007.

Introduction générale

Les moteurs à courant continu sont tout à fait adaptés aux fonctionnements en régime variable mais ils présentent de gros inconvénients liés à la présence du contact mobile entre balais et collecteur (entretien, problèmes de commutation ...).

On a donc cherché à les remplacer par des machines qui utilisent un commutateur électronique à la place du commutateur mécanique : ceux sont les moteurs à courant continu sans balais (BLDC), ils possèdent quelques avantages par rapport aux moteurs à courant continu classique et moteurs à induction. Certaines de ces derniers sont une meilleure vitesse, la réponse dynamique élevée, le rendement élevé, la longue durée de fonctionnement, l'opération silencieuse et des gammes de vitesse plus élevées. En outre, les moteurs de BLDC sont fiables, faciles à commander, et peu coûteux.

Le moteur BLDC se trouve dans différentes applications exigeant des performances élevées en terme de fiabilité et sécurité de fonctionnement tel que les véhicules électriques et hybrides, les engins spatiaux, l'aéronautique, la robotique, ainsi dans les petites applications telles que l'entraînement des disques durs et des lecteurs de CD-ROM, d'automatisation industrielle ,... etc.

D'autre part, les circuits FPGA (Field Programmable Gate Array), qui sont des circuits programmables adaptables à des besoins divers, deviennent incontournables dans les applications nécessitant un temps de développement rapide et une modularité garantie. Ils sont surtout utilisés dans les systèmes embarqués (avionique, automobile, espace,...) et tendent à se généraliser dans le domaine des applications on chip.

L'objet de ce projet est présenté en cinq chapitres :

Le premier chapitre est consacré à l'étude du moteur BLDC associé à son commutateur et sa logique de commande.

Le deuxième chapitre est consacré à l'étude des caractéristiques du moteur.

Le troisième chapitre présente une étude sur la logique floue et la commande en cascade de vitesse par un régulateur flou, (dans cette structure on trouve la vitesse comme variable de la boucle externe et le courant dans la branche continue comme variable secondaire), aussi nous présentons la structure d'un régulateur flou, le principe de fonctionnement de l'hacheur, éventuellement des résultats de simulation, ainsi que les tests de la robustesse du régulateur de vitesse.

Le quatrième chapitre présente des généralités sur la technologie des circuits FPGA.

Le 5^{eme} et le dernier chapitre est dédié à l'implantation sur une carte à FPGA de l'algorithme de la commande floue.

En fin, nous terminons par une conclusion générale qui résume tous les résultats obtenus dans ce présent mémoire.

Chapitre 1 Notion de base sur le moteur a courant continu sans balais

1.1 Introduction

Récemment, des moteurs à courant continue ont été graduellement remplacés par des moteurs BLDC puisque les applications industrielles exigent des actionneurs plus puissants ont de faible encombrement. La souplesse à la commande, la longue durée de vie due à l'absence de la commutation mécanique ainsi qu'il présente moins de bruit acoustique sont les principaux avantages de ce moteur.

Leur principal inconvénient est la nécessité d'avoir une électronique de contrôle. Mais cet inconvénient est dépassé grâce au progrès réalisé dans les domaines de l'électronique de puissance et l'arrivée des nouveaux calculateur numériques très sophistiqués ce qui permet de réaliser des variateurs performants. Dans ce présent chapitre on va étudier le principe de fonctionnement de ce moteur.

1.2 Constitution du moteur BLDC

Les moteurs BLDC sont constitués de 3 éléments principaux :

- Une partie fixe, le stator, muni de trois groupes de bobines, appelées les trois phases du moteur. Ces bobines fonctionnent comme des électro- aimants et permettent de générer diverses orientations du champ magnétique régulièrement réparties autour de l'axe central du moteur.

- Une partie tournante, le rotor, muni d'aimants permanents. Comme l'aiguille d'une boussole, ces aimants vont en permanence entraîner le rotor pour tenter de s'aligner sur le champ magnétique du stator. Pour une durée de vie optimale du moteur, le rotor est monté sur roulements à billes.

Trois capteurs magnétiques à " effet Hall ". Ces capteurs permettent de connaître à chaque instant la position des aimants du rotor [1].

La machine à courant continu sans balais se présente généralement munie de ses capteurs de position. Ses connexions extérieures comprennent donc les bornes des enroulements et les bornes des capteurs.

Remarque

Il existe quelques réalisations des moteurs BLDC dans lesquelles l'électronique effectuant les commutations est intégrée. La machine ne comporte alors que deux fils comme un moteur à courant continu classique.

On peut trouver également des moteurs BLDC qui ne possèdent pas des capteurs.

C'est le cas des moteurs désigné pour la commande sans capteurs appelé aussi « Sensorless Control », dans ce cas le contrôleur utilise la force contre électromotrice créée par la rotation du rotor au passage devant les bobines pour déterminer la position du rotor. La machine ne comporte alors que trois fils comme une machine synchrone à aimant permanent [1].

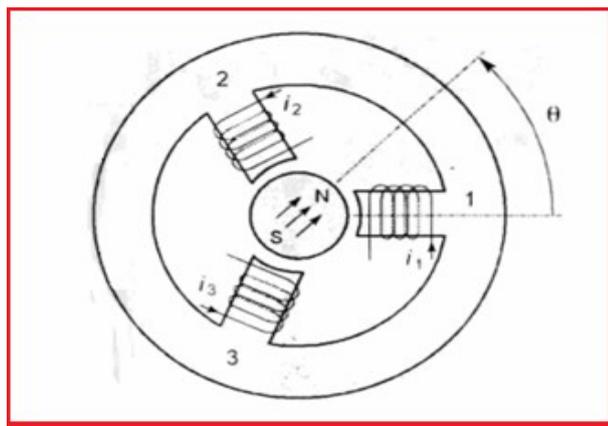


Figure 1.1: Structure élémentaire d'un moteur à courant continu sans balais

1.3 Principe de fonctionnement du moteur BLDC

1.3.1 Principe de base d'un moteur électrique

D'une manière générale, une machine électrique est utilisée comme convertisseur d'énergie électrique en une énergie mécanique. Le couplage est réalisé par l'énergie magnétique, selon la loi d'Ampère [1].

$$\vec{F} = \vec{J} * \vec{B} \quad (1.1)$$

Dans le cas d'une machine électrique utilisée en mode moteur, une force mécanique \mathbf{F} est produite en couplant des charges électriques en mouvement \mathbf{J} dans un champ magnétique \mathbf{B} . Plusieurs types d'architectures différentes existent pour exploiter la loi d'Ampère (moteurs à induction, à courant continu, à aimant permanent, etc.). Dans le cas d'une machine électrique utilisée en mode générateur, une force contre-électromotrice est produite \mathbf{E} lorsqu'un conducteur se déplace dans un champ magnétique \mathbf{B} [1]. Dans le cas du BLDC, un champ magnétique est créé par un aimant permanent placé sur le rotor et le courant est injecté seulement dans le bobinage du stator de telle façon à créer des pôles. Les pôles magnétiques de l'aimant permanent du rotor sont ainsi attirés et repoussés par les pôles du stator. En alternant dans le temps et dans l'espace le signe (Nord/Sud) de ces pôles et en variant l'intensité du courant, le rotor peut tourner avec une vitesse ω_m , produire un couple \mathbf{C}_e et créer ainsi une puissance mécanique de sortie :

$$P = C_e \cdot \omega_m \quad (1.2)$$

Puisqu'il n'y a aucun moyen de contrôler le champ magnétique créé par l'aimant Permanent, la seule chose qui reste à contrôler pour faire tourner le rotor est le courant nécessaire à produire un couple sur celui-ci [3].

1.3.2 Principe de fonctionnement du BLDC avec l'électronique de pilotage

L'étude du moteur lui-même est indissociable de celle de l'électronique qui lui est associée. On a en fait une machine alternative qui est transformée en moteur à courant continu quand on y inclut les circuits électroniques assurant les commutations.

L'électronique de pilotage détermine la position du rotor à partir des capteurs à effet Hall. Elle en déduit l'orientation à donner au champ magnétique du stator. Au cours de la rotation, elle commande les trois bobines pour ajuster régulièrement l'orientation du champ à la position du rotor, de façon à entraîner celui-ci dans le sens choisi par l'utilisateur [1].

1.3.3 Caractéristiques du BLDC

Considérons le cas le plus courant, la machine triphasée. Lorsque le rotor est en mouvement, les enroulements sont le siège de forces contre-électromotrices induites. Leur variation dépend de la forme des aimants et de la répartition des conducteurs le long de l'entrefer ainsi que la position du rotor.

Dans les moteurs à courant continu sans balais, on souhaite alimenter les enroulements par des courants en créneaux pour simplifier l'électronique correspondante. Alors, les constructeurs font en sorte qu'à vitesse fixe, la variation des forces contre-électromotrices en fonction de la position du rotor soit trapézoïdale afin d'obtenir un couple constant figure (1.2) [2].

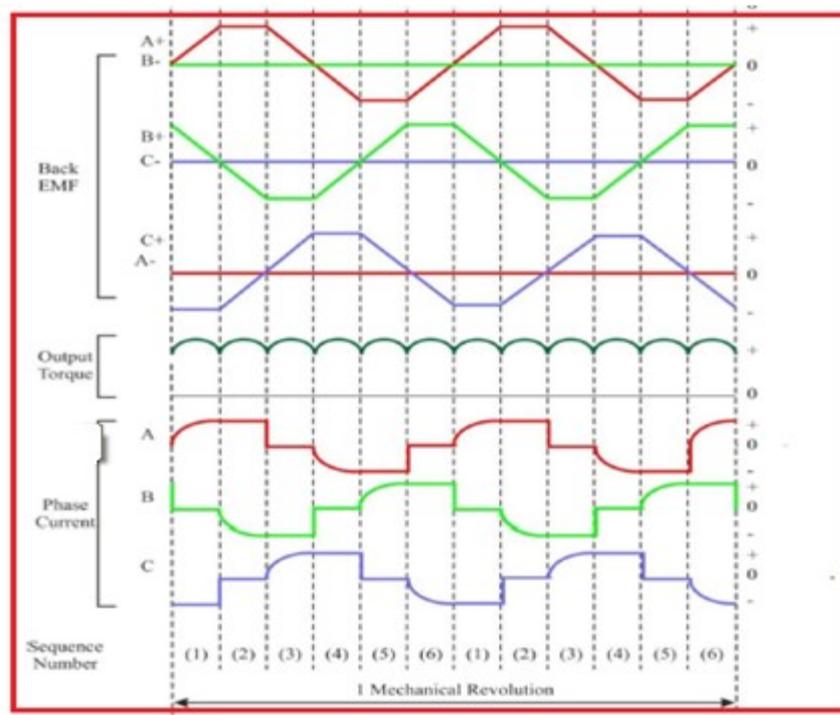


Figure 1.2 : Courbe des forces électromotrices, le couple et des courants dans une machine triphasée.

1.4 Technologies du moteur BLDC

1.4.1 Le stator

Le stator porte les enroulements qui peuvent prendre plusieurs formes :

- bobines concentrées autour de pôles saillants. (figure1.3 a).
- enroulements repartis dans des encoches. (figure1.3 b).
- enroulements repartis le long de l'entrefer sans encoches. (figure1.3 c).

La première solution, qui correspond à celle qui en général est employée pour les rotors des moteurs à courant continu ou pour les stators de moteur pas à pas est peu utilisée pour les moteurs sans balais. Les machines obtenues ont des performances modestes et le couple présente des irrégularités dues aux variations de réluctance.

La deuxième solution est la plus classique. Par une répartition adaptée des conducteurs dans les encoches, on s'approche de la forme trapézoïdale des variations des forces électromotrices induites. Une géométrie bien choisie des dents permet de minimiser les irrégularités de couple.

Dans le troisième cas, on élimine toute variation de couple due aux modifications de réluctance. Cependant, la construction est moins solide et le comportement thermique est moins bon. Cette solution est néanmoins intéressante pour les petits moteurs [2].

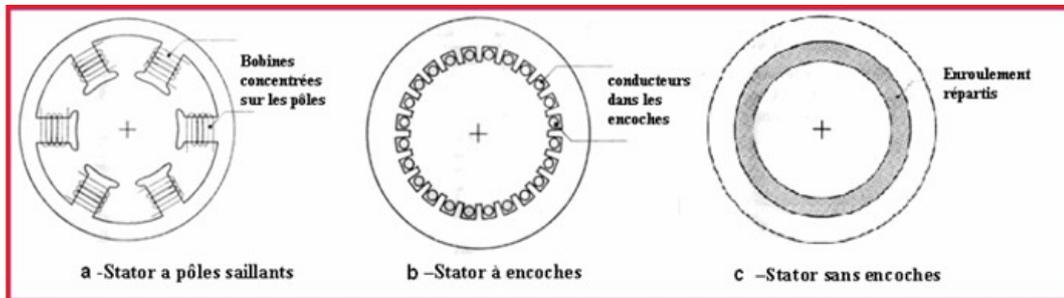


Figure 1.3 : Différents types des stators.

1.4.2 Le rotor :

Le rotor porte des aimants tels qu'en Samarium-Cobalt ou en Néodyme-Fer-Bore. Le nombre de pôles magnétiques est toujours pair. Les aimants peuvent se présenter de différentes manières :

- aimants en tuile sur un support ferromagnétique (figure 1.4 a).
- aimants en anneau fabriqué dans un matériau isotrope (figure 1.4 b).
- aimants munis de pièces polaires (figure 1.4 c).

La première solution est la plus classique, la seconde peut être intéressante pour les petites machines. Le troisième cas peut permettre une concentration du flux si la surface de la pièce polaire en regard de l'entrefer est inférieure à la surface présentée par l'aimant. On obtient ainsi des champs magnétiques élevés qui conduisent à des machines de grandes performances [2].

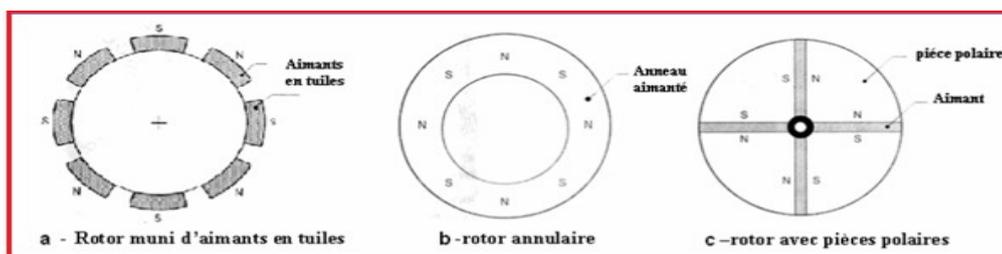


Figure 1.4 : Différents types des rotors.

1.5 Différents types du BLDC

Il existe de nombreuses configurations de moteur BLDC. La raison principale de cette diversité est que les utilisateurs ont tous des besoins spécifiques. Par exemple, pour la conception d'un servomoteur, l'inertie devra être la plus faible possible, afin de permettre une accélération et une décélération, la plus rapide possible. Il sera donc préférable d'utiliser un moteur à rotor intérieur avec des aimants à hautes énergies. Tandis que, pour une application où la vitesse doit rester la plus constante possible, telle que la motorisation d'un disque magnétique, un moteur à rotor extérieur du fait de son inertie, est préférable. Le choix du type de moteur est donc la décision la plus fondamentale dans un travail de conception.

Trois grandes familles existent, les moteurs à rotor intérieur, à rotor extérieur ou à entrefer axial. Ces familles se décomposent elles-mêmes en deux parties: les machines à forme d'onde sinusoïdale ou trapézoïdale [4].

1.5.1 Moteur à rotor intérieur

Le moteur à rotor intérieur a une configuration classique. Le rotor est constitué d'aimants assemblés sur une douille magnétique. Le stator est similaire à celui des machines triphasées à induction. D'ailleurs, les stators des machines à induction sont parfois directement utilisés dans la fabrication des machines BLDC, pour des raisons d'économie [4].

La figure 1.1 présente un exemple de moteur à rotor intérieur.

Cette structure présente l'avantage d'un ratio couple / vitesse élevé. Néanmoins une difficulté de construction existe, il faut maintenir les aimants, même à vitesse élevée. Le plus souvent, ils sont collés et frétés. La puissance volumique d'une machine BLDC à aimants ferrites est à peu près équivalente à un moteur à induction, mais son rendement et son facteur de puissance sont généralement bien meilleurs [4].

1.5.2 Moteur à rotor extérieur

La figure suivante présente le schéma d'un moteur à rotor extérieur :

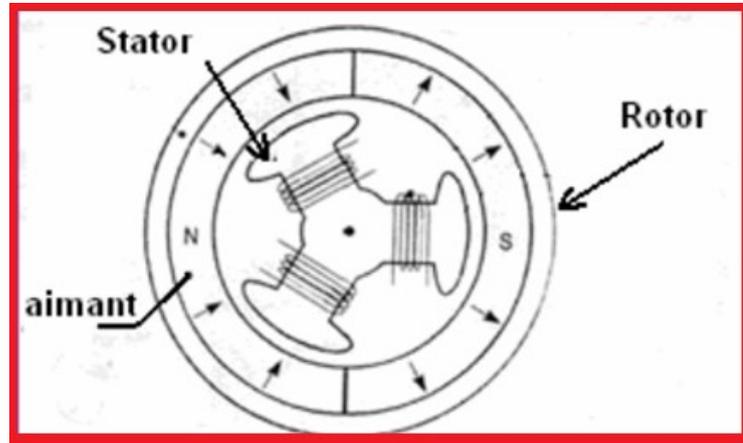


Figure 1.5 Moteur à rotor externe.

Les tôles utilisées au stator ressemblent à celles de l'induit de la machine à courant continu à balais. Ce type de stator est simple à bobiner, car le fil se bobine directement sur le stator section par section. Le rotor est constitué d'aimants montés dans une cloche magnétique permettant la continuité du champ. Dans cette configuration, il n'y a plus de problème pour maintenir les aimants, car ils sont plaqués sur la cloche par l'action de la force centrifuge [1].

Cette structure est celle utilisée pour l'entraînement des lecteurs CD-ROM et elle est aussi fréquemment utilisée dans les applications de ventilation pour son faible coût et sa facilité de fabrication. Cependant, les machines à rotor extérieur ne sont utilisées que pour des applications à vitesse constante car le rotor a une inertie importante de par sa position externe [1].

1.5.3 Moteur à entrefer axial

Dans certaines applications à encombrement réduit, des machines à entrefer axial ont fait leur apparition (moteur de disque dur d'ordinateur, table tournantes, ventilateur) .Les aimants sont montés sur un disque magnétique mobile. Les bobines

sont fixées en général sur le circuit imprimé avec les composants électroniques. Sous le circuit une plaque métallique permet au champ de se refermer. La figure 1.6 représente le schéma d'un moteur à entrefer axial.

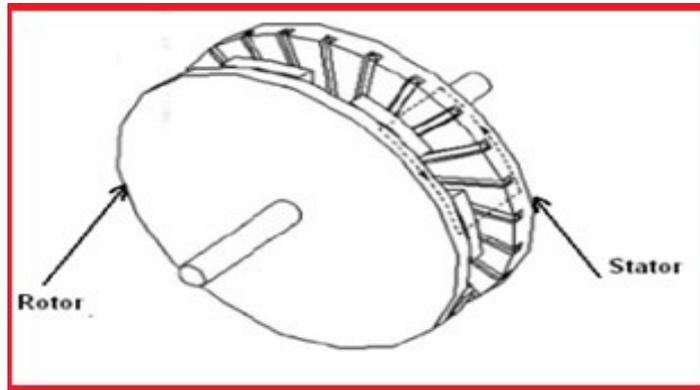


Figure 1.6. : Moteur à entrefer axial.

Ces machines ont en général un grand nombre de pôles et fonctionnent à basse vitesse (inférieure à 1000tr/min) afin d'éviter un échauffement excessif dû au courant de Foucault [1].

Les avantages du moteur à entrefer axial sont : un faible coût, sa forme plate et l'absence de couple réactant à faible vitesse [4].

On peut également faire d'autre classement des moteurs BLDC en fonction du nombre des enroulements dans le stator. On trouve les moteurs biphasés, triphasés, et aussi à quatre phases... etc. [2]

1.6 L'alimentation électronique du BLDC :

La structure globale d'un moteur à courant continu sans balais comprend :

- Des capteurs, munis éventuellement des circuits de mise en forme, qui permettent de connaître la position du rotor à tout instant en général des capteurs à effet hall.
- un circuit logique qui élabore les commutations à appliquer à différentes phases du moteur compte tenu des signaux issus des capteurs.

- un commutateur de puissance qui alimente les enroulements en fonction des informations reçues par la logique de commande.
- une commande de la tension ou du courant permettant d'agir sur la vitesse du moteur.

Les trois premières fonctions permettent d'effectuer les commutations assurées par le système balais-collecteur dans un moteur à courant continu classique tandis que la dernière est analogue à celle qui est utilisée pour la commande de ces mêmes moteurs [2].

