

République Algérienne Démocratique et populaire  
Ministère de l'enseignement Supérieur et de la recherche scientifique

Université A.Mira Béjaïa  
Faculté de technologie  
Département de Génie Electrique



**Mémoire de fin d'étude**  
**En vue de l'obtention du diplôme de master en électrotechnique**

**Filière : Electrotechnique**

**Spécialité: Automatismes industriels**

*Thème*

**Implémentation de la Commande SVM d'un Onduleur Triphasé de  
Tension sur Arduino.**

**Présenté par :**

Mr AIT CHALALT Fares

Mr AROUN Rabah

**Dirigé par :**

Mr Taib Nabil

*Année universitaire 2022/2023*

## **Remerciement**

*Nous tenons d'abord à remercier dieu tout puissant qui nous aide et nous donne la patience et le courage durant ces années d'études et durant ce travail*

*En premier lieu, il est important de souligner que ce travail n'aurait pas atteint une telle richesse et n'aurait pas pu voir le jour sans l'aide et l'encadrement précieux de M. TAIB Nabil. Nous le remercions chaleureusement pour la qualité exceptionnelle de son encadrement, sa patience, sa rigueur et sa disponibilité tout au long de notre préparation de ce mémoire.*

*Nous souhaitons également exprimer notre gratitude envers nos professeurs de l'Université A. Mira de Bejaia, qui nous ont dispensé un enseignement de haute qualité au cours des cinq dernières années.*

## Sommaire :

Remerciement	
Liste des figures	
Liste des tableaux	
Liste des nomenclatures	
<b>Introduction générale :</b>	<b>1</b>
<b>Chapitre1 : Généralités sur les onduleurs et leurs applications</b>	
<b>I-1-Introduction</b>	<b>3</b>
<b>I-2-Description d'onduleur</b>	<b>3</b>
<b>I-3-Classification des onduleurs</b>	<b>3</b>
<b>I-3-1-Onduleurs non autonomes</b>	<b>4</b>
<b>I-3-2-Onduleurs à résonance</b>	<b>4</b>
<b>I-3-3-Onduleurs autonomes</b>	<b>4</b>
<b>I-4-Types d'onduleurs autonomes de tension</b>	<b>4</b>
<b>I-4-1-Les onduleurs monophasés</b>	<b>4</b>
<b>I-4-1-1- Onduleur monophasé en demi-pont</b>	<b>5</b>
<b>I-4-1-1-1- Schéma de principe</b>	<b>5</b>
<b>I-4-1-1-2-Principe de fonctionnement</b>	<b>5</b>
<b>I-4-1-2-Onduleur monophasé en pont</b>	<b>5</b>
<b>I-4-1-2-1-Schéma de principe</b>	<b>5</b>
<b>I-4-1-2-2-Principe de fonctionnement</b>	<b>6</b>
<b>I-4-2-Les onduleurs triphasés</b>	<b>6</b>
<b>I-4-2-1-Schémas de principe</b>	<b>6</b>
<b>I-4-2-2-Principe de fonctionnement</b>	<b>7</b>
<b>I-5-Choix des composants</b>	<b>7</b>
<b>I-6-Les applications des onduleurs</b>	<b>8</b>
<b>I-6-1-Réglage de la vitesse de rotation d'un moteur synchrone</b>	<b>8</b>
<b>I-6-2-Alimentation de secours</b>	<b>9</b>
<b>I-6-3-Filtres actifs</b>	<b>9</b>
<b>I-6-4-Compensateur de puissance réactive</b>	<b>10</b>
<b>I-7-Conclusion</b>	<b>10</b>
<b>Chapitre2 : Les stratégies de commande des onduleurs</b>	
<b>II-1-Introduction</b>	<b>11</b>
<b>II-2-Commandes à ondes carrés</b>	<b>11</b>
<b>II-2-1-Commande pleine onde</b>	<b>11</b>
<b>II-2-2-Commande décalé (120°)</b>	<b>12</b>
<b>II-3-Commande par hystérésis des courants:</b>	<b>12</b>
<b>II-4-Commande par modulation de largeur d'impulsion (MLI)</b>	<b>14</b>
<b>II-4-1-Commande MLI Triangulo-Sinusoidale (MLI-TS)</b>	<b>15</b>
<b>II-4-2-Commande MLI Vectorielle</b>	<b>16</b>
<b>II-5-Conclusion :</b>	<b>16</b>
<b>Chapitre3 : Description de la commande SVM d'un onduleur</b>	
<b>III- 1-Introduction</b>	<b>17</b>
<b>III- 2-Principe de la MLI vectorielle</b>	<b>17</b>

III- 3-La transformation de Clark :.....	18
III- 4-Les séquences de commutation de l'onduleur.....	19
III- 5-Algorithmes de modulation de la largeur d'impulsion (MLI) vectorielle .....	22
III-5-1-Schémas de modulation .....	22
III- 5-1-1-La séquence alignée à droite (SVM1).....	22
III- 5-1-2- Séquence symétrique (SVM2) .....	23
III- 5-1-3-Séquence vectorielle nulle alternée (SVM3) .....	23
III- 5-1-4-Séquence sans commutation de courant le plus élevé (SVM4) .....	23
III- 6-La simulation de la commande SVM sous Matlab Simulink.....	24
III- 6-1-Débit sur charge RL .....	24
III- 6-1-1-Représentation des tensions de référence .....	25
III- 6-1-1-1-Résultats de simulation .....	25
III- 6-1-2-Détermination des tensions dans le repère ( $\alpha$ , $\beta$ ) .....	26
III- 6-1-2-1-Résultats de simulation .....	27
III- 6-1-3-Détermination du numéro de secteur et phi .....	27
III- 6-1-3-1-Résultats de simulation : .....	29
III- 6-1-4-Détermination des durées de conduction ( $T_1$ , $T_2$ , $T_0$ ) .....	29
III- 6-1-4-1-Résultats de simulation .....	30
III- 6-1-5-Détermination des signaux de commande des interrupteurs .....	31
III- 6-1-5-1-Résultats de simulation .....	31
III- 6-1-6-Visualisation des tensions simples et composées avec courants de sorties .....	33
III- 6-1-6-1-Résultats de simulation .....	33
III- 6-1-7-Interprétations des résultats :.....	35
III-6-2-Débit sur charge MAS .....	36
III- 6-2-1-La commande scalaire de la charge (machine asynchrone) .....	37
III- 6-2-2-La simulation de la commande scalaire sous Matlab Simulink .....	37
III- 6-2-2-1-Résultats de simulation :.....	38
III- 6-2-3-Interprétation des résultats .....	44
III- 7-Conclusion : .....	44

