

الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم العالي والبحث العلمي

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITE MOHAMED SEDDIK BENYAHIA JIJEL

Faculté des sciences et de la technologie

Département d'électrotechnique

N° :...../2023

MEMOIRE DE MASTER

DOMAINE: Sciences et Technologies

FILIERE: Électrotechnique

SPECIALITE : Commande Électrique

Thème

*Commande D'un Convertisseur En pont,
Implémentation Sur Une Carte ARDUINO*

Présenté Par : Benayad Mustapha

Encadré Par : Lalili Djafer

Boumaaraf Seyf Eddine

Date de soutenance : 26/06/2023

Jury de Soutenance

Président : Maouche Abdelaziz

Grade : MCB

Univ MSB jijel

Encadreur : Lalili Djafer

Grade : Professeur

Univ MSB jijel

Examineur: Kihal Mouhamed Cherif

Grade : MCB

Univ MSB jijel

Promotion : 2022 /2023

Remerciement

*Avant tout, nous remercions **Dieu** le tout puissant de nous avoir donné courage, patience et force durant toutes ces années d'étude et que grâce à lui ce travail a pu être réalisé.*

*Nous remercions tout particulièrement notre encadrant Mr **LALILI Djafer**, qui a proposé et dirigé ce travail pour ses précieux conseils, orientations, patience qui ont constitué un apport considérable sans lesquels ce travail n'aurait pas pu être mené au bon port. Qu'il trouve dans ce travail un hommage vivant à ses hautes qualités humaines.*

Nous remercions également les membres du jury qui ont fait l'honneur de bien vouloir examiner ce travail.

*Nous tenons à remercier les responsables et tout le personnel du **département d'électrotechnique** de l'université Mohammed Seddik Benyahia – Jijel pour les facilités qu'ils nous ont accordés pour terminer ce travail.*

Sans oublier tous ceux qui ont contribué de près ou de loin à la réalisation de ce modeste mémoire.

DEDICACE

Au nom d'ALLAH tout puissant pour tous les bienfaits qu'il ne cesse d'accomplir dans nos vies, Que la paix et le salut soient son messager Mohamed, sa famille et ses compagnons.

Je dédie ce modeste travail :

À mes très chers parents, qui ont tout le crédit.

À ma sœur Aya.

À toute ma famille, des oncles et des oncles.

À mes amis : Mouhammed, Taher, Chakib, Ahmed, Zakaria, Aymen, Ammardjia et mon binôme Seyf Eddine.

À mes collègues de la promotion commande électrique 2023.

Avec un grand MERCI pour notre prof Mr LALILI DJAAFER,

Benayad Mustapha

DEDICACE

*Au nom d'ALLAH tout puissant pour tous les bienfaits qu'il ne cesse d'accomplir dans nos vies,
Que la paix et le salut soient son messager Mohamed, sa famille et ses compagnons.*

Je dédie ce modeste travail :

À mes très chers parents, qui ont tout le crédit.

À mes grands-parents, Fatima, Manina, Lakhdar et Rabah Allah yarhimo.

À mon cher amie « Wahibou » Allah yarhimo.

À mes frères et sœurs : Abbas et Moussaab, Ayam et Anfel.

À toute ma famille, des oncles et des oncles, et surtout Kouahi Haroun.

À mes cousins, et surtout Moussa, Yacine et Aymen.

À mes amis : El Hadj, Maamar, Hichem, Djihad, Houcine, Nabil, Chakib, Ahmed, Ibrahim, Ammardjia et mon binôme Mustapha.

À mon maître, Cheikh Mohammed El-Ssaleh Hadeef.

À mes collègues de la promotion commande électrique 2023 et surtout Mouhamed Baba.

À l'âme du grand penseur « Malek Bennabi », le philosophe de la civilisation.

À « Oum El Fadhl », la femme qui sera la mère de mes fils et filles incha'alah.

Avec un grand MERCI pour notre prof Mr LALILI DJAAFER.

Boumaaraf Seyf Eddine

TABLES DES MATIERES

Liste des figures

Liste des tableaux

Notations

Introduction générale 1

Chapitre I : Onduleur Et Sa Commande

I.1 Introduction.....	3
I.2 Définition de l'onduleur.....	3
I.3 Classification des onduleurs	3
I.3.1 Classification selon la nature de l'alimentation.....	3
I.3.1.1 Onduleur de tension	3
I.3.1.2 Onduleur du courant	4
I.3.2 Classification selon la nature de la commande	4
I.3.2.1 Onduleur autonome.....	4
I.3.2.2 Onduleur non autonome.....	5
I.3.3 Classification selon la nature de la charge	5
I.3.3.1 Onduleur monophasé	5
I.3.3.2 Onduleur triphasé.....	6
I.3.3.3 Autres structures des onduleurs	7
I.4 Stratégies de commande de l'onduleur monophasé.....	7
I.4.1 Commande en pleine onde symétrique	8
I.4.2 Commande « Décalée »	10
I.4.3 La commande en MLI triangulo-sinusoidale	14
I.4.3.1 Caractéristiques de la MLI.....	15
I.4.3.2 MLI Bipolaire	15
I.4.3.3 MLI unipolaire	20
I.4.4 La commande en courant par hystérésis	31
I.5 Conclusion :	34

TABLES DES MATIERES

Chapitre II : Carte ARDUINO, Onduleur Et Accessoires De Mesure

II.1 Introduction	35
II.2 Carte ARDUINO	35
II.2.1 Définition et Historique de la carte ARDUINO	35
II.2.2 Différent type.....	36
II.2.3 Critère de Choix d'une carte Arduino	36
II.2.4 Carte Arduino UNO.....	38
II.2.4.1 Description d'une carte Arduino UNO.....	38
II.2.4.2 Structure de la carte ARDUINO.....	39
II.2.4.3 Caractéristique de la carte Arduino UNO.....	41
II.2.5 L'environnement de programmation de l'ARDUINO	41
II.2.5.1 Structure d'un programme Arduino IDE.....	42
II.2.5.2 Structure et Principaux commandes d'un projet ARDUINO C	43
II.3 Onduleur à commander	45
II.3.1 Module L298N	45
II.3.1.1 Définition et fonctionnement.....	45
II.3.1.2 Structure et brochage du module L298N.....	45
II.3.1.3 Caractéristiques du module L298N	47
II.3.2 Module BTS7960, ou BT_2	48
II.3.2.1 Définition et fonctionnement.....	48
II.3.2.2 Structure et brochage du module BTS7960.....	48
II.3.2.3 Caractéristiques du module BTS7960	49
II.4 Accessoires de mesure.....	50
II.4.1 Convertisseur analogique numérique ADS1115	50
II.4.1.1 Définition et Pinnage	50
II.4.1.2 Modes de mesure de tension.....	51
II.4.1.3 Caractéristiques principales du module ADS1115.....	51

TABLES DES MATIERES

II.4.1.4 Raccordement du Module ADS1115 avec ARDUINO	52
II.4.1.5 Programmation et paramétrage de l'ADS1115	53
II.4.2 Capteur de tension et de courant INA219	53
II.4.2.1 Définition et Pinnage du module	53
II.4.2.2 Caractéristiques principales du module	54
II.4.2.3 Raccordement du Module INA219 avec l'ARDUINO	55
II.4.2.4 Programmation et paramétrage de l'INA219	55
II.5 Conclusion	56
 <u>Chapitre III : Implémentation De La Commande De L'onduleur Sur Une Carte Arduino</u>	
III.1 Introduction	57
III.2 Schéma de la commande et banc d'essai	57
III.3 Application de la commande en pleine onde sur le module L298N	59
III.4 Application de la commande en pleine onde sur le module BTS7960	59
III.5 Application de la commande en pleine onde décalée sur le module L298N	60
III.6 Application de la commande en pleine onde décalée sur le module BTS7960	61
III.7 Application de la commande MLI triangulo-sinusoïdale bipolaire sur le module L298N	61
III.8 Application de la commande MLI triangulo-sinusoïdale bipolaire sur le module BTS7960 ...	62
III.9 Application de la commande MLI unipolaire à deux références sur le module L298N	63
III.10 Application de la commande MLI unipolaire à deux références sur le module BTS7960	63
III.11 Application de la commande MLI unipolaire à deux porteuses sur le module L298N	64
III.12 Application de la commande MLI unipolaire à deux porteuses sur le module BTS7960	65
III.13 Conclusion.....	65
 Conclusion générale	66
Bibliographie	68
Annexes	71

LISTES DES FIGURES

N°: Figure	Intitulé	page
<i>Figure (I.1)</i>	Structure Générale de l'onduleur	3
<i>Figure (I.2)</i>	L'onduleur de tension	4
<i>Figure (I.3)</i>	L'onduleur du courant	4
<i>Figure (I.4)</i>	Structure principal d'un onduleur monophasé en demi-pont	5
<i>Figure (I.5)</i>	Structure d'un onduleur monophasé en pont complet	6
<i>Figure (I.6)</i>	Structure d'un onduleur triphasé	7
<i>Figure (I.7)</i>	Principe de commande du bras de l'onduleur	8
<i>Figure (I.8)</i>	Signal de commande de la commande pleine onde	9
<i>Figure (I.9)</i>	Configurations possibles de la commande pleine onde	9
<i>Figure (I.10)</i>	Tension et courant de sortie de la commande pleine onde	11
<i>Figure (I.11)</i>	L'analyse spectrale de la tension pour la commande pleine onde	11
<i>Figure (I.12)</i>	Signal de commande de la Commande décalée	12
<i>Figure (I.13)</i>	Tension et courant de sortie de la commande décalée pour $\theta=\pi/2$	12
<i>Figure (I.14)</i>	L'analyse spectrale de la tension pour la commande décalée pour $\theta=\pi/2$	12
<i>Figure (I.15)</i>	Tension et courant de sortie de la commande décalée pour $\theta=\pi/12$	13
<i>Figure (I.16)</i>	L'analyse spectrale de la tension pour la commande décalée pour $\theta=\pi/12$	13
<i>Figure (I.17)</i>	Variation du fondamental et des harmoniques en fonction de « θ »	14
<i>Figure (I.18)</i>	principe de La MLI bipolaire	16
<i>Figure (I.19)</i>	Tension et courant de sortie pour la MLI bipolaire (m=21, r=0.9)	16
<i>Figure (I.20)</i>	L'analyse spectrale de la tension pour la MLI bipolaire (m=21, r=0.9)	17
<i>Figure (I.21)</i>	Tension et courant de sortie pour la MLI bipolaire (m=42, r=0.9)	17
<i>Figure (I.22)</i>	L'analyse spectrale de la tension pour la MLI bipolaire (m=42, r=0.9)	17
<i>Figure (I.23)</i>	Tension et courant de sortie pour la MLI bipolaire (m=21, r=0.4)	18
<i>Figure (I.24)</i>	L'analyse spectrale de la tension pour la MLI bipolaire (m=21, r=0.4)	18
<i>Figure (I.25)</i>	Tension et courant de sortie pour la MLI bipolaire (m=21, r=0.7)	18
<i>Figure (I.26)</i>	L'analyse spectrale de la tension pour la MLI bipolaire (m=21, r=0.7)	19

LISTES DES FIGURES

<i>Figure (I.27)</i>	L'évolution du fondamentale et du THD en fonction de « r » pour la MLI bipolaire	19
<i>Figure (I.28)</i>	L'évolution du fondamentale et du THD en fonction de « m » pour la MLI bipolaire	19
<i>Figure (I.29)</i>	Principe de la Commande unipolaire avec deux références	20
<i>Figure (I.30)</i>	Tension et courant de sortie pour la MLI unipolaire avec deux références (m=21, r=0.9)	21
<i>Figure (I.31)</i>	L'analyse spectrale de la tension pour la MLI unipolaire avec deux références (m=21, r=0.9)	22
<i>Figure (I.32)</i>	Tension et courant de sortie pour la MLI unipolaire avec deux références (m=42, r=0.9)	22
<i>Figure (I.33)</i>	L'analyse spectrale de la tension pour la MLI unipolaire avec deux références (m=42, r=0.9)	22
<i>Figure (I.34)</i>	Tension et courant de sortie pour la MLI unipolaire avec deux références (m=21, r=0.4)	23
<i>Figure (I.35)</i>	L'analyse spectrale de la tension pour la MLI unipolaire avec deux références (m=21, r=0.4)	23
<i>Figure (I.36)</i>	Tension et courant de sortie pour la MLI unipolaire avec deux références (m=21, r=0.7)	23
<i>Figure (I.37)</i>	L'analyse spectrale de la tension pour la MLI unipolaire avec deux références (m=21, r=0.7)	24
<i>Figure (I.38)</i>	l'évolution du fondamentale et du THD en fonction de « r » pour la MLI unipolaire avec deux références	24
<i>Figure (I.39)</i>	l'évolution du fondamentale et du THD en fonction de « m » pour la MLI unipolaire avec deux références	24
<i>Figure (I.40)</i>	Principe de la Commande unipolaire avec deux porteuses	25
<i>Figure (I.41)</i>	Tension et courant de sortie pour la MLI unipolaire avec deux porteuses (m=21, r=0.9)	26
<i>Figure (I.42)</i>	L'analyse spectrale de la tension pour la MLI unipolaire avec deux porteuses (m=21, r=0.9)	27
<i>Figure (I.43)</i>	Tension et courant de sortie pour la MLI unipolaire avec deux porteuses (m=42, r=0.9)	27
<i>Figure (I.44)</i>	L'analyse spectrale de la tension pour la MLI unipolaire avec deux porteuses (m=42, r=0.9)	27
<i>Figure (I.45)</i>	Tension et courant de sortie pour la MLI unipolaire avec deux porteuses (m=21, r=0.4)	28
<i>Figure (I.46)</i>	L'analyse spectrale de la tension pour la MLI unipolaire avec deux porteuses (m=21, r=0.4)	28
<i>Figure (I.47)</i>	Tension et courant de sortie pour la MLI unipolaire avec deux porteuses (m=21, r=0.7)	28
<i>Figure (I.48)</i>	L'analyse spectrale de la tension pour la MLI unipolaire avec deux porteuses (m=21, r=0.7)	29
<i>Figure (I.49)</i>	L'évolution du fondamentale et du THD en fonction de « r » pour la MLI unipolaire avec deux porteuses	29

LISTES DES FIGURES

<i>Figure (I.50)</i>	L'évolution du fondamentale et du THD en fonction de « m » pour la MLI unipolaire avec deux porteuses	29
<i>Figure (I.51)</i>	Modulation hystérésis	31
<i>Figure (I.52)</i>	Principe de la commande par Hystérésis	31
<i>Figure (I.53)</i>	Tension et courant de sortie pour la modulation à hystérésis (h=0.05)	32
<i>Figure (I.54)</i>	L'analyse spectrale du courant de sortie pour la modulation à hystérésis (h=0.05)	32
<i>Figure (I.55)</i>	Tension et courant de sortie pour la modulation à hystérésis (h=0.07)	33
<i>Figure (I.56)</i>	L'analyse spectrale du courant de sortie pour la modulation à hystérésis (h=0.07)	33
<i>Figure (I.57)</i>	Tension et courant de sortie pour la modulation à hystérésis (h=0.1)	33
<i>Figure (I.58)</i>	L'analyse spectrale du courant de sortie pour la modulation à hystérésis (h=0.1)	34
<i>Figure (II.1)</i>	Premier modèle de la carte Arduino	36
<i>Figure (II.2)</i>	Description de la carte Arduino-UNO	38
<i>Figure (II.3)</i>	l'interface de l'Arduino IDE	42
<i>Figure (II.4)</i>	Structure d'un projet Arduino IDE	43
<i>Figure (II.5.a)</i>	Schéma bloc du module L298N	46
<i>Figure (II.5.b)</i>	Brochage du module L298N	46
<i>Figure (II.6.a)</i>	Structure interne d'un demi-pont du module BTS7960	49
<i>Figure (II.6.b)</i>	Brochage du module BTS7960	49
<i>Figure (II.7)</i>	Convertisseur analogique numérique ADS1115	50
<i>Figure (II.8)</i>	Raccordement de l'ADS1115 avec ARDUINO	52
<i>Figure (II.9)</i>	Capteur de courant et de tension INA219	54
<i>Figure (II.10)</i>	Raccordement de l'INA219 avec ARDUINO	55
<i>Figure (III.1)</i>	schéma de la commande de l'onduleur monophasé	58
<i>Figure (III.2)</i>	Photographie du montage réalisé	58
<i>Figure (III.3)</i>	Tension de sortie correspondante à $V_{in} = 5V$	59
<i>Figure (III.4)</i>	Tension de sortie correspondante à $V_{in} = 25V$	59
<i>Figure (III.5)</i>	Tension de sortie correspondante à $V_{in} = 5V$	59
<i>Figure (II.6)</i>	Tension de sortie correspondante à $V_{in} = 25V$	59
<i>Figure (III.7)</i>	Tension de sortie correspondante à $V_{in} = 5V$	60
<i>Figure (III.8)</i>	Tension de sortie correspondante à $V_{in} = 25V$	60

LISTES DES FIGURES

<i>Figure (III.9)</i>	Tension de sortie correspondante à $V_{in} = 5V$	61
<i>Figure (III.10)</i>	Tension de sortie correspondante à $V_{in} = 25V$	61
<i>Figure (III.11)</i>	Tension de sortie correspondante à $V_{in} = 5V$	61
<i>Figure (III.12)</i>	Tension de sortie correspondante à $V_{in} = 25V$	62
<i>Figure (III.13)</i>	Tension de sortie correspondante à $V_{in} = 5V$	62
<i>Figure (III.14)</i>	Tension de sortie correspondante à $V_{in} = 25V$	62
<i>Figure (III.15)</i>	Tension de sortie correspondante à $V_{in} = 5V$	63
<i>Figure (III.16)</i>	Tension de sortie correspondante à $V_{in} = 25V$	63
<i>Figure (III.17)</i>	Tension de sortie correspondante à $V_{in} = 5V$	63
<i>Figure (III.18)</i>	Tension de sortie correspondante à $V_{in} = 25V$	63
<i>Figure (III.19)</i>	Tension de sortie correspondante à $V_{in} = 5V$	64
<i>Figure (III.20)</i>	Tension de sortie correspondante à $V_{in} = 25V$	64
<i>Figure (III.21)</i>	Tension de sortie correspondante à $V_{in} = 5V$	65
<i>Figure (III.22)</i>	Tension de sortie correspondante à $V_{in} = 25V$	65

LISTES DES TABLEAUX

N°: Tableau	Intitulé	page
<i>Tableau (I.1)</i>	Les cas de commutation des interrupteurs et la tension de sortie de la commande pleine onde	10
<i>Tableau (I.2)</i>	les cas de commutation des interrupteurs et la tension de sortie de la commande décalée	11
<i>Tableau (I.3)</i>	l'analyse spectrale du fondamental et des harmoniques pour différents valeur de « θ »	13
<i>Tableau (I.4)</i>	Evolution du fondamental et du THD en fonction de « r » pour la MLI bipolaire	16
<i>Tableau (I.5)</i>	Evolution du fondamental et du THD en fonction de « m » pour la MLI bipolaire	16
<i>Tableau (I.6)</i>	Evolution du fondamentale et du THD en fonction de « r » pour la MLI unipolaire avec deux références	21
<i>Tableau (I.7)</i>	Evolution du fondamentale et du THD en fonction de « m » pour la MLI unipolaire avec deux références	21
<i>Tableau (I.8)</i>	Evolution du fondamentale et du THD en fonction de « r » pour la MLI unipolaire avec deux porteuses	26
<i>Tableau (I.9)</i>	Evolution du fondamentale et du THD en fonction de « m » pour la MLI unipolaire avec deux porteuses	26
<i>Tableau (II.1)</i>	Critère de choix de la carte Arduino	37
<i>Tableau (II.2)</i>	Caractéristique de la carte Arduino-UNO	41
<i>Tableau (II.3)</i>	Liste des raccourcis sur l'écran de l'Arduino IDE	43
<i>Tableau (II.4)</i>	Liste des instructions usuelles sur l'Arduino IDE	44
<i>Tableau (II.5)</i>	Instructions d'exploitation du module ADS1115	53
<i>Tableau (II.6)</i>	Instructions d'exploitation du module INA219	56

NOTATIONS

U_c	La tension aux bornes de la charge
I_c	Le courant de la charge
E	La tension de la source continue
IGBT	Insolated gate bipolar transistor
MOSFET	Transistor à effet de champs
S_p	Signal de la porteuse
S_{cst}	Signal constant
S_{var}	Signal variable
U_{cmoy}	Valeur moyenne de tension aux bornes de la charge
α	Rapport cyclique
R	Résistance
L	L'inductance
MLI	Modulation de largeur de l'impulsion(PWM)
f	La fréquence
f_p	La fréquence de porteuse
h	La bande d'hystérésis

Introduction Générale

Introduction générale :

Dans le cadre de la conversion d'énergie électrique, l'électronique de puissance occupe une place particulière [1]. Elle a fait des avancées significatives dans la conception et la réalisation de circuits électroniques de plus en plus performants. Les convertisseurs en pont sont parmi ces circuits. Ils sont couramment utilisés dans des applications telles que la commande de moteurs électriques, les onduleurs solaires, les alimentations électriques à découpage, etc.

Les progrès accomplis dans le domaine de la microinformatique ont permis de réaliser des commandes complexes impossible en analogique avec une grande robustesse [2][3]. En effet, au lieu d'être implémentés sur des portes logiques et des amplificateurs opérationnels, les commandes sont implémentées à travers des programmes sauvegardés et exécutés sur des espaces mémoires, puis communiqués aux systèmes réels à travers des interfaces entrées sorties.

La carte Arduino est développée dans l'objectif de simplifier l'implémentation des programmes de commande, et de fournir une solution intégrale, à un coût faible, pour l'implémentation en pratique des différentes techniques de commandes.

Dans ce travail, on s'intéresse à l'implémentation de la commande de l'onduleur monophasé en pont complet en utilisant une carte Arduino Uno. Afin de bien situer notre travail, nous l'avons structuré suivant trois chapitres :

Le premier chapitre est consacré à une présentation de la constitution élémentaire, le principe de fonctionnement et les différentes classifications de l'onduleur monophasé. Ensuite quelques méthodes de sa commande et leurs simulations avec MATLAB/Simulink.

Dans le deuxième chapitre, nous allons projeter la lumière sur la carte Arduino, nous allons citer les différents types de cette dernière. On va présenter les deux aspects liés à la carte ; la constitution matérielle et la programmation de la carte. Nous allons également donner une description de deux modules utilisés en tant qu'onduleur monophasé en pont complet, le module L298N et le module BTS7960. Ces deux modules servent essentiellement au contrôle des machines à courant continu de faible puissance. Ils sont bâtis autour des pont H. On va les exploiter pour avoir un onduleur monophasé en pont. En fin du chapitre on va présenter les éléments de mesures du courant et de la

INTRODUCTION GENERALE

tension, qui sont requis pour la commande de l'onduleur et la visualisation des signaux de sortie résultants de cette commande.

Le dernier chapitre se concentre sur la mise en œuvre pratique des différentes stratégies de commandes de l'onduleur en pont complet sur la carte Arduino Uno. On va utiliser les deux modules L298N et BTS7960 en tant qu'onduleur monophasé. Nous allons comparer les résultats expérimentaux avec celles de la simulation.

Enfin, nous allons terminer notre travail par une conclusion générale et des perspectives du travail.

Chapitre I



Onduleur Et Sa Commande



I.1 Introduction

L'onduleur est un dispositif important dans le domaine de l'électronique de puissance. C'est l'élément principal dans la conversion de la puissance électrique. Il est indispensable pour la commande et l'alimentation des moteurs à courant alternatif. Dans ce chapitre, nous donnons une introduction générale aux onduleurs et leurs stratégies de commande, en particulier l'onduleur de tension monophasé.

I.2 Définition de l'onduleur

Un onduleur est un dispositif permettant de transformer en alternatif une énergie électrique de type continu (Figure I.1) [4]. C'est un convertisseur statique assurant la conversion continue – alternative. Alimenté en continu, il modifie de façon périodique les connexions entre l'entrée et la sortie pour permettre d'obtenir l'alternatif à la sortie [5].

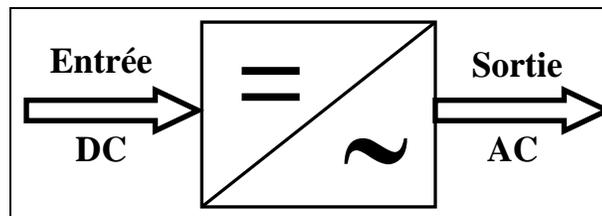


Figure (I.1) : Structure Générale de l'onduleur

La plupart des applications des onduleurs se résument aux : Commande des moteurs alternatifs, filtrage actif, systèmes d'alimentations sans interruption (ASI) utilisés par exemple pour réaliser des alimentations de sécurité, on s'efforce alors de maintenir la tension de sortie constante quelle que soit la charge [6].

I.3 Classification des onduleurs

Les onduleurs peuvent être classés selon différents critères.

I.3.1 Classification selon la nature de l'alimentation

I.3.1.1 Onduleur de tension

On dit onduleur de tension, un onduleur qui est alimenté par une source de tension continue (Figure I.3), La nature de la source continue impose la nature du récepteur alternatif. Puisque la plupart des applications électriques actuelles fonctionnent en courant alternatif, l'utilisation d'un onduleur de tension est alors nécessaire [7].

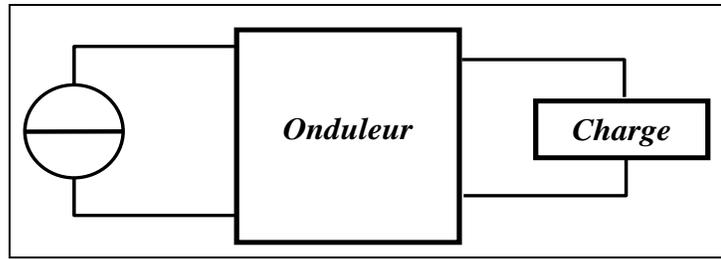


Figure (I.2) : onduleur de tension

I.3.1.2 Onduleur du courant

L'onduleur de courant permet le transfert de puissance entre une source de courant continu et une source de tension alternative (Figure I.3), monophasé ou triphasé, en imposant la loi d'évolution du courant alternatif [8].

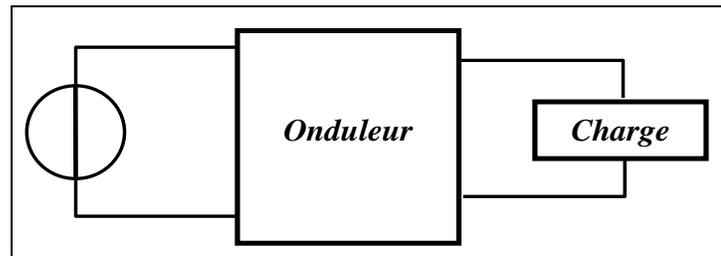


Figure (I.3) : onduleur du courant

I.3.2 Classification selon la nature de la commande

I.3.2.1 Onduleur autonome

Un onduleur autonome est un système de commutation à transistors ou à thyristors, dont les instants de commutation sont imposés par des circuits externes. La fréquence et la forme d'onde de la tension de sortie sont imposées à la charge, c'est-à-dire elles sont propres à l'onduleur. La tension de sortie de l'onduleur autonome peut être [9] :

- ✓ Soit imposé (fixe ou réglable), par une commande électronique autonome
- ✓ Soit imposée constant par la source elle-même (onduleur assisté débité sur le réseau).
- ✓ Soit imposée variable par la source elle-même (onduleur alimentant une machine synchrone, piloté par la fréquence de la f.é.m. de la machine) [8].

I.3.2.2 Onduleur non autonome

Un onduleur non autonome ou assisté est un système de commutation à thyristors, dont les instants de commutation sont imposés par la charge [9]. Dans ce cas, la fréquence et la forme d'onde de la tension de sortie sont imposées par le réseau alternatif sur lequel débite l'onduleur. Ce type d'onduleur est utilisé pour :

- Le transport d'énergie en courant continu, on peut transporter de grandes puissances à très haute tension (de l'ordre du million de volts) par lignes aériennes sur des distances élevées (500-1000 Km) pour lesquelles, les lignes à courant alternatif posent des problèmes délicats (stabilité, pertes...). De plus, le transport sur des plus faibles distances par câbles souterrains dont la capacité très élevée, limite leur emploi en courant alternatif [10].
- Le freinage par récupération : pendant le freinage de la machine à courant continu qui peut fonctionner en génératrice, l'énergie fournie dans ce cas peut être récupérée et transférée au réseau alternatif par un onduleur non autonome (domaine de la traction électrique) [11].

I.3.3 Classification selon la nature de la charge

I.3.3.1 Onduleur monophasé

Deux classes d'onduleurs monophasés sont à distinguer, suivant leur topologie, onduleur monophasé en demi-pont et onduleur monophasé en pont-complet.

- **Onduleur monophasé en demi-pont**

Il est constitué principalement d'un seul bras qui se compose de deux interrupteurs de puissance notés Q_1 et Q_2 à commande complémentaire (Figure I.4).

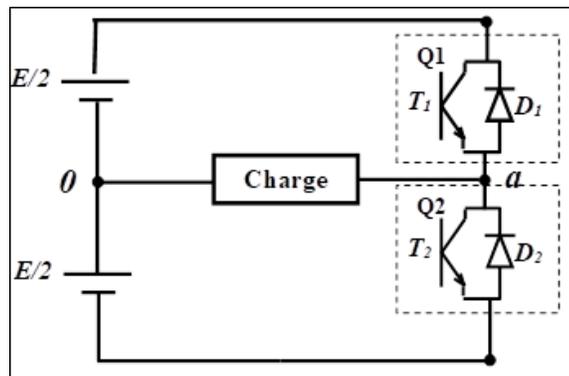


Figure (I.4) : Structure principale d'un onduleur monophasé en demi-pont

- **Onduleur monophasé en pont complet**

L'onduleur de tension monophasé le plus utilisé, car n'incluant pas dans son principe de transformateur ou de diviseur capacitif, est l'onduleur en pont complet (Figure I.5). Il est formé par quatre « interrupteurs » réversibles en courant, commandables à la fermeture et à l'ouverture. On remarque qu'il a la même structure que l'hacheur quatre quadrants ; il doit en effet assurer les mêmes réversibilités [12].