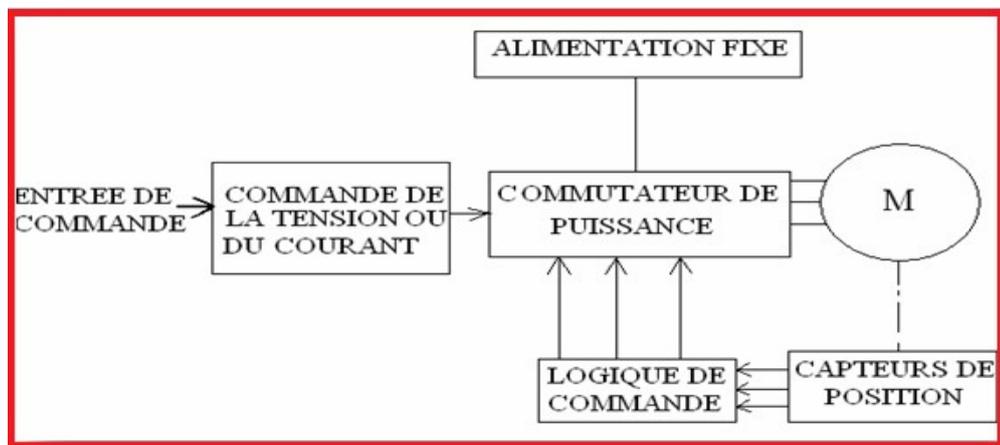


Figure 1.7: Structure générale de l'alimentation d'un moteur BLDC.

1.6.1 Commutateur de puissance

On prévoit en général l'inversion du sens du courant dans les phases (alimentation Bidirectionnelle) car la machine est alors mieux utilisée. Les enroulements peuvent être couplés en étoile ou en triangle.

On rencontre aussi parfois des alimentations unidirectionnelles, mais uniquement lorsque les phases sont couplées en étoile avec neutre Sortie [2].

a Alimentation unidirectionnelle :

Dans ce cas, on a besoin d'un transistor et d'une diode par phase figure (1.8). Lorsqu'un transistor est saturé, la tension d'alimentation est appliquée aux bornes de la phase. Quand on bloque le transistor, la force électromotrice induite dans l'inductance du circuit impose la mise en conduction de la diode de roue libre qui court-circuite alors l'enroulement [2].

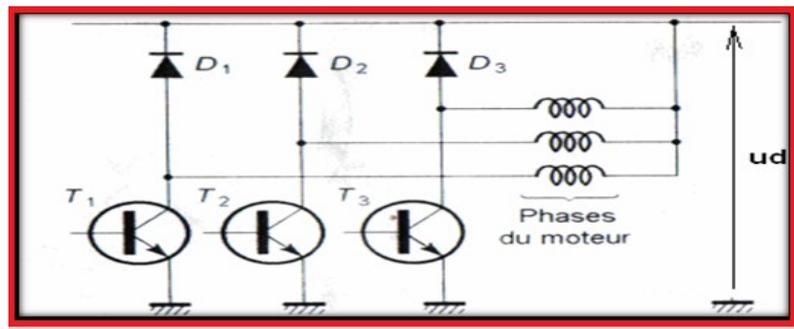


Figure 1.8 : Alimentation unidirectionnelle.

b Alimentation bidirectionnelle

Il faut ici deux transistors et deux diodes par phase. Le montage est identique, que le moteur soit en étoile ou en triangle. On sature simultanément deux transistors, l'un du groupe T_1, T_2, T_3 et l'autre du groupe T'_1, T'_2, T'_3 . Dans le cas du moteur triphasé en étoile, la tension U_d est appliquée aux bornes de deux enroulements en série. Dans le cas du moteur triphasé en triangle, cette tension est appliquée à une phase en parallèle avec les deux autres phases en série. Les six diodes assurent la fonction de roue libre dans tous les cas de la figure (1.9) [2].

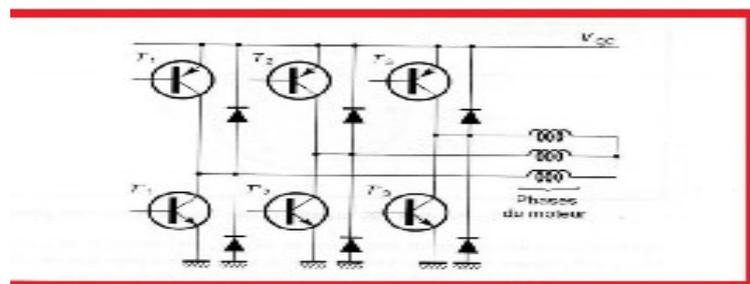


Figure 1.9 : Alimentation bidirectionnelle d'un moteur en étoile.

1.6.2 Capteurs Capteur à effet Hall

Pour connaître la position instantanée du rotor, il faut disposer sur le stator d'un certain nombre de capteurs. On utilise soit des capteurs magnétiques à effet hall, soit des capteurs optiques. Pour une machine triphasée, il faut des capteurs régulièrement espacés figure (1.10).

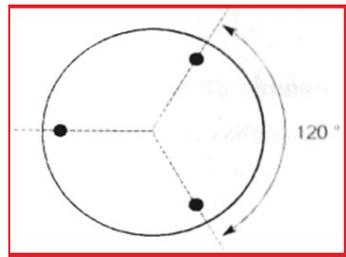


Figure 1.10 : Position des capteurs pour un moteur à trois phases.

Après mise en forme par un comparateur adapté, on obtient trois signaux logiques **h1**, **h2** et **h3** figure (1.11). Pour un moteur à quatre phases on se contente de deux capteurs décalés de 90° (deux capteurs supplémentaires donneraient des signaux inverses, ce qui ne sert à rien) [2].

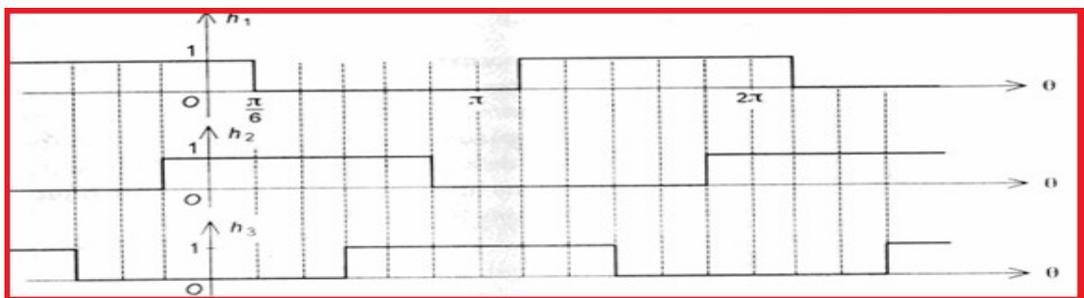


Figure 1.11 : Signaux logiques issus des capteurs pour une machine triphasée.

Lorsqu'un conducteur est parcouru par un courant électrique est placé dans un champ magnétique il se produit une force (force de Lorentz) perpendiculaire à la direction du courant ,cette force va avoir tendance à déplacer ces charges vers l'un des cotes du conducteur, ce déplacement des électrons va donc produire un différence de potentielle dont l'intensité est fonction du courant i et de l'induction magnétique B .

La *DDP* créée entre les deux cotés du conducteur est donnée par [5] :

$$U_H = \frac{i \cdot B}{nqd} \quad (1.3)$$

Avec :

i : Courant appliqué.

d : Epaisseur du conducteur.

n : Densité des charges électriques.

q : Charge électrique d'un électron.

1.6.3 Logique de commande

Ce circuit reçoit les informations logiques issues des capteurs et délivre les signaux de commande des interrupteurs nécessaires au fonctionnement du circuit de puissance. Par exemple, dans le cas du moteur triphasé à alimentation bidirectionnelle, on dispose de trois signaux h_1 , h_2 , et h_3 et on veut élaborer six signaux logiques c_1 , c_2 , c_3 , c'_1 , c'_2 et c'_3 destinés à la commande des six transistors pour obtenir les courants souhaités dans les trois phases du moteur figure (1.11). On constate qu'il faut assurer les équations logiques suivantes [2]:

$$C_1 = h_1 \overline{h_2} \quad \check{C}_1 = \overline{h_1} h_2$$

$$C_2 = h_2 \overline{h_3} \quad \check{C}_2 = \overline{h_2} h_3$$

$$C_3 = h_3 \overline{h_1} \quad \check{C}_3 = \overline{h_3} h_1$$

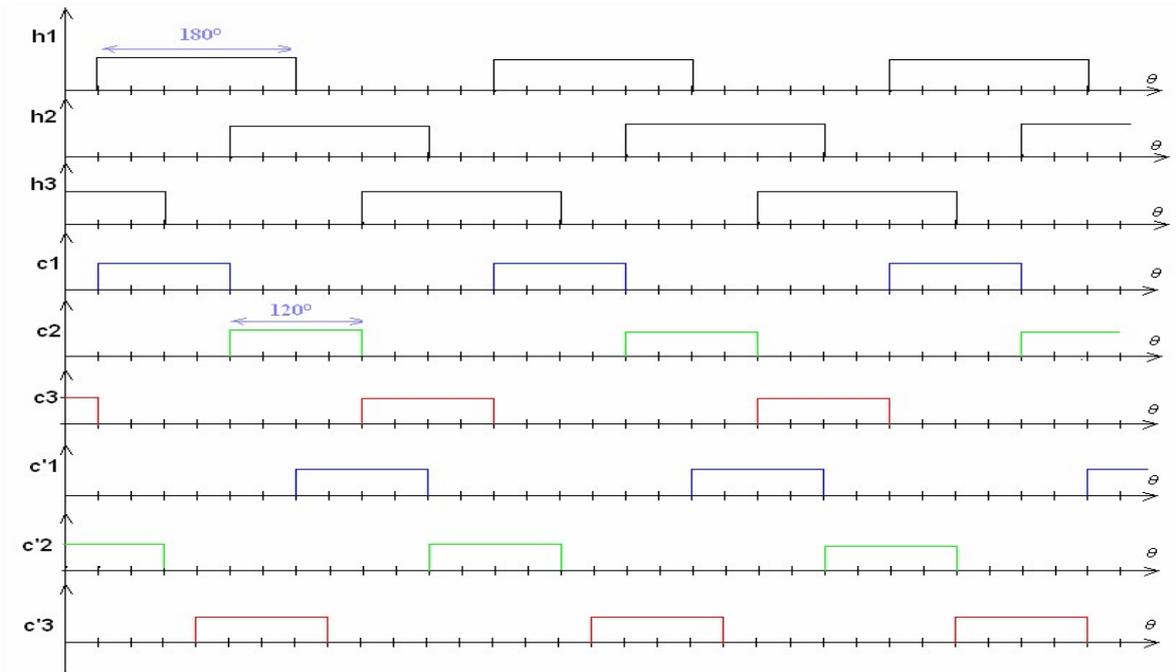


Figure 1.12: Signaux de commande pour une alimentation bidirectionnelle triphasée

Cette logique de commande peut être facilement réalisée avec des circuits élémentaires figure (1.12), elle est en général intégrée dans des circuits spécialisés pour moteurs à courant continu sans balais [2].

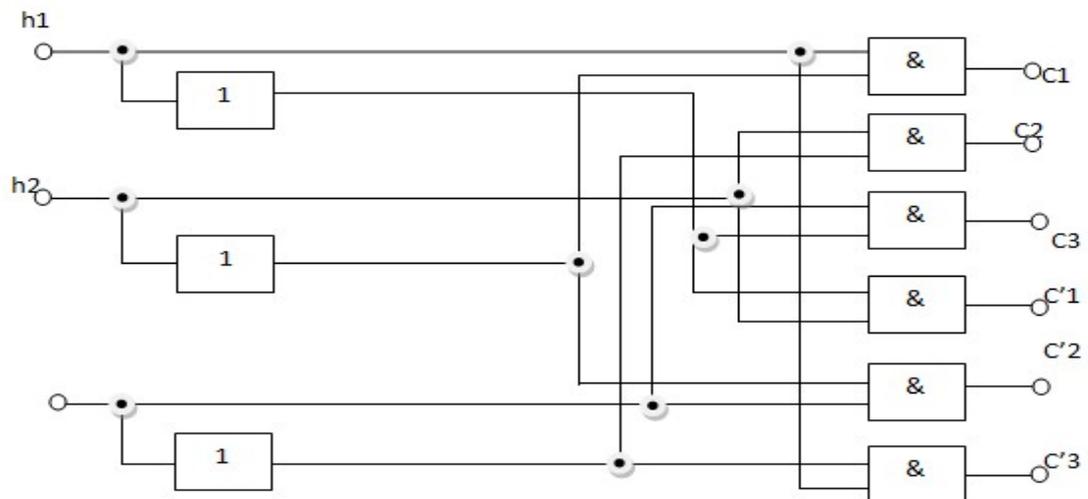


Figure 1.13: Génération des impulsions pour une alimentation bidirectionnelle triphasée.

1.6.4 Commande en tension ou en courant

a Alimentation en tension

On a vu que le moteur associé à son commutateur de puissance avec sa logique de commande et ses capteurs a un comportement analogue à celui d'une machine à courant continu. On peut donc le commander en tension en faisant varier l'alimentation U_d (appliquée au commutateur. Pour avoir un rendement correct, on utilise en général un Hacheur pour cette opération. Son fonctionnement est identique à celui du moteur à courant continu [2].

b Alimentation en courant

Il est possible, comme pour une machine classique, d'utiliser une source de courant plutôt qu'une source de tension. De plus, dans le cas du moteur sans balais, il y a deux possibilités de réglage de courant, on trouve le réglage du courant total à l'entrée du commutateur, ou encore on régule le courant dans les phases par découpage pour s'approcher de la forme idéale de ses variations lors des commutations [2].

1.7 Conclusion

On a vu dans ce chapitre les différents types de moteur BLDC, ainsi on a donné l'information globale concernant sa logique de commutation. On a concentré l'étude sur le moteur triphasé puisque c'est le plus utilisé à cause de sa structure élémentaire et puisque l'étude qui va venir est consacrée à ce type de moteur. Dans ce chapitre, on a saisi bien le principe de fonctionnement de ce moteur, ce qui nous permet d'aiguiller l'étude vers la modélisation en vue de la commande.

Chapitre 2 Modélisation du moteur à courant continu sans balais

2.1 Introduction

Afin de mieux voir les avantages du moteur à courant continu sans balais et en vue de tester ces performances on doit disposer d'un modèle dynamique sous forme des équations différentielles.

Ce modèle permet de simuler son comportement et de reproduire sa vitesse, Son couple et les courants ainsi que les Fem dans chaque phase. Le modèle est issu des équations mécaniques et électriques de l'ensemble moteur-commutateur. Dans ce chapitre, on va donner le modèle correspondant à chaque séquence d'allumage sous forme des équations différentielles ainsi que la simulation du comportement dynamique de l'ensemble moteur commutateur à base de ces modèles.

2.2 Hypothèses simplificatrices

L'analyse est faite selon les hypothèses suivantes :

➤ Les six transistors $T_1, T_2, T_3, T_1', T_2', T_3'$ ont des caractéristiques identiques, Dans l'état-**off** ils sont modélisés comme des impédances infinies et dans l'état-**on** ils sont représentés comme des contacts fermés. Dans notre cas on néglige la chute de tension dans les transistors. Les mêmes suppositions sont valables pour les diodes de roue libre notées $D_1, D_2, D_3, D_1', D_2', D_3'$

- Les trois phases du moteur sont entièrement symétriques et produisent des Fem

Induites périodiques de formes trapézoïdales.

- Les paramètres du moteur sont constants.
- Le temps de commutation entre les phases est séparé avec 60° électrique [1].
- Le circuit magnétique est linéaire (absence de saturation).
- Le circuit de base est illustré par la figure (2.1)

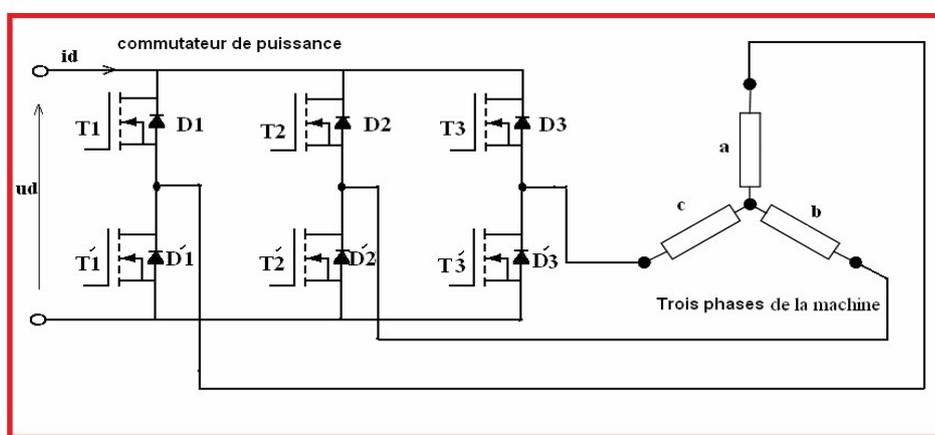


Figure 2.1 : Circuit de base.

2.3 Schéma équivalent d'un moteur BLDC

On considère le moteur triphasé en étoile avec alimentation bidirectionnelle deux bobines sont branchées en série au borne de générateur figure (2.1) .

Pour un enroulement symétrique et un système équilibré figure (2.2), la matrice des tensions aux bornes des trois phases sont donné par l'équation (2-1) [8] :

$$\begin{pmatrix} V_a \\ V_b \\ V_c \end{pmatrix} = \begin{pmatrix} R & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & R \end{pmatrix} \begin{pmatrix} i_a \\ i_b \\ i_c \end{pmatrix} + \frac{d}{dt} \begin{pmatrix} L_{aa} & L_{ab} & L_{ac} \\ L_{ba} & L_{bb} & L_{bc} \\ L_{ca} & L_{cb} & L_{cc} \end{pmatrix} \begin{pmatrix} i_a \\ i_b \\ i_c \end{pmatrix} + \begin{pmatrix} e_a \\ e_b \\ e_c \end{pmatrix} \quad (2-1)$$

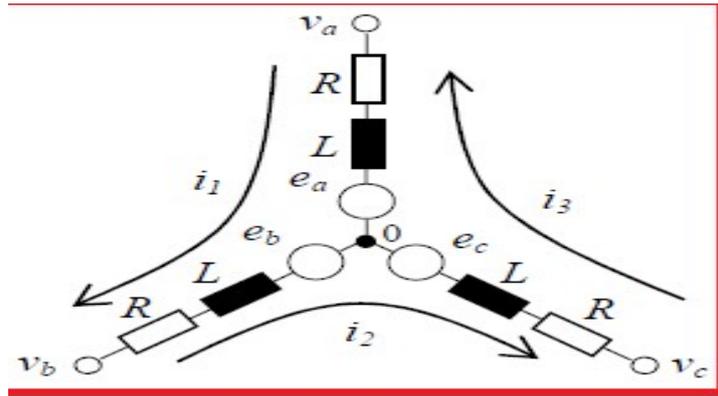


Figure 2.2 : Modèle électrique simplifié du BLDC.

La position électrique θ_r du rotor se déduit directement de la position mécanique du rotor θ_m qui est mesurée à l'aide d'un capteur placé sur l'arbre du rotor [6]. La relation entre les deux grandeurs est donnée par l'équation (2-2) :

$$\theta_r = p * \theta_m \quad (2-2)$$

Avec :

p : Le nombre de paires de pôles du moteur.

Par conséquent, l'expression de la vitesse électrique est :

$$\omega_r = \frac{d\theta}{dt} = p \cdot \frac{d\theta_m}{dt} = p \omega_m \quad (2-3)$$

Les inductances et les mutuelles sont constantes pour les aimants permanents montés sur la surface du rotor cylindrique, et lorsque les enroulements sont symétriques donc [7] :

$$L = L_{aa} = L_{bb} = L_{cc} \quad (2-4)$$

$$M = L_{ab} = L_{ac} = L_{bc} = L_{ba} = L_{ca} = L_{cb} \quad (2-5)$$

L'équation sous forme matricielle :

$$\begin{pmatrix} V_a \\ V_b \\ V_c \end{pmatrix} = \begin{pmatrix} R & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & R \end{pmatrix} \begin{pmatrix} i_a \\ i_b \\ i_c \end{pmatrix} + \frac{d}{dt} \begin{pmatrix} L & M & M \\ M & L & M \\ M & M & L \end{pmatrix} \begin{pmatrix} i_a \\ i_b \\ i_c \end{pmatrix} + \begin{pmatrix} e_a \\ e_b \\ e_c \end{pmatrix} \quad (2-6)$$

Puisque les enroulements du stator sont couplés en étoile, la somme des trois courants de phase est nulle [10].

$$i_a + i_b + i_c = 0 \quad (2-7)$$

donc :

$$i_a + i_b = -i_c \quad (2-8)$$

Par conséquent, la tension prend la forme suivante:

$$\begin{pmatrix} V_a \\ V_b \\ V_c \end{pmatrix} = \begin{pmatrix} R & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & R \end{pmatrix} \begin{pmatrix} i_a \\ i_b \\ i_c \end{pmatrix} + \frac{d}{dt} \begin{pmatrix} L-M & 0 & 0 \\ 0 & L-M & 0 \\ 0 & 0 & L-M \end{pmatrix} \begin{pmatrix} i_a \\ i_b \\ i_c \end{pmatrix} + \begin{pmatrix} e_a \\ e_b \\ e_c \end{pmatrix} \quad (2-9)$$

Avec :

$$e_a = E \cdot f_a(\theta) : \text{force électromotrice de la phase } \mathbf{a}. \quad (2-10-a)$$

$$e_b = E \cdot f_b(\theta) : \text{Force électromotrice de la phase } \mathbf{b}. \quad (2-10-b)$$

$$e_c = E \cdot f_c(\theta) : \text{Force électromotrice de la phase } \mathbf{c}. \quad (2-10-c)$$

Et :

$$E = k_e \cdot \omega_m \quad (2-11)$$

k_e : le coefficient de la force électromotrice.

$f_a(\theta)$, $f_b(\theta)$ et $f_c(\theta)$ sont des fonctions dépendent de la position du rotor [9].