## Chapitre4 : Implémentation sur Arduino

IV -1- Introduction .....	45
IV -2- La carte ARDUINO Uno .....	45
IV -2-1- Caractéristiques de la carte Arduino Uno .....	46
IV -3- Implémentation de la commande SVM sur Proteus .....	46
IV -3-1- Implémentation de la commande scalaire sous Proteus .....	47
IV -4- Le séquencement du programme Arduino .....	48
IV -4-1- Exemples des démarches suivies .....	49
IV -5- Résultats de simulation .....	49
IV -5-1- Détermination de $V_a$ , $V_b$ et $V_c$ en fonction de $f_{ref}$ .....	49
IV -5-2- La mesure de Phi en fonction de $V_{\alpha}$ et $V_{\beta}$ .....	50
IV -6- Visualisation des signaux de commande des interrupteurs .....	51
IV -6-1- Interprétation des résultats .....	55
IV -7- Conclusion .....	55
Conclusion générale .....	56
Références bibliographiques .....	57
Annexe	
Résumé	

## **Liste des figures :**

<b>Figure I-1</b> : Schéma de principe de la conversion DC-AC.....	3
<b>Figure I-2</b> : Schéma électrique d'un onduleur monophasé en demi-pont. ....	5
<b>Figure I-3</b> : <i>Schéma de principe d'un onduleur monophasé en pont.</i> .....	6
<b>Figure I-4</b> : Schéma de principe d'un onduleur triphasé en pont.....	6
<b>Figure I-5</b> : Choix des composants selon la puissance et la fréquence. ....	8
<b>Figure I-6</b> : Réglage de la vitesse de rotation d'un moteur synchrone. ....	8
<b>Figure I-7</b> : Alimentation de secours. ....	9
<b>Figure I-8</b> : Filtre actif parallèle (FAP).....	10
<b>Figure I-9</b> : Filtre actif série (FAS). ....	10
<b>Figure I-10</b> : Structure d'un compensateur de puissance réactive.....	10
<b>Figure I-11</b> : Diagramme vectoriel. ....	10
<b>Figure II-1</b> : Commande 180°. ....	11
<b>Figure II-2</b> : Commande 120°. ....	12
<b>Figure II-3</b> : Schéma de principe du contrôle par hystérésis d'un bras de l'onduleur. ....	13
<b>Figure II-4</b> : Bande d'Hystérésis ....	13
<b>Figure II-5</b> : Principe du contrôle des courants par PWM. ....	14
<b>Figure II-6</b> : Répartition des commutations sur un bras d'un onduleur. ....	14
<b>Figure II-7</b> : Onduleur triphasé de tension en pont. ....	15
<b>Figure II-8</b> : Formes d'onde de tension porteuse, de référence et de sortie d'un onduleur triphasé contrôlé par PWM Triangulo- sinusoïdale. ....	16
<b>Figure III-1</b> : Les vecteurs de tension de l'onduleur dans le plan ( $\alpha,\beta$ ). ....	17
<b>Figure III-2</b> : Emplacement des huit vecteurs stationnaires pour un onduleur de tension à deux niveaux.....	18
<b>Figure III-3</b> : Principe de construction du vecteur de tension $V_{ref}$ .....	20
<b>Figure III-4</b> : Formes des largeurs d'impulsion du secteur 1.....	21
<b>Figure III-5</b> : Formes des largeurs d'impulsions pour secteur 2. ....	21
<b>Figure III-6</b> : Formes des largeurs d'impulsion du secteur 3.....	21
<b>Figure III-7</b> : Formes des largeurs d'impulsions pour secteur 4. ....	21
<b>Figure III-8</b> : Formes des largeurs d'impulsion du secteur 5.....	21
<b>Figure III-9</b> : Formes des largeurs d'impulsion du secteur 6. ....	21
<b>Figure III-10</b> : La séquence alignée à droite (SVM1).....	22
<b>Figure III-11</b> : Séquence symétrique (SVM2). ....	23
<b>Figure III-12</b> : Séquence vectorielle nulle alternée (SVM3).....	23
<b>Figure III-13</b> : Séquence sans commutation de courant le plus élevé (SVM4).....	24
<b>Figure III-14</b> : Schéma global de la commande SVM d'un onduleur sous Matlab Simulink.....	25
<b>Figure III-15</b> : Schéma de bloc des tensions de référence. ....	25
<b>Figure III-16</b> : Programme pour calcul des tensions de référence.....	25
<b>Figure III-17</b> : Allure des tensions de référence.....	26
<b>Figure III-18</b> : Bloc du $V_{\alpha}$ , $V_{\beta}$ .....	26
<b>Figure III-19</b> : Programme pour calcul de $V_{\alpha}$ et $V_{\beta}$ .....	26