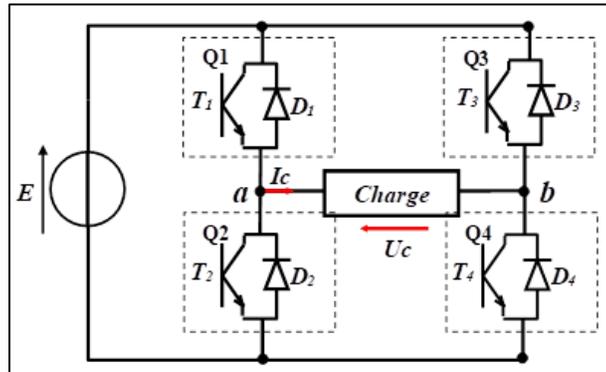


Figure (I.5) : Structure d'un onduleur monophasé en pont complet

Dans le schéma de la figure I.5 :

T_1, T_2, T_3, T_4 : Transistors de puissance.

D_1, D_2, D_3, D_4 : Diodes de puissance.

Q_1, Q_2, Q_3, Q_4 : Transistors avec une diode antiparallèle.

E : Tension continue.

U_c, I_c : La tension et le courant de la charge.

I.3.3.2 Onduleur triphasé

On peut réaliser un onduleur triphasé en regroupant, en parallèle, trois onduleurs monophasés en demi-pont et commander les interrupteurs de chacun d'eux pour obtenir à la sortie trois phases décalées de 120° . En fait, en regroupant trois demi ponts monophasés, on obtient un onduleur en pont triphasé à six interrupteurs représenté par la Figure I.6 dont les interrupteurs du même bras de l'onduleur doivent être complémentaires pour que la tension continue E ne soit jamais en court-circuit et pour que les circuits des courants I_{c1} , I_{c2} et I_{c3} ne soient jamais ouverts. Afin que les interrupteurs puissent imposer les tensions de sortie, quelques soient les courants de charge, il faut que ces derniers soient bidirectionnels en courant. Chacun d'eux est formé d'un semi-conducteur à ouverture et à fermeture commandées et d'une diode montée en antiparallèle. Ces interrupteurs peuvent être

réalisées, suivant la puissance à contrôler, avec des transistors **MOS** ou bipolaires, des **IGBTs** ou des **GTOS** associés à une diode en antiparallèle pour obtenir la réversibilité en courant [13].

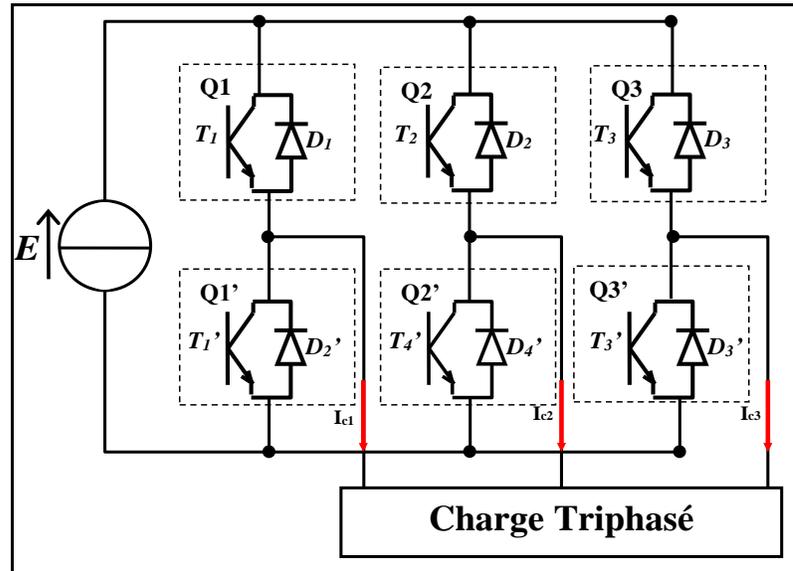


Figure (I.6) : Structure d'un onduleur triphasé

I.3.3.3 Autres structures des onduleurs

Des nouvelles topologies des convertisseurs ont été développées, tels que les onduleurs multi-niveaux, les onduleurs multi-phases et les onduleurs matricielles.

Dans la suite de notre travail, on se concentre à la commande de l'onduleur monophasé en pont complet. Ce choix est fait vu la possibilité d'utiliser des modules de commandes des machines à courant, construits à la base d'un pont complet, en tant qu'onduleurs, et par conséquent d'avoir la possibilité d'appliquer les différentes techniques de commande sur ces modules.

✓ Remarque

Y a d'autres critères de classification des onduleurs, comme la fréquence de commutation, ainsi que la puissance transmettre.

I.4 Stratégies de commande de l'onduleur monophasé

Nous allons étudier et simuler les types de commandes suivants :

- ✓ Les commandes à rapport cyclique fixe (pleine onde, décalé), où le rapport cyclique de chacune des cellules de commutation est maintenu constant.

- ✓ Les commandes à modulations de largeur d’impulsion (MLI), où le rapport cyclique est variable, sinusoïdalement pour un grand nombre d’applications.
- ✓ La Commande en courant par hystérésis.

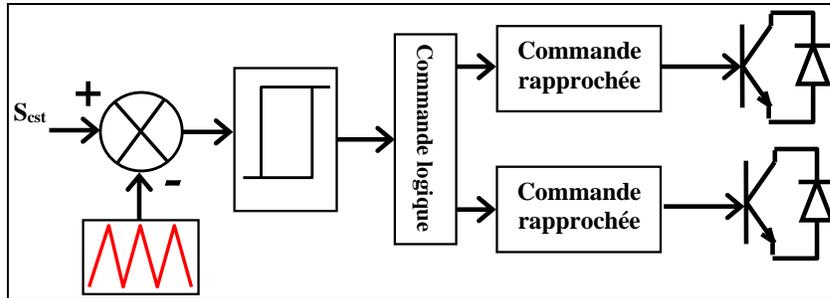


Figure (I.7) : Principe de commande du bras de l’onduleur

Le principe de commande d’un onduleur monophasé est présenté sur la Figure I.7 valable pour un bras de commutation [14].

Le principe de la commande est de comparer un signal de référence à un signal de type « triangle » appelé la porteuse. Le résultat de cette comparaison crée un signal de commande pour les gâchettes des interrupteurs. Pour obtenir la commande de deux bras, on doit avoir deux comparaisons. La commande décrite ici est sous sa forme la plus complexe, dans la mesure où les deux bras de commutations de convertisseur peuvent être commandés de manière indépendante. Quand bien même la porteuse est identique pour les deux bras, deux signaux de commandes indépendants sont générés pour commander les deux bras [14].

I.4.1 Commande en pleine onde symétrique

La commande en pleine onde consiste à commander chacun des deux bras de l’onduleur en fermeture pendant une demi-période de commutation, et en ouverture pendant l’autre demi-période. Le rapport cyclique d’un interrupteur étant le rapport entre la durée de fermeture et la période, la commande en pleine onde correspond alors à un rapport cyclique $\alpha=0.5$.

En se référant à la Figure I.5, la commande en pleine onde symétrique consiste à commander les interrupteurs Q_1 et Q_3 d’une manière complémentaire, et de commander les interrupteurs Q_2 et Q_4 d’une manière complémentaire.

Notant aussi que pour éviter le court-circuit de la charge et le circuit ouvert de la source, on doit commander les interrupteurs Q_1 et Q_2 d'une manière complémentaire, et de même que les interrupteurs Q_3 et Q_4 .

Les commandes sont déduites de la comparaison entre le signal constant S_{cst} et de porteuse S_p (Figure I.8.a), Le résultat de cette comparaison donne un signal de commande S_{cm} (Figure (I.8.b)).

Tout croisement des signaux de S_{cst} avec la porteuse S_p se traduit par un changement de l'état des interrupteurs du bras de commutation.

Cela donne deux configurations possibles pour l'onduleur (Figure I.9) :

$S_{cst} \geq S_p$ l'interrupteur Q_1 et Q_4 amorcés, Q_2 et Q_3 bloqués.

$S_{cst} \leq S_p$ l'interrupteur Q_2 et Q_3 amorcés, Q_1 et Q_4 bloqués.

En appliquant cette commande, la tension de la charge U_c a pour valeur moyenne :

$$U_{cmoy} = T [E\alpha T - E (1 - \alpha) T] = (2\alpha - 1) E$$

Avec α : rapport cyclique, qui détermine donc, la valeur moyenne de la tension U_c .

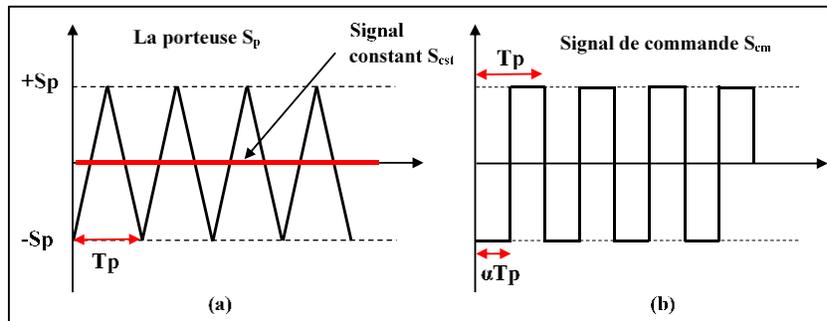
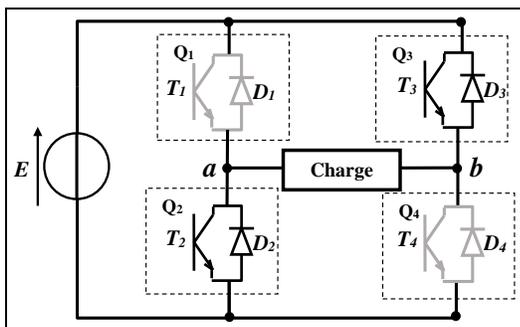
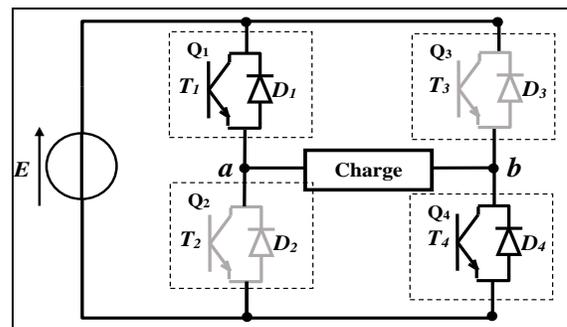


Figure (I.8) : Signal de commande de la commande pleine onde



Cas 1 : Accumulation alternance négative.



Cas 2 : Accumulation alternance positive.

Figure (I.9) : Configurations possibles de l'onduleur commandé en pleine onde

Le tableau suivant montre les cas possibles de commutation des interrupteurs de la commande symétrique :

	Bras 1		Bras2		Tension U_c
	Q_1	Q_2	Q_3	Q_4	
Cas 1	Passent	Bloqué	Bloqué	Passent	$U_c = E$
Cas 2	Bloqué	Passent	Passent	Bloqué	$U_c = -E$

Tableau (I.1) : Les cas de commutation des interrupteurs et la tension de sortie de la commande pleine onde

À l'aide de logiciel de simulation *MATLAB- Simulink*, on trace l'allure de la tension et du courant de sortie, ainsi que l'analyse spectrale de la tension de sortie par l'application de la commande pleine onde comme montre sur les figures I.10 et I.11.

Remarques :

- ✓ La figure (I.10) montre que la tension U_c bascule entre deux valeur (+E et -E), ainsi que le courant de sortie augmente et diminue selon la tension d'une façon exponentielle à cause de l'effet de l'inductance.
- ✓ La figure (I.11) montre qu'en plus du fondamentale, la tension U_c est composée des harmoniques impaires : la troisième, la cinquième, la septième...etc.

I.4.2 Commande « Décalée »

En se référant à la figure I.5, par rapport à la commande symétrique, la commande décalée, consiste à décaler d'un certain angle θ , la commande de l'interrupteur Q_4 par rapport à la commande de l'interrupteur Q_1 , et de décaler du même angle θ , la commande de l'interrupteur Q_3 par rapport à la commande de l'interrupteur Q_2 .

Un tel algorithme permet d'imposer trois niveaux de tension, à savoir $U_c = E$, $U_c = -E$ ainsi que $U_c = 0$, ce dernier niveau de tension correspond à une phase de roue libre, sans échange d'énergie entre les deux sources. Suivant le type de commande adopté, ce troisième niveau peut être ou non mis à contribution. Le tableau I.2 donne les combinaisons possibles des états des interrupteurs los de la commande décalée.

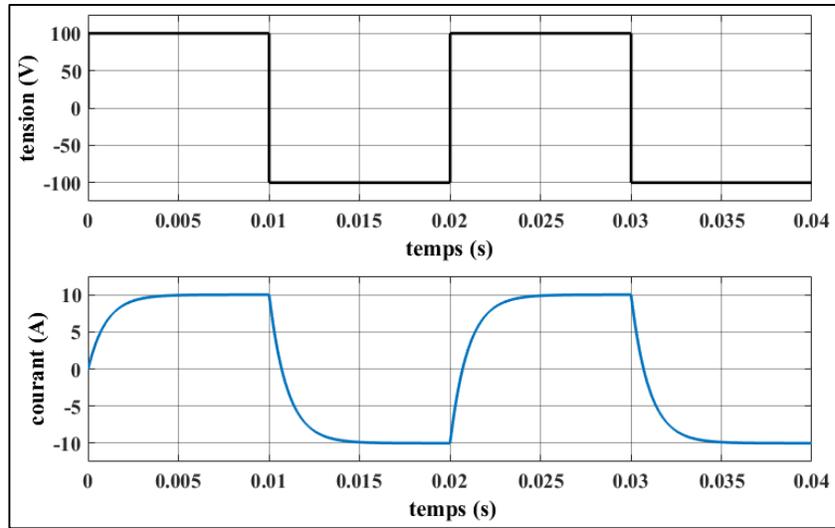


Figure (I.10) : Tension et courant de sortie pour la commande pleine onde

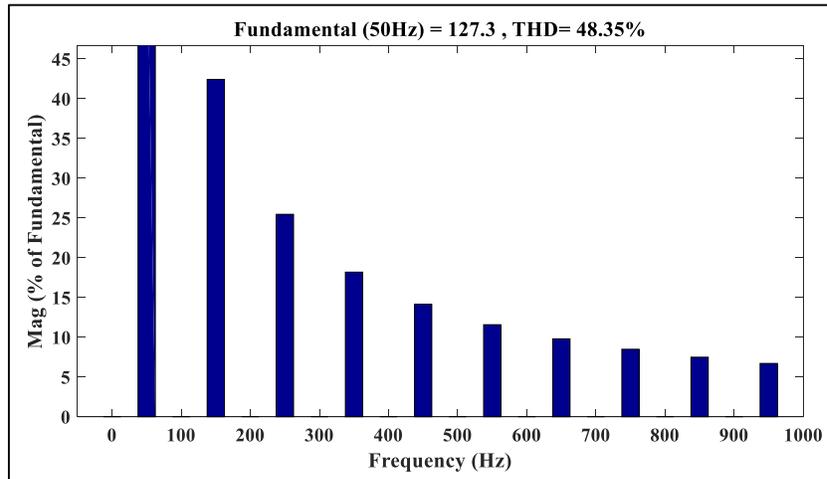


Figure (I.11) : L'analyse spectrale de la tension pour la commande pleine onde

	Q_1	Q_2	Q_3	Q_4	Tension U_c
Cas 1	Passant	Bloqué	Passant	Bloqué	0
Cas 2	Passant	Passant	Bloqué	Passant	E
Cas 3	Bloqué	Passant	Passant	Bloqué	-E
Cas 4	Bloqué	Passant	Bloqué	Passant	0

Tableau (I.2) : Les cas de commutation des interrupteurs et la tension de sortie pour la commande décalée

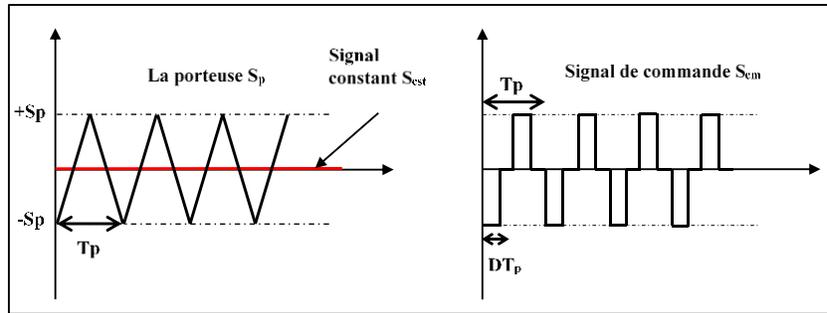


Figure (I.12) : Signal de commande de la Commande décalée

À l'aide de logiciel de simulation *MATLAB (Simulink)*, on trace l'allure de la tension et du courant de sortie, ainsi que l'analyse spectrale de la tension de sortie par l'application de la commande décalée pour différentes valeurs de l'angle de décalage θ comme présenter dans les figures I.13, I.14, I.15 et I.16.

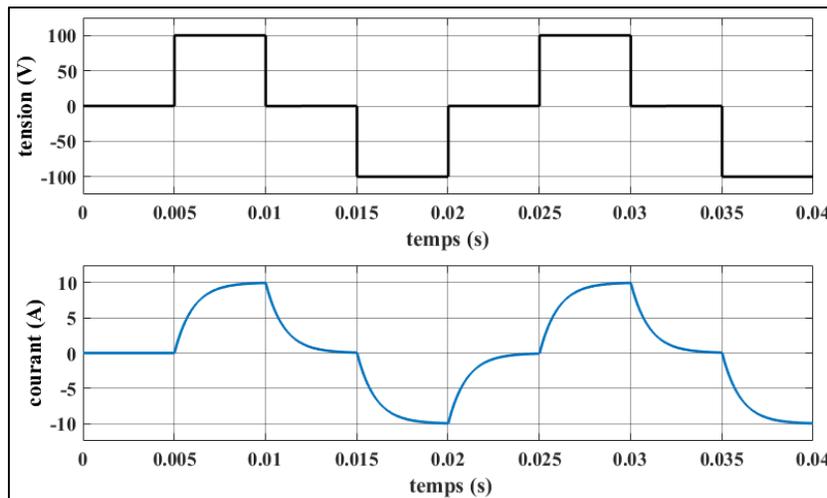


Figure (I.13) : Tension et courant de sortie de la commande décalée pour $\theta=\pi/2$

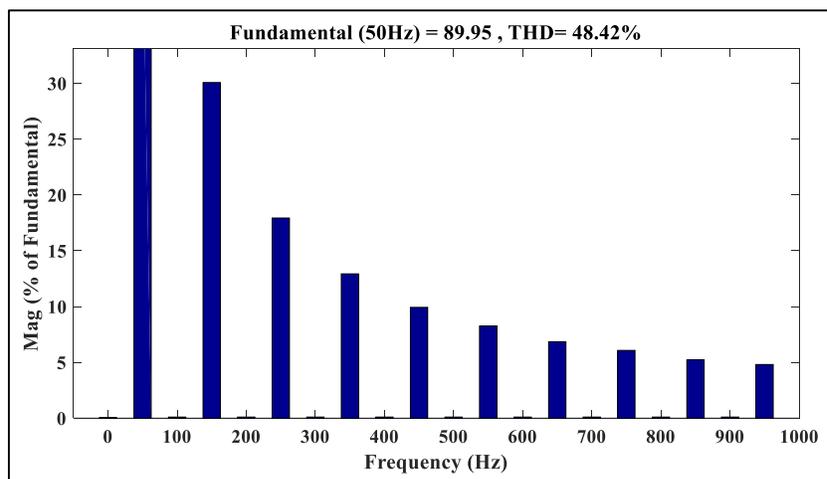


Figure (I.14) : L'analyse spectrale de la tension pour la commande décalée pour $\theta=\pi/2$

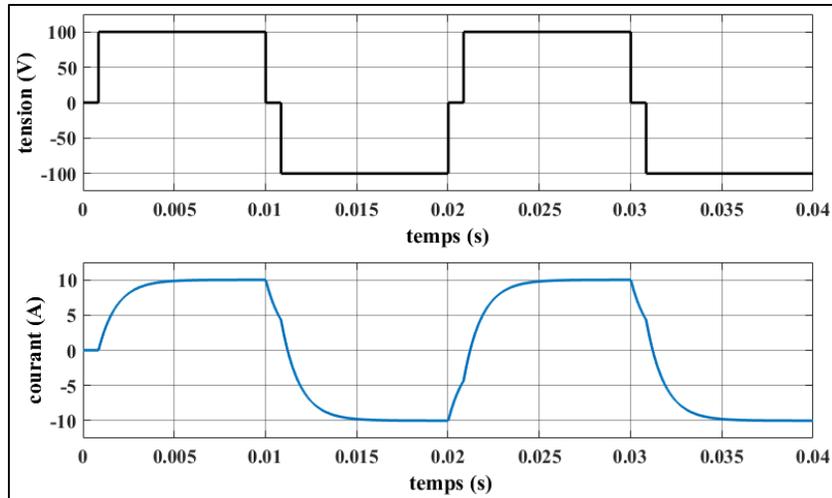


Figure (I.15) : Tension et de courant de sortie pour la commande décalée pour $\theta=\pi/12$

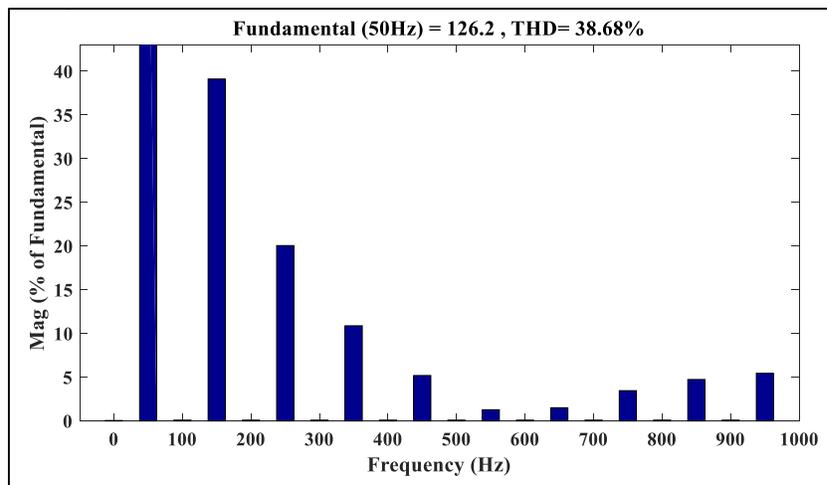


Figure (I.16) : L'analyse spectrale de la tension pour la commande décalée pour $\theta=\pi/12$

L'analyse spectrale pour différentes valeurs de l'angle de décalage « θ » est résumé dans le tableau

I.3 :

θ	Fundamental	THD	H3	H5	H7	H9	H11	H13
15°	126.2	38.68	39.12	20.06	10.89	5.20	1.28	1.51
30°	122.94	31.87	29.89	6.43	4.87	10.12	11.22	9.42
45°	117.59	28.99	16.15	9.84	16.84	13.03	4.34	3.84
60°	110.15	31.15	0.23	22.17	15.63	0.23	10.14	8.36
75°	100.91	37.96	16.40	25.22	2.21	13.13	6.91	6.09
90°	89.96	48.42	30.08	17.94	12.93	9.93	8.25	6.85

Tableau (I.3) : L'analyse spectrale du fondamental et des harmoniques pour différentes valeurs de θ

La Figure I.17 représente la variation du fondamental et des harmoniques en fonction de « θ » :

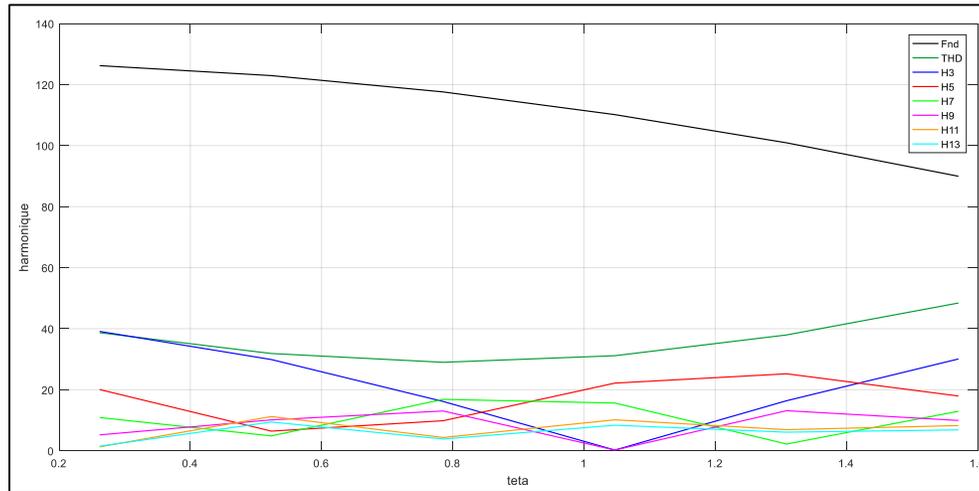


Figure (I.17) : Variation du fondamental et des harmoniques en fonction de « θ »

Remarques :

- ✓ Les figure (I.13) et (I.15) montrent que la tension U_c bascule entre trois niveaux de tension (0 , $+E$ et $-E$).
- ✓ Les figures (I.14) et (I.16) montrent que Les harmoniques contenus dans le spectre sont des harmoniques impairs.
- ✓ Le tableau (I.3) et la figure (I.17) montrent que L'amplitude du fondamental varie d'une façon inversement proportionnelle avec l'angle de décalage θ .
- ✓ Le tableau (I.3) et la figure (I.17) montrent que la variation du THD et des amplitudes des harmoniques en fonction de l'angle est non linéaire.

I.4.3 La commande en MLI triangulo-sinusoidale

La modulation de largeur d'impulsion sinus triangle est réalisée par comparaison d'une onde modulante basse fréquence (tension de référence) à une onde porteuse haute fréquence de forme triangulaire. Les instants de commutation sont déterminés par les points d'intersection entre la porteuse et la modulante, la fréquence de commutation des interrupteurs est fixée par la porteuse [15].

Il existe deux modes de modulation, la modulation unipolaire et la modulation bipolaire. Pour la modulation unipolaire, la tension de sortie ne change pas sa polarité durant toute une demi période : elle bascule entre $+E$ à 0 pendant une demi période, et entre $-E$ à 0 pendant la deuxième demi période.

Pour la modulation bipolaire, la tension de sortie change sa polarité à chaque période de commutation : elle bascule régulièrement entre $+E$ ou $-E$: lorsque le signal de référence est plus grand que la porteuse, les interrupteurs Q_1 et Q_4 (**Figure I.5**) sont actifs, et la tension de sortie est $+E$, et lorsque le signal de référence est plus petit que la porteuse, les interrupteurs Q_3 et Q_2 seront activés, et la tension de sortie est alors $-E$. Les caractéristiques de MLI sont choisies de manière à respecter les normes IEEE 1547 [16], qui imposent certains critères et exigences pour connecter une source électrique au réseau d'énergie électrique.

I.4.3.1 Caractéristiques de la MLI

Les paramètres importants de la commande en MLI sont :

- *Indice de modulation de fréquence*

C'est le rapport entre la fréquence de la porteuse f_p et celle de la modulante f_m :

$$m = \frac{f_p}{f_m}$$

- *Indice de modulation d'amplitude*

C'est le rapport entre l'amplitude de modulante V_0^* et celle de la porteuse V_p :

$$r = \frac{V_0^*}{V_p}$$

I.4.3.2 MLI Bipolaire

La MLI bipolaire fonctionne en comparant une tension de référence sinusoïdale $V_{\text{réf}}$ avec une onde de porteuse S_p de fréquence plus élevée (Figure I.18). La comparaison entre les deux signaux génère des impulsions de largeur variable, qui sont ensuite appliquées aux interrupteurs de l'onduleur.

À l'aide de logiciel de simulation **MATLAB (Simulink)**, on trace l'allure de la tension et du courant de sortie ainsi que l'analyse spectrale de la tension de sortie par l'application de la commande MLI bipolaire pour différents valeur des indices de modulation de fréquence et d'amplitude « m » et « r » comme il montre dans les figures I.19, I.20, I.21, I.22, I.23, I.24, I.25, et I.26.