2.4 La force électromotrice

La force électromotrice induite est déterminée par la position angulaire de l'aimant, C'est à dire du rotor, il est très important de remarquer que la valeur crête de la Fem est directement proportionnelle à la vitesse angulaire instantanée du rotor [1] :

$$E = k_e \cdot \omega_r \quad (2-12)$$

Les fonctions des forces électromotrices Fem sont supposées de formes trapézoïdales et le courant de la phase a commence à 30° (angle électrique)
 Les formes typiques des Fem induites et des courants dans les trois phases sont illustrées dans la figure (2.3) :

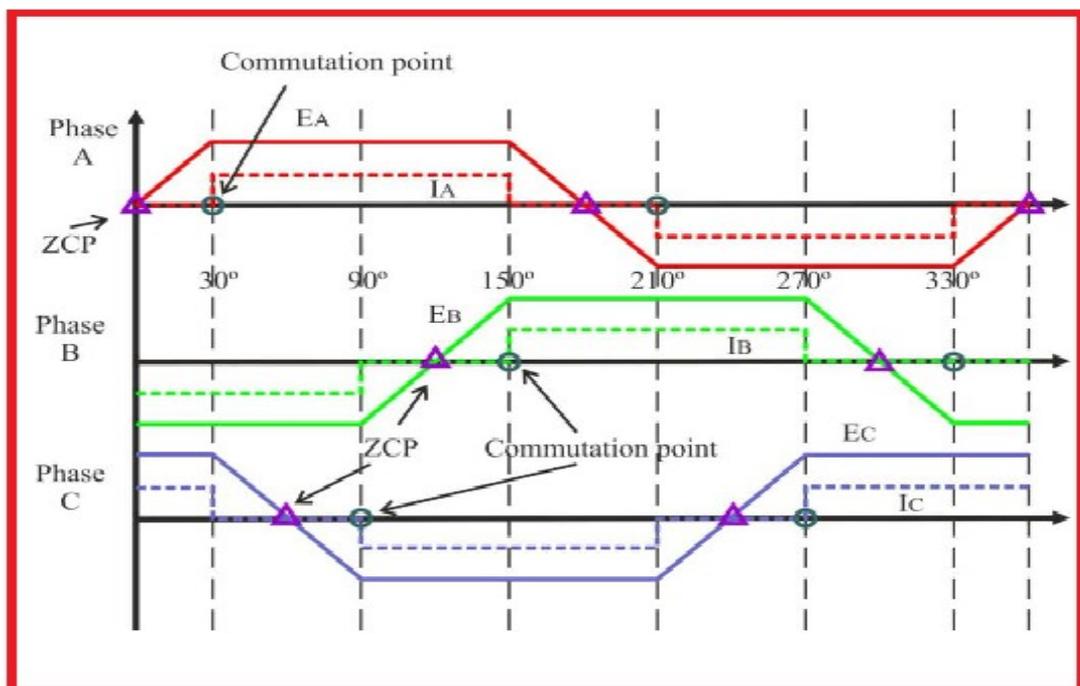


Figure 2.3: Les formes typiques des courants et des FEM .

2.5 L'équation mécanique du mouvement

2.5.1 Vitesse du moteur

Le modèle mécanique du moteur BLDC est donné par l'équation (2.13) où la dynamique de position du rotor $\frac{d\theta_r}{dt}$ est déterminée par son inertie J , son accélération angulaire $\frac{dw_r}{dt}$, le couple électromagnétique développé c_e et le couple dû à la charge c_r . Alors l'équation mécanique est donnée par [1] :

$$J \frac{dw_r}{dt} = c_e + f \cdot w_r + c_r \quad (2-13)$$

Avec :

w_r : La vitesse angulaire [rad · s⁻¹].

c_e : Le couple électromagnétique développé [N.m].

c_r : Le couple de charge [N.m].

f : Le coefficient de frottement ([N.m]/[rad · s⁻¹]).

2.5.2 Couple électromagnétique

Le couple produit dans le BLDC est dû à l'alignement des aimants. L'alignement des aimants se fait à la position où la force électromotrice est nulle. Pour obtenir un couple dans le BLDC il est nécessaire de fournir un courant dans les phases au moment où la force électromotrice est présente. Donc le couple électrique développé par le moteur BLDC est donné par [1] :

$$c_e = \frac{e_a \cdot i_a + e_b \cdot i_b + e_c \cdot i_c}{w_r} \quad (2 - 14)$$

2.6 Modèle du BLDC alimenté par commutateur

Comment on détermine les signaux de commandes des transistors ? :

L'aimant permanent est supposé dans une position quelconque, alors il va créer des Fem dans les phases du stator, cette Fem est supposée trapézoïdale, on sait que le signe de la Fem est imposé par les lois de la physique et dépend essentiellement du sens de rotation du rotor. Donc, on va agir sur les courants pour imposer le régime de fonctionnement, on impose un courant de même sens que la fem de cette phase et de plus au moment où la fem présente sa partie plate (figure .2.4) ce qui assure le

fonctionnement moteur. Pour ce faire on doit commander le transistor qui nous donne le courant voulu. [1]

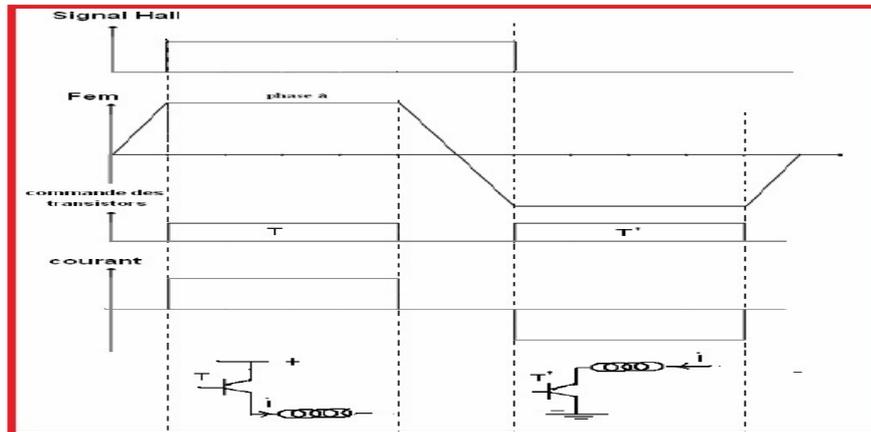


Figure 2.4. Méthode d'élaboration des signaux de commande des transistors.

2.6.1 Signaux des transistors et la séquence des phases alimentées

On peut alors distinguer six intervalles dont chacun peut se diviser en deux sous intervalles selon la conduction ou non de la diode de roue libre correspondante comme le montre la figure (2.5) [1].

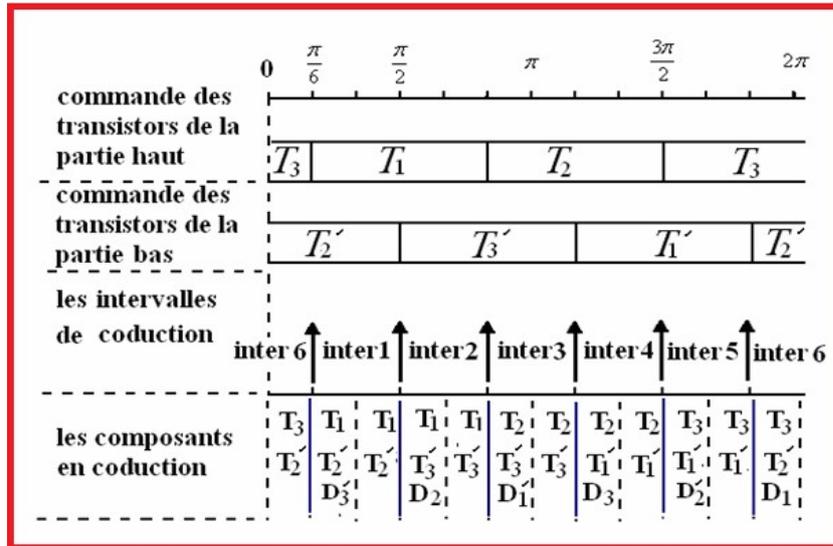


Figure 2.5 : Signaux des transistors pour le sens direct.

La séquence des phases alimentées sur et les états de six pas de commutation sur une période est donnée au tableau (2.1) :

Les intervalles	Phases Excitées	Les états des transistors
$0 - \pi/(6)$	<i>CB</i>	$S = [1\ 0\ 1]$
$\pi/6 - \pi/(2)$	<i>AB</i>	$S = [1\ 0\ 0]$
$\pi/2 - 5\pi/(6)$	<i>AC</i>	$S = [1\ 1\ 0]$
$5\pi/6 - 7\pi/(6)$	<i>BC</i>	$S = [0\ 1\ 0]$
$7\pi/6 - 3\pi/(2)$	<i>BA</i>	$S = [0\ 1\ 1]$
$3\pi/2 - 11\pi/(6)$	<i>CA</i>	$S = [0\ 0\ 1]$
$11\pi/6 - 2\pi$	<i>CB</i>	$S = [1\ 0\ 1]$

Tableau2.1 : Séquence des phases allumées et les états des transistors sur une période.

2.6.2 Expressions des forces électromotrices

Alors les expressions des *Fem* peuvent être déduites de la figure (2.5) et elles sont exprimées comme l'indique le tableau (2.2).

L'angle θ	$Fem(e_a)$	$Fem(e_b)$	$Fem(e_c)$
$0 < \theta \leq \frac{\pi}{6}$	$\frac{6E}{\pi}\theta$	$-E$	E
$\frac{\pi}{6} < \theta \leq \frac{\pi}{2}$	E	$-E$	$-\frac{6E}{\pi}\theta + 2E$
$\frac{\pi}{2} < \theta \leq \frac{5\pi}{6}$	E	$\frac{6E}{\pi}\theta + 4E$	$-E$
$\frac{5\pi}{6} < \theta \leq \frac{7\pi}{6}$	$-\frac{6E}{\pi}\theta + 6E$	E	$-E$
$\frac{7\pi}{6} < \theta \leq \frac{3\pi}{2}$	$-E$	E	$\frac{6E}{\pi}\theta + 8E$
$\frac{3\pi}{2} < \theta \leq \frac{11\pi}{6}$	$-E$	$-\frac{6E}{\pi}\theta + 10E$	E
$\frac{11\pi}{6} < \theta \leq 2\pi$	$\frac{6E}{\pi}\theta - 12E$	$-E$	E

Tableau2.2 : Expressions des FEM des 3 phases (e_a, e_b, e_c) sur une période.

2.6.3 Expressions des courants et les tensions des phases

a les tentions

- Pour chaque phase :

$$v_a = R_{ia} + (L - M) \frac{di_a}{dt} + e_a \quad (2-15)$$

$$v_b = R_{ib} + (L - M) \frac{di_b}{dt} + e_b \quad (2-16)$$

$$v_c = R_{ic} + (L - M) \frac{di_c}{dt} + e_c \quad (2-17)$$

- entre deux phases :

$$v_{ab} = 2R_{ia} + 2(L - M) \frac{di_a}{dt} + e_{ab} \quad (2-18)$$

$$v_{bc} = 2R_{ib} + 2(L - M) \frac{di_b}{dt} + e_{bc} \quad (2-19)$$

$$v_{ca} = 2R_{ic} + 2(L - M) \frac{di_c}{dt} + e_{ca} \quad (2-20)$$

b les courants .

$$\frac{di_a}{dt} = \frac{-R}{(L-M)} i_a + \frac{1}{2(L-M)} (v_{ab} - e_{ab}) \quad (2-21)$$

$$\frac{di_b}{dt} = \frac{-R}{(L-M)} i_b + \frac{1}{2(L-M)} (v_{bc} - e_{bc}) \quad (2-22)$$

$$\frac{di_c}{dt} = \frac{-R}{(L-M)} i_c + \frac{1}{2(L-M)} (v_{ca} - e_{ca}) \quad (2-23)$$

Remarque 1 : Comme le récepteur dans tous les premiers sous intervalles est triphasé équilibré à neutre isolé alors le vecteur $V = [V_a \ V_b \ V_c]$ peut se calculer par la relation suivante [9] :

$$V = M_c S U_d \quad (2-24)$$

$$V = \begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} \quad M_c = \frac{1}{3} \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix} \quad S = \begin{bmatrix} S_a \\ S_b \\ S_c \end{bmatrix} \quad (2-25)$$

S: Désigne les états des bras :

Si la phase j est branchée avec un transistor ou une diode de la partie supérieure, alors $S_j=1$ avec $J = (a,b,c)$

Si la phase j est branchée avec un transistor ou une diode de la partie inférieure, alors $S_j=0$, Pour ce cas, on a le vecteur $s^T [1 \ 0 \ 0]$ puisque la phase 1 est branchée au U_d transistor $T1$ ce qui donne :

$$\frac{di_a}{dt} = \frac{1}{(L-M)} \left[-R_{ia} - \frac{(e_a - e_b)}{2} + \frac{U_d}{2} \right] \quad (2-26)$$

Remarque 2 :

Le coefficient de mutuelle induction M peut être exprimé en fonction du coefficient L .

Quand les conducteurs d'enroulement sont massés dans la même encoche [1]:

$$M = -L/3 \quad (2-27)$$

2.7 Simulation en boucle ouverte

Nous avons établi les équations de fonctionnement de l'ensemble commutateur -Moteur. Un programme de simulation de la dynamique du moteur BLDC est développé sous l'environnement Matlab/Simulink permettant à la fois de

reproduire le couple du moteur, la vitesse, les courants dans les trois phases, les F_{em} induites, ainsi que les signaux de commande des transistors

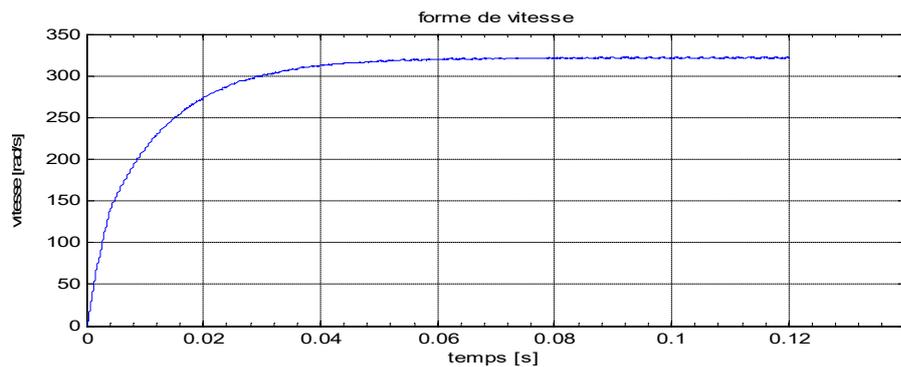
cette simulation comporte deux parties.

- Le régime de démarrage.
- Influence des paramètres.

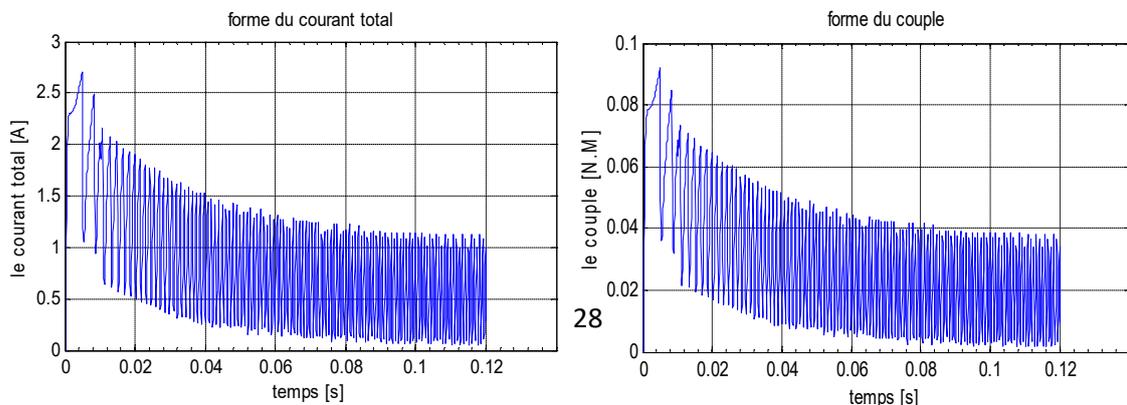
Les simulations sont effectuées avec un couple résistant proportionnelle à la vitesse.

2.7.1 Démarrage du moteur

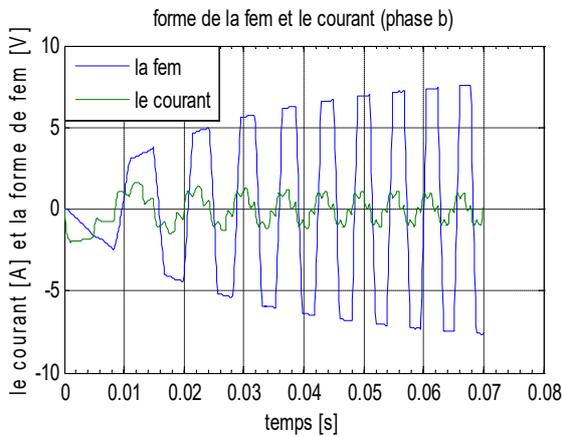
Le moteur étant à l'arrêt, lorsqu'on applique une tension continue a l'entrée du commutateur $U_d = 24V$. Le commutateur est connecté aux enroulements (a, b, c) selon la séquence directe. On obtient la formes de la vitesse du moteur qui démarre de la valeur nulle et se stabilise à 320 Rad/sec selon la figure (2.6.a), il apparaît une faible ondulation due à la présence de la discontinuité au niveau du courant i_d . Pour atténuer son effet il faudrait surtout réduire la durée de cette discontinuité, le courant total i_d figure (2.6.b) Il apparaît que le couple C_e suit la même forme que celle du courant continu i_d selon la figure (2.6.c), le courant de phase(b) et sa fem selon la figure (2.6.d) et la position du rotor θ selon la figure (2.6.e)



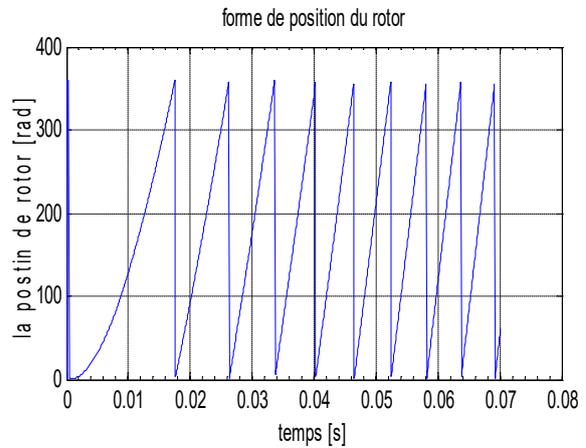
-a- Vitesse



-b- Courant total .



-c- Cuouple.



-d- Courant et Fem de phase (b) .

-e- Position de rotor.

Figure 2.6: Simulation pour le démarrage du moteur.

a Analyses et commentaires

-On remarque que le courant et sa *Fem* sont en phase.

-On remarque que le courant total est de forme continue, mais il présente une discontinuité ce qui donne un couple de même forme, a fin d'avoir un couple lissé, il faut créer les conditions qui atténuent la durée de la discontinuité.

Les ondulations du couple se répercutent directement sur la vitesse qui présente aussi des ondulations.

2.7.2 Influence des paramètres de la machine sur la forme du courant

Pour voir les facteurs influençant sur la forme des courants, on agit sur la résistance ainsi que l'inductance de phase, en examinant par exemple le courant de la phase a, on aura les résultats suivants :

a Influence de résistance

Si n change la valeur de résistance $R=4$ par une valeur $R=12$ (Ohm), On remarque que lorsque la résistance de phase R est grande l'ondulation diminue et la forme du courant se rapproche du courant idéale, ainsi que l'amplitude du courant diminue.

-pour la résistance $R=12\Omega$, l inductance $L=0.0014$ H ;

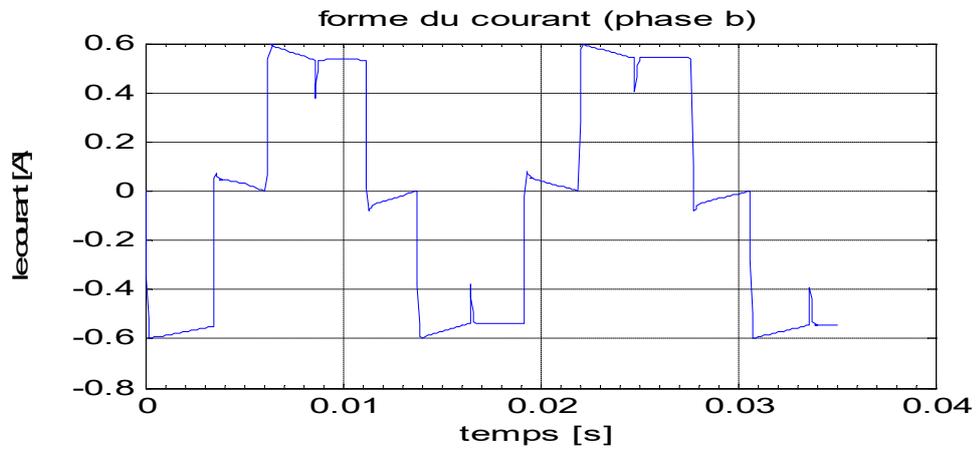


Figure 2.7: Courant de phase (b)

b Influence de l'inductance

Si On change la valeur de l'inductance $L=0.0014$ H par une valeur $L=0.008$ H, On constate que lorsque l'inductance de phase L est grande la forme du courant se déforme et devient ondulatoire et donc la forme du courant de phase s'éloigne de sa forme idéale.