<b>Figure III-20</b> : Allure de $V_{\alpha}$ et $V_{\beta}$ .....	27
<b>Figure III-21</b> : Code pour calcul de $\phi$ et nombre de secteur.....	28
<b>Figure III-22</b> : Schéma de bloc pour numéro de secteur et $\phi$ . ....	28
<b>Figure III-23</b> : Le déplacement du vecteur de référence entre les secteurs en fonction du temps.....	29
<b>Figure III-24</b> : L'angle du vecteur de référence. ....	29
<b>Figure III-25</b> : Les allures des durées de conduction ( <b>a</b> : $T_1$ , <b>b</b> : $T_2$ , <b>c</b> : $T_0$ ).....	30
<b>Figure III-26</b> : Code pour détecter le secteur1. ....	31
<b>Figure III-27</b> : Les signaux de commande des interrupteurs (S1, S3, S5) de secteur 1.....	32
<b>Figure III-28</b> : Les courants ( $I_1$ , $I_2$ et $I_3$ ) avec charge RL.....	33
<b>Figure III-29</b> : Allures des tensions simples avec charge RL. ....	34
<b>Figure III-30</b> : Allure des tensions composées avec charge RL.....	35
<b>Figure III-31</b> : Schéma global de simulation de la commande SVM avec une charge (MAS). ....	36
<b>Figure III-32</b> : Schéma de principe de la commande scalaire de la MAS.....	37
<b>Figure III-33</b> : Schéma du bloc de la commande scalaire sous Matlab Simulink. ....	38
<b>Figure III-34</b> : Codes pour calcul des tensions de référence. ....	38
<b>Figure III-35</b> : Allure des tensions simples avec charge MAS .....	39
<b>Figure III-36</b> : Allure de la tension composée $U_{12}$ avec charge MAS.....	40
<b>Figure III-37</b> : Allure de la tension composée $U_{23}$ avec charge MAS.....	41
<b>Figure III-38</b> : Allure de la tension composée $U_{31}$ avec charge MAS.....	42
<b>Figure III-39</b> : Allure de la vitesse de rotation de la MAS. ....	43
<b>Figure III-40</b> : Allure du courant statorique $I_a$ . ....	43
<b>Figure IV- 1</b> : Carte Arduino Uno. ....	45
<b>Figure IV- 2</b> : Schéma global de simulation au niveau de Proteus. ....	47
<b>Figure IV-3</b> : Organigramme décrivant les démarches suivies pour la construction du code Arduino. ....	48
<b>Figure IV- 4</b> : la mise à échelle de la fréquence de référence.....	49
<b>Figure IV- 5</b> : Valeurs de $V_a, V_b$ et $V_c$ en fonction de $f_{ref}$ et $f_{refn}$ . ....	50
<b>Figure IV- 6</b> : Valeurs de $\phi$ en fonction de $V_{\alpha}$ et $V_{\beta}$ . ....	51
<b>Figure IV- 7</b> : Signaux (S1, S3 et S5) du secteur 1.....	52
<b>Figure IV- 8</b> : Signaux (S1, S3 et S5) du secteur 2.....	52
<b>Figure IV- 9</b> : Signaux (S1, S3 et S5) du secteur 3.....	53
<b>Figure IV- 10</b> : Signaux (S1, S3 et S5) du secteur 4.....	53
<b>Figure IV- 11</b> : Signaux (S1, S3 et S5) du secteur 5.....	54
<b>Figure IV- 12</b> : Signaux (S1, S3 et S5) du secteur 6.....	54

**Liste des tableaux :**

**Tableau III-1** : Table de vérité pour un onduleur de tension triphasé à deux niveaux. ....19  
**Tableau III-2** : Les paramètres de la MAS. ....36

## **Liste de nomenclature :**

**CA** : Courant Alternatif.

**GTO** : Gate Turn Off.

**IGBT** : Insulated Gate Bipolar Transistor.

**MOSFET** : Metal Oxide Semiconductor Field Effect Transistor.

**DC** : Direct Current.

**AC** : Alternating Current.

**MLI** : Modulation de Largeur d'Impulsion.

**SVM** : Space Vector Modulation.

**PWM** : Pulse Width Modulation.

**V<sub>r</sub>** : Tension de référence.

**W<sub>p</sub>** : Porteuse.

**MS** : Machine Synchrone.

**RMS** : Root Mean Square.

**MLI-TS** : Modulation de Largeur d'Impulsion Triangulo-Sinusoidale.

**SSV** : Switching State Vectors.

**E/S** : Entrées/Sorties.

**USB** : Universal Serial Bus.

**ICSP** : In-Circuit Serial Programming.

**SRAM** : Static Random Access Memory.

**EEPROM** : Electrically-Erasable Programmable Read-Only Memory.

**IDE** : Integrated Development Environment.

**PIC** : Peripheral Interface Controller.

## **Introduction générale**

La croissance de l'industrie a entraîné une augmentation de l'utilisation de différents types de moteurs électriques, qui diffèrent les uns des autres par plusieurs facteurs, tels que la nature de leurs alimentations (alternative ou continue) et leurs fonctionnements en tant que moteurs ou générateurs. Dans cette étude, nous nous concentrons sur l'étude des onduleurs et comment fournir du courant alternatif (CA) à partir d'une source continue et comment les commander. Pour résoudre ce problème, nous utilisons des circuits électriques, qui sont composés principalement d'un onduleur, que nous allons examiner dans cette thèse. L'onduleur est un circuit électrique basé sur l'électronique de puissance comprenant des éléments tels que des diodes, thyristor GTO, Transistor (IGBT, MOSFET), etc. Ce circuit permet de convertir du courant continu en courant alternatif à partir d'une source d'alimentation continue (telle que des piles, un redresseur ou une plaque photovoltaïque) en utilisant la commande de semi-conducteurs (interrupteurs).

La commutation des interrupteurs se fait à des intervalles très courts, de l'ordre de millisecondes ou de microsecondes. Cette vitesse de commutation ne peut pas être contrôlée uniquement par des méthodes traditionnelles, mais elle nécessite l'utilisation des techniques modernes comme des circuits intégrés tels que les PIC et les Arduino, en fonction du temps d'impulsion envoyé pour activer les interrupteurs de l'onduleur. Nous pouvons utiliser différents types de contrôles tels que la modulation de largeur d'impulsion pré-calculée (MLI) et la MLI sinusoïdale-triangulaire. D'autres types de contrôles, tels que la commande en onde carrée, peuvent également être ajoutés. Dans cette étude, nous utilisons des circuits modernes pour obtenir le contrôle d'un onduleur triphasé de tension tel que la carte électronique programmable Arduino.