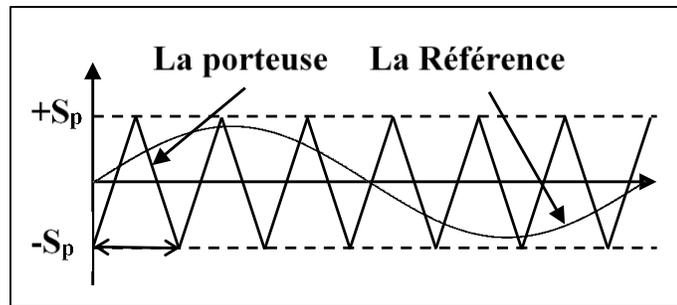


Figure (I.18) : Principe de La MLI bipolaire

Les tableaux I.4 et I.5 représentent l'évolution du fondamentale et du THD en fonction des paramètres « m » et « r » :

✓ **m = 21** :

r	0.2	0.4	0.7	0.9
fondamental	20.28	39.52	69.10	90
THD %	690.21	343.57	178.57	121.19

Tableau (I.4) : Evolution du fondamental et du THD en fonction de l'indice « r » pour la MLI bipolaire

✓ **r = 0.9** :

m	7	14	21	42
fondamental	90.40	90.74	90	89.78
THD %	120.28	119.53	121.19	121.70

Tableau (I.5) : Evolution du fondamental et du THD en fonction de l'indice « m » pour la MLI bipolaire

✓ « m=21 » et « r=0.9 »

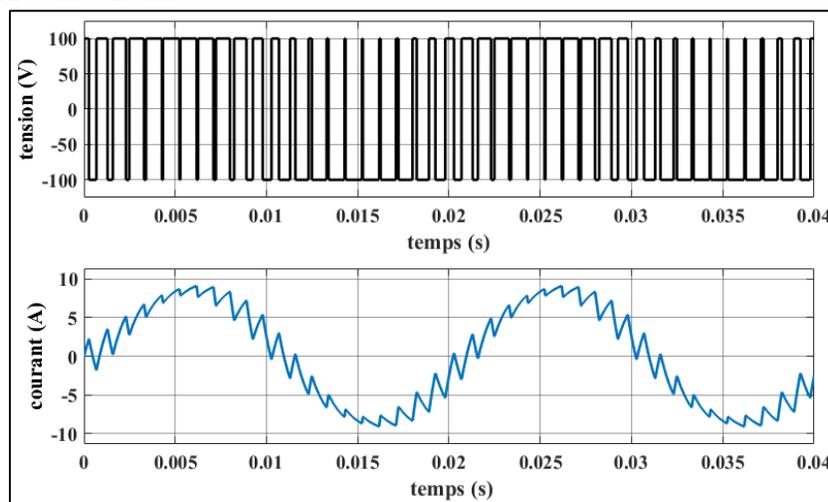


Figure (I.19) : Tension et courant de sortie pour la MLI bipolaire (m=21, r=0.9)

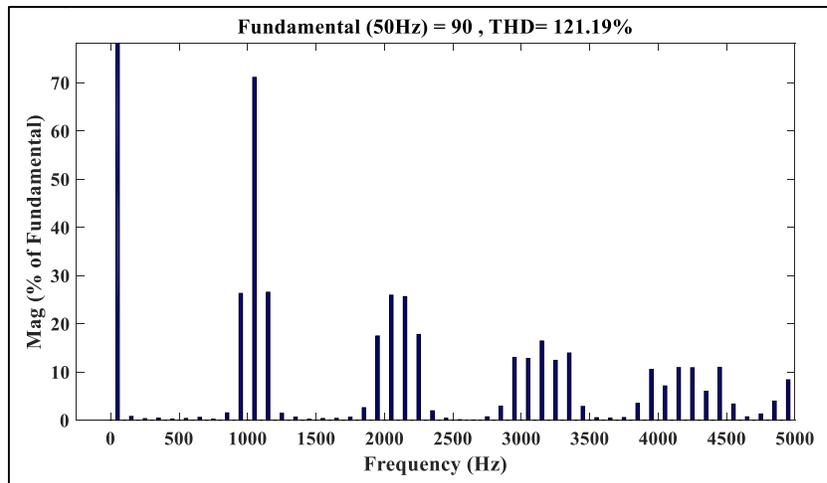


Figure (I.20) : L'analyse spectrale de la tension pour la MLI bipolaire (m=21, r=0.9)

✓ « m=42 » et « r=0.9 »

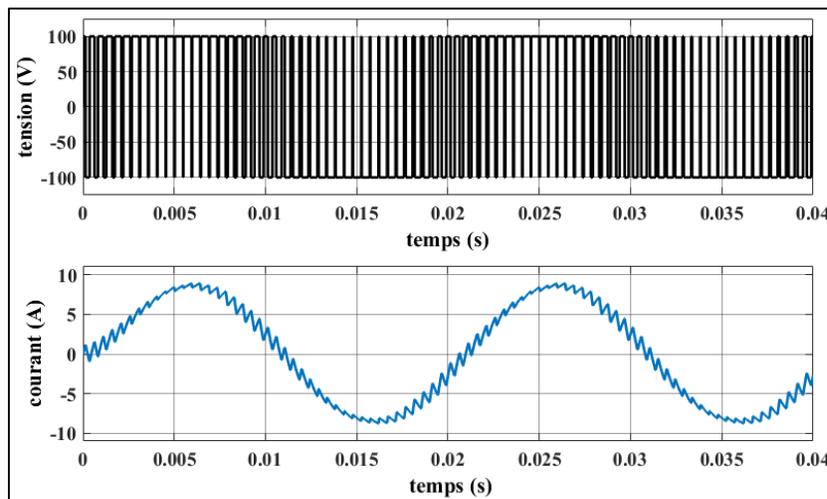


Figure (I.21) : Tension et courant de sortie pour la MLI bipolaire (m=42, r=0.9)

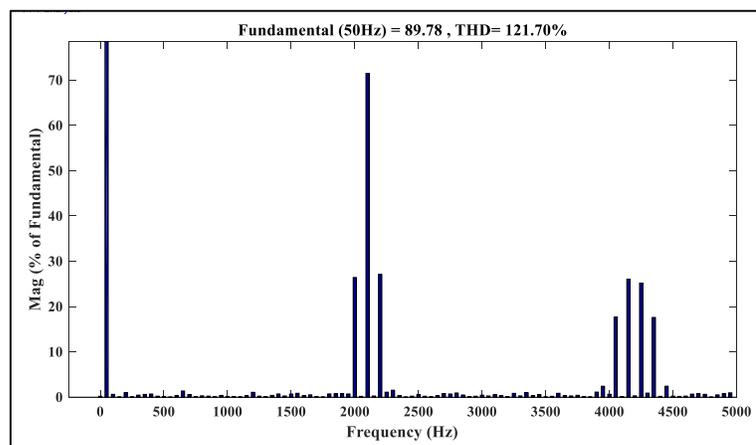


Figure (I.22) : L'analyse spectrale de la tension pour la MLI bipolaire (m=42, r=0.9)

✓ « $m=21$ » et « $r=0.4$ »

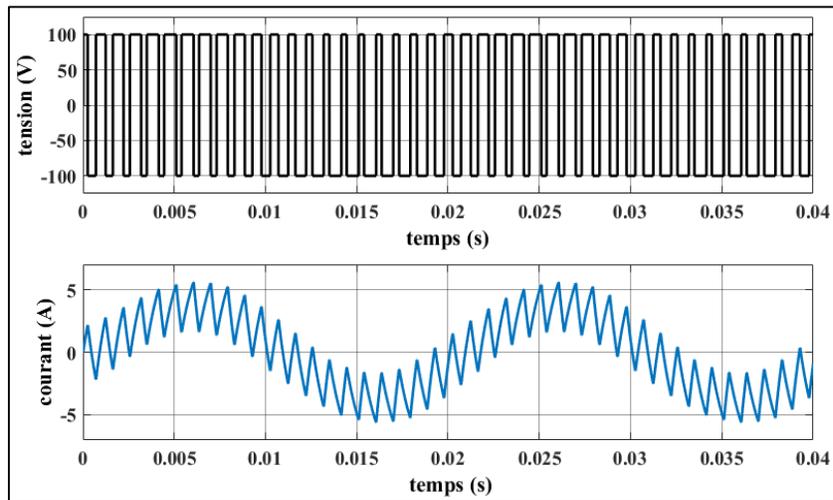


Figure (I.23) : Tension et courant de sortie pour la MLI bipolaire ($m=21$, $r=0.4$)

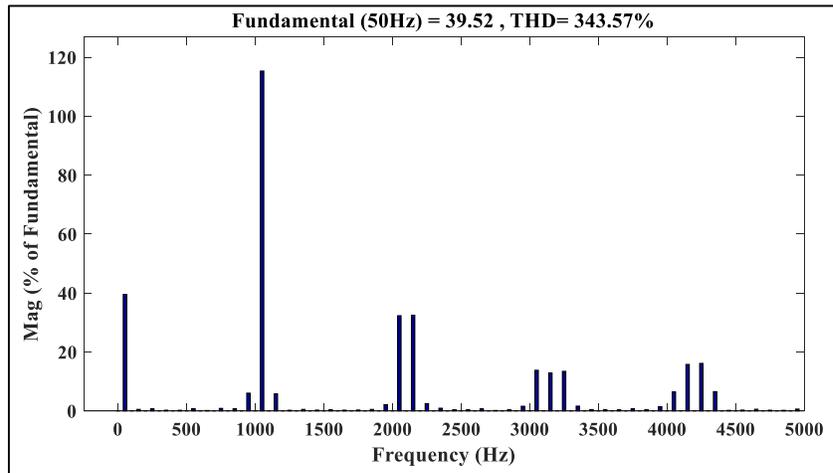


Figure (I.24) : L'analyse spectrale de la tension pour la MLI bipolaire ($m=21$, $r=0.4$)

✓ « $m=21$ » et « $r=0.7$ »

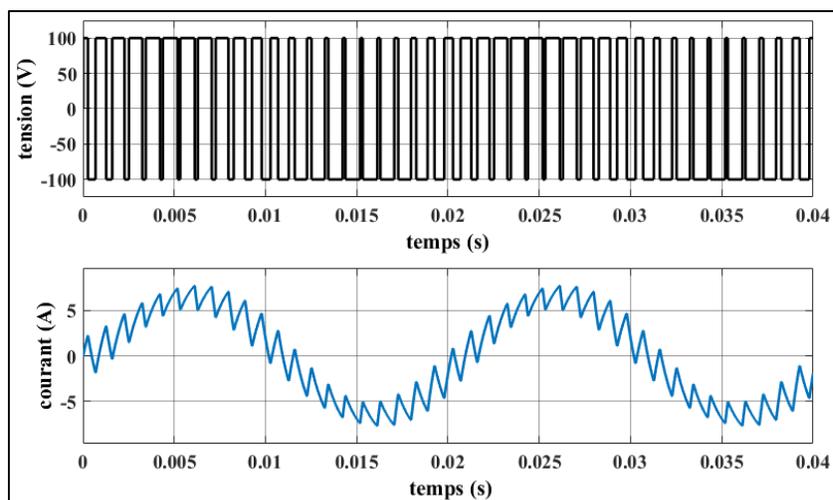


Figure (I.25) : Tension et courant de sortie pour la MLI bipolaire ($m=21$, $r=0.7$)

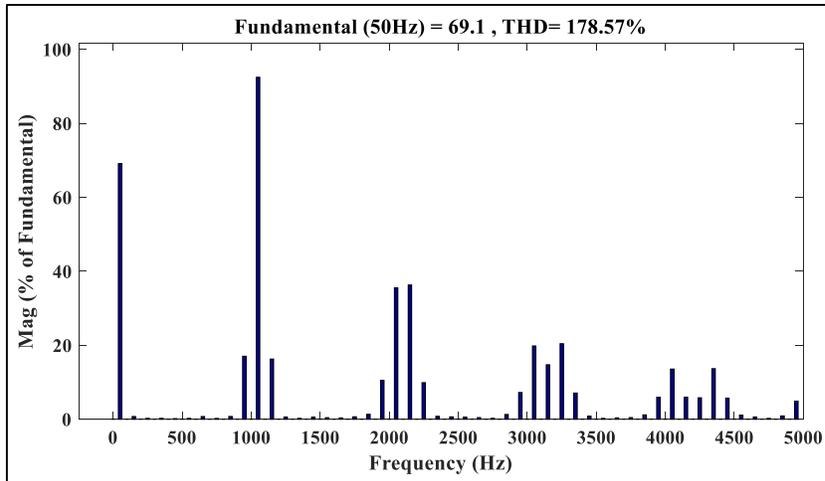


Figure (I.26) : L'analyse spectrale de la tension pour la MLI bipolaire ($m=21, r=0.7$)

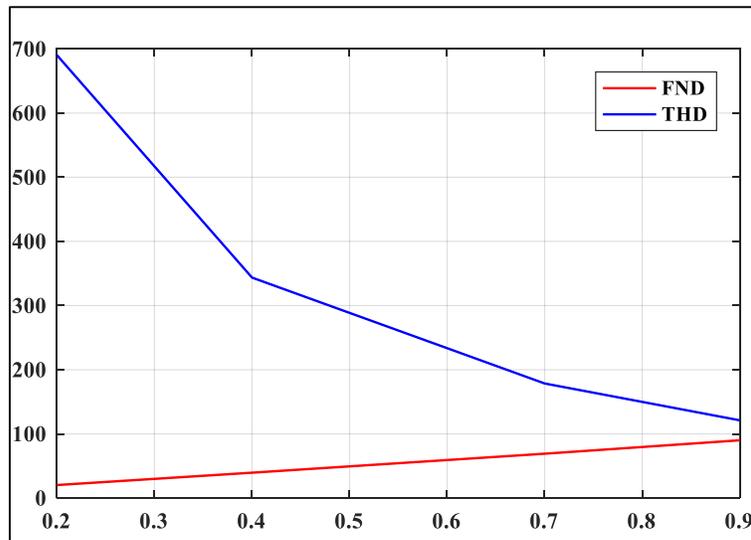


Figure (I.27) : Evolution du fondamentale et du THD en fonction de « r » pour la MLI bipolaire

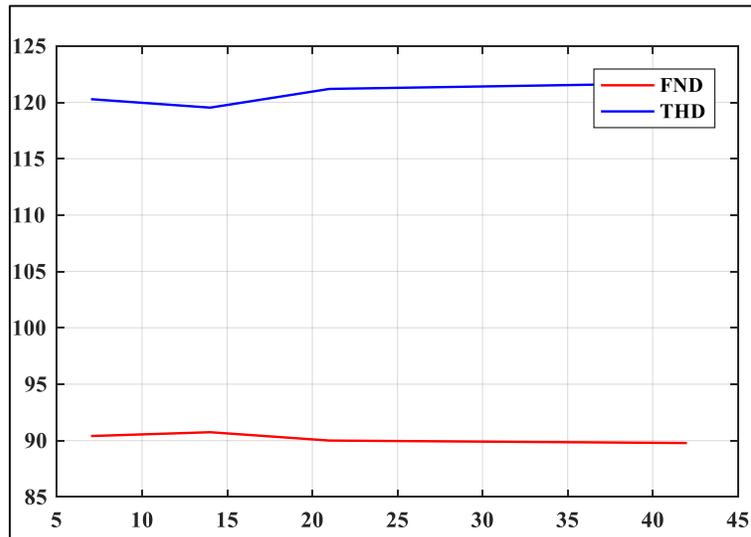


Figure (I.28) : Evolution du fondamentale et du THD en fonction de « m » pour la MLI bipolaire

Remarques :

- ✓ Les figures (I.19-21-23-25) montrent que la tension bascule entre deux niveau $+E$ et $-E$, et que le courant est quasiment sinusoïdal avec des bruits provoqués par la commutation des interrupteurs.
- ✓ Les figures (I.20-22-24-26) montrent que les harmoniques de faible fréquence sont quasiment supprimées, et celle de haute fréquence sont groupées autour des multiples de la fréquence de la porteuse.
- ✓ La figure (I.27) et le tableau (I.4) montrent que l'amplitude du fondamentale est proportionnelle à l'indice « r », et que le THD est inversement proportionnelle avec « r ».
- ✓ La figure (I.28) et le tableau (I.5) montrent que l'amplitude du fondamentale n'est pas affectée par « m », et que le THD augmente légèrement avec « m ».

I.4.3.3 MLI unipolaire

Dans le but de réduire le stress sur les interrupteurs lors de la commutation, on peut modifier la façon de générer les signaux de commande de manière à avoir une tension de charge qui garde sa polarité durant toute une demi-période. Ce mode de fonctionnement est nommé : MLI bipolaire. On présente deux variantes de la MLI unipolaire : la MLI unipolaire à deux références, et la MLI unipolaire à deux porteuses.

a. MLI unipolaire à deux références

Cette variante consiste à générer les signaux de commande par la comparaison de deux signaux de références sinusoïdales de faible fréquence avec un signal de porteuse triangulaire de haute fréquence, comme indiqué à la figure I.29.

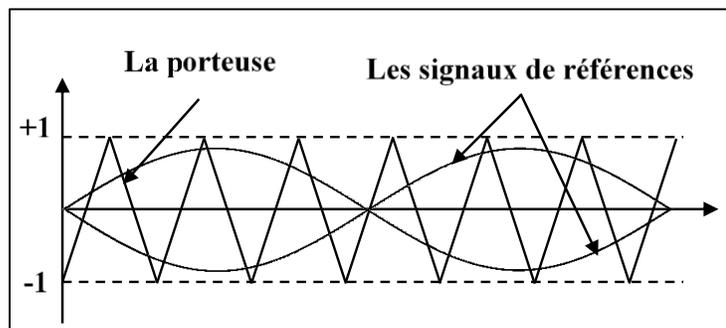


Figure (I.29) : Principe de la Commande unipolaire avec deux références

On utilise *Matlab/Simulink* pour tracer les allures de tension et du courant, ainsi que l'analyse spectrale de la tension de sortie pour la MLI unipolaire avec deux références et un signal porteuse pour différents valeur de « **m** » et « **r** » comme il montre dans les figures I.30, I.31, I.32, I.33, I.34 I.35, I.36, et I.37.

Les tableaux I.6 et I.7 et les figures I.38 et I.39 représentent l'évolution du fondamentale et du THD en fonction de différentes valeurs de « **m** » et « **r** ».

✓ **m = 21** :

r	0.2	0.4	0.7	0.9
fondamentale	20.28	39.52	69.10	90
THD %	229.65	148.79	92.49	64.75

Tableau (I.6) : Evolution du fondamentale et du THD en fonction de « **r** » pour la MLI unipolaire avec deux références

✓ **r = 0.9** :

m	7	14	21	42
fondamentale	90.40	90.74	90	89.78
THD %	64.91	63.54	64.75	64.73

Tableau (I.7) : Evolution du fondamentale et du THD en fonction de « **m** » pour la MLI unipolaire avec deux références

✓ « **m=21** » et « **r=0.9** »

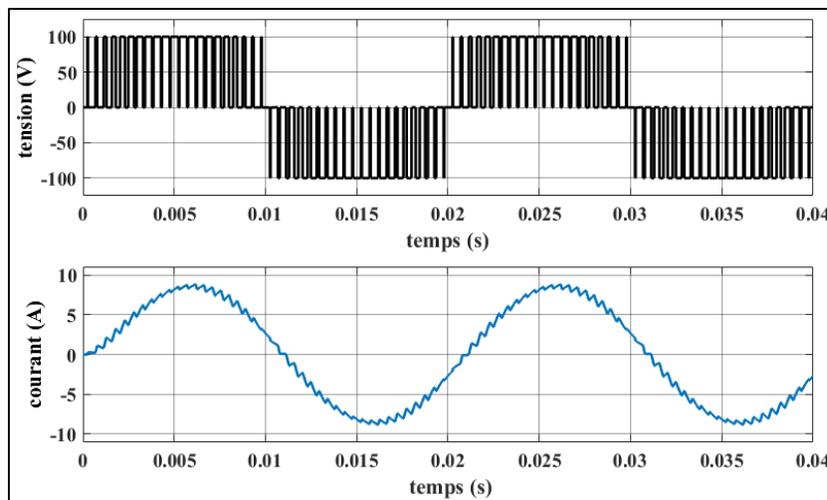


Figure (I.30) : Tension et courant de sortie pour la MLI unipolaire avec deux références ($m=21, r=0.9$)

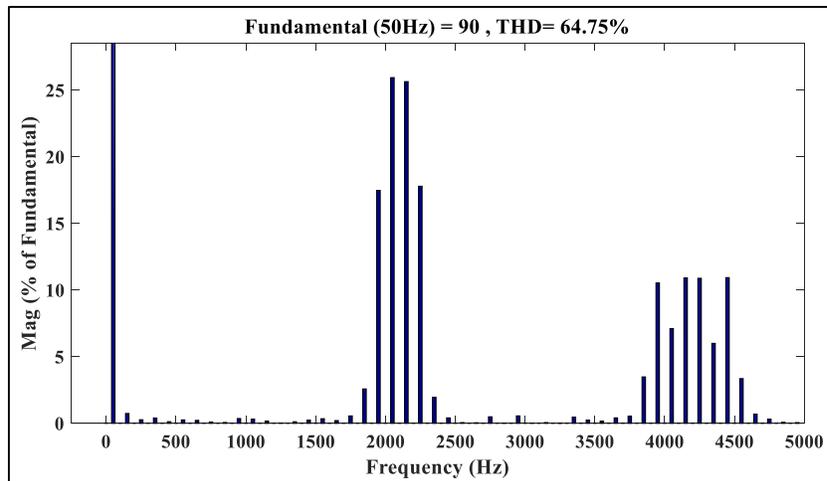


Figure (I.31) : L'analyse spectrale de la tension pour la MLI unipolaire avec deux références ($m=21$, $r=0.9$)

✓ « $m=42$ » et « $r=0.9$ »

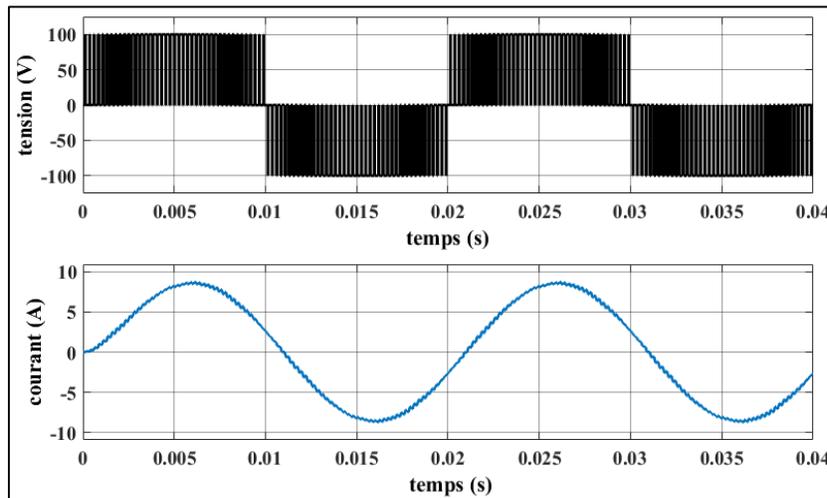


Figure (I.32) : Tension et courant de sortie pour la MLI unipolaire avec deux références ($m=42$, $r=0.9$)

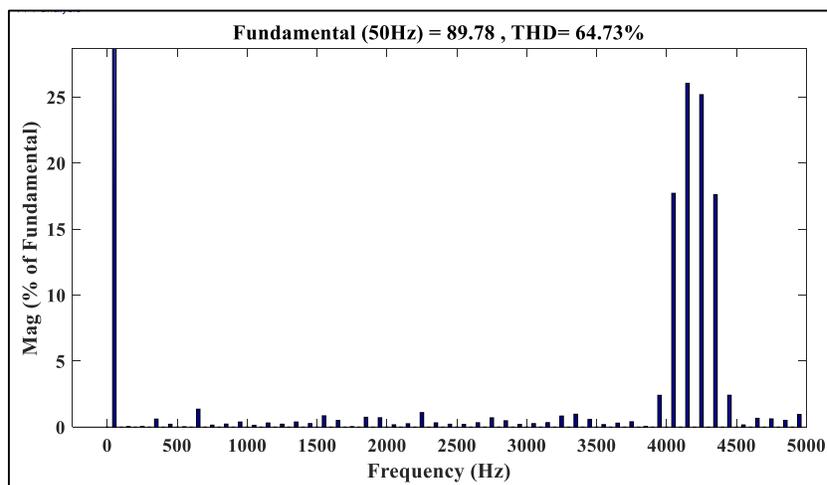


Figure (I.33) : L'analyse spectrale de la tension pour la MLI unipolaire avec deux références ($m=42$, $r=0.9$)

✓ « $m=21$ » et « $r=0.4$ »

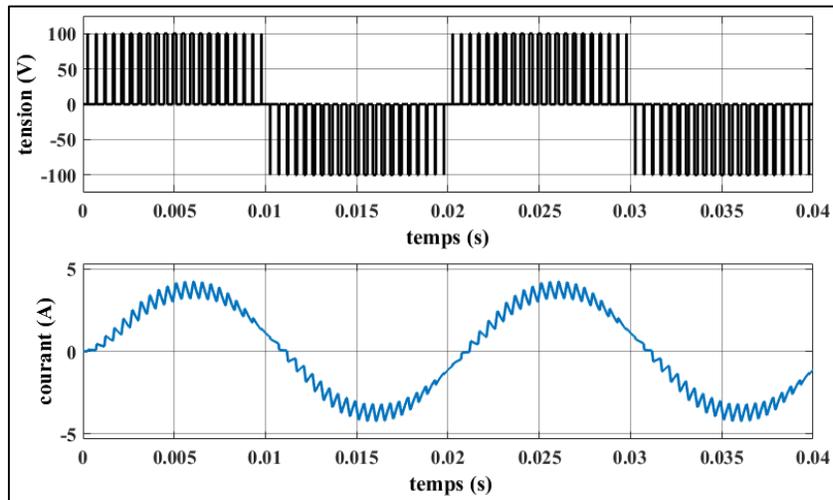


Figure (I.34) : Tension et courant de sortie pour la MLI unipolaire avec deux références ($m=21$, $r=0.4$)

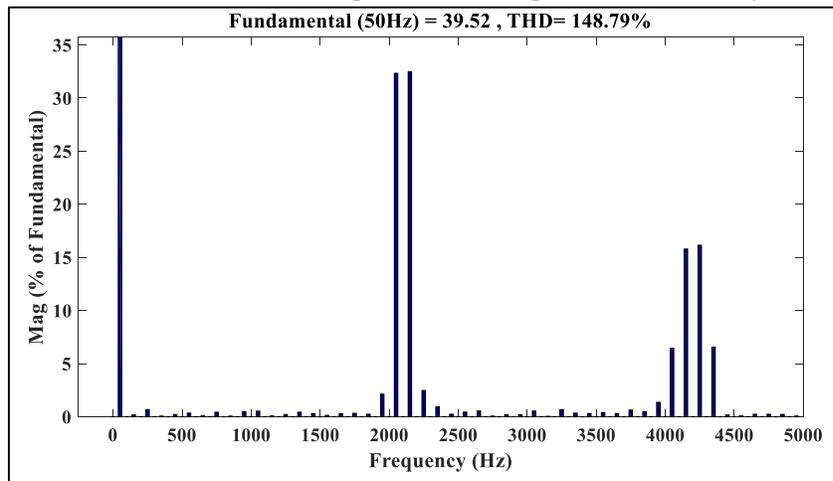


Figure (I.35) : L'analyse spectrale de la tension pour la MLI unipolaire avec deux références ($m=21$, $r=0.4$)

✓ Pour : « $m=21$ » et « $r=0.7$ »

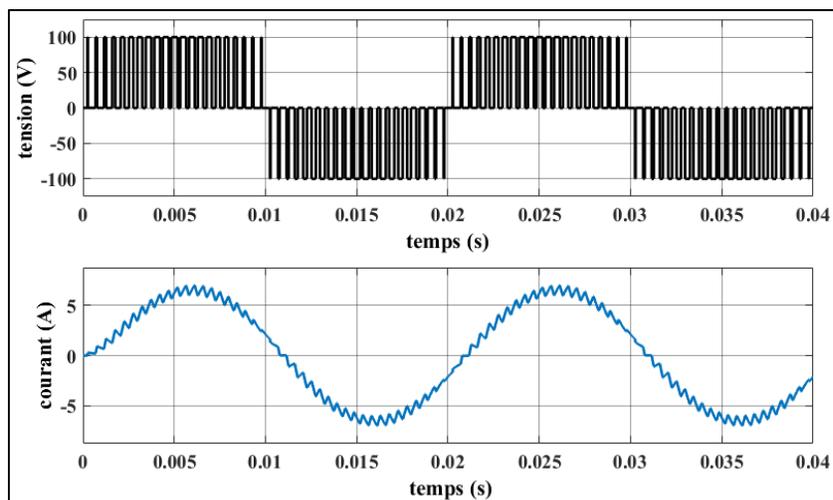


Figure (I.36) : Tension et courant de sortie pour la MLI unipolaire avec deux références ($m=21$, $r=0.7$)

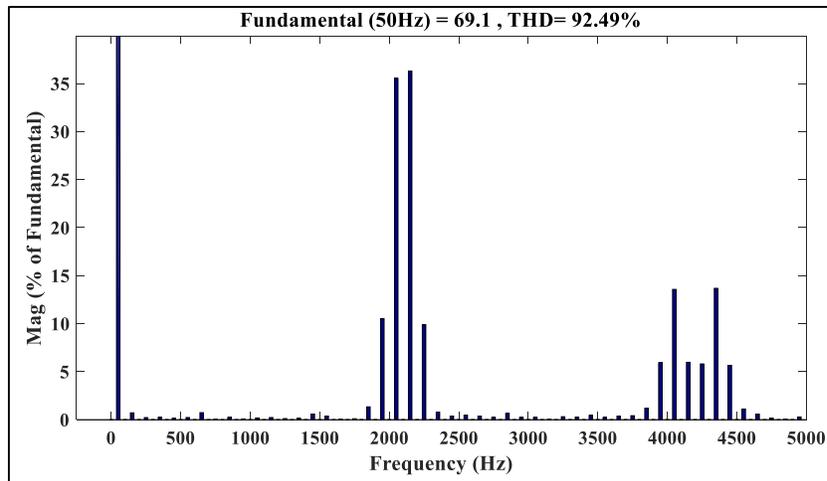


Figure (I.37) : L'analyse spectrale de la tension pour la MLI unipolaire avec deux références ($m=21, r=0.7$)

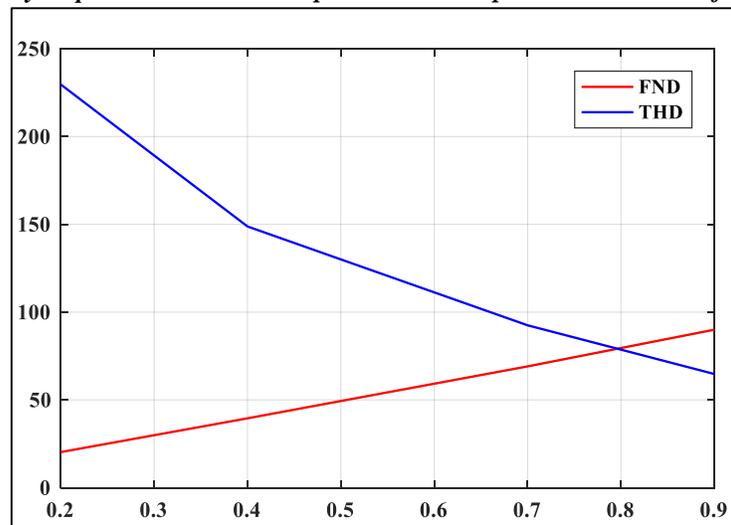


Figure (I.38) : Evolution du fondamentale et du THD en fonction de « r » pour la MLI unipolaire avec deux références

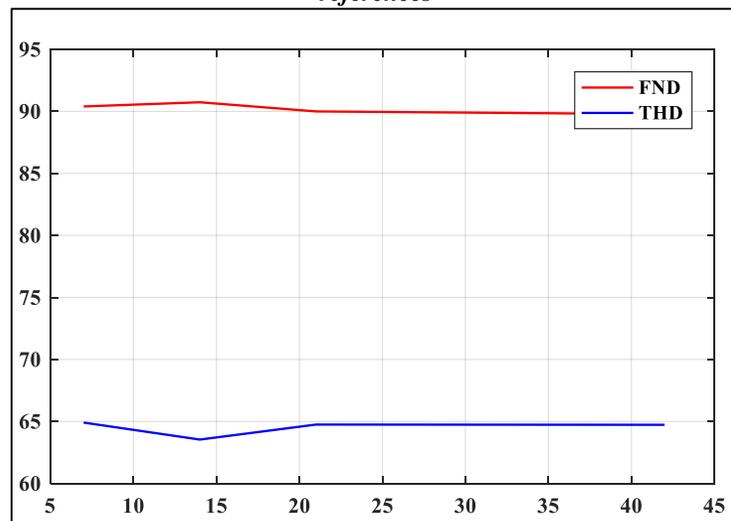


Figure (I.39) : Evolution du fondamentale et du THD en fonction de « m » pour la MLI unipolaire avec deux références

Remarques :

- ✓ Les figures (I.30, 32, 34, 36) montrent que la tension U_c bascule entre trois niveaux 0 , $+E$ et $-E$: elle varie entre 0 , $+E$ durant la première demi période, et entre 0 , $-E$ durant la deuxième demi période. Le courant de sortie est quasiment sinusoïdal avec des bruits provoqués par la commutation des interrupteurs.
- ✓ Les figures (I.31, 33, 35,37) montrent que les harmoniques de faible fréquence sont quasiment supprimées, et celle de haute fréquence sont groupées autour des multiples de la fréquence de la porteuse.
- ✓ La figure (I.38) et le tableau (I.6) montrent que l'amplitude du fondamentale est proportionnelle à « r », et que le THD est inversement proportionnelle à « r ».
- ✓ La figure (I.39) et le tableau (I.7) montrent que l'amplitude du fondamentale n'est pas affectée par « m », et que le THD augmente légèrement avec « m ».

a. MLI unipolaire à deux porteuses

Cette variante consiste à générer les signaux de commande par la comparaison d'un signal de référence sinusoïdale de faible fréquence avec deux signaux de porteuses triangulaires de haute fréquence, comme indiqué à la figure I.40.