-pour la résistance $R= 4 \Omega$, l inductance $L=0.008$ H ;

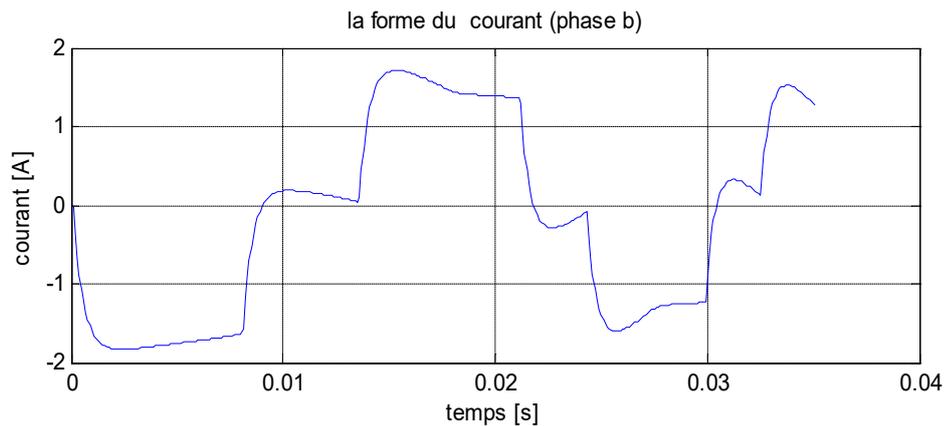


Figure 2.8: Courant de phase (b)

C Analyse et commentaires

On remarque que a l'inverse de la résistance R , plus l'inductance L augmente plus il y'a des ondulations (i.e plus R augmente moins on aura des ondulations).

Ce qui explique le fait que les constricteurs produisent des moteurs de grande résistances et faible inductance.

2.8 Conclusion

Dans ce chapitre nous avons étudié le fonctionnement du moteur *BLDC* en boucle ouverte dans le sens direct. Au début, on a modélisé le commutateur, puis on a établi les équations d'états du moteur *BLDC*. Afin de voir les différents signaux (couple, vitesse, les courants....).

Finalement, On a montré que les paramètres de la machine peuvent influencer sur le comportement dynamique du BLDC.

Chapitre 3 Commande en vitesse par la logique

floue

3.1 Commande en vitesse

3.1.1 Introduction

L'objectif principal de cette partie est de présenter une solution d'asservissement en vitesse de la machine BLDC permettant d'atteindre les performances exigées d'un servo-entraînement. On abordera successivement la modélisation de la machine BLDC où deux enroulements sont toujours alimentés en vue de synthétiser les régulateurs de vitesse, avant d'étudier son alimentation par variateur de courant continu.

Un modèle dynamique de ce dernier sera obtenu. Il sera alors possible de se pencher sur la question de l'asservissement de la vitesse pour lesquels la solution classique de régulation en cascade sera proposée. Pour cette structure la variable principale est la vitesse et la variable auxiliaire est le courant. Pour ce dernier, une stratégie de commande est proposée qui consiste à régler le courant continu à l'entrée du commutateur.

3.1.2 Modèle du BLDC lorsque deux phases sont alimentées

On peut supposer qu'à chaque instant on a seulement deux phases alimentées. En régime dynamique, il faut tenir compte de la mutuelle induction entre les enroulements [1], dans ce cas la tension appliquée aux bornes des phases alimentées (a et b) et le courant i_d qui les traverse sont liés par :

$$u_d = (e_a - e_b) + 2R_{ph} i_d + 2L_{ph} \frac{di_d}{dt} - 2M \frac{di_d}{dt} \quad (3-1)$$

Ou :

$$u_d = (e_a - e_b) + 2R_{ph} i_d + (2L_{ph} - 2M) \frac{di_d}{dt} \quad (3-2)$$

Tel que :

$$e_a = E \quad (3-3-a)$$

$$e_b = -E \quad (3-3-b)$$

Où

u_d : est la tension à l'entrée du commutateur laquelle est appliquée aux bornes des phases (a) et (b) mise en série.

i_d : est le courant à l'entrée du commutateur qui passe à travers les deux phases (a) et (b) mise en série.

L_{ph} et R_{ph} sont respectivement l'inductance et la résistance de phase.

E : est la valeur crête de la Fem induite dans chaque phase, dont l'expression est proportionnelle à la vitesse :

$$E = k_e w_m \quad (3-4)$$

Donc le modèle devient :

$$u_d = 2E + 2R_{ph} i_d + (2L_{ph} - 2M) \frac{di_d}{dt} \quad (3-5)$$

on pose :

$$\begin{aligned} R &= 2R_{ph} \\ L &= 2(L_{ph} - 2M) \\ E_p &= 2E \end{aligned}$$

Alors le modèle devient :

$$u_d = E_p + R i_d + L \frac{di_d}{dt} \quad (3-6)$$

avec :

$$E_p = 2E = 2k_e w_m = k_v w_m \quad (3-7)$$

L'équation (3.6) est identique à celle d'une machine à courant continu classique.

Alors l'expression du couple est :

$$C_e = \frac{(E_a - E_b)}{w_m} i_d \quad (3-8)$$

$$C_e = \frac{2.k_e .w_m}{w_m} i_d \quad (3-9)$$

Alors :

$$C_e = k_v . i_d \quad (3-10)$$

3.1.3 Stratégie d'asservissement du courant total

On a vu que la machine BLDC associée à son commutateur de puissance et sa logique de commande, a un comportement analogue à celui d'une machine à courant continu classique. On peut donc la commander en courant en faisant varier la tension d'alimentation continue appliquée au commutateur, pour avoir un rendement correct on utilise en général un hacheur. Pour avoir le fonctionnement dans les deux quadrants pour chaque un des deux sens on doit utiliser un hacheur réversible en courant. La tension totale à l'entrée du commutateur étant commandée par un

régulateur flu, la sortie de ce dernier est la tension de commande u_c qui est comparée avec une porteuse de type dents de scie pour générer les signaux d'excitation des transistors du hacheur [2]. Ce schéma de commande est représenté à la figure (3.1) où les différentes parties du schéma seront détaillées par la suite.

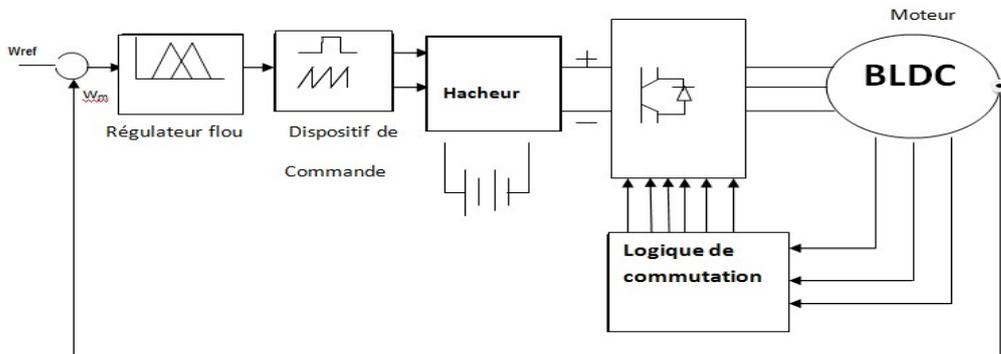


Figure 3.1 : Schéma synoptique de la commande.

a Principe du hacheur

L'étage de puissance utilisé est nommé hacheur car son rôle est de découper une tension [2]

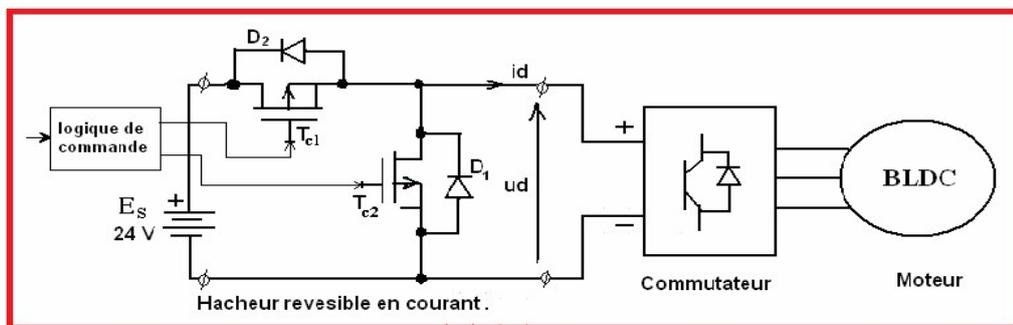


Figure 3.2 : Hacheur réversible en courant.

On dispose d'une tension continue fixe E_s (issue d'un redresseur branché sur le secteur ou d'une batterie), deux transistors T_{c1}, T_{c2} , deux diodes D_1, D_2 et un circuit électronique (dispositif de contrôle), qui sera précisé par la suite, impose des basculements périodiques aux transistors. Deux modes de fonctionnement pour ce hacheur peuvent être décrits :

Le mode normal «moteur » que soit dans le sens direct ou dans le sens inverse quand la vitesse et le couple sont de même signe, le transistor T_{c1} est commandé périodiquement avec un rapport cyclique α_1 . Il est facile de montrer que T_{c2} (n'est jamais fermé) est bloqué par la conduction de D1 et la diode D2 est bloquée par la conduction de T_{c1} . Le courant (i_d) est positif à chaque instant (dans la mesure où la conduction est continue et qu'il ne s'annule pas). Avec la même hypothèse, la tension moyenne aux bornes du commutateur est positive et vaut $U_d = \alpha_1 \cdot E_s$.

Le mode freinage, que soit dans le sens direct ou dans le sens inverse quand la vitesse et le couple sont de signe opposé c'est T_{c2} qui est commandé périodiquement avec le rapport cyclique α_1 . T_{c2} Et D1 ne participent pas au fonctionnement du montage. Le courant (i_d) est négatif et la tension moyenne aux bornes du commutateur est positive et Vaut $U_d = (1 - \alpha_2) \cdot E_s$. [1]

b Commande en modulation de largeur d'impulsion

On a vu que la vitesse du moteur est modifiée par le rapport cyclique du signal de commande des commutations, la fréquence étant fixe. Il faut donc créer un circuit électronique délivrant une tension rectangulaire adaptée à l'attaque du transistor, de fréquence donnée et de rapport cyclique commandé par la tension. On parle de modulation de largeur d'impulsions (MLI ou PWM pour *pulse width modulation* en anglais). Le schéma de principe (figure 3.3) comporte un générateur de signaux délivrant une tension u_T en dents de scie, de fréquence fixe f_d qu'on compare au signal de commande u_c (figure 3.4).

À l'échelle des temps utilisée, u_c n'a pas le temps de varier. On obtient alors en sortie un signal rectangulaire de fréquence f_d et de rapport cyclique α_1 (alors que $\alpha_1 = (1 - \alpha_2)$).

L'équation de u_t dans l'intervalle $[0, T]$ est [2] :

$$u_t = \frac{2.u_h}{T} \cdot T - u_h \quad (3.11)$$

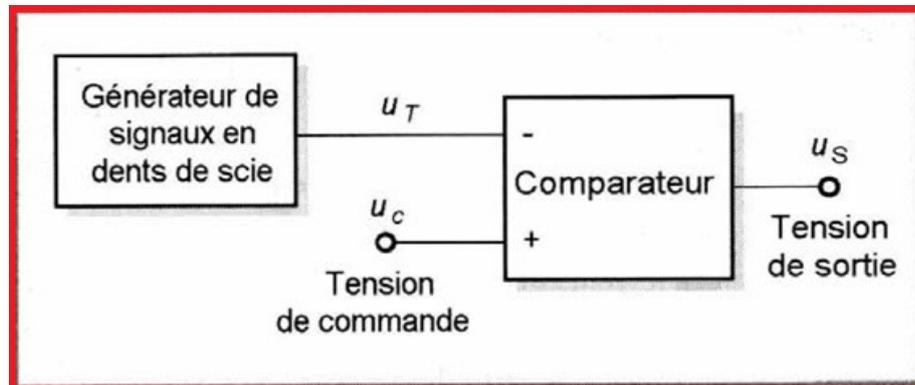


Figure 3.3 : Principe d'un modulateur de largeur d'impulsions.

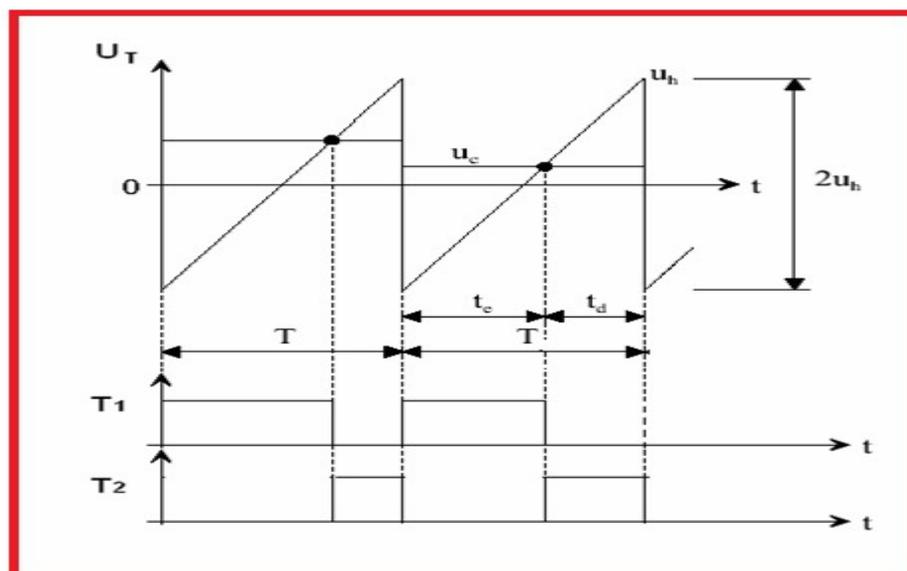


Figure 3.4 : Impulsions d'allumage de T_{c1} , T_{c2} obtenues par une MLI.

Le basculement de u_s se produit à l'instant $\alpha_1 T$ où u_T atteint la valeur u_c

$$u_c = \frac{2.u_h}{T} \cdot (\alpha_1 T) - u_h \quad (3.12)$$

On en déduit :

$$\alpha_1 = \frac{u_c + u_h}{2.T} \quad (3.13)$$

Si la tension u_c varie théoriquement de 0 à u_h , le rapport cyclique α_1 prend des valeurs entre 0 et 1. Le circuit étudié est facilement réalisable avec des composants ordinaires (amplificateurs opérationnels, résistances et condensateurs), mais en pratique, il est intégré dans les circuits spécifiques à la commande des moteurs BLDC [2].

3.2 Réglage de la vitesse par la logique floue

3.2.1 Introduction

Les soucis d'une vaste utilisation de l'intelligence humaine a poussé les chercheurs au développement des algorithmes flous, en raison de leur simplicité d'implémentation, de mise au point et de la non nécessité de modélisation mathématique du processus à commander, ils sont devenus de plus en plus appliqués dans plusieurs domaines tel que le domaine industriel, médical et des transports et ce quelle que soient les complexités des systèmes.

En commande floue, la plupart des algorithmes utilisés sont basés sur le schéma proposé par Mamdani. La majorité de ses algorithmes ont été appliqués surtout sur des processus industriels plus au moins lents tels que la régulation de température d'une chaudière.

3.2.2 Historique

Les prémisses de la logique floue sont apparues avant les années 1940, avec les premières approches, par des chercheurs Américains, du concept d'incertitude. Il a fallu attendre 1965, pour que le concept de sous-ensemble flou (*fuzzy* en Anglais) soit proposé par L.A.Zadeh, automaticien de réputation internationale, professeur à l'université de Berkeley en Californie, qui a contribué à la modélisation des phénomènes sous forme floue, en vue de palier les limitations dus aux incertitudes des modèles classiques à équations différentielles. en 1974, M. Mamdani expérimentait la théorie énoncée par Zadeh sur une chaudière à vapeur, introduisant ainsi la commande floue dans la régulation d'un processus industriel [11].

Grâce au chercheur Japonais M. Sugeno, la logique floue était introduire au Japon dès 1985 [11], les sociétés Japonaises commencent à utiliser la logique floue dans des produits industriels pour résoudre des problèmes de réglage et de commande [12].

3.2.3 Généralité sur la logique floue

a Variables linguistiques et ensemble flous

La description d'une certaine situation, d'un phénomène ou d'un procédé contient en général des expressions floues (quelque, souvent, chaud, froid, rapide, lent...). Les expressions de ce genre forment les valeurs des variables linguistiques de la logique floue. Afin de permettre un traitement numérique, il est indispensable de les soumettre à une définition à l'aide de fonctions d'appartenance [12].

b Les fonctions d'appartenance

Chaque sous-ensemble flou peut être représenté par sa fonction d'appartenance. Si l'ensemble de référence est un ensemble fini, les valeurs d'appartenance sont des valeurs discrètes définies dans $[0,1]$. Si l'ensemble est infini, on peut représenter ces valeurs comme des fonctions d'appartenance continues. La

fonction d'appartenance et notée par $u_a(x)$, l'argument x se rattache à la variable linguistique et l'indice A désigne l'ensemble concerné [13].

➤ **Fonction triangulaire :**

Elle est définie par trois paramètres $\{a, b, c\}$ qui déterminent les ordonnées des trois sommets.

$$\mu(x) = \max\left(\min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0\right) \quad (3-14)$$

➤ **Fonction trapézoïdale :**

Elle est définie par quatre paramètres $\{a, b, c, d\}$:

$$u(x) = \max\left(\min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right) \quad (3-15)$$

➤ **Fonction sigmoïdale :**

Elle est définie par deux paramètres $\{a, c\}$:

$$\mu(x) = \frac{1}{1 + \exp(-a(x-c))} \quad (3-16)$$

➤ **Fonction gaussienne :**

Elle est définie par deux paramètres $\{m, \sigma\}$

$$u(x) = \exp\left(-\frac{(x-m)^2}{2\sigma^2}\right) \quad (3-17)$$

Elle peut être monotone, triangulaire, trapézoïdale, ou en forme de cloche comme la présentation de la figure (3.5) [17].

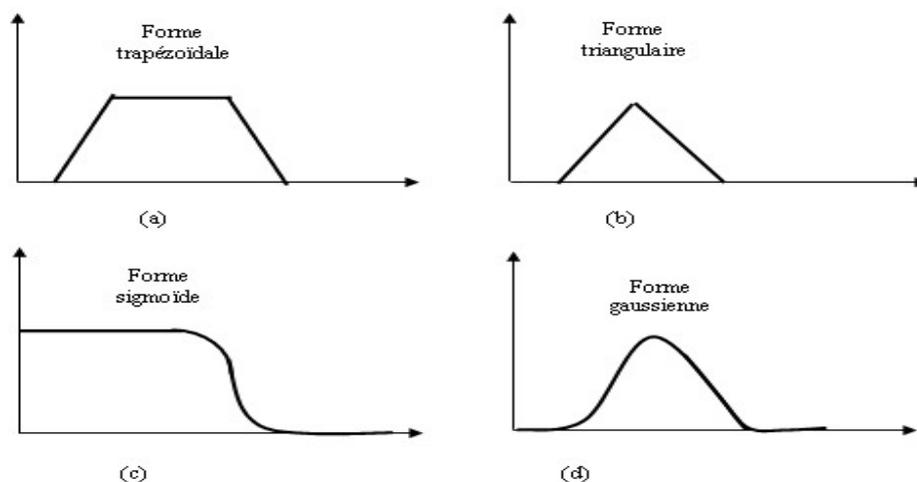


Figure 3.5 : Fonction d'appartenance.

c Univers de discours et classes d'appartenances

Un des premiers pas dans la conception d'une application floue est de définir l'ensemble de référence ou univers de discours pour chaque variable linguistique. L'ensemble des prédicats peut être caractérisé par un ensemble de sous-ensembles flous. Chaque sous ensemble flou dans l'univers de discours représente une valeur linguistique [13].

3.2.4 Les opérateurs de la logique floue

Comme pour les ensembles classiques, On peut définir différentes opérations sur les ensembles flous. Soient A et B deux ensembles flous dans l'univers U . pour $x \in U$ on a : [17]

- Complémentation : $U_{\bar{A}}(x) = 1 - U_A(x)$
- Intersection : $U_{A \cap B} = \min(U_A(x), U_B(x))$
- Union : $U_{A \cup B} = \max(U_A(x), U_B(x))$

3.2.5 Les systèmes flous

La structure conventionnelle d'un contrôleur flou est représentée par la figure (3.6). Il est composé de quatre blocs [16].

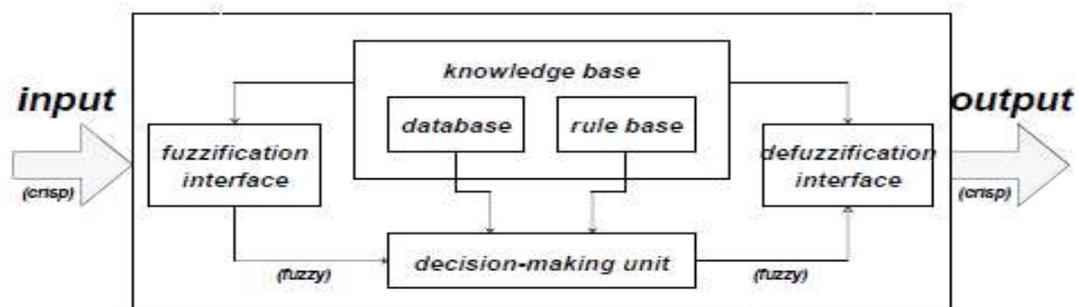


Figure 3.6: structure d'un contrôleur flou.