La carte Arduino est utilisée dans cette étude puisque elle donne la possibilité d'allier les performances de la programmation à celles de l'électronique pour commander des systèmes électroniques comme notre système étudié. L'avantage apporté par l'utilisation de cette carte c'est réduire les coûts de réalisation électroniques de système, aussi gagner de temps de la réalisation de schémas électroniques compliqué, et la facilité de commande des systèmes difficile à contrôler avec les méthodes classiques. Nous discutons également de la théorie de fonctionnement et de contrôle d'un onduleur dans le cadre de cette étude.

Pour aboutir aux objectifs de ce travail, notre mémoire sera structuré en quatre chapitres présentés comme suit :

Dans la première partie nous avons présenté la vue d'ensemble des onduleurs et de leurs applications, nous avons parlé de la classification des onduleurs et du principe de fonctionnement de ces derniers, puis nous avons également parlé sur leurs divers domaines d'application et d'utilisations.

La deuxième partie a été consacrée à l'étude des différentes techniques de commande, notamment la commande à ondes carrées, la commande par modulation de largeur d'impulsion (MLI) et la commande MLI vectorielle. Chacune de ces techniques a été expliquée en détail, mettant en évidence leurs avantages et leurs limites.

Dans la troisième partie, nous avons approfondi sur le principe de la MLI vectorielle, en abordant des concepts tels que la transformation de Clark et les séquences de commutation de l'onduleur. Nous avons également présenté des algorithmes de modulation de la largeur d'impulsion (MLI) vectorielle, en fournissant des schémas de modulation pour une meilleure compréhension.

La quatrième partie était pour l'implémentation de la commande SVM dans la carte Arduino, nous avons traduit et implémenté ce programme dans la carte Arduino via ISIS-Proteus en tenant compte les contraintes de fonctionnement de cette carte et les conditions de branchement de ses entrées et ses sorties.

Enfin, on termine par une conclusion, mettant l'importance de la commande SVM sur les onduleurs et l'intérêt de la programmation de cette commande sur la plateforme Arduino.

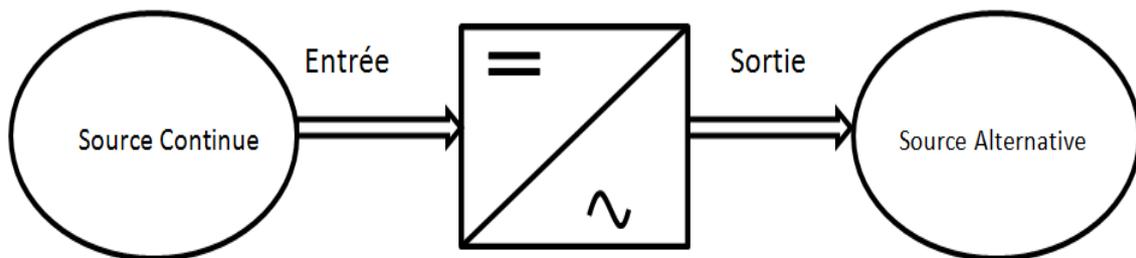
# **Chapitre I : Généralités sur les onduleurs et leurs applications**

## **I-1-Introduction :**

Les onduleurs sont des dispositifs électroniques essentiels pour de nombreuses applications, offrant une conversion efficace du courant continu en courant alternatif, ce que on va détailler dans ce chapitre.

## **I-2-Description d'onduleur :**

Un onduleur est un convertisseur statique qui transforme l'électricité du courant continu (DC) en courant alternatif (AC). En fait, cette conversion d'énergie est effectuée par un dispositif de contrôle (semi-conducteur). Ceci permet d'obtenir une tension alternative aux bornes du récepteur avec une fréquence et une valeur efficaces réglables, à l'aide de séquences de commandes appropriées [1].



*Figure I-1 : Schéma de principe de la conversion DC-AC.*

La structure de l'onduleur dépend essentiellement du type d'alimentation sur laquelle l'onduleur est installé. L'alternance de l'alimentation devant être respectée lors de l'accès, on distingue :

- Un onduleur de tension qui connecte une alimentation en tension continu à une alimentation en courant alternatif,
- Un onduleur de courant qui est placé entre une alimentation en courant continu et une alimentation en tension alternative.

La nature des sources est caractérisée du point de vue de la commutation [2].

## **I-3-Classification des onduleurs :**

Les onduleurs se distinguent en trois familles principales qui sont :

### **I-3-1-Onduleurs non autonomes :**

Un onduleur est considéré comme non autonome si l'énergie nécessaire à la commutation des thyristors provient du réseau alternatif, c'est-à-dire du réseau actif. C'est le cas pour le fonctionnement en onduleur des redresseurs. La fréquence et la forme d'onde de la tension sont déterminées par la source d'alimentation (CA).

### **I-3-2-Onduleurs à résonance :**

Un onduleur résonant est un onduleur de tension ou de courant avec une fente par courant alternatif qui fonctionne dans certaines conditions. La charge doit être un circuit résonant mal amorti. L'interrupteur est commandé par une fréquence proche de la fréquence de résonance de la charge. Si cela change, la fréquence de contrôle doit être modifiée. L'onduleur doit donc être piloté par la charge et n'est plus autonome.

### **I-3-3-Onduleurs autonomes :**

Un onduleur autonome est un convertisseur statique dont son alimentation continue modifie périodiquement la connexion entre l'entrée et la sortie pour fournir une alimentation CA à la sortie. Pour un onduleur autonome, ce qui compte vraiment, c'est la nature du générateur et du récepteur auxquels il est attaché. Cela fait les distinctions suivantes :

- Onduleur de tension.
  
- Onduleur de courant [3].