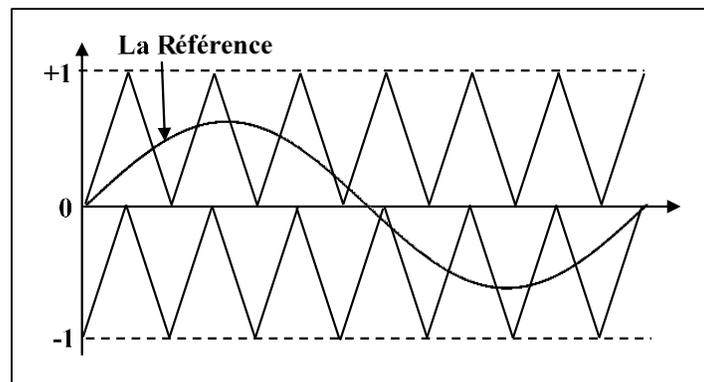


Figure (I.40) : Principe de la Commande unipolaire avec deux porteuses

On utilise *Matlab/Simulink* pour tracer les allures de tension et du courant, ainsi que l'analyse spectrale de la tension de sortie de la MLI unipolaire avec deux porteuses et un signal de référence pour différents valeur de « m » et « r » comme il montre dans les figures I.41, I.42, I.43, I.44, I.45 I.46, I.47, et I.48.

Les tableaux I.8 et I.9 et les figures I.49 et I.50 représentent l'évolution du fondamentale et du THD en fonction de différentes valeurs de « m » et « r ».

✓ **m = 21 :**

r	0.2	0.4	0.7	0.9
fondamentale	19.70	40.10	69.76	89.89
THD %	234.35	147.37	90.56	64.25

Tableau (I.8) : l'évolution du fondamentale et du THD en fonction de « r » pour la MLI unipolaire avec deux porteuses

✓ **Pour r = 0.9 :**

m	7	14	21	42
fondamentale	90.31	90.15	89.89	89.90
THD %	61.35	63.72	64.25	64.53

Tableau (I.9) : l'évolution du fondamentale et du THD en fonction de « m » pour la MLI unipolaire avec deux porteuses

✓ « m=21 » et « r=0.9 »

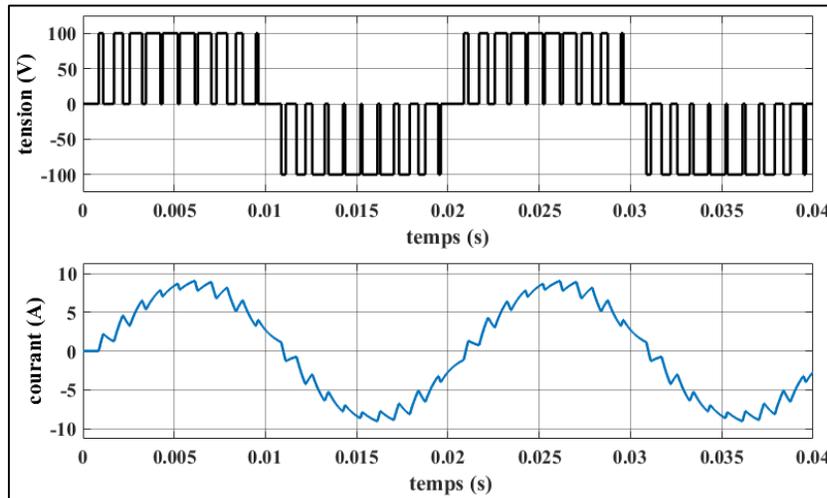


Figure (I.41) : Tension et courant de sortie pour la MLI unipolaire avec deux porteuses (m=21, r=0.9)

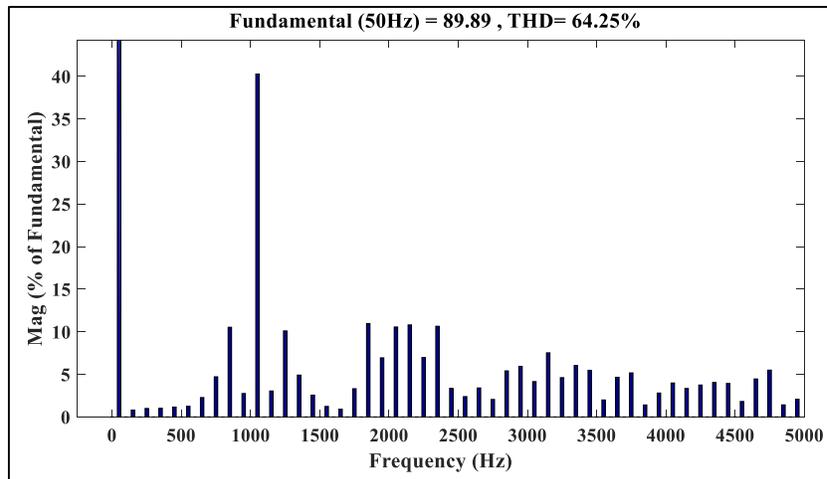


Figure (I.42) : L'analyse spectrale de la tension pour la MLI unipolaire avec deux porteuses ($m=21$, $r=0.9$)

✓ « $m=42$ » et « $r=0.9$ »

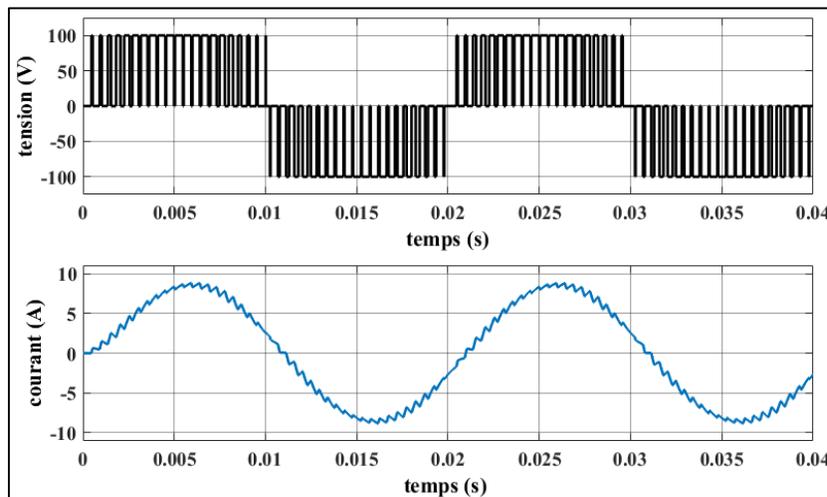


Figure (I.43) : Tension et courant de sortie pour la MLI unipolaire avec deux porteuses ($m=42$, $r=0.9$)

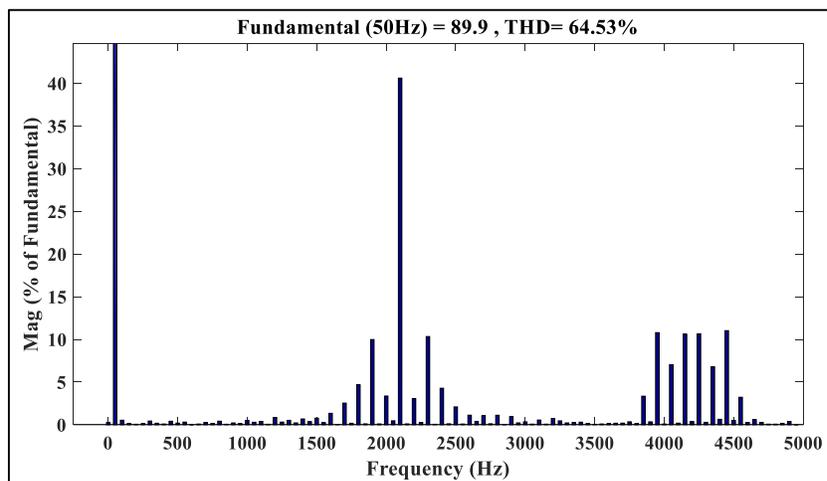


Figure (I.44) : L'analyse spectrale de la tension pour la MLI unipolaire avec deux porteuses ($m=42$, $r=0.9$)

✓ « $m=21$ » et « $r=0.4$ »

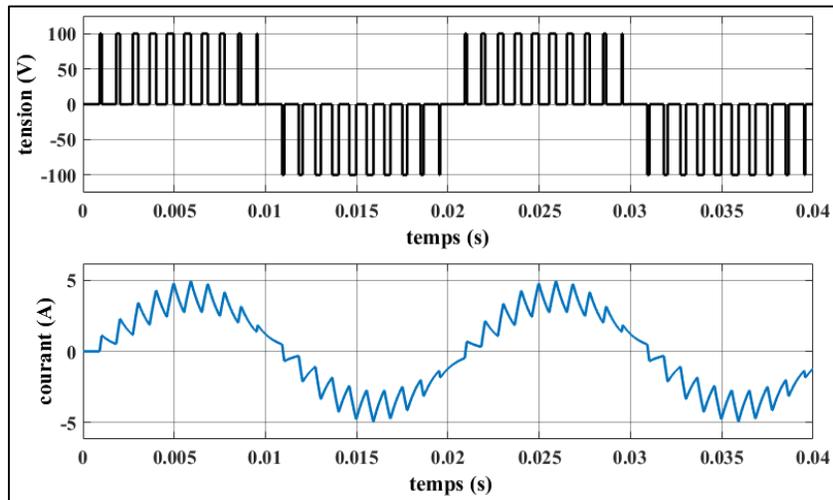


Figure (I.45) : Tension et courant de sortie pour la MLI unipolaire avec deux porteuses ($m=21$, $r=0.4$)

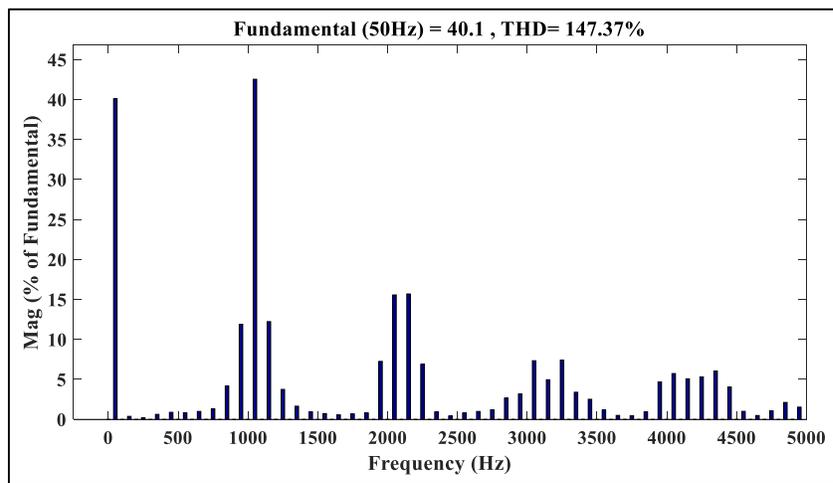


Figure (I.46) : L'analyse spectrale de la tension pour la MLI unipolaire avec deux porteuses ($m=21$, $r=0.4$)

✓ « $m=21$ » et « $r=0.7$ »

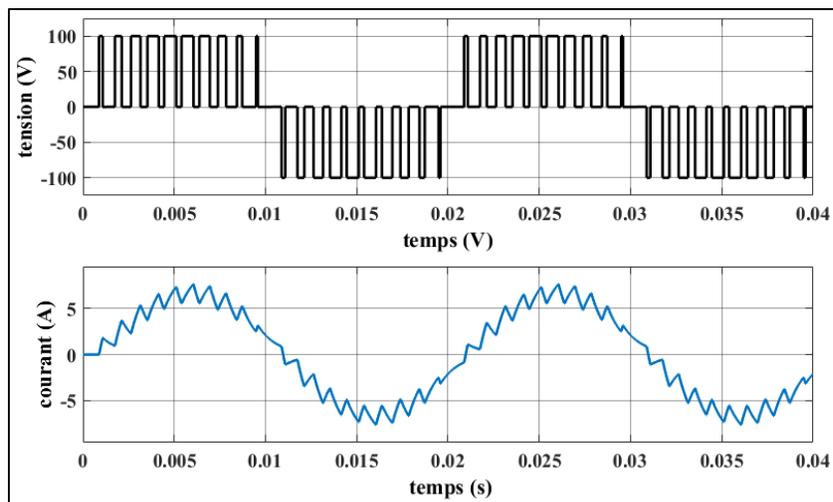


Figure (I.47) : Tension et courant de sortie pour la MLI unipolaire avec deux porteuses ($m=21$, $r=0.7$)

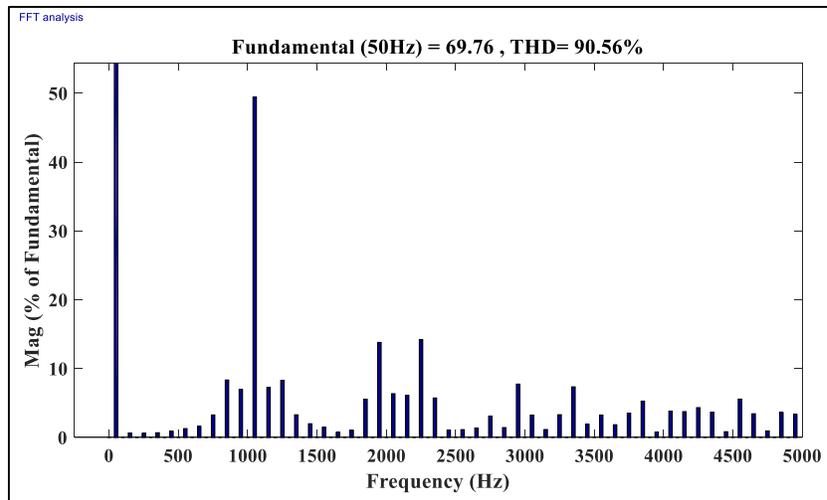


Figure (I.48) : L'analyse spectrale de la tension pour la MLI unipolaire avec deux porteuses ($m=21, r=0.7$)

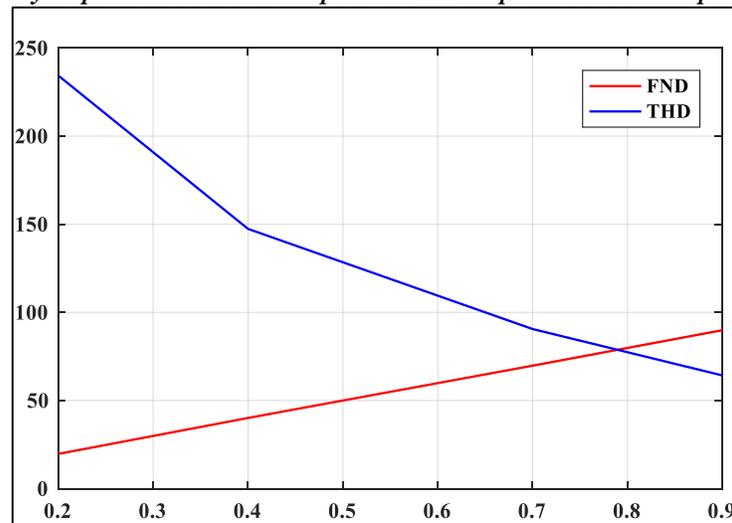


Figure (I.49) : l'évolution du fondamentale et du THD en fonction de « r » pour la MLI unipolaire avec deux porteuses

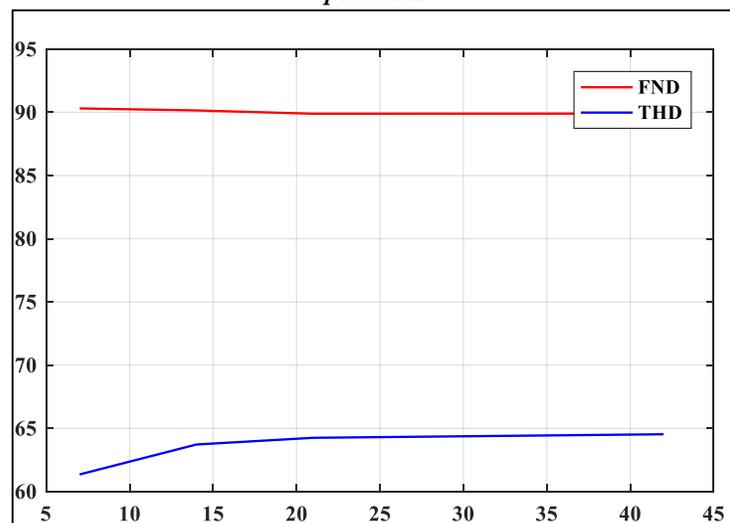


Figure (I.50) : l'évolution du fondamentale et du THD en fonction de « m » pour la MLI unipolaire avec deux porteuses

Remarques :

- ✓ Les figures (I.41, 43, 45, 47) montrent que la tension U_c bascule entre trois niveaux 0 , $+E$ et $-E$: elle varie entre 0 , $+E$ durant la première demi période, et entre 0 , $-E$ durant la deuxième demi période. Le courant de sortie est quasiment sinusoïdal avec des bruits provoqués par la commutation des interrupteurs.
- ✓ Les figures (I.42, 44, 46, 48) montrent que les harmoniques de faible fréquence ont des amplitudes très faibles, et que les harmoniques de haute fréquence sont groupées autour des multiples de la fréquence de la porteuse.
- ✓ La figure (I.49) et le tableau (I.8) montrent que l'amplitude du fondamentale est proportionnelle à « r », et que le THD est inversement proportionnelle à « r ».
- ✓ La figure (I.50) et le tableau (I.9) montrent que l'amplitude du fondamentale n'est pas affectée par « m », et que le THD augmente légèrement avec « m ».

Comparaison des résultats de la MLI bipolaire et unipolaire :

- ✓ La tension de sortie et le courant de sortie sont légèrement de meilleure qualité de point de vue THD pour la MLI unipolaire par rapport à la MLI bipolaire.
- ✓ Les pertes par commutation sont plus réduites pour la MLI unipolaire, par rapport à la MLI bipolaire. En effet si on applique la MLI unipolaire, chaque interrupteur de l'onduleur se commute seulement durant chaque demi-période. Par contre si on applique la MLI bipolaire, tous les interrupteurs de l'onduleur se commutent durant toute la période de commutation, alors les pertes par commutation sont plus importantes pour la MLI bipolaire.
- ✓ En comparant les deux variantes de la MLI unipolaire, on constate que la MLI unipolaire à deux tensions de référence Par la MLI donne une tension de sortie de sortie de meilleure qualité de point de vue THD, par rapport à la MLI unipolaire à deux porteuses.
- ✓ En contrepartie, la MLI à deux porteuses provoque moins des pertes par commutation par rapport à la MLI à deux références, car le nombre de commutations, à paramètres identiques, est plus faible.

I.4.4 La commande en courant par hystérésis

La commande par hystérésis, appelé aussi commande en tout ou rien, est une commande non linéaire qui utilise l'erreur existante entre le courant de référence et le courant produit par l'onduleur, cette erreur est comparée à un gabarit « h » appelé bande d'hystérésis [17].

L'approche la plus simple utilisée pour cette fin, est la stratégie de contrôle qui réalise la comparaison entre le courant mesuré et le courant de référence à l'aide d'un comparateur à hystérésis. Celui-ci, produit des impulsions d'amorçage et de blocage des interrupteurs de l'onduleur, de façon à limiter le courant dans une bande d'hystérésis autour du courant de référence, comme illustré à la figure I.51 et I.52 [17].

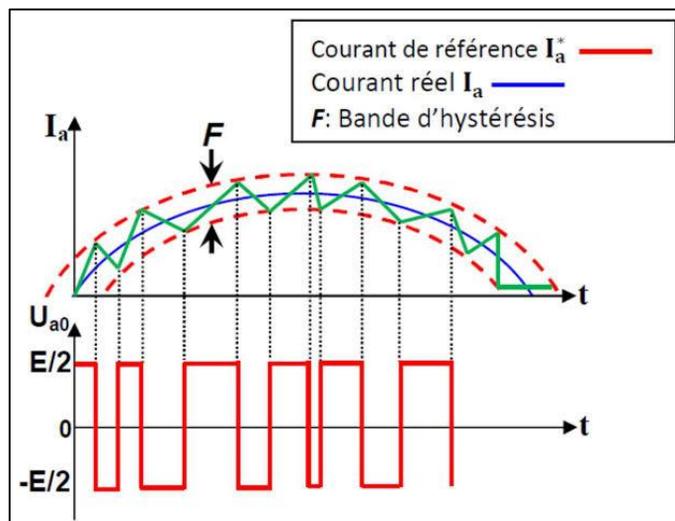


Figure (I.51) : Modulation hystérésis

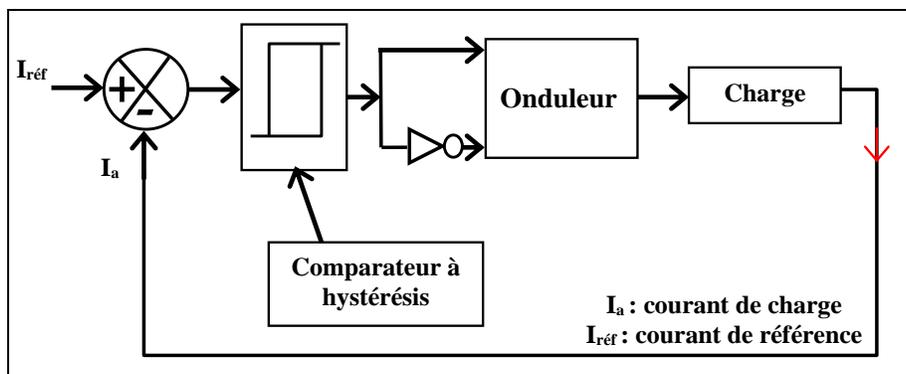


Figure (I.52) : Principe de la commande par Hystérésis

On simule la méthode à l'aide de *Matlab/Simulink* et on trace les allures de tension et du courant, ainsi que l'analyse spectrale du courant de sortie, pour différentes grandeurs de la largeur de la bande d'hystérésis h .

✓ $h=0.05$:

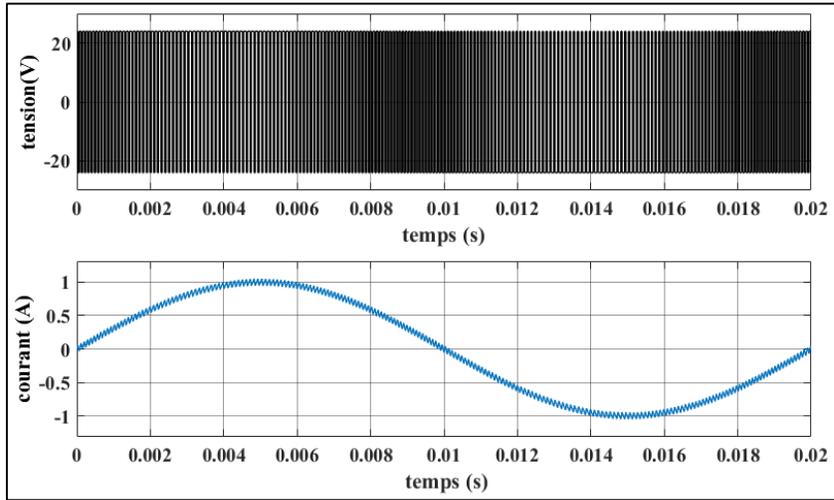


Figure (I.53) : Tension et courant de sortie pour la modulation à hystérésis ($h=0.05$)

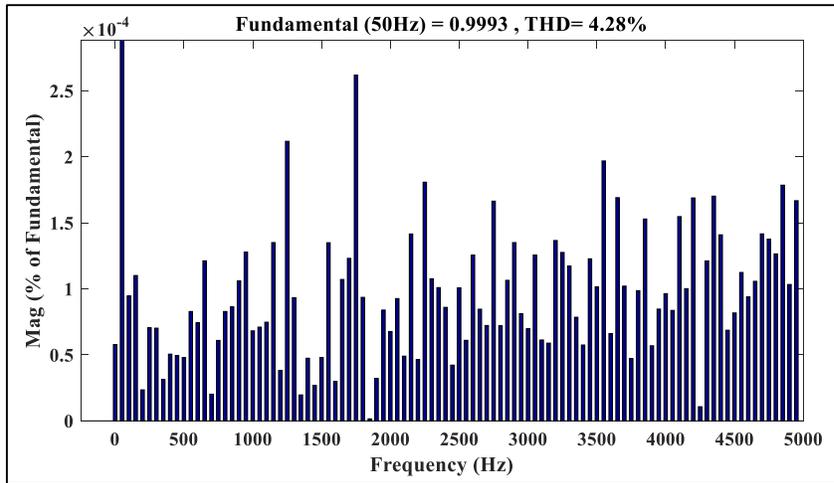


Figure (I.54) : L'analyse spectrale du courant de sortie pour la modulation à hystérésis ($h=0.05$)

✓ $h=0.07$:

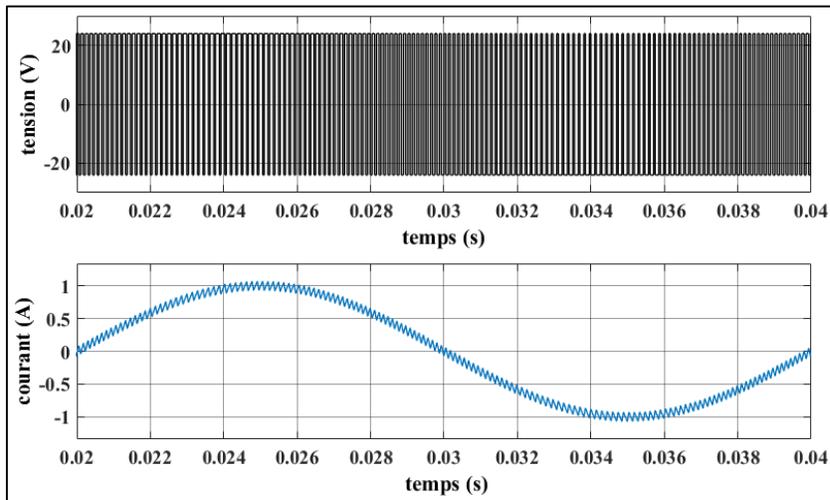


Figure (I.55) : Tension et courant de sortie pour la modulation à hystérésis ($h=0.07$)

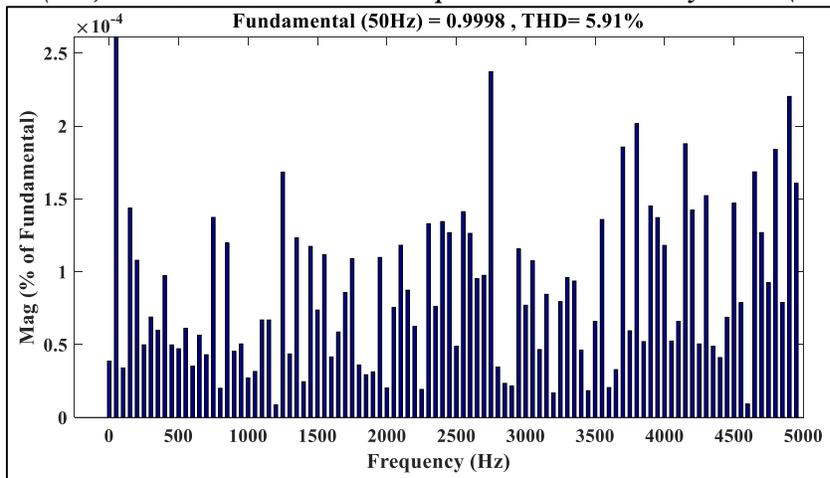


Figure (I.56) : L'analyse spectrale du courant de sortie pour la modulation à hystérésis ($h=0.07$)

✓ $h=0.1$:

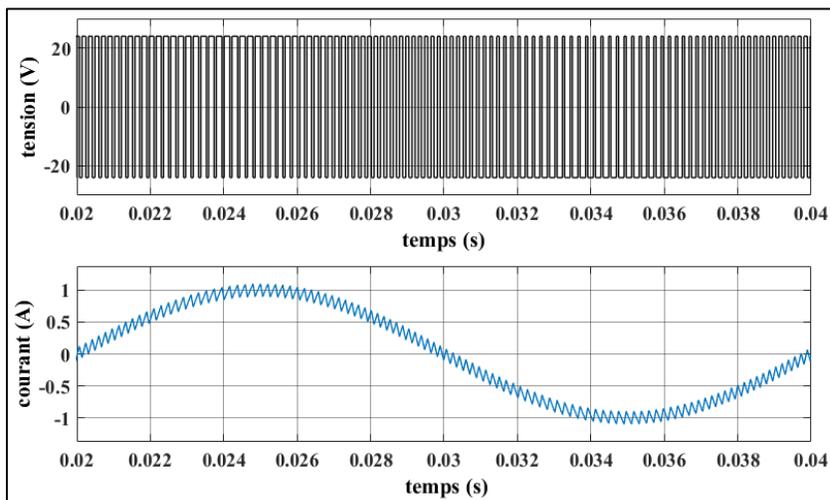


Figure (I.57) : Tension et courant de sortie pour la modulation à hystérésis ($h=0.1$)

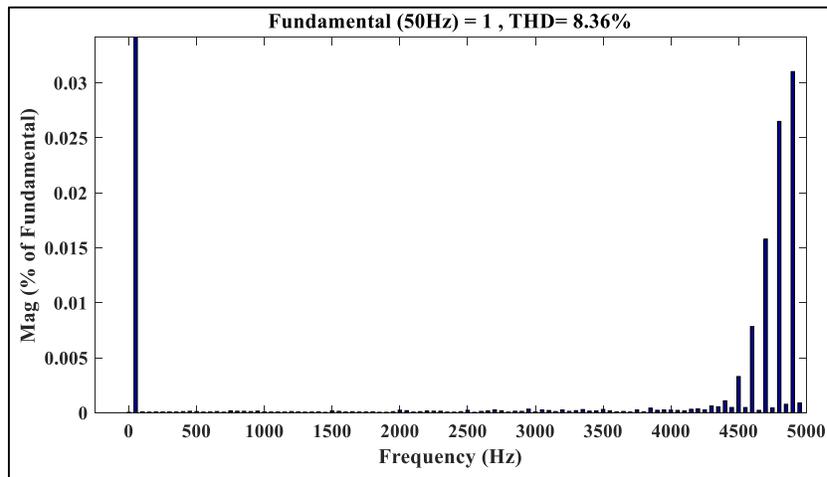


Figure (I.58) : L'analyse spectrale du courant de sortie pour la modulation à hystérésis ($h=0.1$)

Remarques :

- ✓ En se référant aux figures (I.52, 54, 56), le courant de charge bascule autour du courant de référence, en évoluant à l'intérieur de la bande d'hystérésis imposée.
- ✓ Les spectres de courant indiqué dans les figures (I.53, 55, 57) montrent que le THD est proportionnelle à la largeur de la bande d'hystérésis.