- ✓ Bloc de règles ou base de connaissances (règles et paramètres des fonctions D'appartenances).
- ✓ Bloc de décision ou Moteur d'inférence (inférence des opérations sur les règles).
- ✓ Bloc de fuzzification (transformation des entrées précises en degrés d'appartenance)
- ✓ Bloc de defuzzification (transforme les résultats flous en sorties précise).

a Fuzzification

La fuzzification consiste à attribuer à la valeur réelle de chaque entrée, au temps t , sa fonction d'appartenance à chacune des classes préalablement définies, donc à transformer l'entrée réelle en un sous-ensemble flou [11].

b Les règles floues

L'idée principale des systèmes basés sur la logique floue est d'exprimer la connaissance humaine sous forme de règles linguistiques de forme Si...alors...Chaque règle à deux parties :

- Partie antécédente (prémisse ou condition), exprimée par Si...
- Partie conséquence (conclusion), exprimée par Alors...

La partie antécédente est la description de l'état du système. La partie conséquence exprime l'action que l'opérateur qui contrôle le système doit exécuter [13].

➤ **Les règles de Mamdani :**

La forme typique d'une règle de Mamdani s'écrit :

Si (x_1 est E_1) et (x_2 est E_2) ... et (x_n est E_n) **Alors** (u_1 est C_1) et (u_2 est C_2) et...et (u_m est C_m)

Ou :

$x_1 \dots x_m$: Les variables d'entrées.

$u_1 \dots u_n$: Les variables de sorties.

$E_1 \dots E_n$ est $C_1 \dots C_m$: Ensembles flous.

➤ **Les règles de Takagi Sugeno :**

Les règles de Takagi Sugeno sont similaires à celles de Mamdani, elles diffèrent dans la deuxième partie (conséquence). La forme typique de ces règles s'écrit :

Si (x_1 est E_1) et (x_2 est E_2) et... et (x_n est E_n) **Alors** $u_1 = f(x_1 \dots x_n)$, $u_2 = f(x_1 \dots x_n)$... $u_m = f(x_1 \dots x_m)$.

Les fonctions ($f_1, f_2 \dots f_m$) sont des fonctions réelles, elles peuvent être linéaires ou non linéaires.

c Inférences

Les inférences lient les grandeurs mesurées et les variables de sorties par des règles linguistiques. Ces règles sont combinées en utilisant les connexions (et) et (ou) [13]. Cette partie décisionnelle où le contrôleur décide l'action similaire à celle d'un opérateur humain, elle présente et englobe toutes les connaissances de l'expert humain sur la dynamique, et les caractéristiques du système.

La description des inférences peut se faire par des expressions linguistiques, Symboliques, par tableau ou matrice d'inférence [12]. Soit un régulateur flou dont l'entrée et la dérivée de l'erreur, avec une sortie de Commande u . La notion de méthode d'inférence détermine la réalisation des différents opérateurs dans une inférence, pour le réglage par logique floue, on utilise en générale une des méthodes suivante :

- Méthode d'inférence max-min (méthode de Mamdani).
- Méthode d'inférence max-produit (méthode de Larsen).
- Méthode de Sugeno.

d Défuzzification

La défuzzification permet de passer d'un degré d'appartenance d'une commande à la détermination de la valeur à donner à cette commande. Il existe plusieurs méthodes de défuzzification, dont la plus utilisée est celle du centre de gravité [11].

➤ **Méthode du maximum**

Cette stratégie génère une commande qui représente la valeur maximale de la fonction d'appartenance résultante issue de l'inférence. Cependant, cette méthode présente un certain inconvénient lorsqu'il existe plusieurs valeurs pour lesquelles la fonction d'appartenance résultante est maximale.

➤ **Méthode de la moyenne des maxima**

Cette méthode génère une commande qui représente la valeur moyenne de toutes les valeurs pour lesquelles la fonction d'appartenance résultante est maximale.

➤ **Méthode du centre de gravité**

Cette méthode est la plus utilisée dans les contrôleurs flous, elle génère le centre de gravité de la fonction d'appartenance issue de l'inférence.

L'intégrale au dénominateur donne la surface, tandis que l'intégrale au numérateur correspond au moment de la surface.

$$y = \frac{\int yu(y)dy}{\int u(y)dy} \quad (3-18)$$

➤ **Méthode de la somme pondérée**

Il s'agit d'un compromis entre les deux méthodes précédentes. On calcule individuellement les sorties relatives à chaque règle selon le principe de la moyenne des maxima, puis on réalise leur moyenne pondérée [14].

3.2.6 Avantages et désavantages du réglage par logique floue

Les avantages principaux des régulateurs flous sont les suivants [13]:

- Il n'est pas nécessaire de faire un modèle mathématique du système à régler.
- On peut implémenter des connaissances de l'opérateur du processus.
- On peut maîtriser les systèmes non linéaires sont difficile à modéliser.
- Grâce à des processus dédiés, On peut régler des processus rapides.

Les désavantages sont les suivants :

- Manque de directives précises pour la conception d'un régulateur.
- Approche artisanale et non systématique (il est souvent très difficile d'implémenter les connaissances de l'opérateur).
- Impossibilité de démontrer la stabilité du circuit de réglage en toute généralité (en absence d'un modèle valable).
- Précision du réglage en général peu élevée.
- Cohérence des inférences non garantie à priori (possible apparition de règles d'inférence contradictoire).

3.2.7 Réglage de la vitesse d'un moteur BLDC par un régulateur flou à 49 règles

a Description du régulateur flou

La majorité des contrôleurs développés utilisent le schéma simple proposé par Mamdani pour les systèmes mono-entrée / mono-sortie. Ce schéma est représenté comme la suit :

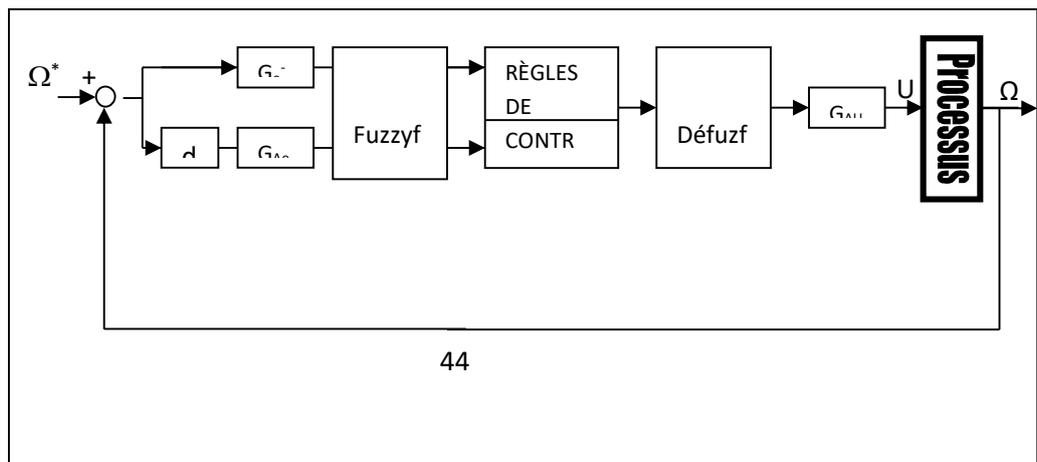


Figure 3.7 : Schéma d'un contrôleur flou typique.

D'après le schéma, le contrôleur est composé :

- D'un bloc de calcul de la variation de l'erreur au cours du temps (Δe).
- Des facteurs d'échelle (normalisation) associés à l'erreur, sa variation et le gain associé à la variation de la commande ($G_{\Delta u}$).
- D'un bloc de fuzzification de l'erreur et de sa variation.
- Des règles de contrôle flou. Ainsi qu'un bloc d'évaluation des règles du contrôle flou (inférence).
- D'un bloc de défuzzification utilisé pour convertir la variation de la commande floue en valeur numérique.
- G_e , $G_{\Delta e}$ et $G_{\Delta u}$ représentent les gains d'adaptation et de normalisation, ils jouent un rôle extrêmement important. En effet, ce sont ces derniers qui fixeront les performances de la commande.

b Loi de commande

Cette loi est fonction de l'erreur et de sa variation telle que $U = f(e, \Delta e)$. Par conséquent, l'activation de l'ensemble des règles de décision associées donne la variation de la commande Δu nécessaire, permettant ainsi l'ajustement d'une telle commande. Dans les cas simples, cette variation de la commande est obtenue par une simple lecture d'une table de décision définie hors ligne [18].

L'erreur est calculée de manière suivante [9]:

$$e[n] = \omega_{ref}[n] - \omega_m[n] \quad (3.19)$$

Et la dérivée de L'erreur est calculée de manière suivant :

$$de[n] = e[n] - e[n - 1] \quad (3.20)$$

tel que :

ω_{ref} : La vitesse référentielle.

ω_m : La vitesse de rotor.

La forme la plus générale de cette loi de commande est :

$$U_{K+1} = U_K + G_{\Delta U} \cdot \Delta U_{K+1} \quad (3.21)$$

Où :

- $G_{\Delta U}$:

Gain associé à la commande U_{K+1} en général choisi pour assurer la stabilité du système.

- ΔU_{K+1} : variation de la commande.

c synthèse du régulateur flou a 49 règles

L'intervalle d'intérêt de chaque variable d'entrée et sortie est subdivisé en sept (7) classes, comme c'est montré à les figure (3.7) pour l'erreur, la figure (3.8) pour la dérive de l'erreur finalement la sortie (u) sur la figure (3.9)

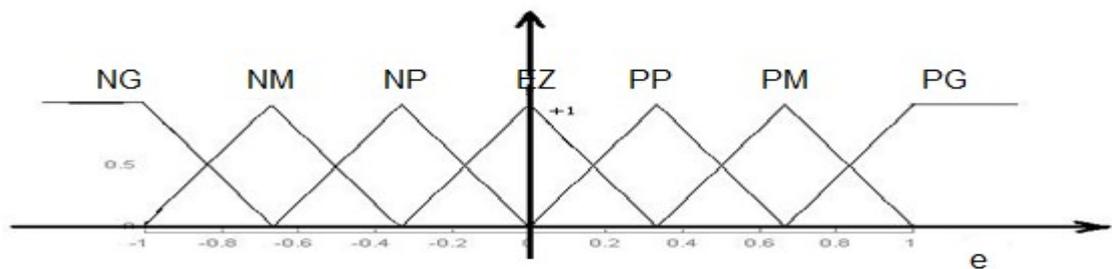


Figure 3.8 : L'erreur

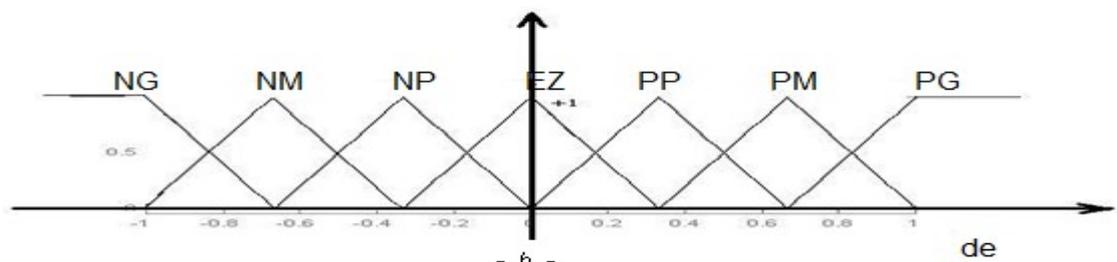


Figure 3.9 : La dérive de l' erreur

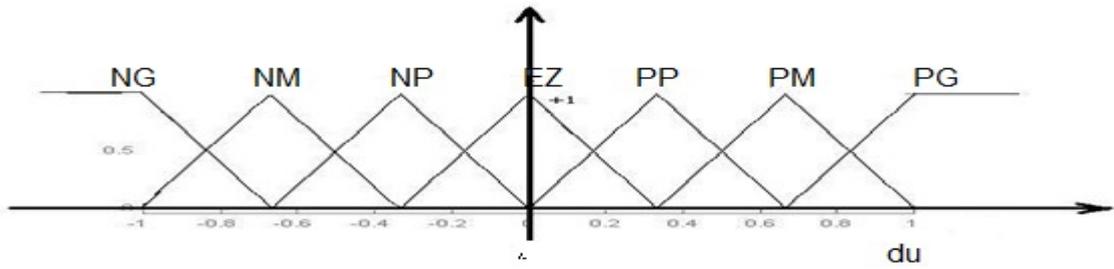


Figure 3.10 : La sortie (u)

Les 49 règles se trouvent au niveau du tableau suivant:

<i>Entre (de)</i>							
<i>Entre (e)</i>	<i>NG</i>	<i>NM</i>	<i>NP</i>	<i>EZ</i>	<i>PP</i>	<i>PM</i>	<i>PG</i>
<i>NG</i>	<i>PG</i>	<i>PG</i>	<i>PM</i>	<i>PM</i>	<i>PP</i>	<i>PP</i>	<i>EZ</i>
<i>NM</i>	<i>PG</i>	<i>PM</i>	<i>PM</i>	<i>PP</i>	<i>PP</i>	<i>EZ</i>	<i>NP</i>
<i>NP</i>	<i>PM</i>	<i>PM</i>	<i>PP</i>	<i>PP</i>	<i>EZ</i>	<i>NP</i>	<i>NP</i>
<i>EZ</i>	<i>PM</i>	<i>PP</i>	<i>PP</i>	<i>EZ</i>	<i>NP</i>	<i>NP</i>	<i>NM</i>
<i>PP</i>	<i>PP</i>	<i>PP</i>	<i>EZ</i>	<i>NP</i>	<i>NP</i>	<i>NP</i>	<i>NM</i>
<i>PM</i>	<i>PP</i>	<i>EZ</i>	<i>NP</i>	<i>NP</i>	<i>NM</i>	<i>NM</i>	<i>NG</i>
<i>PG</i>	<i>EZ</i>	<i>NP</i>	<i>NP</i>	<i>NM</i>	<i>NM</i>	<i>NG</i>	<i>NG</i>

Tableau 3.1 : Tableau des règles (49) de décision floue (inférences)

Les classes sont notées comme suite :

- NG* : négatif grand , *PP* : positif petit
- NM* : négatif moyen , *PM* : positif moyen
- NP* : négatif petit , *PG* : positif grand
- EZ* : enivrent zéros

Les règles de commandes s'écrivent comme suit :

- Si (e est NG) et (de est NG) alors Δu est PG Ou*
- Si (e est NG) et (de est NM) alors Δu est PG Ou*

Si (e est NG) et (de est NP) alors Δu est PM Ou

Si (e est NG) et (de est EZ) alors Δu est PM Ou

-
-
-
-

Ou *Si (e est PG) et (de est PG) alors Δu est NG*

Pour l'implication, on a utilisé la méthode de *Mamdani*. La défuzzification se fait par relation dite *du centre de gravité*.

3.2.8 Simulation et résultats obtenus

Nous avons appliqué un réglage de vitesse du moteur à courant continu sans balais (BLDC) associée à un contrôleur flou.

Afin d'étudier le comportement de notre système pour différents régimes de fonctionnement, trois simulations ont été réalisées.

La simulation contient trois parties :

- Simulation pour le sens direct de la vitesse.
- Avec inversion de sens de rotation.
- Test de robustesse du régulateur de la vitesse.

a Simulation pour le sens direct de la vitesse

La figure (3.11), présente la simulation pour le sens de rotation direct, la figure (3.11.a) présente l'évolution de la vitesse, nous remarquons que la vitesse suit parfaitement sa consigne avec temps de réponse $t=0.05$ s.

En suite, la figure (3.11.b) présente le couple, le courant total selon la figure(3.11.c), le courant d'une phase avec sa FEM correspondante soit par exemple la phase **a** (figure .3.11.d) et la position du rotor (figure .3.11.e) en fonction de temps.

-L'erreur de réglage :

Dans le programme on utilise l'erreur donnée par la relation suivante :

$$Erreur = w_{ref} - w_m \quad (3.22)$$

-Le temps de simulation :

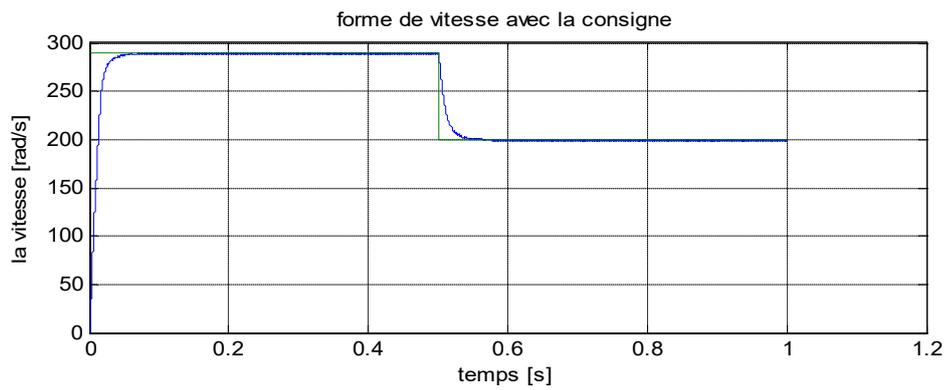
$T_s=1$ s.

La consigne de la vitesse positive:

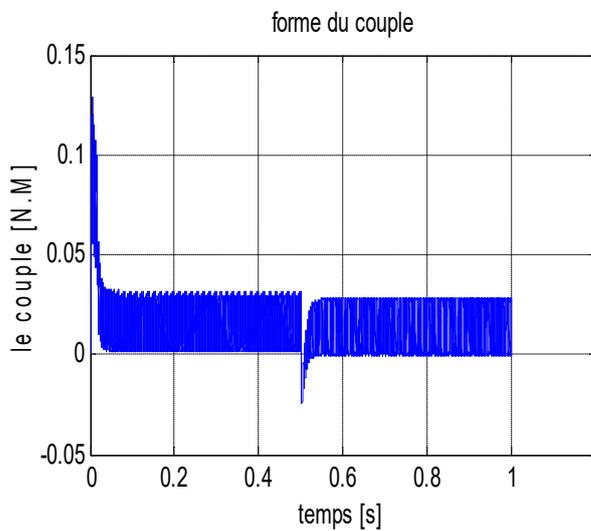
Si $t < 0.5$ $w_{ref} = 290$ Rad/s.

Si $(t > 0.5)$ $w_{ref} = 200$ Rad/s.

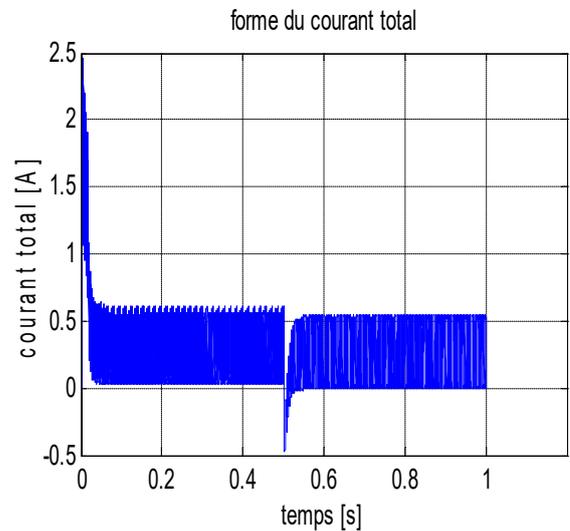
-le couple résistant : $C_r = 0.02$ [n.M].



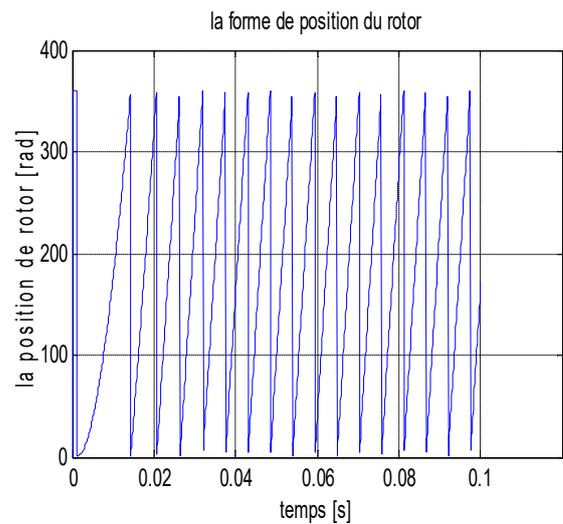
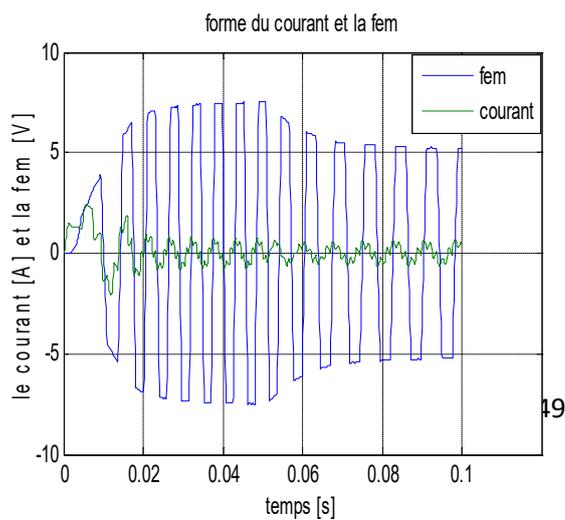
-a- Forme de la vitesse



-b- Forme de du couple C_e



-c- Forme du courant total



-d-Courant i_a et sa Fem_a

-e- Position du rotor

Figure 3.11 : Résultats pour le fonctionnement du moteur pour la vitesse positive.

b Simulation pour les deux sens de rotation

La figure (3.12), présente la simulation pour les deux sens de rotation, la figure (3.12.a) présente l'évolution de la vitesse, nous remarquons que la vitesse suit parfaitement sa consigne avec temps de réponse $t=0.05$ s.