## **I-4-Types d'onduleurs autonomes de tension :**

Il existe deux types d'onduleurs de tension autonomes : les onduleurs monophasés et les onduleurs triphasés :

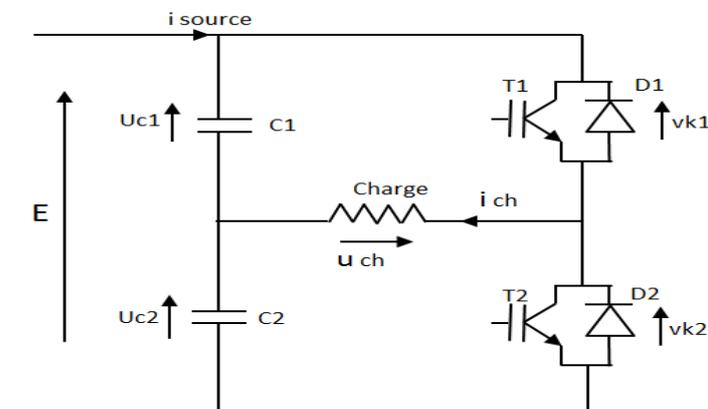
### **I-4-1-Les onduleurs monophasés :**

Pour dériver une tension alternative à partir d'une tension continue à l'aide de deux commutateurs, il faut un point médian soit du côté de la sortie alternative, soit du côté de l'entrée continue. Cela correspond à:

- Un onduleur monophasé avec un transformateur de sortie central, appelé push-inverter-pull.
- Onduleur monophasé avec diviseur de tension capacitif en entrée. C'est ce qu'on appelle un onduleur demi-pont. Si nous voulons modifier la largeur relative des fentes formant l'alternance de la tension de sortie, nous avons besoin de quatre interrupteurs, c'est-à-dire un onduleur à pont monophasé.

### **I-4-1-1- Onduleur monophasé en demi-pont (deux interrupteurs ; à diviseur capacitif) :**

#### **I-4-1-1-1- Schéma de principe :**



*Figure I-2 : Schéma électrique d'un onduleur monophasé en demi-pont.*

#### **I-4-1-1-2-Principe de fonctionnement :**

Il existe une seule source de tension et le point médian est obtenu en connectant deux condensateurs de haute qualité en série. Le courant de charge traverse ces deux condensateurs. Lorsque le transistor  $T_1$  conduit, le condensateur  $C_1$  se décharge et le condensateur  $C_2$  se charge, d'où la tension  $u=E/2$ . De même, lorsque  $T_2$  conduit, le condensateur  $C_2$  se décharge et le condensateur  $C_1$  se charge à la tension  $u=-E/2$ .

### **I-4-1-2-Onduleur monophasé en pont :**

#### **I-4-1-2-1-Schéma de principe :**

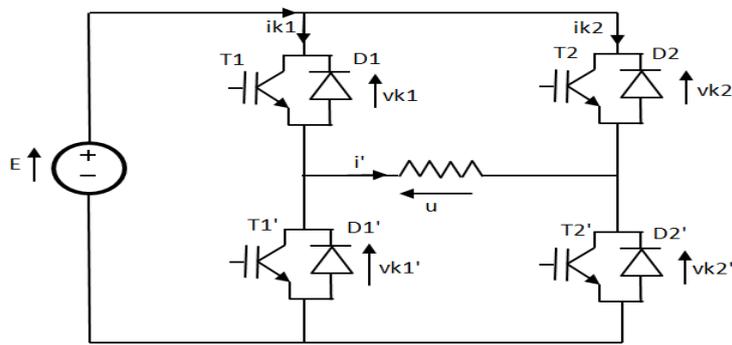


Figure I-3 : Schéma de principe d'un onduleur monophasé en pont.

#### **I-4-1-2-2-Principe de fonctionnement :**

Supposons que la tension  $u$  est égale à  $+E$  dans le premier demi-cycle car les interrupteurs  $T1$  et  $T2$  sont fermés, et  $u$  est égal à  $-E$  dans le second demi-cycle car les interrupteurs  $T1'$  et  $T2'$  sont fermés. Cet ordre est appelé ordre symétrique. Rappelons que les diodes montées juste à côté des transistors et des thyristors sont très importantes pour les protéger et pour dissiper l'énergie stockée dans la self de charge (En anglais, cela s'appelle : "free Wheeling diodes" « diode de roue libre »).

#### **I-4-2-Les onduleurs triphasés :**

Trois demi-ponts monophasés sont immédiatement suivis d'un onduleur de tension triphasé, ce qui donne un onduleur triphasé à six interrupteurs. Chaque demi-pont est constitué d'un thyristor (ou transistor) et d'une diode. Une source de tension continue peut provenir d'une batterie ou d'un pont redresseur.

#### **I-4-2-1-Schéma de principe :**

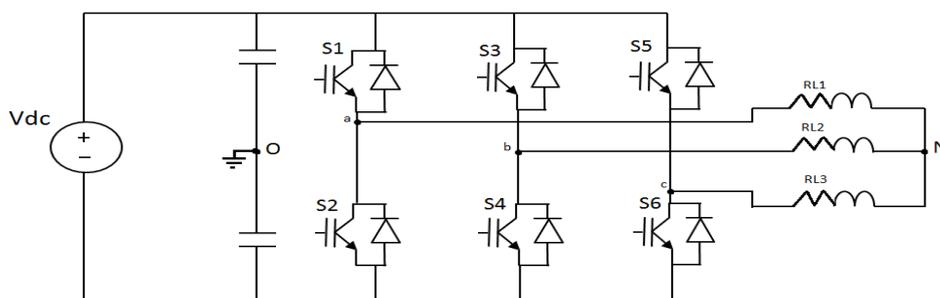


Figure I-4 : Schéma de principe d'un onduleur triphasé en pont.