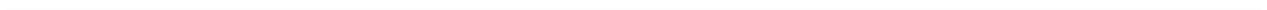
I.5 Conclusion :

Au cours de ce chapitre, nous avons présenté la structure et le fonctionnement général de l'onduleur monophasé, ainsi que les stratégies classiques de sa commande. Dans le chapitre suivant, nous allons présenter les éléments matériels nécessaires pour implanter ces stratégies de commande.

Chapitre II



Carte ARDUINO, Onduleur et Accessoires de Mesure



II.1 Introduction

Dans ce chapitre, Nous donnons la description de la carte "ARDUINO" et des composants matériels requis pour l'application de la commande de l'onduleur monophasé. Nous décrivons les caractéristiques, la structure et la programmation de la carte Arduino. Nous décrivons par suite les deux modules L298N et BTS7960, qui sont fabriqués à la base des convertisseurs en pont H, et qui sont utilisés initialement en tant que drivers des moteurs à courant continu, mais qui peuvent aussi être utilisés en tant qu'onduleurs monophasés. En fin du chapitre nous donnons la description des éléments de mesures du courant et de tension, nécessaires pour l'application de la commande et l'acquisition des signaux engendrés. On va présenter le composant ADS1115, qui est un convertisseur digital analogique et qui peut être utilisé en tant que capteur de tension différentielle, et le composant INA219, qui est un capteur de courant et de tension.

II.2 Carte ARDUINO

II.2.1 Définition et Historique de la carte ARDUINO

La carte ARDUINO est un circuit imprimé spécifiquement conçu pour héberger un microcontrôleur et donner accès à toutes ses entrées et sorties. Elle comprend aussi quelques autres composants électroniques qui permettent de faire fonctionner le microcontrôleur ou d'en étendre les fonctionnalités [18].

L'Arduino est à l'origine un projet d'étudiants de l'école de Design d'Interaction d'Ivrea en Italie. Au début des années 2000, les outils de conception de projets dans le domaine du design d'interaction étaient onéreux, proches d'une centaine d'euros. Ces outils étaient pour la plupart conçus pour le domaine de l'ingénierie et de la robotique. Maîtriser et utiliser ces composants demandait beaucoup de temps et d'apprentissage et ralentissait fortement le processus de création pour ces jeunes étudiants.

Il leur vient alors à l'idée de créer une plateforme plus abordable et plus simple à utiliser, reposant sur l'environnement de développement Processing mis au point en 2001 par des étudiants du MIT. C'est donc en 2003 que, pour un projet de fin d'études, fut conçue « la carte Wiring », ancêtre de l'Arduino. Visant à rendre la plateforme toujours moins chère et plus accessible, une équipe d'étudiants et de professeurs finirent par concevoir la toute première Arduino en 2005. Entièrement open source, l'Arduino présentait l'avantage d'être multiplateforme et d'être en perpétuelle optimisation par la communauté d'utilisateurs [19].

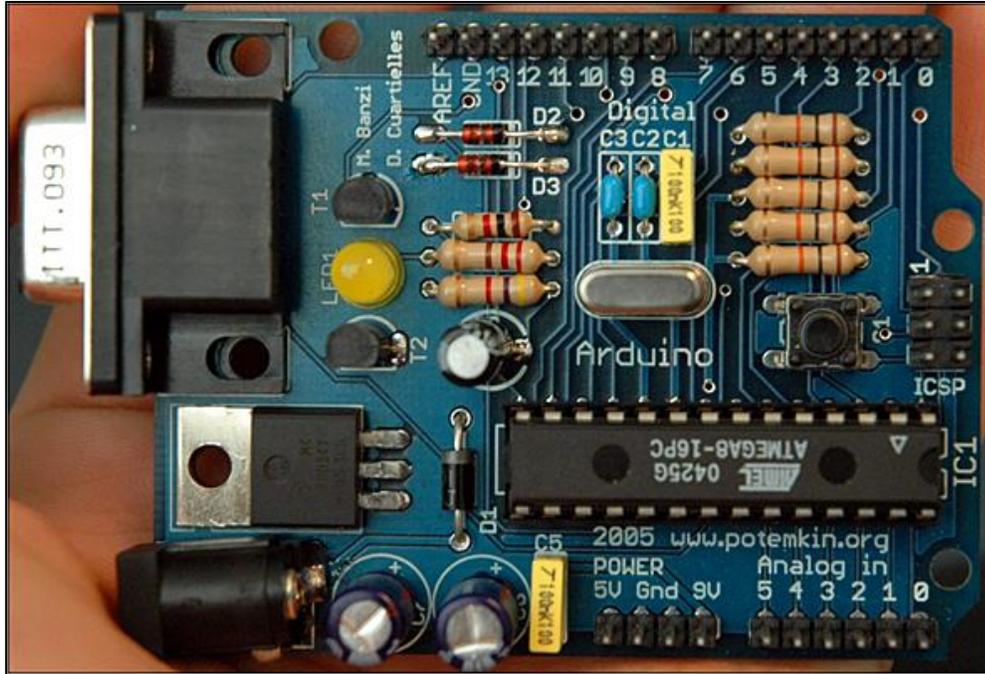


Figure (II.1) : Premier modèle de la carte Arduino [19]

II.2.2 Différent type

Il y a plusieurs types des cartes Arduino. On mentionne les plus fréquents [20].

- ✓ Arduino UNO.
- ✓ Arduino Leonardo.
- ✓ Arduino Mega 2560.
- ✓ Arduino Esplora.
- ✓ Arduino Nano.
- ✓ Arduino LilyPad.
- ✓ Arduino Due.

II.2.3 Critère de Choix d'une carte Arduino

Le choix de la carte Arduino se fera à la base de ses caractéristiques, de son microcontrôleur et de son coût. Le tableau II.1 suivant dresse une comparaison entre les composantes matérielles des différents types des cartes Arduino. Nous avons opté pour l'utilisation de la carte Uno en raison d'être standard et simple en pratique pour les projets comme la commande de convertisseur en pont, ainsi qu'elle offre une mémoire importante.

ARDUINO	Processeur	Flash « ko »	SRAM « ko »	EEPROM « ko »	Broches d'E/S numérique	Broches avec PWM	Broches d'entrée analogique
UNO 	ATmega328p	32	2	1	14	6	6
Leonardo 	ATmega32U4	32	2,5	1	20	7	12
Mega 2560 	AT2560	256	8	4	54	15	16
Esplora 	ATmega32U4	32	2,5	1	/	/	/
Nano 	ATmega168 ou 328	16 ou 32	1 ou 2	512 octets ou 1	14	6	8

Tableau (II.1) : Critères de choix de la carte Arduino [20]

II.2.4 Carte Arduino UNO

II.2.4.1 Description d'une carte Arduino UNO

La carte « Arduino UNO » est un bon compromis entre le cout abordable et les performances suffisantes pour réaliser des projets non complexes tels que celui de la commande en PWM d'un onduleur monophasé.

La figure II.2 indique la forme d'une carte Arduino UNO. La carte se connecte à un ordinateur à travers un port USB. Ce port sert en même temps à l'alimentation de la carte, et à l'échange de données entre la carte et l'ordinateur. En option, on peut alimenter la carte à travers un adaptateur secteur ou une pile à travers le connecteur d'alimentation externe, mais ceci n'est pas nécessaire, l'alimentation par le port USB peut être suffisante.

La carte comporte 14 broches numérique d'entrées/sorties, dont 6 peuvent être utilisée en sorties PWM. Elle comporte en plus 6 sorties analogique.

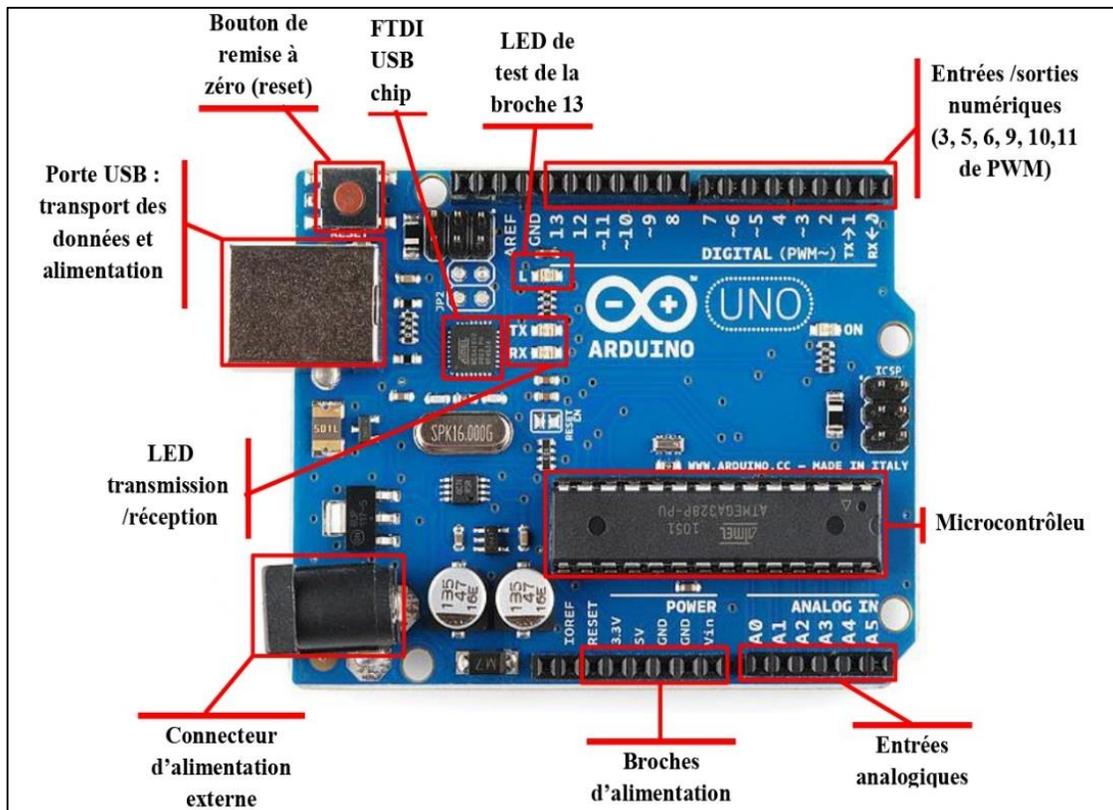


Figure (II.2) : Description de la carte Arduino-UNO

II.2.4.2 Structure de la carte ARDUINO

On présente la description des différentes parties présente sur la carte Arduino UNO :

- **Alimentation**

Le microcontrôleur présent sur les cartes Arduino est en général alimenté par une tension de 5V. En fonction du modèle de la carte, cette tension peut être fournie soit par une des prises d'alimentation présentes sur la carte, soit par la prise USB utilisée pour la connecter à un ordinateur. La valeur de la tension à fournir sur une des prises d'alimentation doit être comprise entre 7 et 12V mais cette tension n'a pas besoin d'être stabilisée en raison de la présence d'un régulateur de tension sur la carte.

Il peut exister deux prises d'alimentation sur une carte Arduino. La première prise est souvent la plus simple à mettre en place. Il s'agit d'une simple prise jack standard, présente sur les cartes Arduino UNO et Arduino Mega, sur laquelle il suffit de raccorder un bloc d'alimentation fournissant la tension requise pour démarrer sa carte. Cette prise n'est pas présente sur toutes les cartes Arduino, notamment pour les cartes les plus petites pour lesquelles ces prises n'étaient pas vitales [19].

- **Mémoires [21]**

- ✓ **Mémoire flash** Elle est de 32 ko dont 0,5 ko sont utilisés pour le bootloader, son programme de démarrage. Cette mémoire est l'équivalent du disque dur pour l'ordinateur. C'est la place que nous avons pour stocker le programme.
- ✓ **SRAM** Équivalent à la mémoire RAM, elle sert à stocker le résultat des variables. Sa taille est de 2 ko. Comme la RAM, cette mémoire est volatile : à l'extinction de la carte, son contenu disparaît.
- ✓ **EEPROM** Elle permet de sauvegarder des valeurs de variables et ceci même à l'extinction de la carte. Sa taille est de 1 ko.

- **Entrées et sorties numériques**

Chacune des 14 broches numériques de la carte UNO (numérotées des 0 à 13) peut être utilisée soit comme une entrée numérique, soit comme une sortie numérique, en utilisant les instructions **pinMode()**, **digitalWrite()** et **digitalRead()** du langage Arduino. Ces broches fonctionnent en 5V. Chaque broche peut fournir ou recevoir un maximum de 40mA d'intensité et dispose d'une résistance interne de "rappel au plus" (pull-up) (déconnectée par défaut) de 20-50 KOhms. Cette résistance interne s'active sur une broche en entrée à l'aide de l'instruction **digitalWrite(broche, HIGH)** [22].

De plus, certaines broches ont des fonctions spécialisées [22]:

- ✓ **Communication Série** : Broches 0 (RX) et 1 (TX). Utilisées pour recevoir (RX) et transmettre (TX) les données sériées de niveau TTL (Transistor-Transistor Logic). Ces broches sont connectées aux broches correspondantes du circuit intégré ATmega8U2 programmé en convertisseur USB-vers-série de la carte, composant qui assure l'interface entre les niveaux TTL et le port USB de l'ordinateur.
- ✓ **Interruptions Externes** : Broches 2 et 3. Ces broches peuvent être configurées pour déclencher une interruption sur une valeur basse, sur un front montant ou descendant, ou sur un changement de valeur : instruction **attachInterrupt()**.
- ✓ **Impulsion PWM** : (largeur d'impulsion modulée): Les broches 3, 5, 6, 9, 10, et 11. Fournissent une impulsion PWM 8-bits à l'aide de l'instruction **analogWrite()**. Ces impulsions PWM correspondent à un niveau de tension compris entre 0 volts et 5 volts.
- ✓ **SPI** (Interface Série Périphérique): Broches 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ces broches supportent la communication SPI (Interface Série Périphérique) disponible avec la librairie pour communication SPI. Les broches SPI sont également connectées sur le connecteur ICSP qui est mécaniquement compatible avec les cartes Mega.
- ✓ **I2C** Broches 4 (SDA) et 5 (SCL). Supportent les communications de protocole I2C (ou interface TWI (Two Wire Interface - Interface "2 fils"), disponible en utilisant la librairie Wire/I2C (ou TWI - Two-Wire interface - interface "2 fils").
- ✓ **LED** Broche 13. Il y a une LED incluse dans la carte connectée à la broche 13. Lorsque la broche est au niveau HAUT, la LED est allumée, lorsque la broche est au niveau BAS, la LED est éteinte.

- **Broches analogiques**

La carte UNO dispose de 6 entrées analogiques (numérotées de 0 à 5), chacune pouvant fournir une mesure d'une résolution de 10 bits (c.à.d. sur 1024 niveaux soit de 0 à 1023) à l'aide de la fonction **analogRead()** du langage Arduino. Par défaut, ces broches mesurent entre le 0 volts (valeur 0) et 5V (valeur 1023), mais il est possible de modifier la référence supérieure de la plage de mesure en utilisant la broche AREF et l'instruction **analogReference()** du langage Arduino [22].

- **Autres broches**

Il y a deux autres broches disponibles sur la carte Arduino UNO [22] :

- ✓ AREF : Tension de référence pour les entrées analogiques (si différent du 5V). Elle est utilisée avec l'instruction **analogReference()**.
- ✓ Reset : Mettre cette broche au niveau BAS entraîne la réinitialisation (le redémarrage) du microcontrôleur de la carte. Typiquement, cette broche est utilisée pour ajouter un bouton de réinitialisation sur le circuit qui bloque celui présent sur la carte.

II.2.4.3 Caractéristique de la carte Arduino UNO

Le tableau suivant résume les caractéristiques principales de la carte Arduino UNO :

Type du Microcontrôleur	ATmega328
Fréquence d'horloge	16 MHz
Tension de service	5 V
Tension d'entrée (recommandée)	7-12 V
Tension d'entrée (limites)	6-20 V
Ports numériques	14 E/S (6 sorties PWM)
Ports analogiques	6 entrées analogiques
Courant maxi. Par broche d'E/S (c.c)	40 mA
Courant maxi. Par broche 3, 3 V	50 mA
Mémoire	32 Ko (Flash), 2 Ko (SRAM), 1 Ko (EEPROM)
Chargeur d'amorçage	0,5 Ko (en mémoire Flash)
Interface	USB
Dimensions	68.6mm x 53.3mm

Tableau (II.2) : Caractéristique de la carte Arduino-UNO [20]

II.2.5 L'environnement de programmation de l'ARDUINO

L'environnement Arduino C est un logiciel de programmation basé sur un code appelé IDE, qui comprend une cinquantaine d'instructions différentes. Ceci est écrit en Java. L'IDE permet d'écrire des programmes, de les modifier et de les convertir en une série d'instructions compréhensibles par la carte. Les programmes saisis ou modifiés au clavier sont transférés via la connexion USB et stockés sur la carte.

II.2.5.1 Structure d'un programme Arduino IDE

A l'ouverture de programme **Arduino IDE**, l'interface visuelle du logiciel est représentée sur la Figure II.3 suivante :

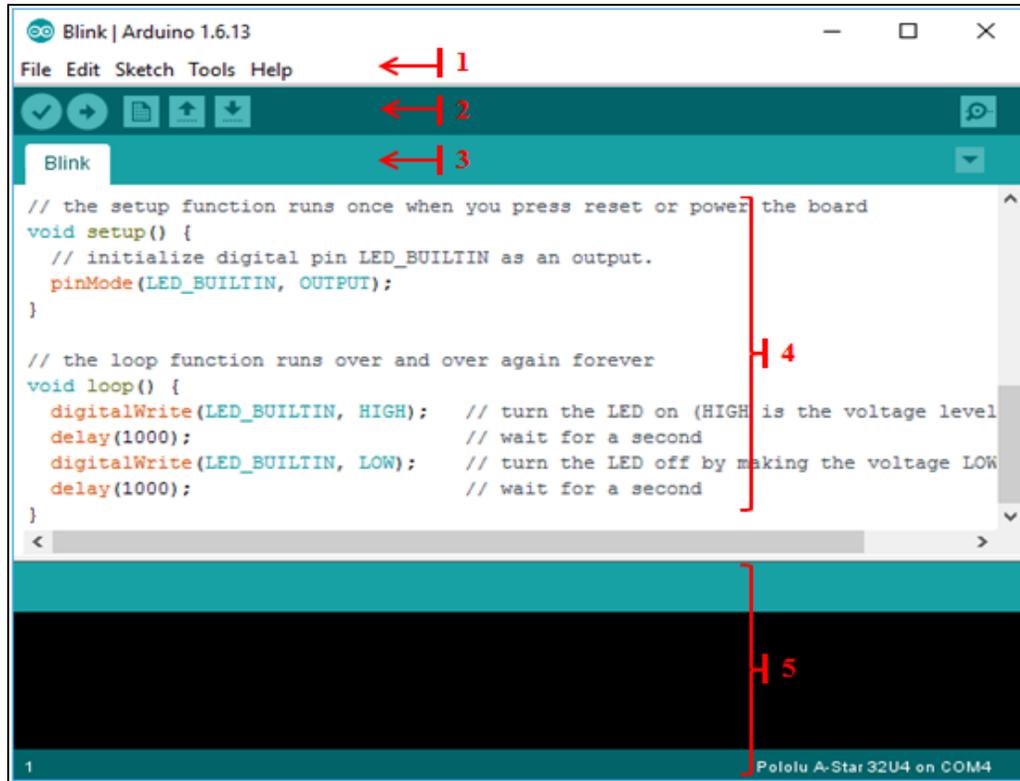


Figure (II.3) : l'interface de l'Arduino IDE

Il est possible de remarquer sur cette fenêtre plusieurs parties essentielles de l'environnement de développement Arduino :

1. une barre de menu, permet d'accéder à l'ensemble des fonctionnalités de l'environnement de développement.
2. une barre d'actions, contenant des raccourcis, qu'il est important de connaître afin de faciliter la prise en main de l'environnement.
3. un ou plusieurs onglets correspondant aux sketches (scripts).
4. zone d'écriture du programme.
5. zone des messages d'erreur ou succès envoyés par le programme lors de la vérification et de la compilation.

Voici la liste des raccourcis qu'il est possible de voir sur la barre d'action :

	Vérifier : cette fonctionnalité permet de compiler le programme Arduino.
	Télé-verser (upload) : cette fonctionnalité permet de transférer le programme compilé sur la carte Arduino
	Nouveau : cette fonctionnalité permet la création de nouveaux sketches ou projets.
	Ouvrir : cette fonctionnalité permet d'ouvrir un sketch ou un projet.
	Enregistrer : cette fonctionnalité permet de sauvegarder le travail.
	Moniteur série : cette fonctionnalité permet de communiquer directement avec la carte Arduino, à travers un affichage numérique ou graphique.

Tableau (II.3) : Liste des raccourcis sur l'écran de l'Arduino IDE

II.2.5.2 Structure et Principaux commandes d'un projet ARDUINO C

Un programme Arduino IDE comporte trois parties principales :

```

light1.ino  light1.ino
1  int led = 13; ← 1
2
3  void setup()
4  {
5      pinMode(led, OUTPUT); 2
6
7  }
8
9  void loop()
10 {
11     digitalWrite(led, HIGH); // turn the led ON
12     delay(5000); // Wait for 5 second(s)
13     digitalWrite(led, LOW); // turn the led OFF
14     delay(2000); // Wait for 2 second(s) 3
15
16 }
    
```

Figure (II.4) : Structure d'un projet Arduino IDE

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

Le tableau suivant présente les instructions principales d'un projet ARDUINO IDE, groupées en trois catégories : les instructions de structure générale, les instructions de déclaration des variables et constantes), et les instructions de fonctions.

Structure générale	Variables Constants	Fonctions
setup () loop () Contrôle et conditions if if...else for switch case while do... while break continue return goto Autres commandes ;(semi colon) { } (curly braces) // (single line comment) /* */ (multi-line comment) #define #include Arithmetic Operators = (assignment operator) + (addition) - (subtraction) * (multiplication) / (division) % (modulo) Opérations de comparaison == (equal to) != (not equal to) < (less than) > (greater than) <= (less than or equal to) >= (greater than or equal to) Operations booléennes && (and) (or) ! (not) Pointer Access Operators * dereference operator & reference operator Bitwise Operators & (bitwise and) (bitwise or)	HIGH LOW INPUT OUTPUT INPUT_PULLUP true false integer constants floating point constants Data Types void Boolean char unsigned char byte int unsigned int word long unsigned long short float double string - char array String - object array Conversion char () byte () int () word () long () float () Variable Scope & Qualifiers variable scope static volatile const Random Numbers random Seed () random () Bits and Bytes low Byte () high Byte () bit Read () bit Write () bit Set ()	E/S numérique pin Mode () digital Write () digital Read () E/S analogique analog Reference () analog Read () analog Write () - PWM Due only analog Read Resolution () analog Write Resolution() E/S Avancé tone () no Tone () shift Out () shift In () pulse In () Gestion du temps Millis () micros () delay () delay Microseconds () Math min () max () abs () constrain () map () pow () sqrt () Trigonometry sin () cos () tan () bit Clear () bit () Interruptions Externes Attach Interrupt () detach Interrupt () no Interrupts ()

Tableau (II.4) : Liste des instructions usuelles sur l'Arduino IDE [14]

II.3 Onduleur à commander

Nous avons opté à utiliser les modules industriels de commande des machines électriques, disponibles sur le marché, en tant qu'onduleurs monophasés. Ces modules sont bâtis autour des ponts H complets en autour des demis ponts ; ces ponts peuvent servir à un fonctionnement en hacheur comme ils peuvent servir à un fonctionnement en onduleur selon la manière de les commander. On a choisi deux parmi ces modules ; le module L298N et le module BTS7960.

II.3.1 Module L298N

II.3.1.1 Définition et fonctionnement

Le module **L298N** est un module de commande bidirectionnel des moteurs électriques, fabriqué par la firme STMicroelectronics. Il est utilisé pour contrôler la vitesse et la direction de rotation de moteurs à courant continu ou des moteurs à courant alternatif à faible puissance. Il est conçu à la base de deux ponts H. Il peut alors Controller deux moteurs en même temps [23].

Suivant le type de sa commande, un pont H peut fonctionner en hacheur onduleur monophasé.

Le module **L298N** est généralement utilisé avec des microcontrôleurs ou des cartes de développement comme l'Arduino pour contrôler les moteurs électriques dans les projets de robotique ou d'automatisation. Dans notre travail, il sera utilisé en tant qu'onduleur monophasé, sur lequel on va implémenter, à travers une carte Arduino UNO, la commande en pleine onde et la commande en PWM [23].

Ce module peut supporter une tension d'entrée de 35volts, et fait circuler un courant qui peut atteindre jusqu'à 2 ampères : il peut alors transférer une puissance de 70 watts.

II.3.1.2 Structure et brochage du module L298N

La figure II.5.a indiqué le schéma bloc du module L298N [24]. Il est composé de deux ponts H complets qui peuvent fonctionner et être commandés d'une manière indépendante l'un de l'autre.

La figure II.5.b indique la forme externe et le brochage du module L298N. Le circuit de puissance à une borne d'entrée qui peut aller jusqu'à 35 volts, et deux groupes des bornes de sorties qui servent à l'alimentation d'un ou deux moteurs électriques. Le circuit de commande a plusieurs broches, qui servent à la sélection et la commande des moteurs connectés au module. Les fonctions des différentes broches sont indiquées dans la liste suivante :

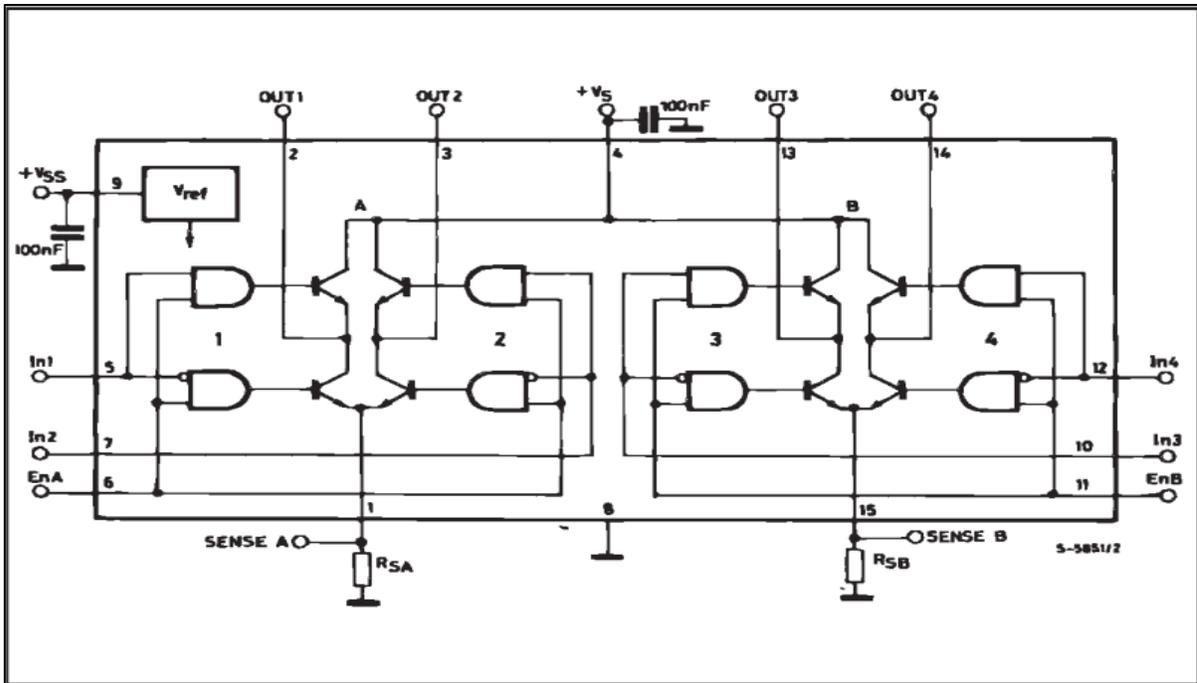


Figure (II.5.a) : Schéma bloc du module L298N [24]

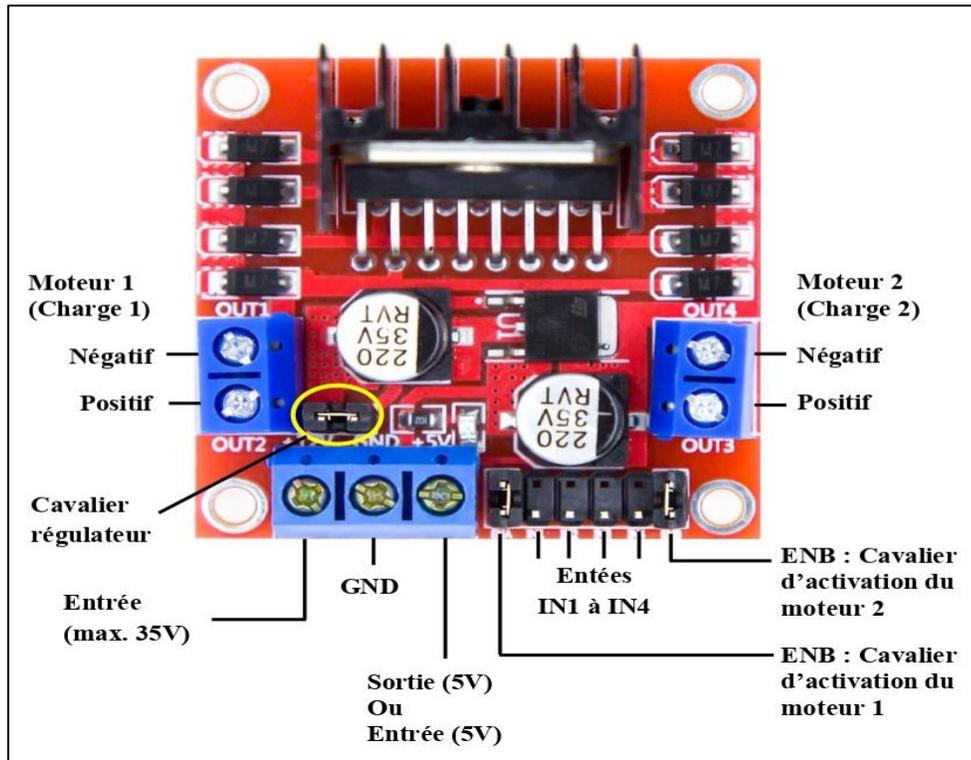


Figure (II.5.b) : Brochage du module L298N

La broche VCC : alimente le moteur. Une tension entre 5 et 35 V peut être appliquée. N'oubliez pas que si le cavalier 5V-EN est en place, vous devez fournir 2 volts supplémentaires par rapport à la tension réelle requise par le moteur, afin de faire fonctionner le moteur à sa vitesse maximale.