En suite, la figure (3.12.b) présente le courant total, le couple selon la figure(3.12.c) et le courant d'une phase avec sa FEM correspondante soit par exemple la phase a (figure .3.12.d) en fonction de temps .

-L'erreur de réglage :

Dans le programme on utilise l'erreur donnée par la relation suivante :

$$Erreur = |w_{ref}| - |w_m| \quad (3.23)$$

-Le temps de simulation :

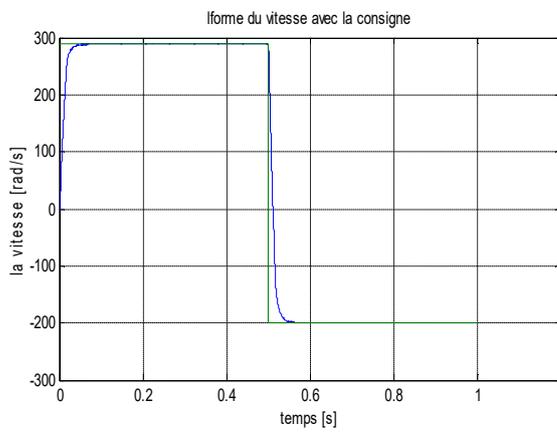
$$T_s = 1 \text{ s.}$$

La consigne de la vitesse positive:

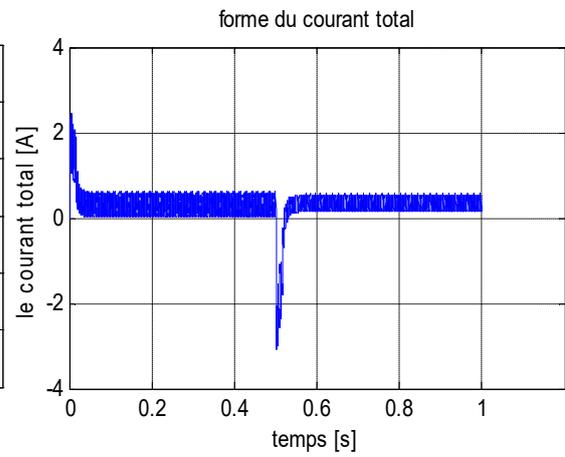
$$\text{Si } t < 0.5 \quad w_{ref} = 290 \text{ Rad/s.}$$

$$\text{Si } (t > 0.5) \quad w_{ref} = -200 \text{ Rad/s.}$$

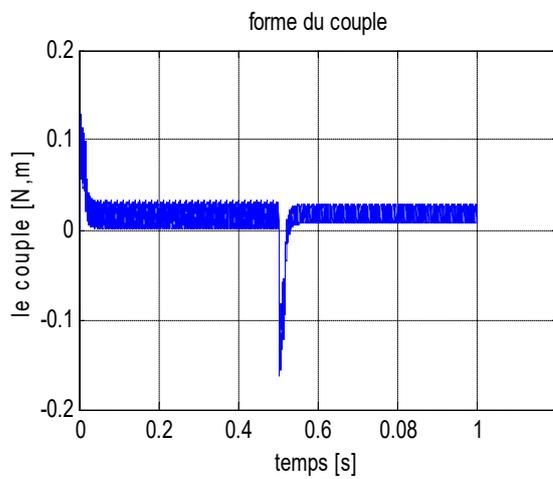
-le couple résistant : $Cr = 0.02$ [n.M].



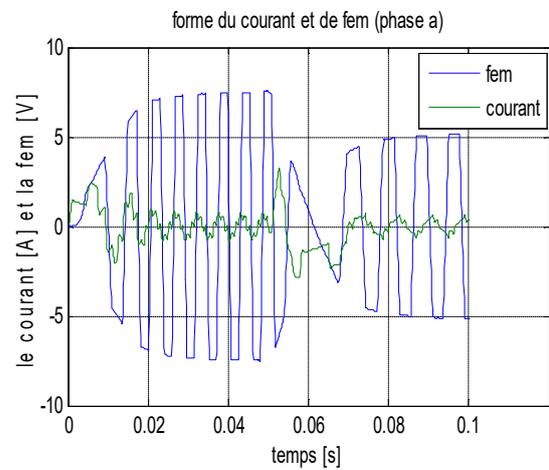
-a- La vitesse et sa consigne



-b- la forme de courant total (i_d)



-c- Forme de couple



-d- le courant i_a et la fem_a

Figure 3.12 : Résultats pour le fonctionnement dans les deux sens de rotation.

c analyse et commentaire

- On remarque que le couple est approximativement constant pour une vitesse constante dans le régime permanent avec la présence des ondulations.
- On remarque qu'il y a une amélioration dans le temps de réponse du système réglé par FLC ceci explique que ce régulateur est efficace.
- On remarque qu'une partie du courant total négatif pendant la durée de freinage du moteur, ceci est due au retour du courant vers la source qui explique que la machine passe en fonctionnement génératrice.
- On remarque que l'inversion de signe entre le courant de phase et f_{em} ceci explique l'inversion de la puissance.
-

d Test de la robustesse du régulateur de la vitesse

Dans cette partie, on démarre le moteur avec un couple résistant nul, On remarque que la vitesse converge parfaitement vers la consigne désirée, L'orsqu on applique un couple résistant $c_r = 0.02$ [N.M] à l' instant $t = 0.5$, On observe une légère détérioration d environ (0.83%) par rapport la consigne désirée, cette détérioration est vite rattrapée avec temps de réponse moins de $t = 0.05$ [s].

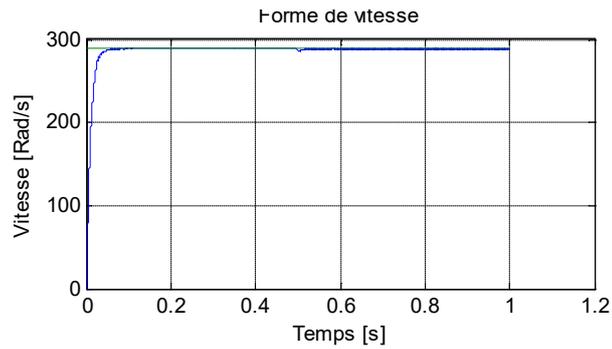


Figure 3.13: Forme de vitesse ($c_r = 0.02$)

Si on applique un couple résistant $c_r = 0.04$ [N.M] à l' instant $t = 0.5$, On observe une légère détérioration d environ (1.33%) par rapport la consigne désirée, cette détérioration est vite rattrapée avec temps de réponse moins de $t = 0.05$ [s]. (figure.3.14).

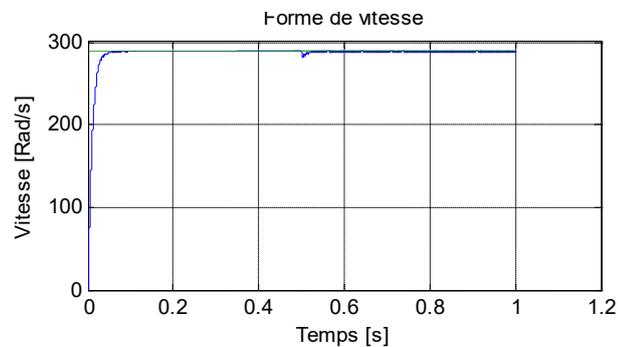


Figure 3.14: Forme de vitesse ($c_r = 0.04$).

Si on augmente la valeur de couple résistant $c_r = 0.04$ [N.M] à l' instant $t = 0.5$, On observe une légère détérioration d environ (2%) par rapport la consigne désirée, cette détérioration est vite rattrapée avec temps de réponse moins de $t = 0.05$ [s]. (figure.3.15).

Ceci explique que la réaction du régulateur est efficace.

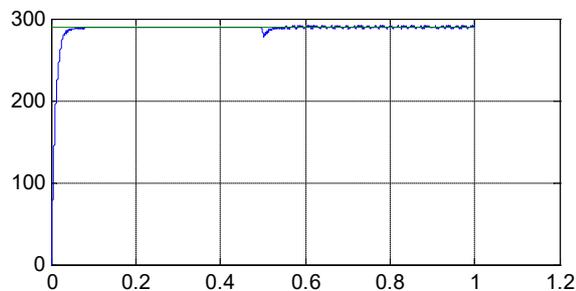


Figure 3.15 : Forme de vitesse ($c_r = 0.06$).

3.2.9 Conclusion

Dans ce chapitre nous avons étudié la commande en vitesse d'un moteur BLDC, On a utilisé un commande linéaire dite cascade, Le réglage du courant totale est réalisé par la commande par modulation de largeur d'impulsion noter qu'il existe d'autres méthodes tel que la commande par hystérésis avec MLI par phase, mais elle nécessite au moins deux capteur de courant. Ainsi nous avons donné des généralités sur la logique floue qui est permet de comprendre que ce dernier n'est pas basé sur des équations physique du système à contrôler mais sur le savoir-faire des opérateurs et des experts du domaine concerné.

Finalement On a montré les performances du réglage par la logique floue dans ce système.

Chapitre 4 Technologie des circuits FPGA

4.1 Introduction

L'histoire des circuits intégrés a commencé après l'invention du transistor en 1947 par les laboratoires *Bell*. Relativement, l'apparition du premier circuit intégré est au cours des années 1958 et 1959 grâce à un jeune ingénieur du nom de *Jack Kilby* de la firme *Texas Instruments* qui a intégré sur un même substrat de silicium plusieurs éléments électroniques (transistors, résistances, capacités) ce qui est qualifié à cette époque le premier circuit intégré.

Juste après, dans les années 60, et plus précisément en 1965, un des fondateurs de la Compagnie *Intel* nommé *Gordon Moore* a fait une étude concernant l'évolution du secteur des circuits intégrés, ce qui a lui permis de prédire que le taux d'intégration des transistors dans ces circuits double tous les deux ans. A nos jours, ce constat reste vérifié sur les circuits *FPGA* qui n'ont pas échappé à cette loi. Jusqu'au

début des années 80 et même à une époque plus récente, la conception d'un système sur puce (SOC) n'était accessible qu'aux firmes et sociétés spécialisées à cause de la complexité des circuits et des fonctions à intégrer qui demandent divers efforts et compétences. Par conséquent des coûts élevés et cette technologie est inaccessible au grand public. [19]

Aujourd'hui, l'avènement des dernières générations d'*FPGA* a permis de mettre la technologie *SOC* à la portée d'un public nettement plus large. Ceci est particulièrement depuis que les *FPGA* sont proposés à un prix très faible et raisonnable. Ce prodigieux essor a été rendu possible grâce aux progrès concernant les technologies de fabrication des transistors et les méthodes de conception assistée par ordinateur (*CAO*). Le rôle des *FPGA* est d'intégrer des circuits logiques complexes. Ces circuits sont susceptibles d'être reconfigurés (Architecture programmée modifiable) partiellement ou entièrement suivant l'application. A cet effet, ce premier chapitre s'inscrit dans un contexte qui traite les *FPGA* et leurs positions au sein des autres systèmes digitaux. Nous allons ensuite décrire l'apport de cette technologie Programmable sur la commande des moteurs.

4.2 Les circuits logiques programmables

De nombreuses familles de circuits programmables et reprogrammables sont apparues depuis les années 70 avec des noms très divers suivant les constructeurs.

La figure 4.1 donne une classification possible des circuits numériques en précisant où se situe chaque type de ces circuits dans cette classification. [20]

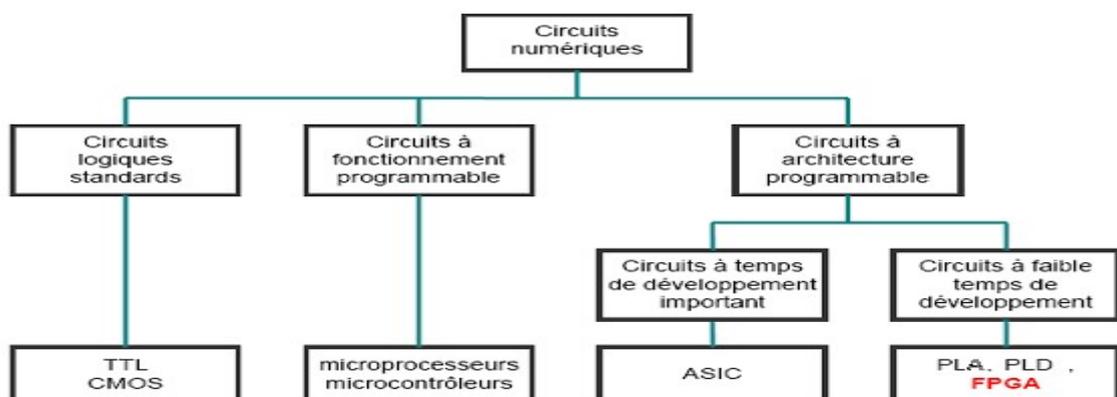


Figure 4.1 : Classification des circuits numériques

4.2.1 Les ASICs (Application Specific Integrated Circuit)

Par définition, les circuits ASIC regroupent tous les circuits dont la fonction peut être personnalisée d'une manière ou d'une autre en vue d'une application spécifique, par opposition aux circuits standards dont la fonction est définie et parfaitement décrite dans le catalogue des composants.

Les ASIC peuvent être classés en plusieurs catégories selon leur niveau d'intégration, en fait un ASIC est défini par sa structure de base (réseau programmable, cellule de base, matrice, etc.). Sous le terme ASIC deux familles sont regroupées, les semi personnalisés (avec des réseaux prédéfinis) et les personnalisés (non préfabriqué, qu'on optimise pour créer son propre composant).

D'une manière générale l'utilisation d'un ASIC conduit à de nombreux avantages provenant essentiellement de la réduction de la taille des systèmes. Il en ressort, une réduction du nombre des composants sur le circuit imprimé. La consommation et l'encombrement s'en trouvent considérablement réduits.

Le concept ASIC par définition assure une optimisation maximale du circuit à réaliser.

Nous disposons alors d'un circuit intégré correspondant réellement à nos propres besoins. La personnalisation du circuit donne une confidentialité au concepteur et une protection industrielle.

Enfin, ce type de composant augmente la complexité du circuit, sa vitesse de fonctionnement et sa fiabilité. Dans l'approche des circuits du type ASIC, l'inconvénient majeur réside dans le fait du passage obligatoire chez le fondeur ce qui implique des frais et un temps de développement élevés du circuit.[21]

4.2.2 Les PLDs (Programmable Logic Device)

Ce sont des chips qui peuvent être programmés pour se comporter comme une

conception arbitraire. Un PLD peut être programmé pour une implémentation aussi simple qu'une opération en logique combinatoire comme il pourrait l'être pour des conceptions beaucoup plus importantes. Il comprend une matrice de portes AND connectée à une matrice de portes OR il est alors représenté sous forme de sommes de produits.

La figure suivante montre deux types de connexions dans les PLDs. [22]

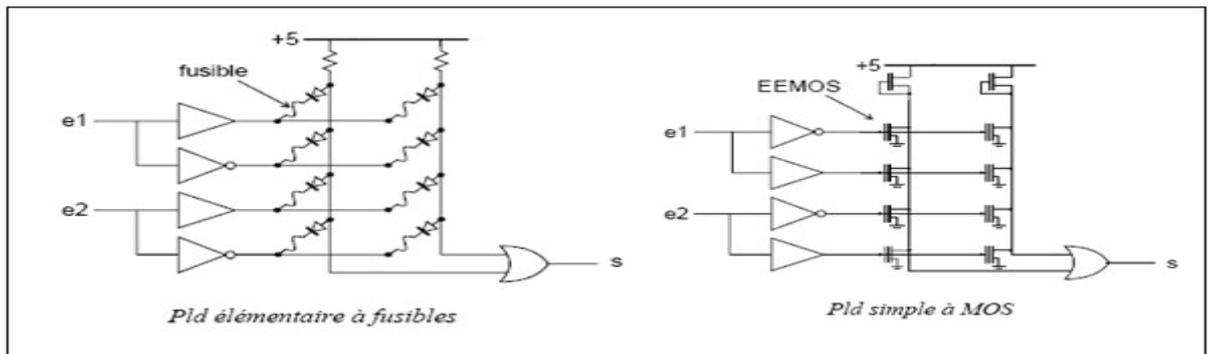


Figure 4.2 : les connexions dans un PLD

4.2.3 Les EPLDs "Erasable PLD" :

Qui sont des circuits effaçables par rayons ultraviolets, ils peuvent être reprogrammés. [22]

4.2.4 Les PALs "Programmable Array Logic"

C'est la partie essentielle des PLD, elle est constituée d'un plan de portes AND programmables suivi d'un plan de portes OR fixes ou figées. Figure(4.3) .[22]

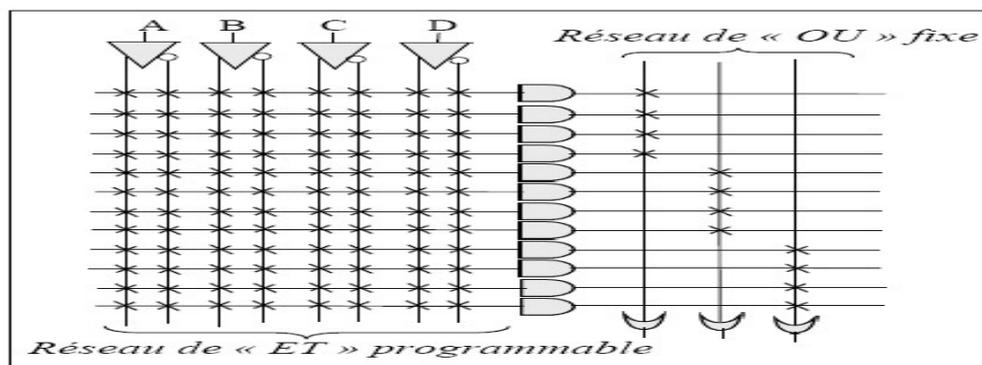


Figure 4.3: les connexions dans un PAL

4.2.5 Les EEPLDs "Electrically Erasable PLD"

Programmables et effaçables électriquement, ils peuvent être reprogrammés sur site. Les limites de l'architecture du PLD résident dans le nombre de bascules, le nombre de signaux d'entrées/sorties, la rigidité du plan logique ET/OU et des interconnexions. Précisons que ces composants très souples d'emploi sont limités à des fonctions numériques et adaptées à des productions de petites séries et ne présentent aucune garantie quant à la confidentialité. [23]

4.2.6 Les FPGA "Field Programmable Gate Arrays" ou "réseaux des blocs programmables"

Les circuits FPGAs sont des composants entièrement reconfigurables ce qui permet de les reprogrammer à volonté afin d'accélérer notablement certaines phases de calculs. Les circuits FPGAs sont constitués d'une matrice de blocs logiques programmables entourés des blocs d'entrée/sortie programmables. L'ensemble est relié par un réseau d'interconnexions programmable.

Les FPGAs ne sont pas optimisés pour une application bien déterminée, par conséquent ils consomment plus d'énergie que les ASICs. Par contre ils sont beaucoup plus simples à programmer et à reprogrammer, ce qui raccourcit les cycles de conception et permet de suivre l'évolution de l'application pour laquelle, ils ont été conçus. Les FPGAs sont plus convenables pour les prototypes et pour les productions en série limitées qui ne sont pas de la qualité des ASICs. [22]

4.3 Les FPGAs "Field Programmable Gate Arrays"

4.3.1 Description des composants FPGA

Les FPGA sont des composants VLSI (*Very Large Scale Integration*). Ils sont programmables par l'utilisateur et essentiellement constitués de trois parties :

- ✓ Une matrice de blocs logiques configurables CLB (*Configurable Logic Bloc*).
- ✓ Des blocs d'entrée/sortie configurables.
- ✓ Un réseau d'interconnexions programmables.

La figure (4.4) présente l'architecture générique d'un circuit FPGA [23] :

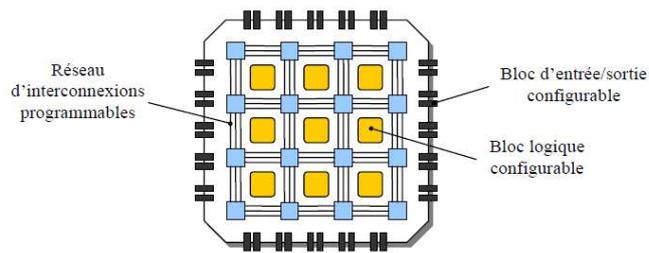


Figure 4.4 : l'architecture générique d'un FPGA

4.3.2 Architecture des circuits FPGA

Les circuits FPGA possèdent une structure matricielle de deux types de blocs (ou cellules). Des blocs d'entrées/sorties et des blocs logiques programmables. Le passage d'un bloc logique à un autre se fait par un routage programmable. Certains circuits FPGA intègrent également des mémoires RAM, des multiplieurs et même des noyaux de processeurs. [22]

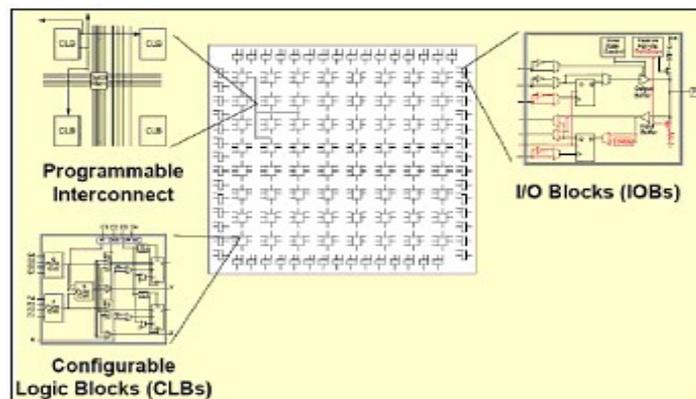


Figure 4.5 : Architecture interne du FPGA

Dans ce qui suit, on va faire une description de l'architecture utilisée par Xilinx, car C'est sur des circuits Xilinx qu'on va implémenter nos programmes.