### **I-4-2-2-Principe de fonctionnement :**

Des schémas d'onduleur sont fournis pour assurer la continuité des courants de sortie AC ( $I_a$ ,  $I_b$  et  $I_c$ ). Les interrupteurs S1 et S2, S3 et S4, S5 et S6 doivent être deux à deux complémentaires et définir  $S_i = (T_i, D_i)$ , donc les tensions de sortie  $V_a$ ,  $V_b$ ,  $V_c$  ont une période  $T$  égale à un tiers de  $L$ . Le point fondamental est que chaque demi-pont doit être commandé avec un retard de  $T/3$  par rapport au demi-pont précédent. Par conséquent, pour le contrôle pleine onde, nous concluons que [4]:

- S1 si  $\omega t = 0$  et S2 si  $\omega t = \pi$ .
- S3 si  $\omega t = \frac{2\pi}{3}$  et S4 si  $\omega t = \pi + \frac{2\pi}{3}$  ;
- S5 si  $\omega t = \frac{4\pi}{3}$  et S6 si  $\omega t = \pi + \frac{4\pi}{3}$  .

### **I-5-Choix des composants :**

Les composants de puissance (commutateurs onduleurs) sont déterminés par le niveau de puissance et la fréquence de commutation. En général, plus le composant est rapide (fréquence de commutation plus élevée), plus la capacité de commutation est faible et inversement. Ce qui suit est pratiquement vrai :

- Transistors MOSFET (Field Effect Transistor) : Ces composants sont très rapides mais consomment relativement peu.
- Transistors bipolaires : Plus lents que les MOSFET, mais avec des avantages plus importants (quelques kHz à quelques dizaines de kW).
- Transistors IGBT : Composants dans la gamme standard (jusqu'à 20 kHz à plusieurs dizaines de kW).
- Thyristor GTO : Commute la puissance élevée très lentement.
- Thyristor : Peut être contrôlé lorsqu'il est ouvert, mais dépend d'un circuit externe pour se fermer. La fonction de la puissance [KW] et de la puissance [KHz] peut être schématisée comme suit [5] :

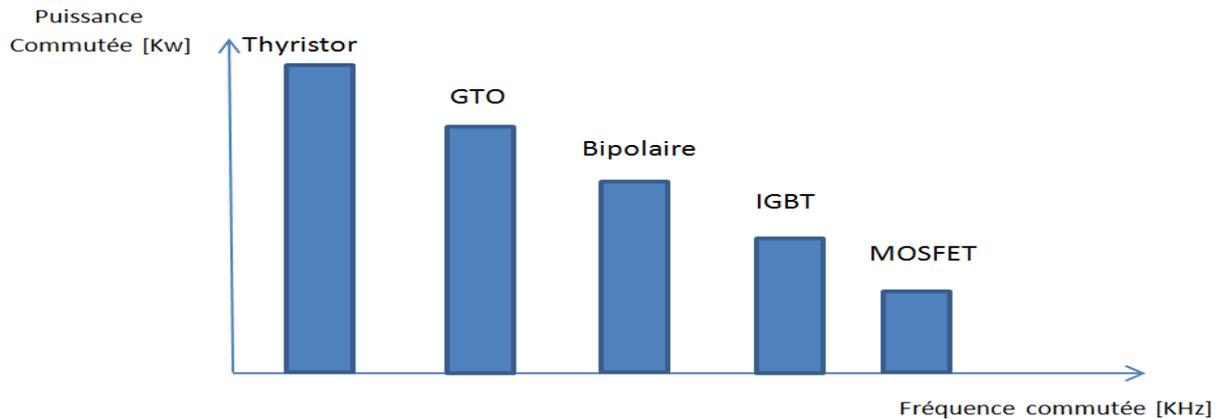


Figure I-5 : Choix des composants selon la puissance et la fréquence.

## I-6-Les applications des onduleurs :

Parmi les divers domaines d'application des onduleurs autonomes figurent les onduleurs à fréquence fixe à commutation forcée. Ceux-ci sont généralement alimentés par des batteries de stockage et jouent généralement le rôle d'une alimentation de sécurité, rendant ainsi disponible le principe des onduleurs autonomes.

Variateur de fréquence à commutation forcée: alimenté par le secteur industriel via un circuit redresseur pour fournir la fréquence et la tension efficaces nécessaires au fonctionnement d'un moteur CA à vitesse variable.

Nous enrichissons notre exploration en citant des exemples concrets d'applications des onduleurs :

### I-6-1-Réglage de la vitesse de rotation d'un moteur synchrone :

La vitesse d'un moteur synchrone est définie par des pulsations de courant statiques. Par conséquent, changer la vitesse nécessite de changer la fréquence de la tension d'alimentation. Ce qui fait, la tension du réseau doit être redressée à la fréquence souhaitée.

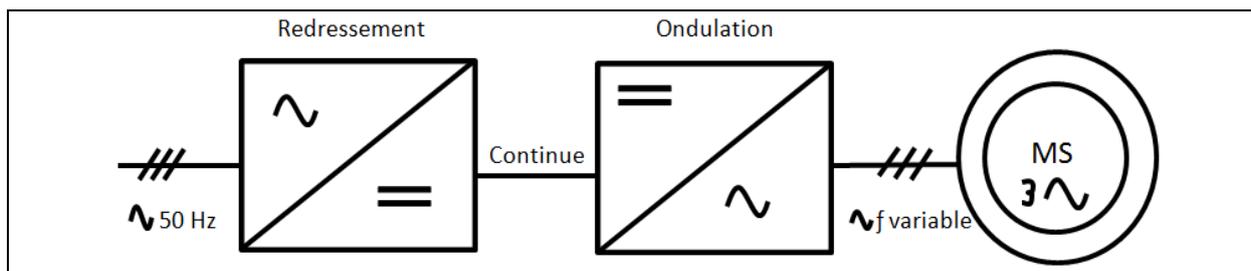
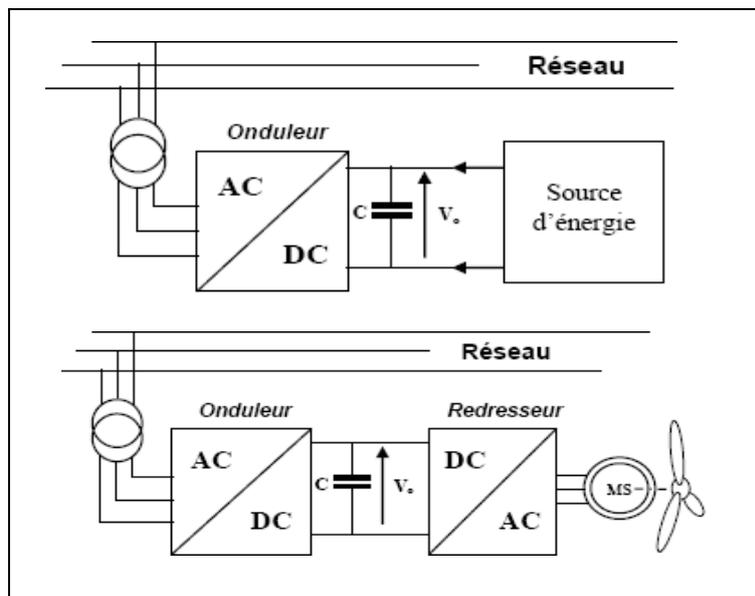