La broche GND : est la broche de masse commune.

La broche 5V : alimente le circuit logique de commutation à l'intérieur du circuit intégré L298N. Si le cavalier 5V-EN est en place, cette broche sert de sortie et peut être utilisée pour alimenter l'Arduino. Si le cavalier 5V-EN est retiré, vous devez le connecter à la broche 5V sur Arduino.

La broche ENA : utilisée pour contrôler la vitesse du moteur A. L'alimentation de cette broche avec la logique HIGH fait tourner le moteur A, l'alimentation avec la logique LOW provoque l'arrêt du moteur. En retirant le cavalier et en connectant cette broche à l'entrée PWM, nous pouvons contrôler la vitesse du moteur A.

Les broches IN1 et IN2 : utilisées pour contrôler la direction du moteur A. Si IN1 est HIGH et IN2 est LOW, le moteur A tourne dans une certaine direction. Pour changer de direction, faites IN1 LOW et IN2 HIGH. Si les deux entrées sont HIGH ou LOW, le moteur A s'arrête.

Les broches IN3 et IN4 : utilisées pour contrôler la direction du moteur B. Si IN3 est HIGH et IN4 est LOW, le moteur B tourne dans une certaine direction. Pour changer de direction, faites IN3 LOW et IN4 HIGH. Si les deux entrées sont HIGH ou LOW, le moteur B s'arrête.

La broche ENB : peut être utilisée pour contrôler la vitesse du moteur B. L'alimentation de cette broche avec le signal HIGH fait tourner le moteur B, l'alimentation LOW provoque l'arrêt du moteur. En éliminant le cavalier et en connectant cette broche aux informations PWM, nous pouvons contrôler la vitesse du moteur B.

Les broches OUT1 et OUT2 : sont connectées au moteur A.

Les broches OUT3 et OUT4 : sont connectées au moteur B [25].

II.3.1.3 Caractéristiques du module L298N

On indique les principales caractéristiques du module L298N :

- Borne d'alimentation (de la partie puissance) VMS : 5 - 35 V.
- Courant crête en sortie : 2 A par pont.
- Tension d'alimentation (de la partie logique): Vss de 4,5 à 5,5 V
- Intensité de courant dans la partie logique 0 à 36 mA.
- Signal de commande : niveau haut = 4.5 à 5.5 V, niveau bas = 0 V.
- Puissance consommée maximale : 20 W.

II.3.2 Module BTS7960, ou BT_2

II.3.2.1 Définition et fonctionnement

Le BTS7960, noté aussi BT_2, est un module de pilotage des moteurs électriques, fabriqué par Infineon Technologies. Il est conçu à la base d'un pont H. Il peut être utilisé pour contrôler un seul moteur électrique à la fois, au lieu de deux comme était le cas pour le module L298N.

Ce module peut supporter une tension d'entrée de 25 volts, et il peut faire circuler un courant qui peut atteindre jusqu'à 43 ampères. La puissance maximale transférée à travers le module BTS7960 est alors aux environs de 1kW, c'est une puissance relativement élevée par rapport à la puissance transférée par le module L298N qui est limitée à 70 watts.

II.3.2.2 Structure et brochage du module BTS7960

La figure II.6.a indiquée la structure interne d'un demi-pont du module BTS7960 [26]. Le demi-pont est composé de deux interrupteurs bidirectionnels. Pour former un pont H complet, deux demi-ponts seront combinés comme illustré sur la figure II.6.b. La figure II.6.b indique le brochage du module BTS7960. Il possède quatre bornes de puissance : deux bornes d'entrée B+ et B-, qui seront connectés à la source d'alimentation continu, et deux bornes de sortie M+ et M-, qui seront connecté au moteur commandé, ou généralement à la charge du pont H, dans le cas de son utilisation dans des applications autre que le control des moteurs.

En plus des bornes de puissance, le module BTS7960 possède plusieurs bornes de commande, dont les fonctions sont indiquées da la liste suivante [27] :

VCC : Fourniture d'alimentation modulaire + 5V.

GND : Terre.

IS-R : signal d'entrée pour détecter le courant élevé – rotation droite.

IS-L : signal d'entrée pour détecter le courant élevé – rotation inverse.

EN-R : Signal de sortie pour contrôler la direction du moteur – rotation droite.

EN-L : Signal de sortie pour contrôler la direction du moteur – rotation inverse.

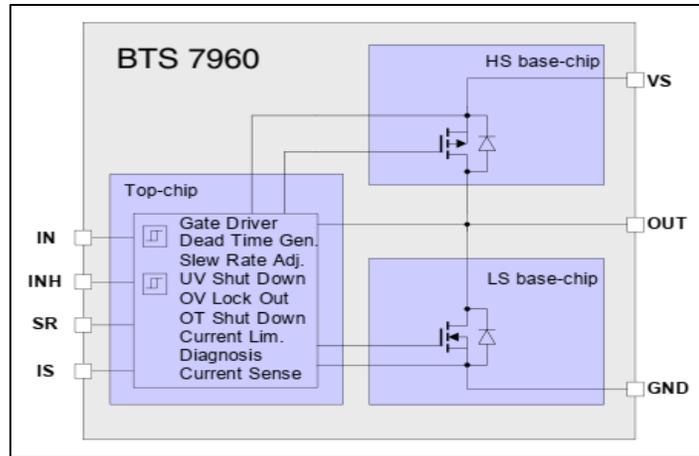


Figure (II.6.a) : Structure interne d'un demi-pont du module BTS7960 [26]

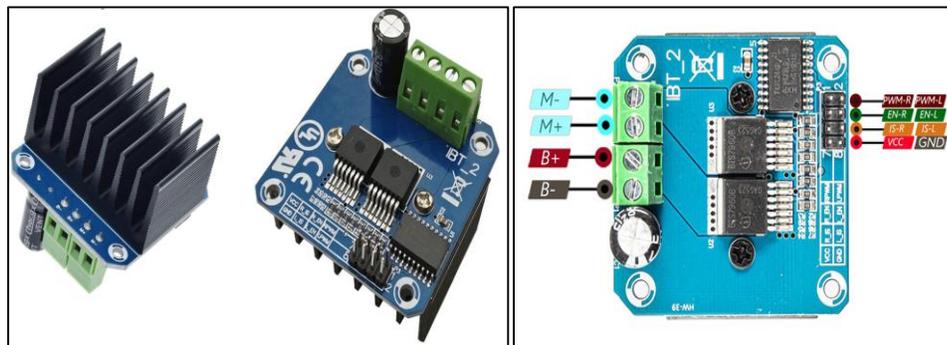


Figure (II.6.b) : Brochage du module BTS7960 [27]

R-PWM : pour le contrôle de la vitesse du moteur – rotation droite.

L-PWM : pour le contrôle de la vitesse du moteur – rotation inverse.

B+ et B- : bornes de puissance pour la connexion de la source continue.

M+ et M- : bornes de puissance pour l'alimentation de la charge.

II.3.2.3 Caractéristiques du module BTS7960

On indique les principales caractéristiques du module BTS7960 [28] :

- Pont H complet à base des transistors MOSFET (x4) avec diodes en antiparallèle
- La tension d'alimentation (partie puissance) : 5.5V à 27V.
- Alimentation partie commande : 3.3-5V.
- Courant maximal (partie puissance) : 43A.
- Fréquence maximale des entrées PWM : DC-25KHz.
- Protection contre surtension/surintensité.
- Module intègre deux capteurs du courant (sens 1 / sens2).

II.4 Accessoires de mesure

Lors de l'application de la commande en PWM sur le pont H, que ce soit celui du module L298N ou du module BTS7960, on a besoin de mesurer la tension et le courant de sa sortie. Ces mesures nécessitent l'utilisation de certains capteurs et instruments, tels que le capteur de courant INA219, et le convertisseur analogique numérique ADS1115 servant en capteur de tension. On présente dans ce qui suit la description et le mode de fonctionnement de ces éléments de mesure.

II.4.1 Convertisseur analogique numérique ADS1115

II.4.1.1 Définition et Pinnage

La forme de ce module est indiquée à la figure II.7. Il s'agit d'une puce électronique, permettant de faire de la conversion analogique numérique. Ainsi, par exemple, il est courant de s'en servir pour faire de la mesure de tension à intervalle régulier, et transmettre ces infos à un microcontrôleur [29].

Sur un module ADS1115 il y a 10 pins, dont les fonctions sont indiquées dans la liste suivante :

- **Vdd** : la tension d'alimentation (de 2 à 5,5 volts).
- **GND** : la masse (0 volt).
- **SCL et SDA** : les lignes d'horloge et de données en série (I2C).
- **ADDR** : un pin servant à définir physiquement l'adresse I2C du module (4 adresses différentes possibles).
- **ALRT** : une sortie « d'alerte », permettant par exemple à l'ADS1115 d'indiquer à un microcontrôleur qu'une tension a été mesurée, et que cette valeur est prête pour être lue.
- **A0, A1, A2, et A3** : les quatre entrées analogiques, sur lesquelles pourront être branchées les bornes des tensions à lire, en mode absolu ou en mode différentiel.

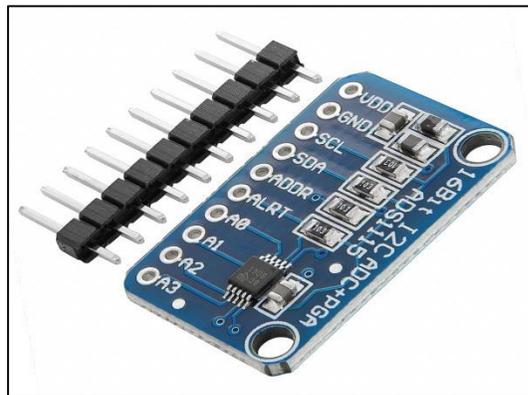


Figure (II.7) : Convertisseur analogique numérique ADS1115 [29]

II.4.1.2 Modes de mesure de tension

L'utilité du convertisseur analogique numérique **ADS1115** dans notre travail est qu'il permet la mesure de tension. Cette mesure de tension peut se faire de deux manières différentes [29] :

- **Lecture absolue** : Dans ce mode de lecture, on mesure une tension fixe positive, dans les limites de l'ADC, par rapport à la masse (0 volts). Le module BTS7960 peut mesurer quatre tensions en parallèle, via les broches A0, A1, A2, et A3. Ces tensions sont mesurées par rapport à la masse (GND).
- **Lecture différentielle** : Dans ce mode de lecture, on mesure une tension flottante, entre deux points quelconques, sans référentiel commun. La mesure se fait entre deux des quatre broches A0, A1, A2, et A3. Par exemple : entre A0 et A1, ou A2 et A3. On peut faire bien d'autres combinaisons, dans les limites de ce que propose la librairie Arduino utilisée.

II.4.1.3 Caractéristiques principales du module ADS1115

On présente les caractéristiques principales du convertisseur numérique analogique ADS1115 [29]:

- **Résolution de 16 bits** : Ce module a une résolution de 16 bits. Pour cette raison, son utilisation est préférable à l'utilisation des entrées analogiques de l'Arduino, car la résolution de l'Arduino n'est que de 10 bits (puisque la valeur retournée se situe entre 0 et 1023). Et Avec l'ADS1115, la résolution est de 15 bits (allant de 0 à 32767).
- **Tension d'alimentation** : La tension d'alimentation recommandée de l'ADS1115 est de 2 à 5,5 volts. La valeur maximale, à ne pas maintenir longtemps, peut aller jusqu'à 7 volts.
- **Plage de mesure** : Il est recommandé de mesurer des tensions qui ne dépassent pas 5 volts. On pourrait mesurer une tension jusqu'à 7 volts pendant un court instant. Les tensions négatives sont mesurables en mode différentielle. On peut utiliser le module BTS7960 pour mesurer des tensions plus importantes, en utilisant un diviseur résistif.
- **Précision** : La précision de mesure de l'ADS1115 dépend du gain de mesure. Ce gain est paramètre réglable à travers la librairie du module. Par exemple, si on utilise l'ADC avec un gain égal à 1, on aura une marge d'erreur plus ou moins égale à environ 0.375 mV.
- **Vitesse de lecture** : Sur l'ads1115, on peut sélectionner l'une des 8 vitesses d'échantillonnage possibles selon ses besoins :
 - ✓ DR=0 : 8 SPS (SPS : Sample Per Seconde, échantillon par seconde). Ici l'ADC aura un « data rate » égal à 8 lectures par seconde, ou 8 échantillons par seconde.

- ✓ DR=1 : 16 SPS
- ✓ DR=2 : 32 SPS
- ✓ DR=3 : 64 SPS
- ✓ DR=4 : 128 SPS. C'est la VALEUR PAR DÉFAUT (soit 128 échantillons mesurés par seconde)
- ✓ DR=5 : 250 SPS
- ✓ DR=6 : 475 SPS
- ✓ DR=7 : 860 SPS. Soit 860 lectures de tension par seconde.

II.4.1.4 Raccordement du Module ADS1115 avec ARDUINO

Pour réaliser un raccordement correct d'un convertisseur analogique-numérique ADS1115 avec une carte Arduino, d'abord nous devons assurer d'avoir installé la bibliothèque ADS1115 sur l'environnement Arduino. Puis, on va connecter notre ADS1115 à Arduino comme suivant :

- ✓ VCC de l'ADS1115 à VCC (3.3V ou 5V) de l'Arduino.
- ✓ GND de l'ADS1115 à GND de l'Arduino.
- ✓ SDA de l'ADS1115 à D18/SDA de l'Arduino.
- ✓ SCL de l'ADS1115 à D19/SCL de l'Arduino.
- ✓ ADDR de l'ADS1115 à GND.

La figure II.8 montre le raccordement des broches du module ADS1115 avec la carte Arduino.

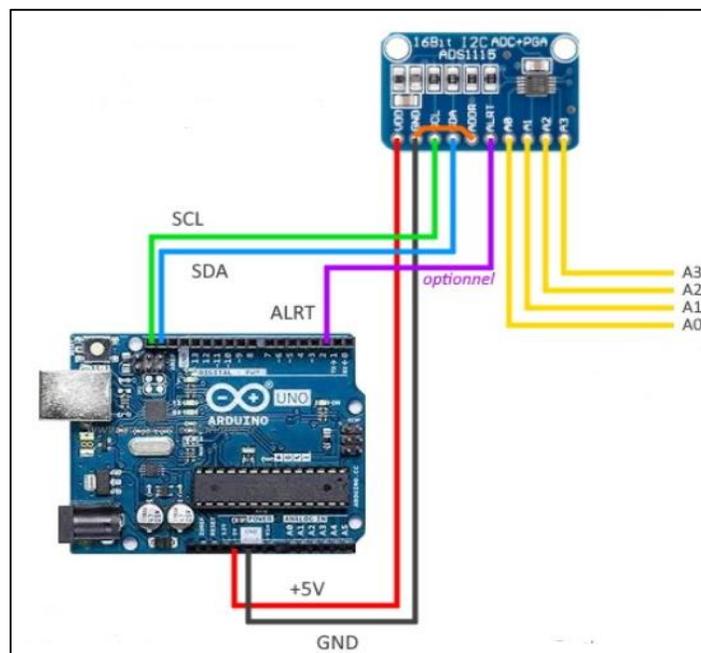


Figure (II.8) : Raccordement de l'ADS1115 avec ARDUINO [29]

II.4.1.5 Programmation et paramétrage de l'ADS1115

Pour pouvoir utiliser le module ADS115, on doit télécharger et intégrer sa librairie dans l'environnement Arduino IDE. Les principales instructions d'appel, de paramétrage et de lecture du module sont groupées dans la table II.5.

Initialisation (appel de la bibliothèque)	<pre>#include "ADS1X15.h" ADS1115 ADS(0x48); // ADS1115 physiquement défini à l'adresse 0x48 void setup() { ADS.begin(); // Paramètres suivants ici }</pre>
Paramétrage de gain	<pre>ADS.setGain(0) ; // ± 6.144 volt (par défaut). ADS.setGain(1) ; // ± 4.096 volt ADS.setGain(2) ; // ± 2.048 volt ADS.setGain(4) ; // ± 1.024 volt ADS.setGain(8) ; // ± 0.512 volt ADS.setGain(16) ; // ± 0.256 volt</pre>
Paramétrage de mode de lecture	<pre>ADS.setMode(1); // 0 = CONTINUOUS, 1 = SINGLE (default)</pre>
Paramétrage de la vitesse de lecture (fréquence d'échantillonnage)	<pre>ADS.setDataRate(7);</pre>
Demande de lecture de tension simple	<pre>int16_t mesure_A0 = ADS.readADC(0); int16_t mesure_A1 = ADS.readADC(1); int16_t mesure_A2 = ADS.readADC(2); int16_t mesure_A3 = ADS.readADC(3);</pre>
Demande de lecture de tension différentielle	<pre>float tension_A0 = ADS.toVoltage(mesure _0); float tension_A1 = ADS.toVoltage(mesure _1); float tension_A2 = ADS.toVoltage(mesure _2); float tension_A3 = ADS.toVoltage(mesure _3);</pre>

Tableau (II.5) : Instructions d'exploitation du module ADS1115 [29]

II.4.2 Capteur de tension et de courant INA219

II.4.2.1 Définition et Pinnage du module

La forme de ce module est indiquée à la figure II.9. C'est un capteur de courant et de tension communiquant avec l'Arduino via une interface I2C. Il est particulièrement adapté à la mesure de la puissance dans une chaîne de puissance, Les principales applications sont le contrôle de charge de batterie, et le suivi de consommation électrique [30]. Il peut mesurer une tension qui peut atteindre 26 volts, et un courant électrique allant jusqu'à 3.2 ampères.

Ce module a les six broches suivantes :

- **VCC** : sert à l'alimentation du module, par une tension de 5 volts.
- **GND** : Mis à la terre du module.
- **SDA** : sert avec la broche SCL à la communication des données vers la carte Arduino.
- **SCL** : sert avec la broche SDA à la communication des données vers la carte Arduino.
- **Vin+** : broche qu'on doit connecter à la borne + de la charge qu'on veut mesurer les variables.
- **Vin-** : broche qu'on doit connecter à la borne - de la charge qu'on veut mesurer les variables.

Les bornes Vin+ et Vin- sont séparées par une résistance shunt de valeur 0.1 Ohms.

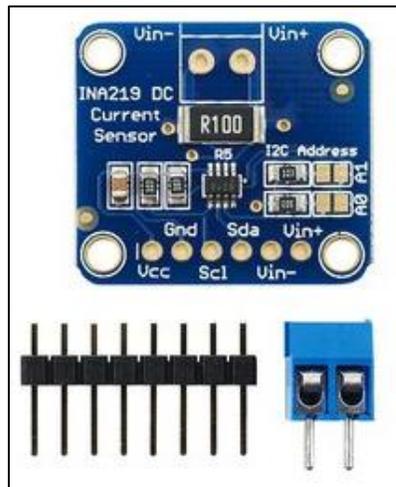


Figure (II.9) : Capteur de courant et de tension INA219 [30]

II.4.2.2 Caractéristiques principales du module

On site les principales caractéristiques du module INA219 [30]:

- **Précision de courant** : La précision de mesure du courant de la puce elle-même est très bonne, elle dépend de la précision de la résistance de détection, et il s'agira d'une précision de 1 %.
- **Précision de tension** : La précision de mesure de tension, sur toute la plage de température, est de 1 %.
- **Résolution de l'INA219** : La largeur de bits physique de L'ADS intégré à ce module est de 12 bits, mais tout au long de la fiche technique, la plage de bits discutée est de 15 bits. Ce qui se passe réellement, c'est que dans certains modes, on applique une moyenne) et obtenir une meilleure résolution, correspondante à 15 bits.

II.4.2.3 Raccordement du Module INA219 avec l'ARDUINO

Pour connecter l'Arduino UNO avec le capteur de courant INA219, nous allons effectuer les connexions suivantes :

- ✓ SDA de l'INA219 avec l'A4 de l'Arduino.
- ✓ SCL de l'INA219 avec l'A5 de l'Arduino.
- ✓ VCC de l'INA 219 avec 5V de l'Arduino.
- ✓ GND de l'INA219 à la masse de l'Arduino.
- ✓ VIN+ de l'INA219 à la source d'alimentation positive que nous allons mesurer.
- ✓ VIN- de l'INA219 à la source d'alimentation négative que nous allons mesurer.

La figure II.10 montre le raccordement des broches du module ADS1115 avec la carte Arduino.

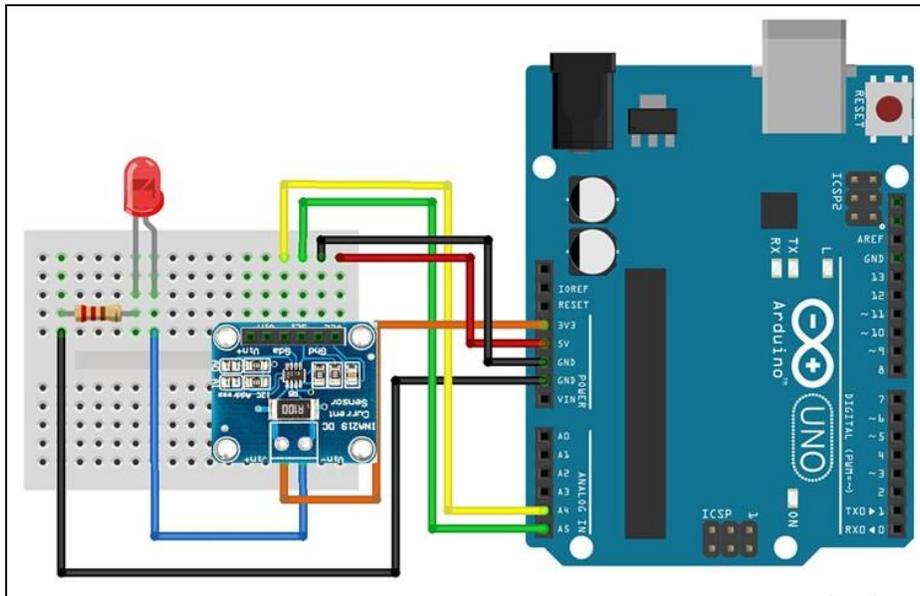


Figure (II.10) : Raccordement de l'INA219 avec ARDUINO [31]

II.4.2.4 Programmation et paramétrage de l'INA219

Pour pouvoir utiliser le module INA219, on doit télécharger et intégrer sa librairie dans l'environnement Arduino IDE. Les principales instructions d'appel, de paramétrage et de lecture du module sont groupées dans la table II.6 :

<p>Initialisation (appel de la bibliothèque)</p>	<pre>#include <Wire.h> #include <Adafruit_INA219.h> Adafruit_INA219 ina219;</pre>
<p>Paramétrage de la précision</p>	<pre>float shuntvoltage = 0; float busvoltage = 0; float current_mA = 0; float loadvoltage = 0; float power_mW = 0; shuntvoltage = ina219.getShuntVoltage_mV(); busvoltage = ina219.getBusVoltage_V(); current_mA = ina219.getCurrent_mA(); power_mW = ina219.getPower_mW(); loadvoltage = busvoltage+(shuntvoltage / 1000);</pre>
<p>Instructions d’affichage (tension, courant, et puissance)</p>	<pre>Serial.print("Bus Voltage: "); Serial.print(busvoltage); Serial.print("Shunt Voltage: "); Serial.print(shuntvoltage); Serial.print("Load Voltage: "); Serial.print(loadvoltage); Serial.print("Current: "); Serial.print(current_mA); Serial.print("Power: "); Serial.print(power_mW);</pre>

Tableau (II.6) : Instructions d’exploitation du module INA219 [31]

II.5 Conclusion

Dans ce chapitre nous avons présenté une description générale de la carte Arduino. On a décrit les différentes broches d’entrée sortie de la carte, ainsi que l’outil de développement de programme compatible à cette carte.

Par suite, nous avons donné une description de deux modules électroniques, le module L298N et le module BTS7960, en guise de leur utilisation en tant qu’onduleurs monophasé en pont complet.

On a présenté en fin du chapitre les éléments de mesures du courant et de la tension, qui sont requis pour la commande de l’onduleur et la visualisation des signaux de sortie résultant de cette commande.

Dans le chapitre suivant, on va implémenter, en utilisant la carte Arduino UNO, différents algorithmes de commande sur les deux modules L298N et BTS7960. Ces commandes imposent un fonctionnement en onduleur sur ces deux module

Chapitre III



Implémentation de la commande de l'onduleur sur une carte ARDUINO



III.1 Introduction

Dans ce chapitre nous allons implémenter les stratégies de commande sur les deux modules L298N et BTS7960, en utilisant la carte Arduino UNO. Ces commandes imposent un fonctionnement en onduleur sur ces deux modules. On va implémenter les techniques de commande simulées dans le chapitre 1, à savoir la commande en pleine onde symétrique et décalée, en plus de la MLI bipolaire et unipolaire. La visualisation des signaux de sortie peut être effectuée soit sur un oscilloscope, soit sur le PC en récupérant les données à travers un capteur de tension.

III.2 Schéma de la commande et banc d'essai

La figure III.1 indique le schéma global et les différents blocs de l'implantation de la commande. La carte Arduino est connectée à un PC. Celui-ci assure d'un côté l'alimentation en puissance de la carte, et permet d'introduire ou de modifier le code exécuté par la carte.

L'onduleur (module L298N ou BTS7960) reçoit les signaux de commande à partir de la carte Arduino. Son alimentation en tension continu est assurée par une source externe à travers les broches V_{in} et GND. Le capteur de tension ADS1115 mesure la tension de sortie de l'onduleur, et envoie cette mesure vers l'Arduino, pour être récupérée et tracée par le PC. Cette tension de sortie peut être visualisée directement sur un oscilloscope.

La figure I.2 indique une photographie des montages réalisés. La figure 1.2.a représente le montage réalisé pour commander le module L298N, tandis que la figure 1.2.b. représente le montage réalisé pour la commande du module BTS7960. Chacune des deux figures montre un PC pour la programmation et l'alimentation de la carte Arduino UNO, la carte elle-même, le module à commander et son alimentation, le capteur de tension ADS1115, et un oscilloscope de type UNT-T UTD2120 CEX+.

On va appliquer les différentes techniques de commande sur les deux modules, en considérant deux cas pour la tension d'alimentation : une tension de 5volts, et une tension 25 volts. Pour le cas d'une alimentation par une tension de 5volts, on présente la tension d'alimentation visualisée par l'oscilloscope, et celle fournie par le capteur de tension ADS1115. Pour le cas d'alimentation par une tension de 25 volts, on présente la tension de sortie visualisée par l'oscilloscope. Dans ce cas, le capteur ADS1115 est incapable de fournir une mesure, car il est destiné à mesurer des tensions inférieures ou égales à 5 volts. Nous avons essayé de résoudre ce problème en utilisant un pont résistif,

mais malheureusement le courant résultant dépasse les limites du capteur est celui-ci capteur a été détérioré.

L'échelle utilisée dans l'oscilloscope pour les résultats obtenus par l'application de 5V est 0.02ms pour un carré de l'axe horizontale, et 2V pour un carré de l'axe vertical. Celle obtenus par l'application de 25V ont une échelle de 0.02ms pour un carré de l'axe horizontale, et 10V pour un carré de l'axe vertical (sauf que la figure III.5 qui a une échelle de 5V pour un carré de l'axe vertical).