L'architecture retenue par Xilinx se présente sous forme de deux couches :

- Une couche appelée **Circuit Configurable**,
- Une couche réseau mémoire **SRAM** (Static Read Access Memory).

Circuit configurable

La couche dite « circuit configurable » est constituée:

- ✓ Tout autour de ces blocs logiques configurables, on trouve des **blocs d'entrées/sorties (IOB)** dont le rôle est de gérer les entrées-sorties réalisant l'interface avec les modules extérieurs (**figure 4.5**).
- ✓ d'une matrice de **blocs logiques configurables (CLB)** permettant de réaliser des fonctions combinatoires et des fonctions séquentielles (**figure 4.6**).

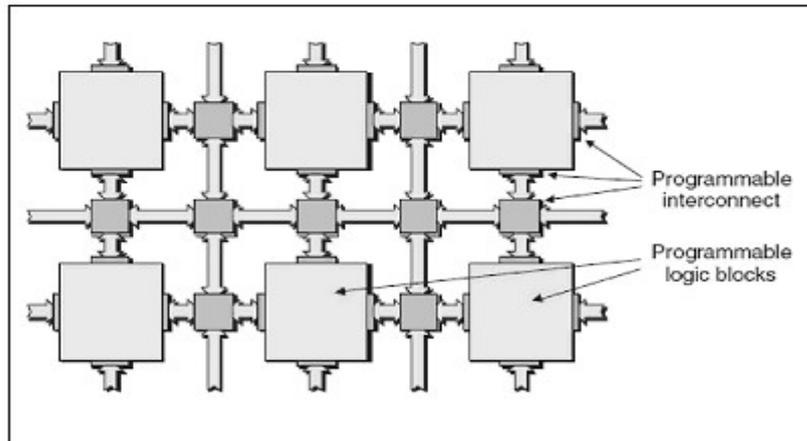


Figure 4.6 : Blocs et Interconnexions programmables

Réseau mémoire SRAM

La programmation du circuit FPGA, appelé aussi LCA (Logic Cells Arrays), consistera en l'application d'un potentiel adéquat sur la grille de certains transistors à effet de champ servant à interconnecter les éléments des CLB et des IOB, afin de réaliser les fonctions souhaitées et d'assurer la propagation des signaux. Ces potentiels sont tout simplement mémorisés dans le réseau mémoire SRAM (**figure 4.7**). [20]

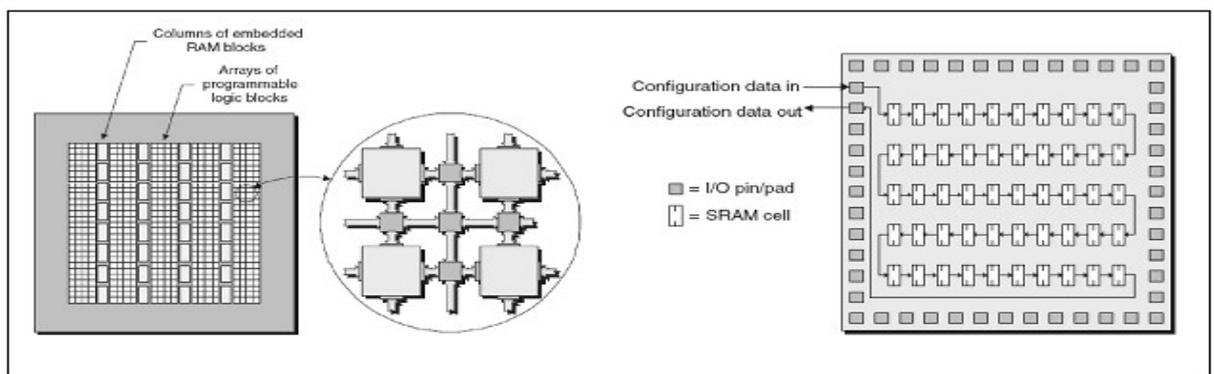


Figure 4.7: Situation du réseau SRAM

La programmation d'un circuit FPGA est volatile, la configuration du circuit est donc mémorisée sur la couche réseau SRAM et stockée dans une ROM externe. Un dispositif interne permet à chaque mise sous tension de charger la SRAM interne (**figure 4.8**) à partir de la ROM. Ainsi on conçoit aisément qu'un même circuit puisse être exploité successivement avec des ROM différentes puisque sa programmation interne n'est jamais définitive. On voit ici tout le profit que l'on peut tirer de cette souplesse en particulier lors d'une phase de mise au point. Une erreur n'est pas rédhibitoire, mais peut aisément être réparée. La mise au point d'une configuration s'effectue en deux temps : Une première étape purement logicielle va consister à dessiner puis simuler logiquement le circuit fini. Dans la seconde étape, on effectuera une simulation matérielle en configurant un circuit réel. On pourra alors vérifier si le fonctionnement réel correspond bien à l'attente du concepteur, et si besoin est identifier les anomalies liées généralement à des temps de transit réels légèrement différents de ceux supposés lors de la simulation logicielle, ce qui peut conduire à des états instables voire même erronés.[22]

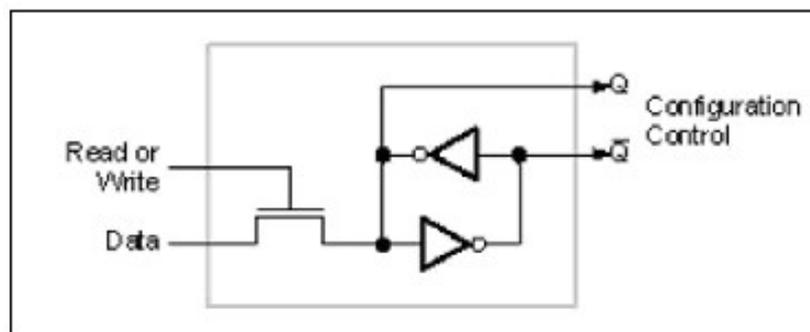


Figure 4.8: Structure d'une cellule SRAM

Les circuits FPGA du fabricant Xilinx utilisent deux types de cellules de base : les cellules d'entrées/sorties appelées IOB (Input Output Bloc), et les cellules logiques appelées CLB (Configurable Logic Bloc). Ces différentes cellules sont reliées entre elles par un réseau d'interconnexions configurable. On décrit dans ce qui suit chacun de ces composants.

a IOB (Input Output Bloc)

Ils constituent l'interface entre les bornes du circuit et les CLB. Le dispositif de routage Versa Ring offre les ressources nécessaires à l'interconnexion des CLB aux IOB. Nous pouvons ainsi modifier le système implanté sur le FPGA sans interférer avec l'attribution des bornes. Cette caractéristique s'avère importante si nous souhaitons développer des nouvelles versions d'un produit tout en conservant la compatibilité au niveau du boîtier.[22]

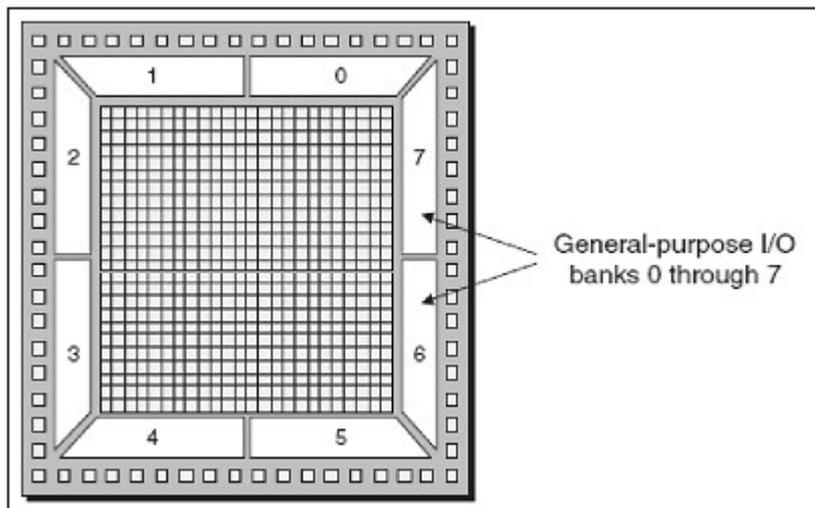


Figure 4.9: Exemple de IOB

Ils sont présents sur toute la périphérie du circuit FPGA. Chaque bloc IOB contrôle une broche du composant et il peut être défini en entrée, en sortie, en signal bidirectionnel ou être inutilisé (état haute impédance). La figure 4.10 présente la structure de ces blocs.

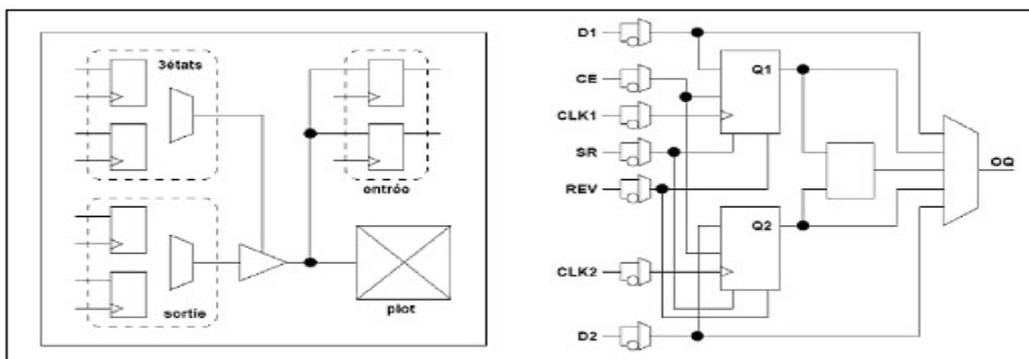


Figure 4.10 : Schéma d'un bloc d'entrée/sortie (IOB).

➤ **Configuration en entrée**

Premièrement, le signal d'entrée traverse un buffer qui, selon sa programmation, peut détecter soit des seuils TTL soit des seuils CMOS. Il peut être routé directement sur une entrée directe de la logique du circuit FPGA ou sur une entrée synchronisée. Cette synchronisation est réalisée à l'aide d'une bascule de type D, le changement d'état peut se faire sur un front montant ou descendant. De plus, cette entrée peut être retardée de quelques nanosecondes pour compenser le retard pris par le signal d'horloge lors de son passage par l'amplificateur. Le choix de la configuration de l'entrée s'effectue grâce à un multiplexeur (program controlled multiplexer). Un bit positionné dans une case mémoire commande ce dernier.

➤ **Configuration en sortie**

Nous distinguons les possibilités suivantes :

- inversion ou non du signal avant son application à l'IOB ;
- synchronisation du signal sur des fronts montants ou descendants d'horloge ;
- mise en place d'un "pull-up" ou "pull-down" dans le but de limiter la consommation des entrées sorties inutilisées ;
- signaux en logique trois états ou deux états. Le contrôle de mise en haute impédance et la réalisation des lignes bidirectionnelles sont commandés par le signal de commande « Out Enable », lequel peut être inversé ou non.

Chaque sortie peut délivrer un courant de 12mA. Ainsi toutes ces possibilités permettent au concepteur de connecter au mieux une architecture avec les périphériques extérieurs.[22]

b CLB (Configurable Logic Bloc)

C'est l'unité fondamentale du bloc logique qui fournit des éléments utilitaires pour la logique combinatoire et la logique synchrone, y compris les éléments du stockage de base : buffers à 3 états à associer avec chaque CLB. Les CLBs incluent quatre parties identiques appelées SLICE et deux buffers à 3 états.

Chaque SILICE est équivalente et contient :

- Deux générateurs de la fonction (F & G).
- Deux éléments du stockage.
- Des portes de la logique arithmétique.
- Grands multiplexeurs.
- Une large fonctionnalité.
- Une logique de retenue rapide.
- Une chaîne de cascade Horizontale (porte OU).

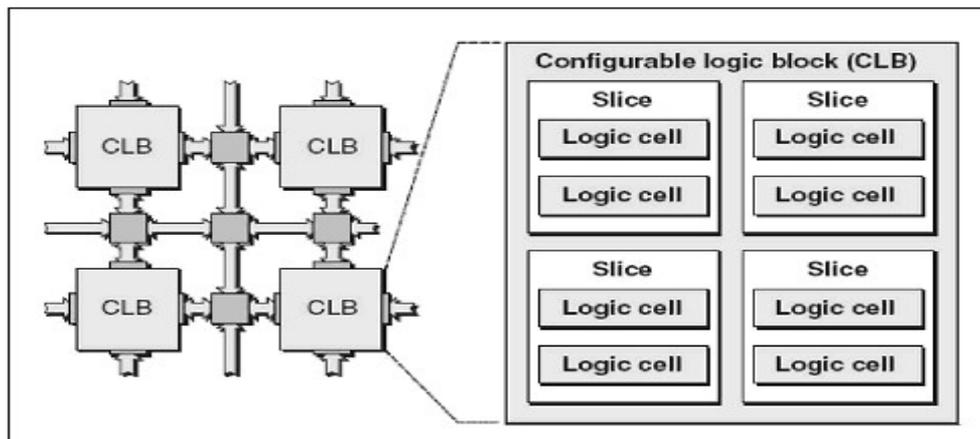


Figure 4.11 : Bloc CLB

Les blocs logiques configurables sont les éléments déterminants des performances du circuit FPGA. Chaque CLB est un bloc de logique combinatoire composé de générateurs de fonctions à quatre entrées (LUT) et d'un bloc de mémorisation/ synchronisation composé de bascules D. Quatre autres entrées permettent d'effectuer les connexions internes entre les différents éléments du CLB.

La LUT (Look Up Table) est un élément qui dispose de quatre entrées, il existe donc $2^4 = 16$ combinaisons différentes de ces entrées. L'idée consiste à mémoriser la sortie correspondant à chaque combinaison d'entrée dans une petite table de 16 bits, la LUT devient ainsi un petit bloc générateur de fonctions. La figure (4.11) montre le schéma simplifié d'un CLB de la famille XC4000 de Xilinx.

4.3.3 Programmation et configuration des FPGAS

La configuration est le processus de charger des données spécifiques à un design dans un ou plusieurs FPGAs pour définir l'opération fonctionnelle des blocs internes ainsi que leur interconnexion, et leurs temps de configuration dépendent du mode de configuration sélectionnée [24]. Dans la littérature on peut distinguer 4 modes de configuration et programmation des circuits FPGAs comme le montre la Figure. 4.12 :

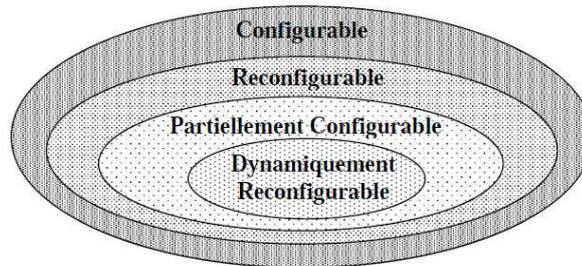


Figure . 4.12 : Classification des circuits FPGAs selon leurs configurations

a Circuit configurable

Un circuit Configurable est un circuit programmé et chargé par différentes données, où les interconnexions d'un FPGA sont programmées afin de donner un fonctionnement spécifique pour un tel circuit.

b Circuit reconfigurable

C'est le même principe de configuration, sauf que cette fois en reconfigurant le circuit FPGA une deuxième fois pour l'utilité dans une autre fonction comme on peut garder la dernière configuration du circuit, et même on peut effacer cette configuration et on config. Le circuit une nouvelle fois.

c Circuit partiellement reconfigurable

Un dispositif ou un circuit est défini comme partiellement reconfigurable (dans la littérature on trouve aussi la terminologie Run Time Reconfiguration RTR globale) s'il est possible de le reconfiguré sélectivement, tandis que l'état de repos du reste du dispositif est inactif, mais il conserve son information configurée. Encore, il ne semble pas y avoir n'importe quel dispositif sur le marché qui soit partiellement reconfigurable, mais non aussi dynamiquement reconfigurable, La reconfiguration partielle permet de rendre un FPGA effectif, multiple fonctions, et change des fonctions pendant le fonctionnement du système.

d Circuit dynamiquement reconfigurable

Un circuit FPGA est reconfigurable dynamiquement (dans la littérature on trouve la terminologie Run Time Reconfiguration RTR local) s'il peut être partiellement reconfiguré durant son fonctionnement, c.-à-d. une partie du circuit correspondant à certaines fonctions logiques et leur interconnexions peut être changé sans affecter le fonctionnement de la logique restante. On peut aussi parler de reconfiguration dynamique dans le cas où plusieurs circuits FPGAs sont connectés entre eux et il s'agit de reconfigurer un seul composant FPGA tout en maintenant les autres circuits en fonctionnement.

4.3.4 Les avantages et les inconvénients des *FPGA* [19]

Les avantages et les inconvénients des *FPGA* sont multiples on trouve :

a Les avantages

- Technologie « facile » à maîtriser
- Idéal pour le prototypage.
- Reprogrammable.
- Parallélisme de traitement.
- Coût peu élevé.
- Parallélisme de traitement.
- Flexibilité et la possibilité de réduire.

b Les Inconvénients

- Performances non optimisées
- Temps de réponse long par rapport aux *ASIC*.

4.4 System Generator

Dans ce paragraphe nous allons présenter un bref aperçu sur le System Generator. Ce dernier est compatible avec le Xilinx 12.3 et matlab 2009a. Le System

Generator de Xilinx est un logiciel qui s'intègre à matlab .il est composé a deux section : une librairie de blocs pour Simulink et un logiciel qui transforme ces blocs en code utilisable pour programmer un FPGA.

La librairie Simulink, fournie par Xilinx, fonctionne selon le même principe que les autres éléments de Simulink, qui contient une série de blocs qui représentent plusieurs fonctions et qui peuvent être reliés entre eux pour former les algorithmes, à la seule différence que ces blocs ne servent pas seulement à simuler, mais à générer également des codes VHDL ou verilog.

4.4.1 Principales fonctions disponibles

Les blocs du System Generator sont séparés en cinq groupes selon leurs fonctions, Ce sont des blocs simples qu'imitent des fonctions déjà présentes dans le Simulink. Dans ce qui suit nous donnons les principaux blocs séparés par sections .ces derniers sont séparées en réalité sur Simulink en dix groupes, dans notre cas on va présenter seulement les sections les plus essentielles.

a Basic Element

System Generator , black Box, register, Mux, Sunc, Delay, Time division ,Multiplexer, Assert ,BitBasher

b Math

Accumulator,AddSub,Cmilt,Constant,Convert,Counter,Expression,Inverter,Logical,MCode,Mult, Negate, Relational ,Scale , Shift,

c Matlab IO

Display, Enable Adapter, Gateway In, Gateway Out, Wave Scope , ModelSim.

d Memory

ROM, Dual Port Ram, Addressable Shift Register, LFSR, FIFO, Shared Memory.

e DSP

DAFIR , DDS ,FFT ,FIR ,LFSR.

Le System Generator offre la possibilité de faire des simulations de l'association hardwar - système continue .et de la simulation hardwar - softwar (Co-Simulation)

4.5 Conclusion

A travers de ce chapitre, nous avons essayé de donner en premier une bref historique sur les circuits intégrés puis on a présenté les différents types de circuits logiques programmable et l'architecture des circuits FPGA, en second lieu nous nous sommes intéressé la présentation d'un outil de conception System Generator de xilinx.

Vu que le system Generator permet de faire des simulation Hardwar-système continue, il convient parfaitement a la description du contrôleur flou et la simulation du l'association Commande –Hacheur –Moteur .

Chapitre 5 Implémentation sur FPGA

5.1 Introduction

Dans ce chapitre, On va présenter une méthode pour l'implémentation d'un régulateur flou sur la carte FPGA. Nous utilisons le System Generator de Xilinx pour la description de l'implémentation sur FPGA, de l'algorithme décrit.