Figure I-6 : Réglage de la vitesse de rotation d'un moteur synchrone.

**N.B :** Afin de maintenir la puissance nominale du moteur lorsque la fréquence fluctue, il est en effet nécessaire de maintenir le rapport (V/f) constant (Lorsque la fréquence augmente, la tension d'alimentation doit être augmentée en conséquence).

### **I-6-2-Alimentation de secours :**

En cas de coupure de courant, l'onduleur assure la continuité de l'alimentation de la machine via la batterie. Dans le domaine de l'informatique professionnelle, un onduleur est indispensable pour éviter les pertes d'informations lors des coupures de courant.



*Figure I-7 : Alimentation de secours.*

### **I-6-3-Filtres actifs :**

Un filtre actif est un onduleur alimenté par une source de courant continu ou de tension. Les onduleurs peuvent produire du courant ou de la tension dont le contenu harmonique ne dépend que de la loi de commande des interrupteurs. Un filtre actif agit donc comme une source de tension ou de courant harmonique déphasée par rapport au secteur, régénérant un courant de source quasi-sinusoïdal. Les filtres actifs peuvent être connectés en série ou en parallèle sur le réseau, selon qu'ils sont destinés à compenser des tensions ou des courants harmoniques.

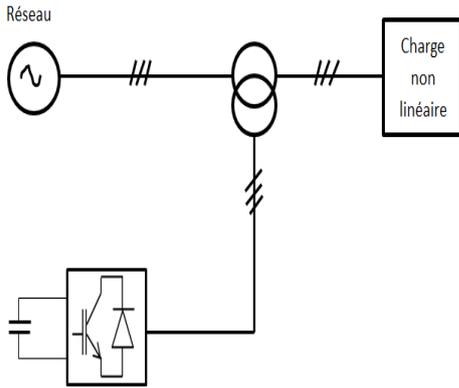


Figure I-8 : Filtre actif parallèle (FAP).

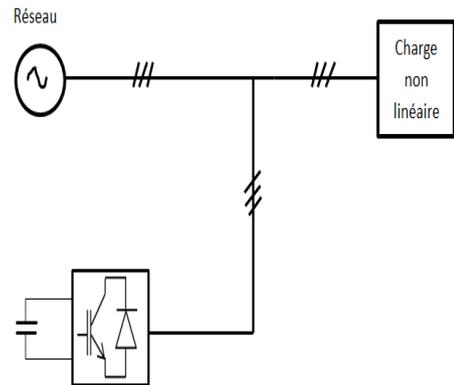


Figure I-9 : Filtre actif série (FAS).

#### I-6-4-Compensateur de puissance réactive :

Un onduleur de tension agit comme un compensateur de puissance réactive. La régulation de l'énergie réactive se fait en contrôlant la tension de sortie  $V_{out}$  de l'appareil. La valeur efficace de la composante fondamentale est supérieure à la valeur RMS de  $V_r$ , et le courant dans l'inductance L est en retard par rapport à  $V_r$  de  $90^\circ$  [6].

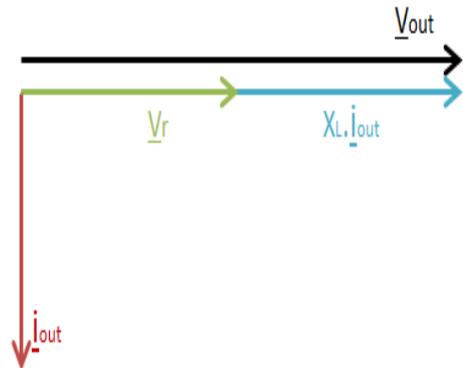
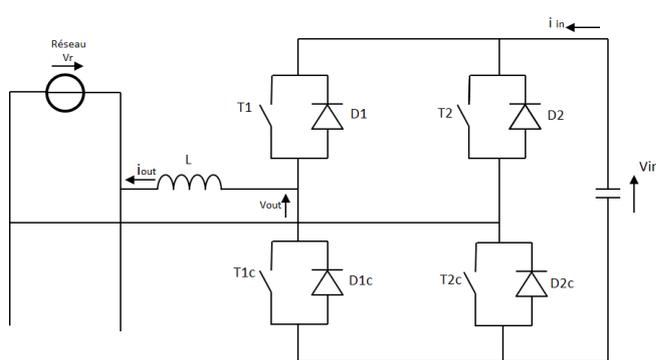


Figure I-10 : Structure d'un compensateur de puissance réactive. Figure I-11 : Diagramme vectoriel.

#### I-7-Conclusion :

Ce chapitre nous a permis d'acquérir une compréhension solide des fondements des onduleurs et de leurs applications dans diverses industries, ce qui sera essentiel pour approfondir notre étude dans les prochains chapitres.