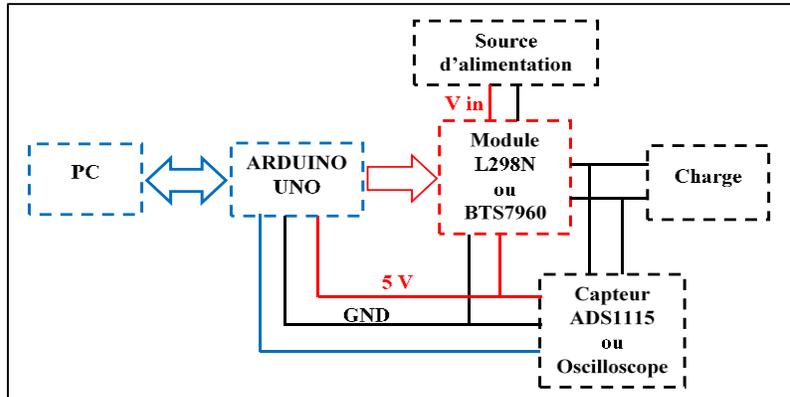
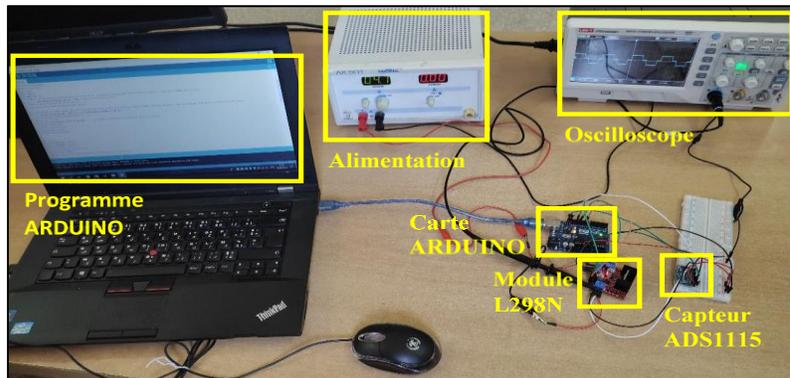
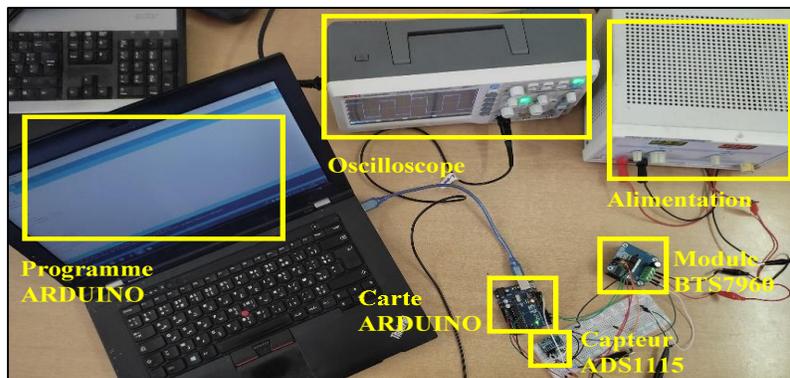


Figure (III.1) : schéma de la commande de l'onduleur monophasé



a. Montage pour la commande du module L298N



b. Montage pour la commande du module BTS7960

Figure (III.2) : Photographie du montage réalisé

III.3 Application de la commande en pleine onde sur le module L298N

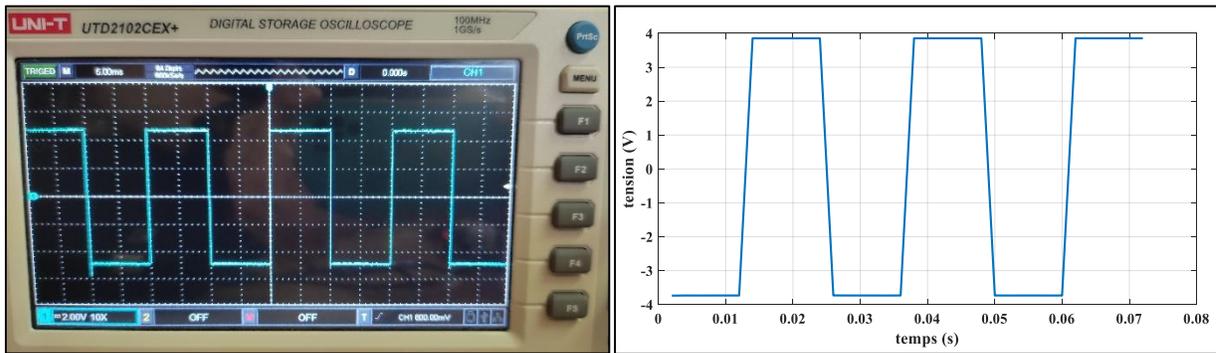


Figure (III.3) : Tension de sortie correspondant à $V_{in} = 5V$

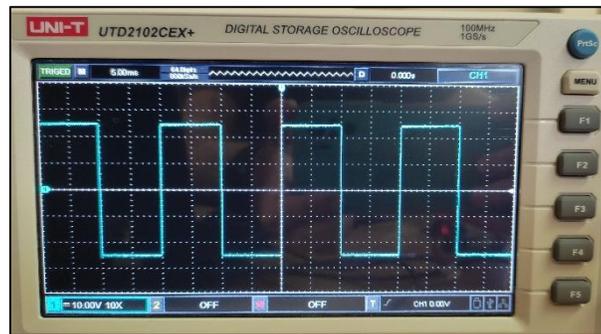


Figure (III.4) : Tension de sortie correspondant à $V_{in} = 25V$

III.4 Application de la commande en pleine onde sur le module BTS7960

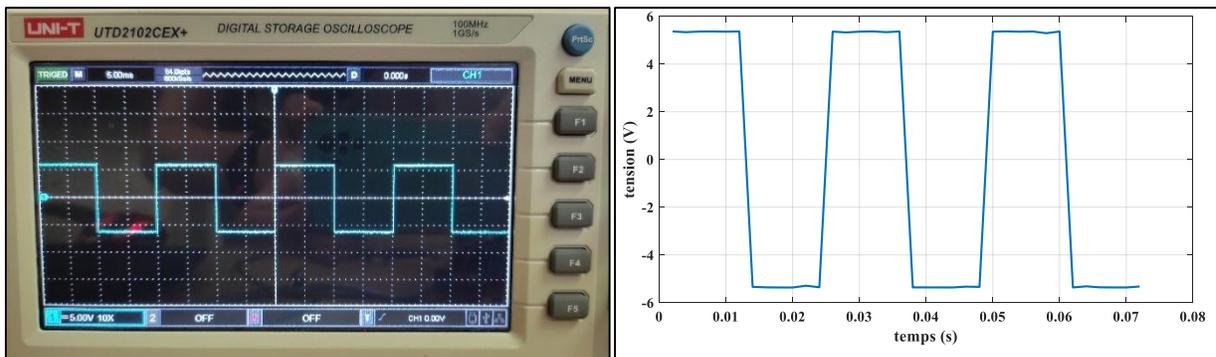


Figure (III.5) : Tension de sortie correspondant à $V_{in} = 5V$

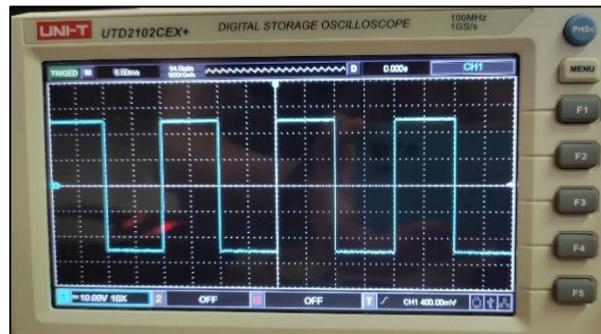


Figure (III.6) : Tension de sortie correspondant à $V_{in} = 25V$

Interprétations des résultats de la commande en pleine onde :

- ✓ Comme prévu, et conformément aux résultats de simulation présentés dans la figure I.10 du chapitre 1, les figures III.3-6 montrent que la tension de sortie de l'onduleur bascule entre deux valeurs $+V_{in}$ et $-V_{in}$.
- ✓ La forme de la tension tracée à partir des mesures fournies par le capteur de tension ads1115 présente une certains pente lors du passage de $+V_{in}$ à $-V_{in}$ et de $+V_{in}$ à $-V_{in}$. Cette pente est expliquée par la vitesse des mesures de capteur, relativement très faible en la comparant avec la vitesse de l'oscilloscope. En effet, la vitesse (la fréquence) maximale d'échantillonnage du capteur est de 860 SPS, soit 860 mesure de tension par seconde. Par contre, la fréquence de l'oscilloscope UNT-T UTD2120 CEX+ est de 100 Mhz.

III.5 Application de la commande en pleine onde décalée sur le module L298N

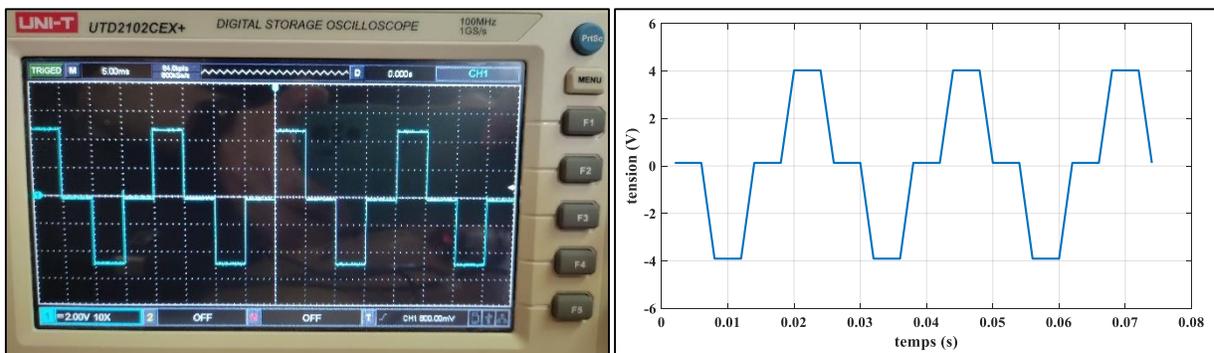


Figure (III.7) : Tension de sortie correspondante à $V_{in} = 5V$

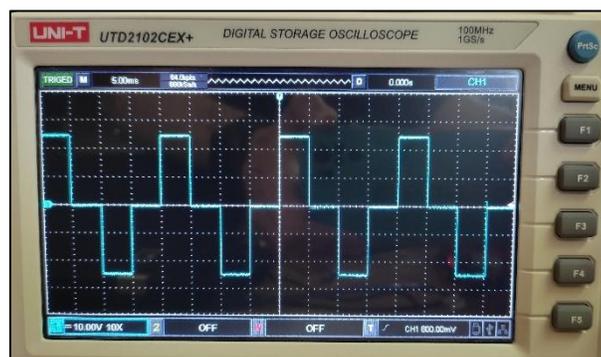


Figure (III.8) : Tension de sortie correspondante à $V_{in} = 25V$

III.6 Application de la commande en pleine onde décalée sur le module BTS7960

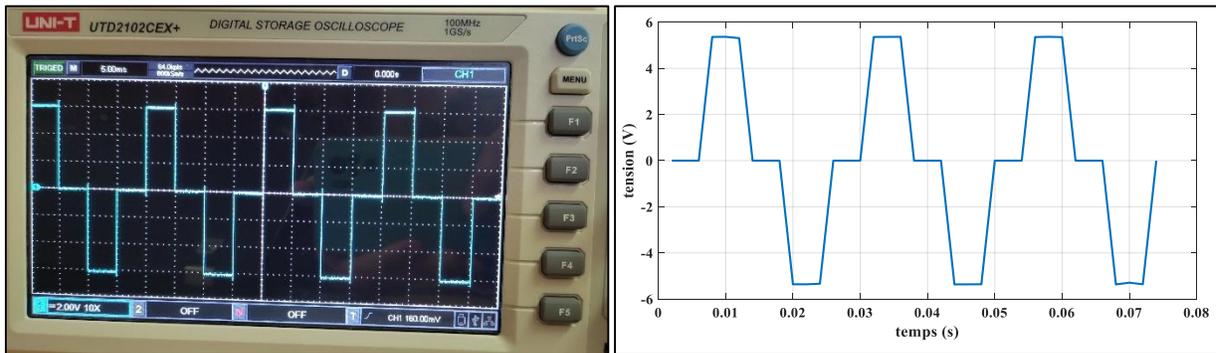


Figure (III.9) : Tension de sortie correspondante à $V_{in} = 5V$

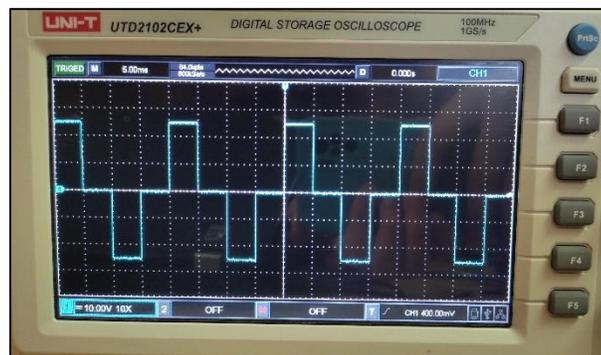


Figure (III.10) : Tension de sortie correspondante à $V_{in} = 25V$

Interprétations des résultats de la commande en pleine onde décalée :

- ✓ Les figures III.7-10 montrent que la tension de sortie bascule entre trois niveaux de tension 0 , $+V_{in}$ et $-V_{in}$, ce qui est conforme avec les résultats de simulation indiqués sur le figure I.13 du chapitre 1.
- ✓ La pente apparente sur la forme de la tension tracée à partir des mesures fournies par le capteur de tension ads1115 est due à la vitesse des mesures de capteur, relativement faible.

III.7 Application de la commande MLI triangulo-sinusoidale bipolaire sur le module L298N

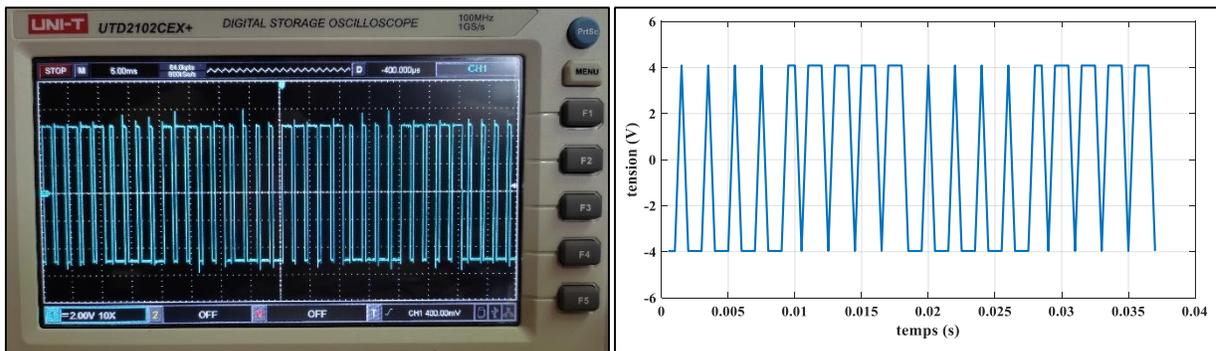


Figure (III.11) : Tension de sortie correspondante à $V_{in} = 5V$

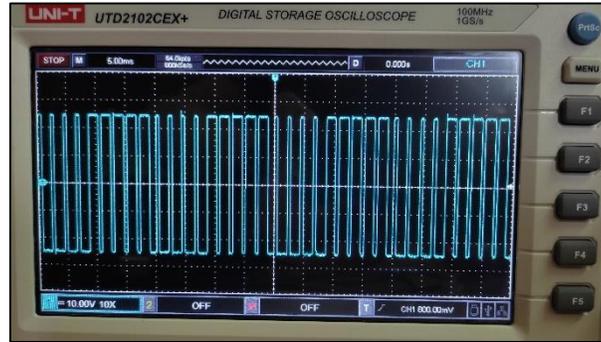


Figure (III.12) : Tension de sortie correspondante à $V_{in} = 25V$

III.8 Application de la commande MLI triangulo-sinusoidale bipolaire sur le module BTS7960

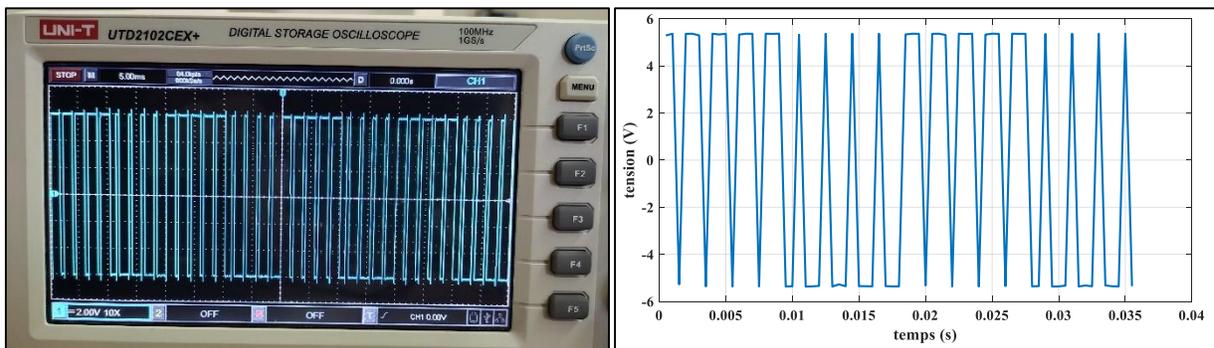


Figure (III.13) : Tension de sortie correspondante à $V_{in} = 5V$

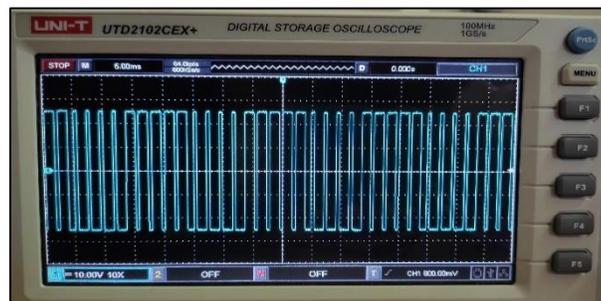


Figure (III.14) : Tension de sortie correspondante à $V_{in} = 25V$

Interprétations des résultats de la commande MLI triangulo-sinusoidale bipolaire :

- ✓ Les figures III.11-14 montrent un basculement à fréquence élevée de la tension de sortie entre deux niveau $+V_{in}$ et $-V_{in}$, conformément aux résultats de simulation indiqués dans la figure I.19 du premier chapitre.
- ✓ Les pentes et les anomalies constatées sur la forme de la tension fournie par le capteur de tension sont dues à sa faible fréquence de mesure.

III.9 Application de la commande MLI unipolaire à deux références sur le module L298N

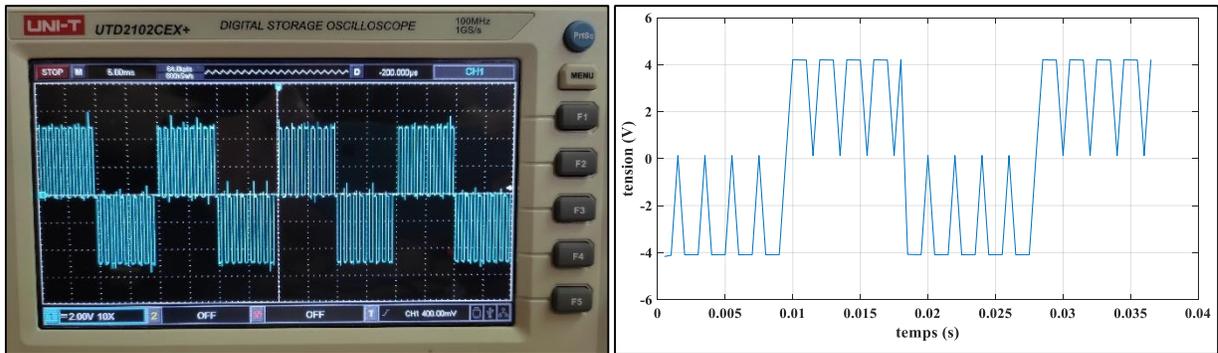


Figure (III.15) : Tension de sortie correspondante à $V_{in} = 5V$

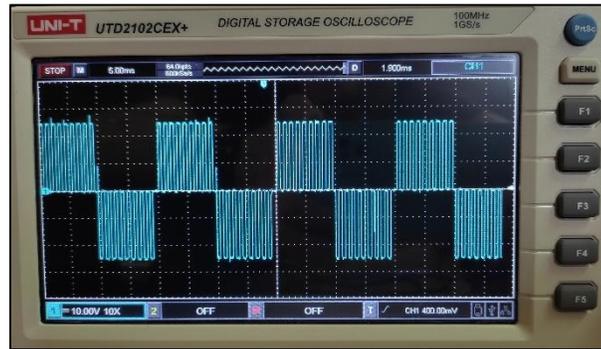


Figure (III.16) : Tension de sortie correspondante à $V_{in} = 25V$

III.10 Application de la commande MLI unipolaire à deux références sur le module BTS7960

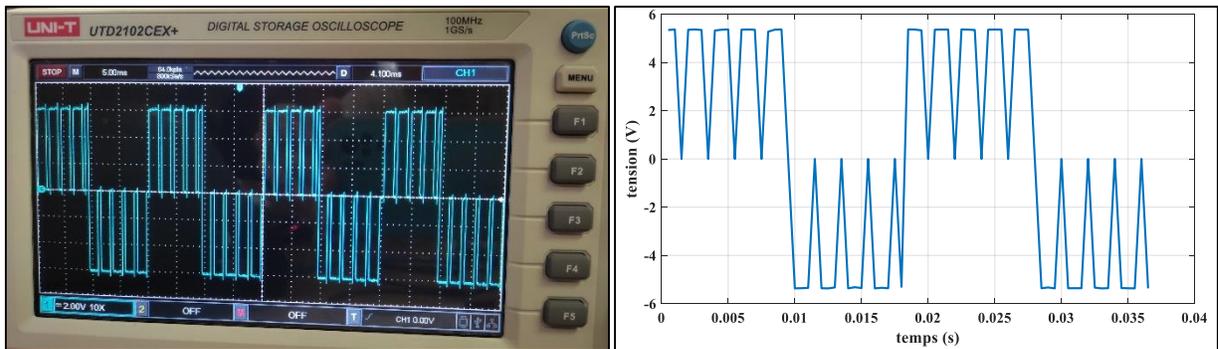


Figure (III.17) : Tension de sortie correspondante à $V_{in} = 5V$

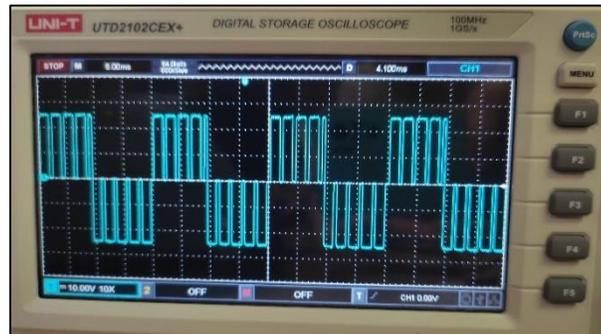


Figure (III.18) : Tension de sortie correspondante à $V_{in} = 25V$

Interprétations des résultats de la commande MLI unipolaire à deux références :

- ✓ Les figures III.15 -18 montrent que pendant la première demi période, la tension de sortie bascule entre 0, + V_{in} et pendant la deuxième demi période, elle bascule entre 0 et $-V_{in}$: Cette remarque est conforme aux résultats de simulation indiqués dans le chapitre 1, sur la figure I.30
- ✓ L'affichage des résultats à travers l'oscilloscope est bien précis que l'affichage à travers le capteur de tension ADS1115.

III.11 Application de la commande MLI unipolaire à deux porteuses sur le module L298N

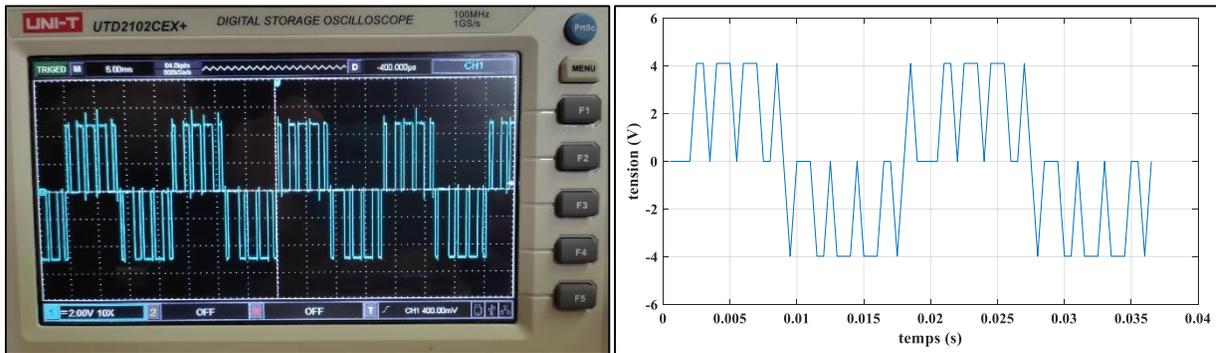


Figure (III.19) : Tension de sortie correspondante à $V_{in} = 5V$

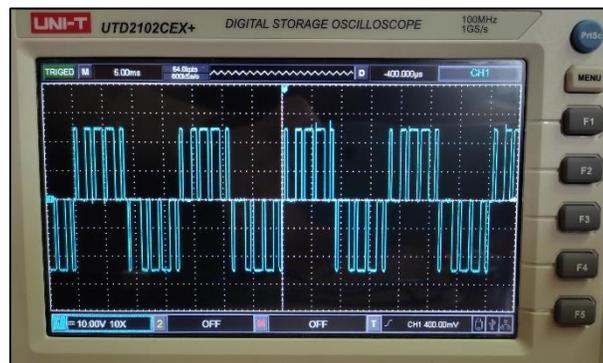


Figure (III.20) : Tension de sortie correspondante à $V_{in} = 25V$

III.12 Application de la commande MLI unipolaire à deux porteuses sur le module BTS7960

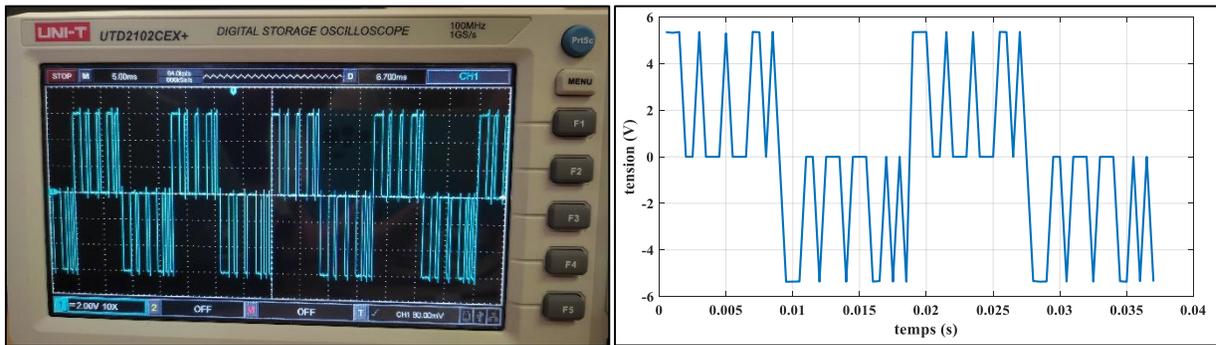


Figure (III.21) : Tension de sortie correspondante à $V_{in} = 5V$

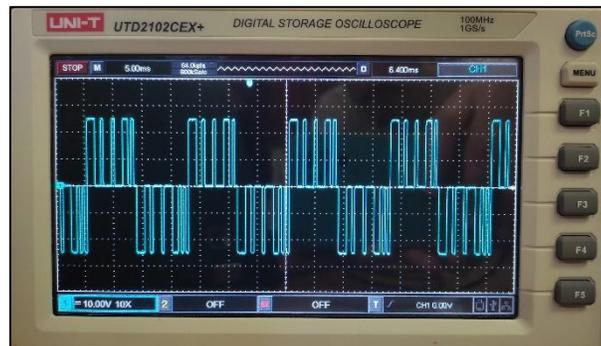


Figure (III.22) : Tension de sortie correspondante à $V_{in} = 25V$

Interprétations des résultats de la commande MLI unipolaire à deux porteuses :

- ✓ Les figures III.19 -22 montrent que la tension de sortie bascule entre 0 , $+V_{in}$ pendant la première demi période, et entre 0 et $-V_{in}$ pendant la deuxième demi période, ce qui est conforme aux résultats de simulation présentés sur la figure I.41 dans le chapitre 1.

III.13 Conclusion

Ce chapitre a été dédié à la mise en œuvre pratique des différentes stratégies de commandes de l'onduleur en pont complet sur la carte Arduino UNO. Mise à part les erreurs de mesures, les résultats obtenus sont conformes à celles obtenus par simulation. La qualité des capteurs et la nature des modules utilisés comme onduleur nous a imposé de faire nos tests pour des faibles niveaux de tension d'alimentation.

Conclusion Générale

Conclusion générale :

Le but visé à travers ce travail était à explorer la mise en œuvre d'un système de contrôle d'un onduleur monophasé en pont complet basé sur une carte Arduino qui génère les impulsions de commande pour piloter les commutateurs du pont en H et réguler la tension et la fréquence de sortie. Quatre stratégies de commande ont été implémenter.

Cette implémentation permet d'avoir un contrôle suffisant d'un onduleur en pont complet à travers un minimum du matériel requis.

En premier lieu, on a commencé, par exposer des notions fondamentales sur l'onduleur monophasé. Ainsi, nous avons présenté quelques ses modes de contrôle et leurs simulations avec MATLAB/Simulink.

Le second chapitre a été consacré à donner la description de la carte Arduino comme étant une plateforme de contrôle, les composants matériels requis pour l'application de la commande du convertisseur en pont complet (L289N et BTS7960). En fin de ce chapitre, nous avons présenté les éléments de mesures du courant et de tension qui ont pour le but de visualiser les signaux de sortie résultants pour chaque type de commande.

Le dernier chapitre a été dédié à implémenter les stratégies de commande de l'onduleur en pont complet que nous l'avons simulé dans le chapitre I sur la carte Arduino UNO.

Après la visualisation des signaux de sortie sur l'oscilloscope et le PC, nous avons remarqué que :

- ✓ Les résultats obtenus dans le chapitre III sont compatibles à celles obtenus par simulation du chapitre I.
- ✓ La qualité des capteurs et la nature des modules utilisés comme onduleur nous a imposé de faire nos tests pour des faibles niveaux de tension d'alimentation.
- ✓ Les résultats obtenus ont démontré la faisabilité et l'efficacité de l'utilisation d'une carte Arduino pour la commande d'un onduleur monophasé en pont.
- ✓ La forme de la tension tracée à partir des mesures fournies par le capteur de tension ADS1115 présente une certaine pente lors du passage de $+V_{in}$ à $-V_{in}$ et de $+V_{in}$ à $-V_{in}$.

CONCLUSION GENERALE

Cette pente est expliquée par la fréquence des mesures de capteur, relativement très faible en la comparant avec la vitesse de l'oscilloscope.

- ✓ L'application de l'alimentation 25V utilisant du convertisseur ADS1115 n'est pas implémentable même avec l'utilisation du pont résistif, puisque ce dernier permet de contrôler seulement la tension, mais le courant résultant a dépassé les limites du capteur, donc l'en damage.
- ✓ La nature du courant de sortie des deux modules récupéré par le capteur de tension et du courant INA219 est perturbée, et pour cette raison, nous n'avons pu d'implémenter la commande en courant par hystérésis.