5.2 Implémentation d'un régulateur flou sur FPGA

Le tableau suivant résume le choix des caractéristiques de régulateur flou implémenté :

Système d'inférence floue	Caractéristiques
Nombre des variables d'entrées	2
Nombre des bits de chaque entrées	12 bits
Nombre des variables de sortie	1
Nombre des bits du variables de sortie	12 bits
Nombre de fonction d'appartenances (MF) de chaque variable	7

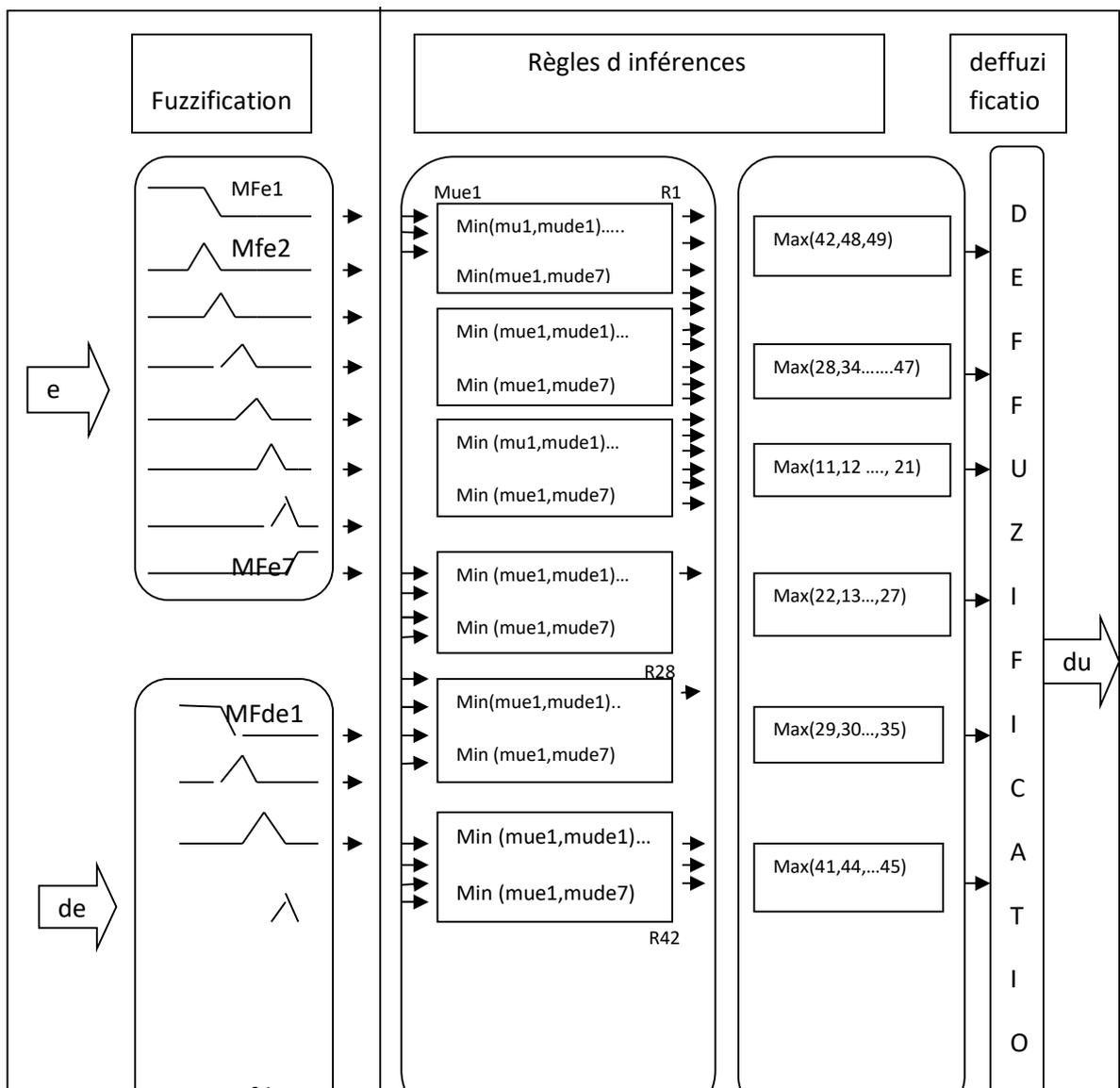
Résolution de chaque (MF) d entrée	8bits
Résolution de chaque (MF) de sortie	12 bits
Méthode d agréation	Max/Min
Méthode de defuzzification	Hauteurs pondérés

Tableau 5.1 : caractéristique du régulateur flou.

La structure générale de l’algorithme du régulateur flou est représentée sur la figure (5.1) . On note que l’horloge est égale a celle de la carte FPGA. Cette implémentation contient 3 modules comme suite :

1. Module de calcul fonction d’appartenances.
2. Module de calcul de l’inférence.
3. Module de la déffuzification.

La structure détaillée est représentée par la figure suivante:



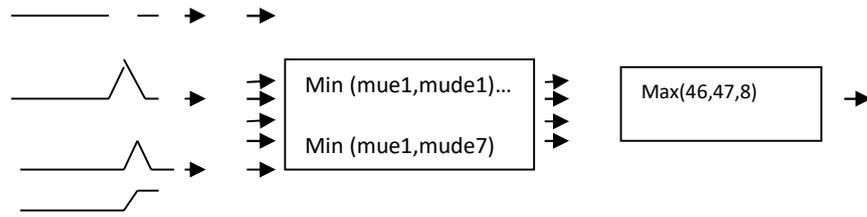


Figure 5.1 : Algorithme général pour l'implémentation du régulateur flou.

5.2.1 Module de calcul de fonctions d'appartenances

a Fuzzification des variables

On choisit cinq intervalles flous et des fonctions d'appartenances de type triangulaire pour les deux variables d'entrée, la première et la dernière sont de type trapézoïdale .L' univers de discours varie dans un intervalle de 12 bits.

b degré d'appartenance

Ce module réalise les sept fonctions d'appartenances pour les deux variables d'entrées. Chaque d'entrée est codée sur 12 bits, la partie positive de chaque fonction est codée sur 8 bits.

c Ensemble d'appartenances

Les ensembles d'appartenances sont nommés par des lettres (NG,NM.....,PG), a cet effet chaque intervalle défini dans la section précédente corresponde a une lettre .ces dernier seront utilisées pour définir les règles d inférences .

La figure (5.2) illustre la structure implémentée sur le System Generator de la fonction d'appartenances de mf de (NG).

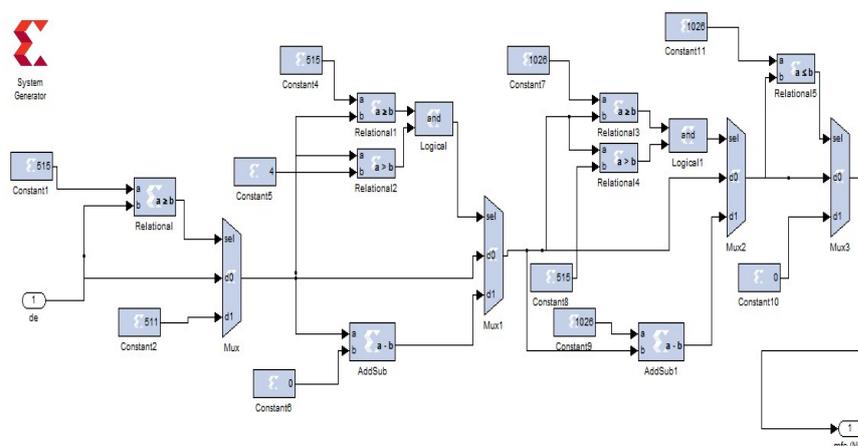


Figure.5.2. Implémentation du module fonction d'appartenances de mfde (NG).

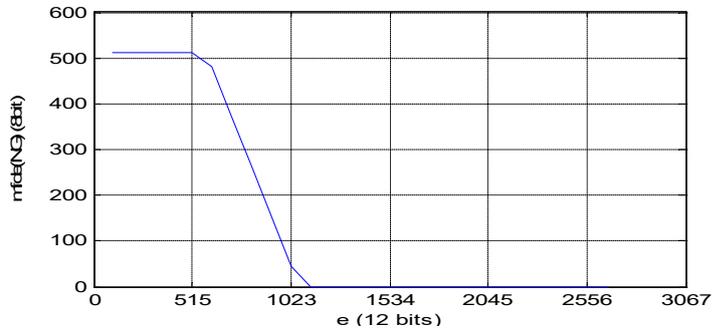


Figure.5.3. Simulation comportementale de fonction d'appartenances (mfde(NG))

5.2.2 Module de calcul de l'inférences

a Les règles d'inférences

L'opérateur **ET** est réalisé par la fonction **Min**, la figure (5.3) donne la réalisation de la septième fonction d'appartenances de la variable erreur **e** combinée aux différentes fonctions de la dérivée de l'erreur **de**, donc nous aurions 49 opérations de **Min**, ce ci comme la suite :

$$R1 = (e(k) \text{ est NG } \mathbf{MIN} \text{ de}(k) \text{ est NG } \mathbf{MIN} \Delta u(k) \text{ est PG})$$

$$R2 = (e(k) \text{ est NG } \mathbf{MIN} \text{ de}(k) \text{ est NM } \mathbf{MIN} \Delta u(k) \text{ est PG})$$

.

.

$$R49 = (e(k) \text{ est PG } \mathbf{MIN} \text{ de}(k) \text{ est PG } \mathbf{MIN} \Delta u(k) \text{ est NG}).$$

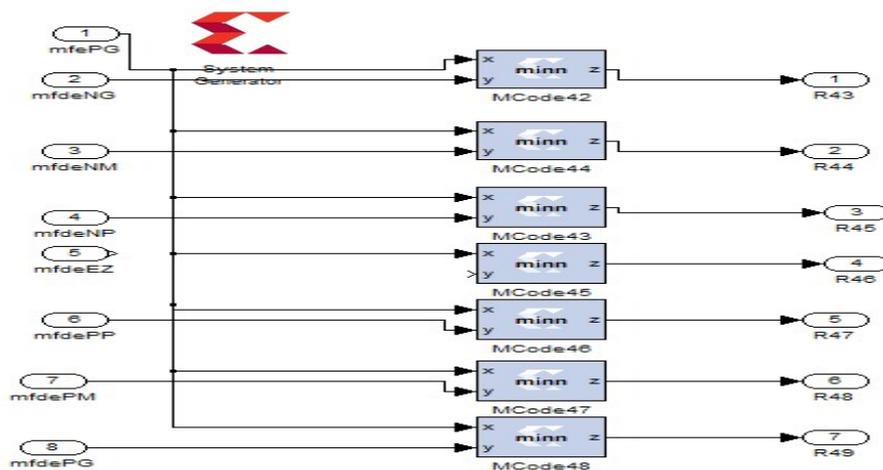


Figure.5.4 : Implémentation de ``si e est mfe (PG) et de est mfde N ``

b Réalisation de la fonction OU :

L'opérateur **OU** réalisé par la fonction **MAX**, elle est appliquée sur les règles qui donnent les mêmes variables floues donne :

$$\mu_{r(NG)} = \text{MAX} (R42, R47, R48)$$

$$\mu_{r(NM)} = \text{MAX} (R28, R34, R35, R41, R46, R47)$$

$$\mu_{r(NP)} = \text{MAX} (R14, R21, R20, R26, R27, R32, R33)$$

$$\mu_{r(EZ)} = \text{MAX} (R7, R13, R19, R25, R31, R42)$$

$$\mu_{r(PP)} = \text{MAX} (R5, R6, R12, R17, R18, R23, R24, R35, R36, R42)$$

$$\mu_{r(PM)} = \text{MAX} (R3, R4, R9, R10, R15, R16, R22)$$

$$\mu_{r(PG)} = \text{MAX} (R1, R2, R8)$$

La figure (5.5) illustre la réalisation du fonction **MAX** pour le résultat $\mu_{r(NP)}$:

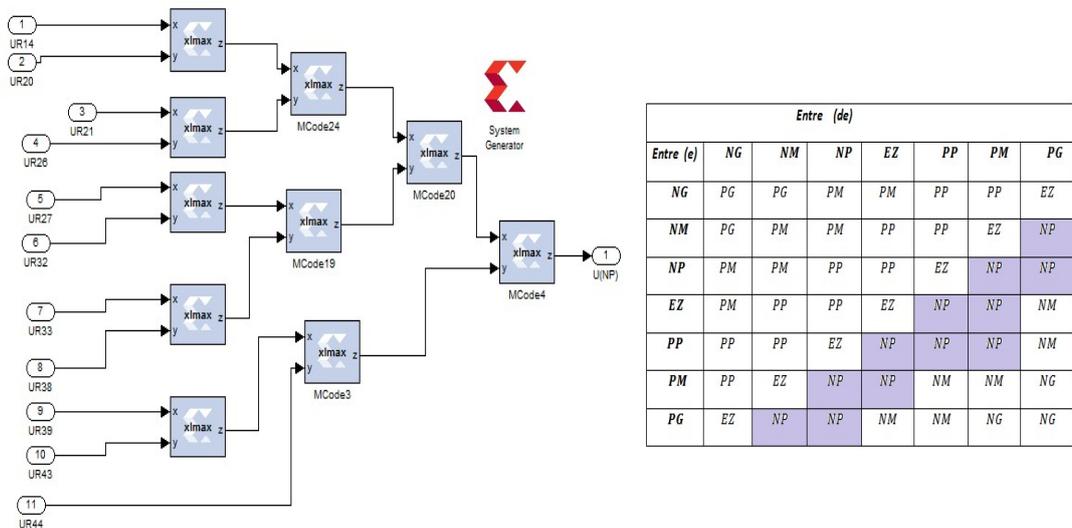


Figure .5.5: implémentation de module de fonction ``OU``, variable(NP)

5.2.3 Réalisation de la déffuzification

Ce module est divisé en trois blocs, bloque pour calcul le numérateur, la deuxième pour calculer le dénominateur et finalement bloque pour calculer la division, Voir la figure (5.6). La méthode utilisée pour cette opération est la méthode des hauteurs pondérées.

$$\Delta u_n(k) = \frac{\sum_{i=1}^n x u_{ri}(x)}{\sum_{i=1}^n u_{ri}(x)} \quad (5.1)$$

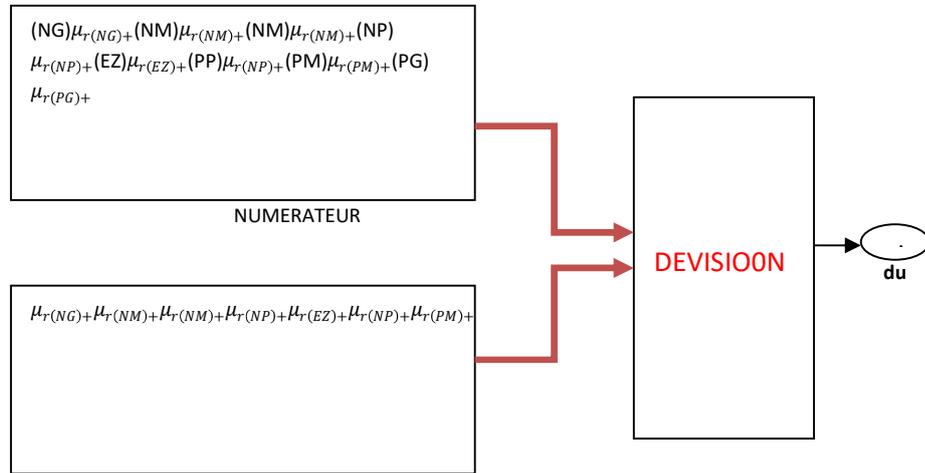


Figure 5.6 : Implémentation de module de la defuzzification

Le bloque numérateur réalise l'implémentation de la fonction multiplication et addition, la figure (5.7) illustre cet implémentation, alors que le dénominateur réalise la somme des degrés d'appartenances (voir la figure (5.8)). La division numérique se réduit à des soustractions et des décalages, nous avons considéré un division non signé.

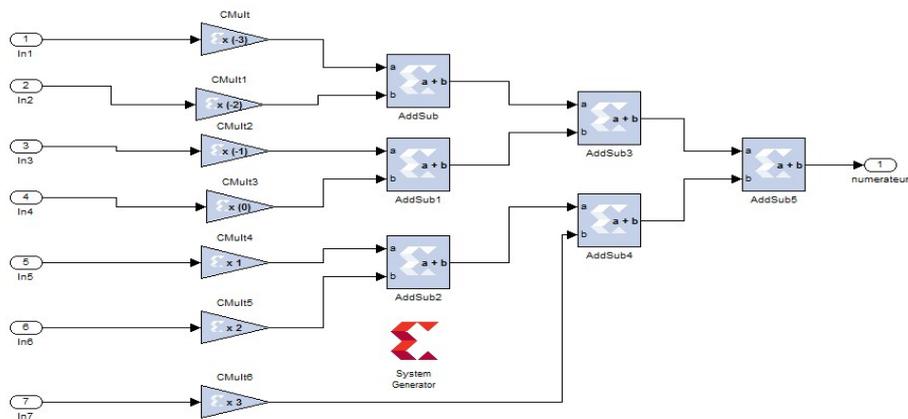


Figure 5.7. Réalisation du numérateur

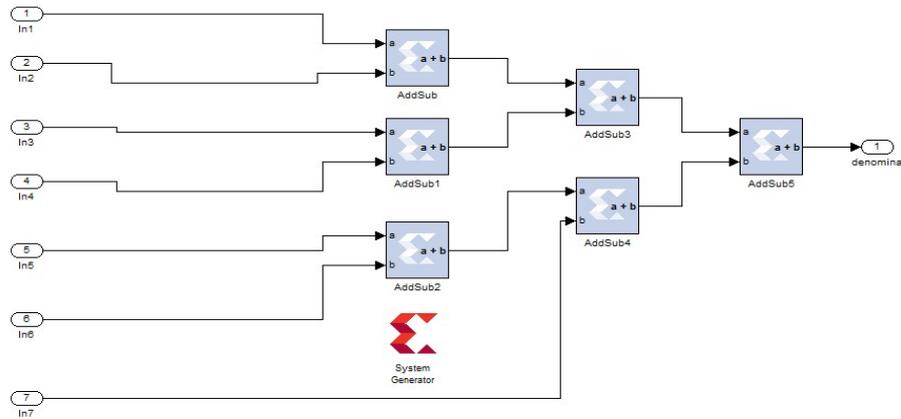


Figure 5.8. Réalisation du dénominateur

5.3 Réalisation d'intégrateur

L'intégral numérique dans le temps de la fréquence f est par definition :

$$\theta(t) = \int f dt = \theta(k) + \theta(k) T_s$$

Avec T_s représente le pas d'intégration.

La figure (5.8) suivante représente la structure d'implémentation sur le Système Générateur :

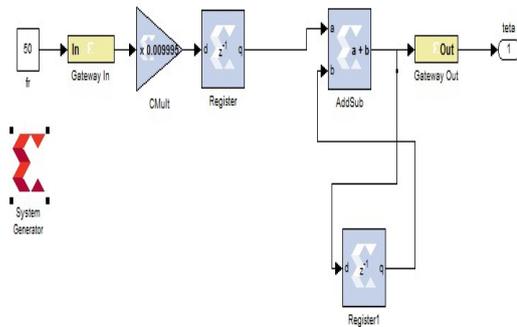


Figure.5.9: Implémentation de l'intégral

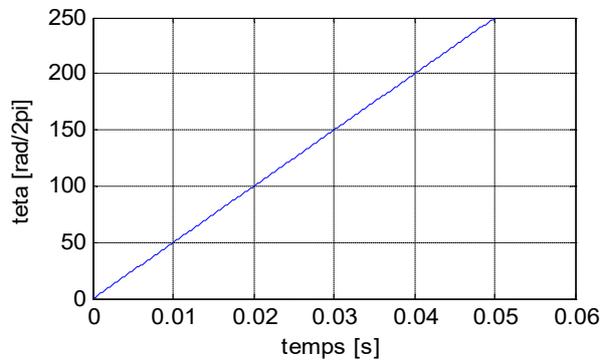


Figure.5.10 : Résultat de simulation

5.4 Conclusion

Dans ce chapitre, nous avons proposé un algorithme simple pour l'implémentation de la fonction d'appartenance. Cet algorithme est basé essentiellement sur une équation. La Synthèse matériel de l'algorithme proposé, est effectué par le System Generator. En deuxième partie, nous avons proposé une architecture simple pour l'implémentation de l'algorithme de régulateur flou à sept fonctions d'appartenance.

L'utilisation de l'outil de conception haut niveau 'System Generator' est très bénéfique pour la vérification du comportement de l'algorithme sur simulink.

Conclusion générale

Le travail qui se fait dans ce mémoire concerne la commande en vitesse d'un moteur BLDC par la logique floue et l'implémentation de cette commande sur une cible FPGA. Nous avons commencé par des informations générales sur le BLDC (principe de fonctionnement, types des BLDC et l'alimentation)

Pour le 2^{ème} chapitre nous avons construit un modèle à partir des équations pour la machine avec son commutateur et après on a fait des tests de simulation pour mieux connaître le fonctionnement de la machine en boucle ouverte. Ainsi nous avons fait une variation paramétrique sur le moteur pour savoir l'influence de ces paramètres sur le fonctionnement du BLDC.

Dans le 3^{ème} chapitre on a fait une étude générale sur la logique floue qui nous a permis de comprendre les principales étapes à suivre dans une conception d'un système flou, puis on a développé la commande en vitesse sur la base de modèle obtenu pour le sens de rotation, ensuite nous avons fait la structure en cascade de régulateur avec le moteur, le réglage du courant continu à l'entrée du commutateur est obtenu par modulation de la tension à l'entrée du commutateur en utilisant un hacheur de courant à deux quadrants. Nous avons effectué la simulation de la commande en vitesse pour les deux sens de rotation.

Dans le 4^{ème} chapitre nous avons présenté les différents circuits logiques programmables. On s'est concentré sur l'étude des FPGAs, les types des circuits logique programmable, architecture interne et présentation l'outil de simulation System Generator.

Le dernier chapitre on a implémenté l'algorithme de régulateur flou fait a l'aide d'un outil de haut niveau a savoir. Le System Generator de xilinx. Cette librairie Simulink permet d'utiliser l'espace matlab pour une simulation comportementale de l'ensemble commande - moteur et par conséquent, permet la validation de l'algorithme avant son implémentation.

Il est extrême demandé de faire une réalisation pratique pour confirmer les résultats ainsi obtenus.