Les perspectives envisagées dans le cadre de ce travail sont :

- ✓ Une réalisation pratique de l'onduleur monophasé en pont complet prise en compte le choix de la carte Arduino à utiliser après.
- ✓ Un choix de forte tension et puissance des éléments de mesures.

Bibliographie

- [1] Hansruedi Bühler « Convertisseurs statiques »,1991
- [2] Melle Rabea Guedouani, « commande d'un onduleur de tension multiniveau, application à la conduite d'une machine synchrone à aimant permanente », Mémoire Magister, Ecole national polytechnique, juin 1998
- [3] BAHI Abdelouahab & DJEDIDI Mohammed Riadh, « Commande d'un Onduleur Triphasé par Différentes Techniques MLI en Utilisant la Carte DSPACE », Mémoire MASTER, Université Echahid Hamma Lakhdar d'El-oued, Mai 2017
- [4] Violaine Didier « Les Onduleurs pour Systèmes Photovoltaïques Fonctionnement, Etat de l'Art et Etude des Performances », Rapport, Août 2007.
- [5] Berroubi fateh & Kharchi ouilid, « technique de commande d'un onduleur à MLI associe à une machine à induction », Mémoire Magister, Université de M'sila, 2006.
- [6] Brahimi Mohamed el Amine, Babah Azmi, « commande par mli vectorielle d'un onduleur alimentant un moteur a cage », Mémoire D'ingénieur D'état, Université de M'sila, 2008.
- [7] Oussama degla & terek sayah lembarek, « Etude par simulation d'un onduleur », Mémoire Master, université Kasdi Merbah Ouargla, 2020.
- [8] Mr: Soyed-Abdessami, Cours d'électronique de puissance, ISET de Nabeul, 2013.
- [9] G. Pinson, "Onduleurs ", Cours de physique appliquée, Académie d'Aix-Marseille, 2008.
- [10] Francis Milsant, « Electrotechnique, Electronique de puissance : Cours et problèmes » Ellipses, 1993.
- [11] S. Bendaikha & A. Djermane, « Etude sur la commande des onduleurs en pont triphasés », Mémoire D'ingénieur D'état, Centre Universitaire d'Oum El-Bouaghi, 2005.
- [12] Guy Séguier, « électronique de puissance, Structures, commandes, applications » 10e édition, août 2015.

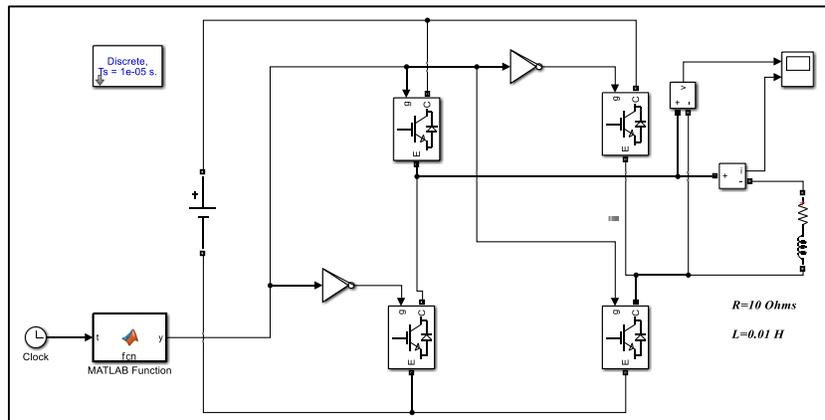
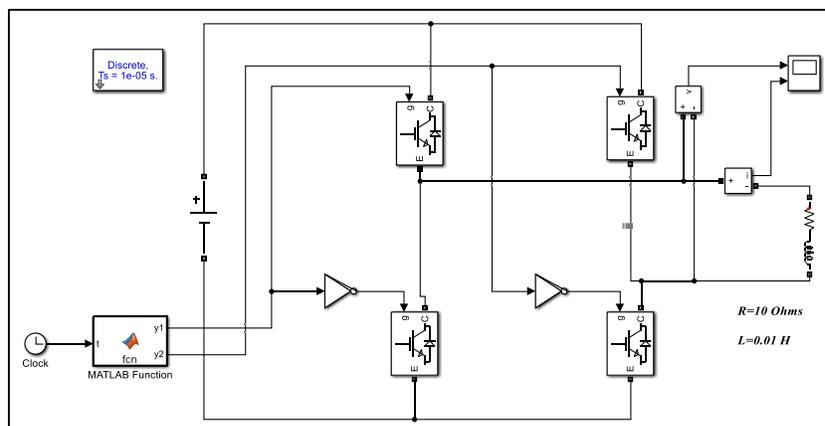
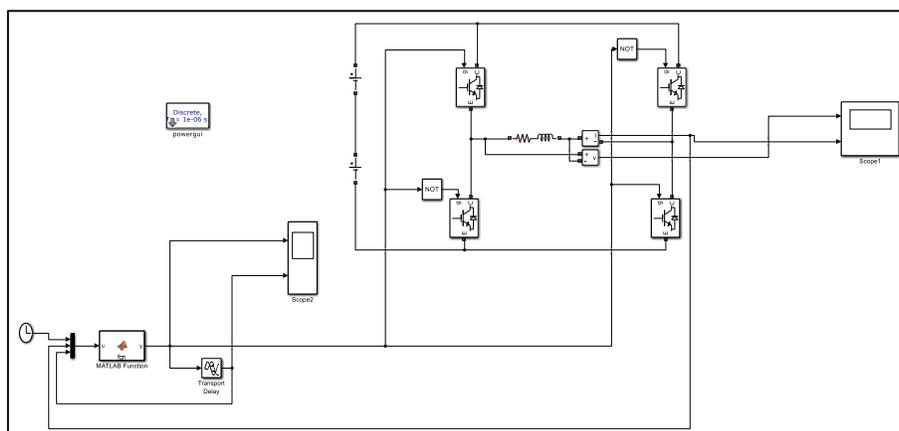
- [13] Boukaroura Abdelkader, « Modélisation et Diagnostic d'un Onduleur Triphasé par l'Approche BondGraph », Mémoire Magister, Université Ferhat Abbas De Setif, 2009.
- [14] Mecheri Salah-Eddine, « Réalisation D'un Onduleur De Tension Monophasé Contrôlé Par Une Carte Arduino », Mémoire Master, Université Constantine I 2014.
- [15] Adjimi Nadia & Belaidi Wahiba « Modélisation et commande d'un onduleur MLI », Mémoire Master, Universitaire Larbi Ben M'hidi Oum El-Bouaghi, 2009
- [16] El Ajjouri Tarik, « Méthode Avancée De Contrôle Commande D'un Onduleur Monophasé Pour Une Réduction De La Banque De Condensateurs, En Vue D'accroître La Durée De Vie Du Convertisseur », Mémoire de Maitrise Université Du Québec À Montréal, DÉCEMBRE 2018.
- [17] Bouteraa Imad Eddine, « Commande en V/f en boucle ouverte du moteur à induction basé sur la modulation de largeur d'impulsion vectorielle hybride », Mémoire Master, Université De M'sila 2013/2014.
- [18] John Nussey, « Arduino pour les nuls », 2e édition, mai 2017.
- [19] Nicolas Goilav & Geoffrey Loi, « Arduino : Apprendre à développer pour créer des objets intelligents », juin 2015.
- [20] Erik Bartmann, « le grand livre d'ARDUINO », 2e édition, 2015.
- [21] Bellili Sana'a, « Conception et Réalisation d'un Robot d'Inspection des Pipelines à base de Carte Arduino UNO », Mémoire De Master, Université Mohamed Khider de Biskra, juillet 2019.
- [22] Structure de la carte Arduino :
<https://www.gcworks.fr/tutoriel/arduino/UNO.html>
- [23] Définition et fonctionnement du module L298N :
<https://www.robotique.tech/tutoriel/commander-un-moteur-par-arduino-et-le-module-l298n/>
- [24] Datasheet du module L298N:
<https://www.alldatasheet.com/view.jsp?Searchword=L298N>
- [25] Structure et brochage du module L289N :
<https://www.etechnophiles.com/l298n-motor-driver-pin-diagram/>

- [26] Datasheet du module BTS7960 :
<https://www.alldatasheet.com/view.jsp?Searchword=BTS7960>
- [27] Structure et brochage du module BTS7960 :
<https://electropeak.com/learn/interfacing-bts7960-43a-high-power-motor-driver-module-with-arduino/>
- [28] Caractéristique du module BTS7960
<https://www.electronique-mixte.fr/onduleur-avec-arduino-1-pont-h-complet-commande-symetrique/>
- [29] Convertisseur analogique numérique ADS1115 :
<https://passionelectronique.fr/tutorial-ads1115/>
- [30] Capteur de tension et de courant INA219 :
<https://www.best-microcontroller-projects.com/ina219.html>
- [31] Raccordement de l'INA219 avec ARDUINO, Instructions d'exploitation du module INA219 :
<https://electropeak.com/learn/interfacing-ina219-current-sensor-module-with-arduino/>



Annexes



Annexe 1 : Schéma bloc de la commande pleine onde et la commande MLI bipolaire**Annexe 2 : Schéma bloc de la commande décalée et la commande MLI unipolaire****Annexe 3 : Schéma bloc de la commande en courant par hystérésis**

Annexe 4 : Code Arduino pour le contrôle du module L298N avec la fonction du capteur de tension ADS1115 :

✓ *La Commande Pleine Onde*

```

#include "ADS1X15.h"
#define T_ms    10
int commande1 = 13;
int commande2 = 12;
int activer = 9;
ADS1115 ADS(0x48);
unsigned long T_alfa;
unsigned long T_0;
float t_m;
float p_m=2;
void mesure (){
  int16_t difference_potentiel_A0_A1 = ADS.readADC_Differential_0_1();
  float tension_volts_A0_A1 = ADS.toVoltage(difference_potentiel_A0_A1);
  Serial.print(" ");
  Serial.print(tension_volts_A0_A1,3);
}
void setup() {
  pinMode(commande1, OUTPUT);
  pinMode(commande2, OUTPUT);
  pinMode(activer, OUTPUT);
  Serial.begin(9600);
  Serial.println("Lecture de tensions différentielles / ADS1115");
  Serial.println("-----");
  Serial.println("Mesure de différence de potentiel entre les entrées A0 et A1 de l'ADC");
  Serial.println("Nota : ADC Range: +/- 6.144V (soit 1 bit = 0.1875 mV)");
  Serial.println("");
  ADS.begin();
  ADS.setGain(0);    ADS.setMode(1);
  ADS.setDataRate(7);
  ADS.readADC(0);
}
void loop() {
  digitalWrite(activer, HIGH);
  digitalWrite(commande1, HIGH);
  digitalWrite(commande2, LOW);
  t_m=0;
  while (t_m<= T_ms){
    mesure ();
    t_m = t_m + p_m;
  }
  digitalWrite(activer, HIGH);
  digitalWrite(commande1, LOW);
  digitalWrite(commande2, HIGH);
  //delay(T_0);
  t_m=0;
  while (t_m<= T_ms){
    mesure() ;
    t_m = t_m + p_m;
  }
}

```

✓ *La Commande Décalée*

```

#include "ADS1X15.h"
#define T_ms    20
#define Alfa    PI
int commande1 = 13;
int commande2 = 12;
int activer = 9;
ADS1115 ADS(0x48);
unsigned long T_alfa;
unsigned long T_0;
float t_m;
float p_m=2;
void mesure (){
    int16_t difference_potentiel_A0_A1 = ADS.readADC_Differential_0_1();
    float tension_volts_A0_A1 = ADS.toVoltage
    Serial.print(" ");
    Serial.print(tension_volts_A0_A1,3);
}
void setup() {
pinMode(commande1, OUTPUT);
pinMode(commande2, OUTPUT);
pinMode(activer, OUTPUT);
    Serial.begin(9600);
    Serial.println("Lecture de tensions différentielles / ADS1115");
    Serial.println("-----");
    Serial.println("Mesure de différence de potentiel entre les entrées A0 et A1 de l'ADC");
    Serial.println("Nota : ADC Range: +/- 6.144V (soit 1 bit = 0.1875 mV)");
    Serial.println("");
    ADS.begin();
    ADS.setGain(0);    ADS.setMode(1);
    ADS.setDataRate(7);    ADS.readADC(0);
    T_alfa=round((double)T_ms*Alfa/(2.0*PI));
    T_0=(T_ms-T_alfa)>>1;
}
void loop() {
digitalWrite(activer, HIGH);
digitalWrite(commande1, LOW);
digitalWrite(commande2, LOW);
//delay(T_alfa>>1);
t_m=0;
while (t_m<= T_alfa>>1){
    mesure ();
    t_m = t_m + p_m;
}
}

```

```
digitalWrite(activer, HIGH);
digitalWrite(commande1, HIGH);
digitalWrite(commande2, LOW);
//delay(T_0);
t_m=0;
while (t_m<= T_0){
  mesure() ;
  t_m = t_m + p_m;
}
digitalWrite(activer, HIGH);
digitalWrite(commande1, LOW);
digitalWrite(commande2, LOW);
//delay(T_alfa>>1);
t_m=0;
while (t_m<= T_alfa>>1){
  mesure ();
  t_m = t_m + p_m;
}
digitalWrite(activer, HIGH);
digitalWrite(commande1, LOW);
digitalWrite(commande2, HIGH);
//delay(T_0);
t_m=0;
while (t_m<= T_0){
  mesure ();
  t_m = t_m + p_m;
}
}
```

✓ *La Commande MLI Bipolaire*

```

#include "ADS1X15.h"
int commande1 = 13;
int commande2 = 12;
int activer = 9;
ADS1115 ADS(0x48);
int g1;
int g2;
int i=0;
float j=0;
float vmax=0.8;
float m=9;
float N=36;
float pas;
int Np;
float Vp;
float Vref;
void mesure (){
    int16_t difference_potentiel_A0_A1 = ADS.readADC_Differential_0_1();
    float tension_volts_A0_A1 = ADS.toVoltage(difference_potentiel_A0_A1);
    Serial.print(" ");
    Serial.print(tension_volts_A0_A1,3);
}
void setup() {
pinMode(commande1, OUTPUT);
pinMode(commande2, OUTPUT);
pinMode(activer, OUTPUT);
    Serial.begin(9600);
    Serial.println("Lecture de tensions différentielles / ADS1115");
    Serial.println("-----");
    Serial.println("Mesure de différence de potentiel entre les entrées A0 et A1 de l'ADC");
    Serial.println("Nota : ADC Range: +/- 6.144V (soit 1 bit = 0.1875 mV)");
    Serial.println("");
    ADS.begin();
    ADS.setGain(0);      ADS.setMode(1);
    ADS.setDataRate(7); ADS.readADC(0);
    pas=2*PI/N;
    Np=N/m;
}
void loop() {
    for (i=0;i<=N ;i++) {
        Vref =(vmax*sin(i*pas));
        j=i%Np;
        if (j<=Np>>1){
            Vp= 4*j/Np-1;
        }
        else{
            Vp= -4*j/Np+3;
        }
        if ((Vref)>(Vp)){
            g1=1; g2=0;
        }else{
            g1=0; g2=1;
        }
        digitalWrite(activer, HIGH);
        digitalWrite(commande1, g1);
        digitalWrite(commande2, g2);
        mesure ();
    } }

```

✓ *La commande MLI unipolaire à deux porteuses :*

```

#include "ADS1X15.h"
int commande1 = 13;
int commande2 = 12;
int activer = 9;
ADS1115 ADS(0x48);
int g1;
int g2;
int i=0;
float j=0;
float vmax=0.8;
float m=9;
float N=36;
float pas;
int Np;
float Vp1;
float Vp2;
float Vref;
void mesure (){
  int16_t difference_potentiel_A0_A1 = ADS.readADC_Differential_0_1();
  float tension_volts_A0_A1 = ADS.toVoltage(difference_potentiel_A0_A1);
  Serial.print(" ");
  Serial.print(tension_volts_A0_A1,3);
}
void setup() {
pinMode(commande1, OUTPUT);
pinMode(commande2, OUTPUT);
pinMode(activer, OUTPUT);
  Serial.begin(9600);
  Serial.println("Lecture de tensions différentielles / ADS1115");
  Serial.println("-----");
  Serial.println("Mesure de différence de potentiel entre les entrées A0 et A1 de l'ADC");
  Serial.println("Nota : ADC Range: +/- 6.144V (soit 1 bit = 0.1875 mV)");
  Serial.println("");
  ADS.begin();
  ADS.setGain(0);      ADS.setMode(1);      ADS.setDataRate(7);  ADS.readADC(0);
  pas=2*PI/N;
  Np=N/m;
}
void loop() {
  for (i=0;i<=N ;i++) {
    Vref =(vmax*sin(i*pas));
    j=i%Np;
    if (j<=Np>>1){
      Vp1= 2*j/Np;
    }
    else{
      Vp1= -2*j/Np+2;
    }
    if (j<=Np>>1){
      Vp2= 2*j/Np-1;
    }
  }
}

```

```
    }
    else{
        Vp2= -2*j/Np+1;
    }
    if ((Vref)>(Vp1)){
        g1=0;
    }else{
        g1=1;
    }
    if ((Vref)>(Vp2)){
        g2=1;
    }else{
        g2=0;
    }
    digitalWrite(activer, HIGH);
    digitalWrite(commande1, g1);
    digitalWrite(commande2, g2);
    mesure ();
} }
```

✓ *La commande MLI unipolaire à deux références :*

```

#include "ADS1X15.h"
int commande1 = 13;
int commande2 = 12;
int activer = 9;
ADS1115 ADS(0x48);
int g1;
int g2;
int i=0;
float j=0;
float vmax=0.8;
float m=9;
float N=36;
float pas;
int Np;
float Vp;
float Vref1;
float Vref2;
void mesure (){
  int16_t difference_potentiel_A0_A1 = ADS.readADC_Differential_0_1();
  float tension_volts_A0_A1 = ADS.toVoltage(difference_potentiel_A0_A1);
  Serial.print(" ");
  Serial.print(tension_volts_A0_A1,3);
}
void setup() {
pinMode(commande1, OUTPUT);
pinMode(commande2, OUTPUT);
pinMode(activer, OUTPUT);
  Serial.begin(9600);
  Serial.println("Lecture de tensions différentielles / ADS1115");
  Serial.println("-----");
  Serial.println("Mesure de différence de potentiel entre les entrées A0 et A1 de l'ADC");
  Serial.println("Nota : ADC Range: +/- 6.144V (soit 1 bit = 0.1875 mV)");
  Serial.println("");
  ADS.begin();
  ADS.setGain(0);   ADS.setMode(1);
  ADS.setDataRate(7);   ADS.readADC(0);
pas=2*PI/N;
  Np=N/m;
}
void loop() {
  for (i=0;i<=N ;i++) {
    Vref1 =(vmax*sin(i*pas));
    Vref2=(-Vref1);
    j=i%Np;
    if (j<=Np>>1){
      Vp= 4*j/Np-1;
    }
    else{
      Vp= -4*j/Np+3
    }
  }
}

```

```
if (Vref1>=0) {
  g2=0;
  if ((Vref1)>=(Vp)){
    g1=1;
  }else{
    g1=0;
  }
}
else {
  g1=0;
  if ((Vref2)>=(Vp)){
    g2=1;
  }else{
    g2=0;
  }
}
digitalWrite(activer, HIGH);
digitalWrite(commande1, g1);
digitalWrite(commande2, g2);
measure ();
} }
```

Annexe 5 : Code Arduino pour le contrôle du module BTS7960 avec la fonction du capteur de tension ADS1115 :

✓ *La commande Pleine Onde :*

```
#include "ADS1X15.h"
#define T_ms 10
int commande1 = 3;
int commande2 = 5;
ADS1115 ADS(0x48);
float t_m;
float p_m=2;
void mesure (){
  int16_t difference_potentiel_A0_A1 = ADS.readADC_Differential_0_1();
  float tension_volts_A0_A1 = ADS.toVoltage(difference_potentiel_A0_A1);
  Serial.print(" ");
  Serial.print(tension_volts_A0_A1,3);
}
void setup() {
  pinMode(commande1, OUTPUT);
  pinMode(commande2, OUTPUT);
  Serial.begin(9600);
  Serial.println("Lecture de tensions différentielles / ADS1115");
  Serial.println("-----");
  Serial.println("Mesure de différence de potentiel entre les entrées A0 et A1 de l'ADC");
  Serial.println("Nota : ADC Range: +/- 6.144V (soit 1 bit = 0.1875 mV)");
  Serial.println("");
  ADS.begin();
  ADS.setGain(0); ADS.setMode(1);
  ADS.setDataRate(7); ADS.readADC(0);
}
void loop() {
  digitalWrite(commande1, HIGH);
  digitalWrite(commande2, LOW);
  t_m=0;
  while (t_m<= T_ms){
    mesure ();
    t_m = t_m + p_m;
  }
  digitalWrite(commande1, LOW);
  digitalWrite(commande2, HIGH);
  t_m=0;
  while (t_m<= T_ms){
    mesure() ;
    t_m = t_m + p_m;
  } }
}
```

✓ *La commande Décalée :*

```

#include "ADS1X15.h"
#define T_ms    20
#define Alfa    PI
int commande1 = 3;
int commande2 = 5;
ADS1115 ADS(0x48);
unsigned long T_alfa;
unsigned long T_0;
float t_m;
float p_m=2;
void mesure (){
  int16_t difference_potentiel_A0_A1 = ADS.readADC_Differential_0_1();
  float tension_volts_A0_A1 = ADS.toVoltage(difference_potentiel_A0_A1);
  Serial.print(" ");
  Serial.print(tension_volts_A0_A1,3);
}
void setup() {
pinMode(commande1, OUTPUT);
pinMode(commande2, OUTPUT);
  Serial.begin(9600);
  Serial.println("Lecture de tensions différentielles / ADS1115");
  Serial.println("-----");
  Serial.println("Mesure de différence de potentiel entre les entrées A0 et A1 de l'ADC");
  Serial.println("Nota : ADC Range: +/- 6.144V (soit 1 bit = 0.1875 mV)");
  Serial.println("");
  ADS.begin();
  ADS.setGain(0);      ADS.setMode(1);
  ADS.setDataRate(7); ADS.readADC(0);
  T_alfa=round((double)T_ms*Alfa/(2.0*PI));
  T_0=(T_ms-T_alfa)>>1;
}
void loop() {
digitalWrite(commande1, LOW);
digitalWrite(commande2, LOW);
//delay(T_alfa>>1);
t_m=0;
  while (t_m<= T_alfa>>1){
    mesure ();
    t_m = t_m + p_m;
  }
digitalWrite(commande1, HIGH);
digitalWrite(commande2, LOW);
//delay(T_0);
t_m=0;
  while (t_m<= T_0){
    mesure() ;
    t_m = t_m + p_m;
  }
digitalWrite(commande1, LOW);
digitalWrite(commande2, LOW);

```

```
//delay(T_alfa>>1);
t_m=0;
while (t_m<= T_alfa>>1){
  mesure ();
  t_m = t_m + p_m;
}
digitalWrite(commande1, LOW);
digitalWrite(commande2, HIGH);
//delay(T_0);
t_m=0;
while (t_m<= T_0){
  mesure ();
  t_m = t_m + p_m;
} }
```

✓ *La Commande MLI Bipolaire :*

```

#include "ADS1X15.h"
int commande1 = 3;
int commande2 = 5;
ADS1115 ADS(0x48);
int g1;
int g2;
int i=0;
float j=0;
float vmax=0.8;
float m=9;
float N=36;
float pas;
int Np;
float Vp;
float Vref;
void mesure (){
int16_t difference_potentiel_A0_A1 = ADS.readADC_Differential_0_1();
float tension_volts_A0_A1 = ADS.toVoltage(difference_potentiel_A0_A1);
  Serial.print(" ");
  Serial.print(tension_volts_A0_A1,3);
}
void setup() {
pinMode(commande1, OUTPUT);
pinMode(commande2, OUTPUT);
  Serial.begin(9600);
  Serial.println("Lecture de tensions différentielles / ADS1115");
  Serial.println("-----");
  Serial.println("Mesure de différence de potentiel entre les entrées A0 et A1 de l'ADC");
  Serial.println("Nota : ADC Range: +/- 6.144V (soit 1 bit = 0.1875 mV)");
  Serial.println("");
  ADS.begin();
  ADS.setGain(0);      ADS.setMode(1);
  ADS.setDataRate(7);
  ADS.readADC(0);
  pas=2*PI/N;
  Np=N/m;
}
void loop() {
  for (i=0;i<=N ;i++) {
    Vref =(vmax*sin(i*pas));
    j=i%Np;
    if (j<=Np>>1){
      Vp= 4*j/Np-1;
    }
    else{
      Vp= -4*j/Np+3;
    }
    if ((Vref)>(Vp)){
      g1=1; g2=0;
    }else{
      g1=0;
      g2=1;
    }
  }
  digitalWrite(commande1, g1);
  digitalWrite(commande2, g2);
  mesure ();
} }

```

✓ *La commande MLI unipolaire à deux porteuses :*

```

#include "ADS1X15.h"
int commande1 = 3;
int commande2 = 5;
ADS1115 ADS(0x48);
int g1;
int g2;
int i=0;
float j=0;
float vmax=0.8;
float m=9;
float N=36;
float pas;
int Np;
float Vp1;
float Vp2;
float Vref;
void mesure (){
int16_t difference_potentiel_A0_A1 = ADS.readADC_Differential_0_1();
float tension_volts_A0_A1 = ADS.toVoltage(difference_potentiel_A0_A1);
Serial.print(" ");
Serial.print(tension_volts_A0_A1,3);
}
void setup() {
pinMode(commande1, OUTPUT);
pinMode(commande2, OUTPUT);
Serial.begin(9600);
Serial.println("Lecture de tensions différentielles / ADS1115");
Serial.println("-----");
Serial.println("Mesure de différence de potentiel entre les entrées A0 et A1 de l'ADC");
Serial.println("Nota : ADC Range: +/- 6.144V (soit 1 bit = 0.1875 mV)");
Serial.println("");
ADS.begin();
ADS.setGain(0); ADS.setMode(1);
ADS.setDataRate(7);
ADS.readADC(0);
pas=2*PI/N;
Np=N/m;
}
void loop() {
for (i=0;i<=N ;i++) {
Vref =(vmax*sin(i*pas));
j=i%Np;
if (j<=Np>>1){
Vp1= 2*j/Np;
}
else{
Vp1= -2*j/Np+2;
}
if (j<=Np>>1){
Vp2= 2*j/Np-1;
}
else{
Vp2= -2*j/Np+1;
}
if (Vref>0) {
g2=0;
if ((Vref)>(Vp1)){
g1=0;
}
}
}
}

```

```
    }else{
      g1=1;
    }
  }
  else {
    g1=0;
    if ((Vref)>(Vp2)){
      g2=1;
    }else{
      g2=0;}
  }
  digitalWrite(commande1, g1);
  digitalWrite(commande2, g2);
  mesure ();
} }
```

✓ *La commande MLI unipolaire à deux références :*

```

#include "ADS1X15.h"
int commande1 = 3;
int commande2 = 5;
ADS1115 ADS(0x48);
int g1;
int g2;
int i=0;
float j=0;
float vmax=0.8;
float m=9;
float N=36;
float pas;
int Np;
float Vp;
float Vref1;
float Vref2;
void mesure (){
int16_t difference_potentiel_A0_A1 = ADS.readADC_Differential_0_1();
float tension_volts_A0_A1 = ADS.toVoltage(difference_potentiel_A0_A1);
  Serial.print(" ");
  Serial.print(tension_volts_A0_A1,3);
}
void setup() {
pinMode(commande1, OUTPUT);
pinMode(commande2, OUTPUT);
  Serial.begin(9600);
  Serial.println("Lecture de tensions différentielles / ADS1115");
  Serial.println("-----");
  Serial.println("Mesure de différence de potentiel entre les entrées A0 et A1 de l'ADC");
  Serial.println("Nota : ADC Range: +/- 6.144V (soit 1 bit = 0.1875 mV)");
  Serial.println("");
  ADS.begin();
  ADS.setGain(0);
  ADS.setMode(1);
  ADS.setDataRate(7);
  ADS.readADC(0);
  pas=2*PI/N;
  Np=N/m;
}
void loop() {
  for (i=0;i<=N ;i++) {
    Vref1 =(vmax*sin(i*pas));
    Vref2=(-Vref1);
    j=i*Np;
    if (j<=Np>>1){
      Vp= 4*j/Np-1;
    }
    else{
      Vp= -4*j/Np+3;
    }
  }
  if (Vref1>=0) {
    g2=0;
    if ((Vref1)>=(Vp)){
      g1=1;
    }else{
      g1=0;
    }
  }
}
}

```

```
else {
  g1=0;
  if ((Vref2)>=(Vp)){
  g2=1;
  }else{
  g2=0;
  }
}
digitalWrite(commande1, g1);
digitalWrite(commande2, g2);
measure ();
} }
```

Résumé

Ce travail est consacré à l'implémentation des stratégies de commande de l'onduleur monophasé en pont complet à travers une carte Arduino. Le logiciel MATLAB est utilisé pour simuler les techniques de commande avant leurs implémentations. Les techniques de commande sont appliquées sur deux modules, L298N et BTS7960, utilisés en tant qu'onduleurs monophasés. Ces modules fonctionnent à faible tension d'alimentation. Les signaux de sortie sont récupérés par un convertisseur analogique numérique appelé ADS1115, utilisé en tant que capteur de tension, et par un oscilloscope numérique.

Mots clés : Arduino – Onduleur monophasé – Commande en MLI– ADS1115 – L298N – BTS7960.

Abstract

This work is devoted to the implementation of the control strategies of the full-bridge single-phase inverter through an Arduino board. The MATLAB software is used to simulate control techniques before implementation. The control techniques are applied on two modules, L298N and BTS7960, used as single phase inverters. These modules operate at low supply voltage. The output signals are recovered by a digital analog converter called ADS1115, used as a voltage sensor, and by a digital oscilloscope.

key words: Arduino – single-phase inverter – PWM control – ADS1115 – L298N – BTS7960.

ملخص

يكرس هذا العمل للتنفيذ العملي لاستراتيجيات التحكم في موج أحادي الطور من خلال بطاقة الاردوينو. يتم استخدام برنامج المطالاب لمحاكاة التحكم قبل تطبيقه. تم استعمال الـ L298N و BTS7960 للعب دور الموجه. هذين الـ وحداتين تشتغلان تحت توتر منخفض. يتم استرداد الإشارات الناتجة عن التحكم بواسطة محول تناظري رقمي يسمى ADS1115 يلعب دور مجس التوتر وبراسم اهتزازات عددي.

كلمات مفتاحية: أردوينو – موج أحادي الطور – التحكم PWM – L298N – BTS7960 – ADS1115.