# **Chapitre II: Stratégies de commande des onduleurs**

### II-1-Introduction :

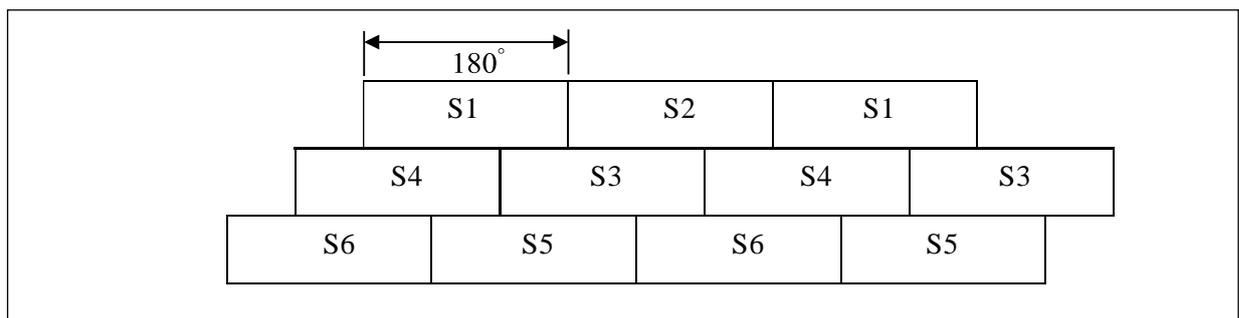
L'objectif de la commande de l'onduleur est de pouvoir reproduire au mieux le courant ou les tensions perturbées de référence, grâce aux ordres de commande appliqués au pilote des interrupteurs de puissance, Les convertisseurs statiques sont regroupés en deux familles principales selon leur mode de commande :

- Commandes à ondes carrés :
  - ✚ Commande pleine onde.
  - ✚ Commande décalé ( $120^\circ$ ).
- Commandes à haute fréquence :
  - ❖ La commande par hystérésis des courants.
  - ❖ La commande par modulation de largeur d'impulsion (Pulse Width Modulation) :
    - ✚ Commande MLI Triangulo-Sinusoidale (MLI-TS),
    - ✚ La commande MLI vectorielle.

### II-2-Commandes à ondes carrés :

#### II-2-1-Commande pleine onde :

Lorsque la séquence de commande de fermeture d'un interrupteur correspond à la séquence de commande d'ouverture d'un interrupteur sur le même bras, on dit qu'il s'agit d'un onduleur à  $180^\circ$ . Dans la première branche de l'onduleur, l'interrupteur **S1** est fermé pendant la moitié du cycle ( $180^\circ$ ) et l'interrupteur **S2** est fermé pendant la seconde moitié du cycle. La même procédure est utilisée pour les deux autres branches de l'onduleur, mais avec des décalages de  $(2\pi/3)$  et  $(4\pi/3)$  par rapport à la première branche.

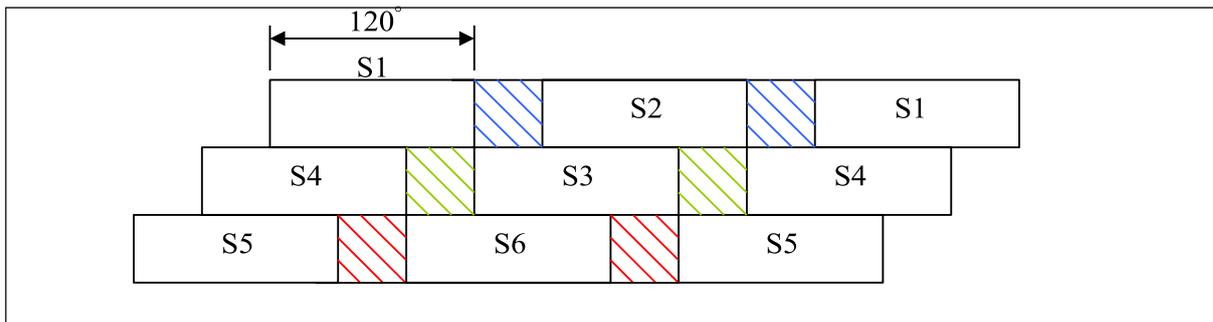


*Figure II- 1 : Commande  $180^\circ$ .*

L'utilisation de cette méthode de contrôle de l'onduleur crée un système de tension alternative triphasé caractérisé par l'absence d'harmoniques multiples de trois.

### II-2-2-Commande décalé (120°) :

Dans ce cas (Figure II-2 ), la commande de fermeture de l'interrupteur ne correspond pas à la commande d'ouverture de l'interrupteur sur le même bras de l'onduleur. Ce type d'onduleur est communément appelé un onduleur à 120°, faisant référence à l'angle de déphasage 60° entre les commandes d'ouverture et de fermeture des interrupteurs.



*Figure II-2 : Commande 120°.*

Dans ce schéma de contrôle, chaque interrupteur fonctionne sur 120 degrés, l'ordre est donc **S1, S3, S5** successivement. De même pour les interrupteurs **S2, S4 et S6** avec un décalage de 60° par rapport à **S1, S3, S5** [1].

### II-3-Commande par hystérésis des courants:

La commande par hystérésis, également connue sous le nom de commande tout ou rien, est une méthode de commande non linéaire qui exploite l'erreur existante entre le courant de référence et le courant produit par l'onduleur. Cette erreur est ensuite comparée à un seuil appelé bande d'hystérésis.

La stratégie de contrôle la plus simple utilisée à cette fin consiste à comparer le courant de phase mesuré avec le courant de référence (Figure II-4) à l'aide d'un comparateur à hystérésis. Ce comparateur génère des impulsions d'amorçage et de blocage des interrupteurs de l'onduleur afin de maintenir le courant de phase dans une plage d'hystérésis autour du courant de référence (Figure II-3 ).

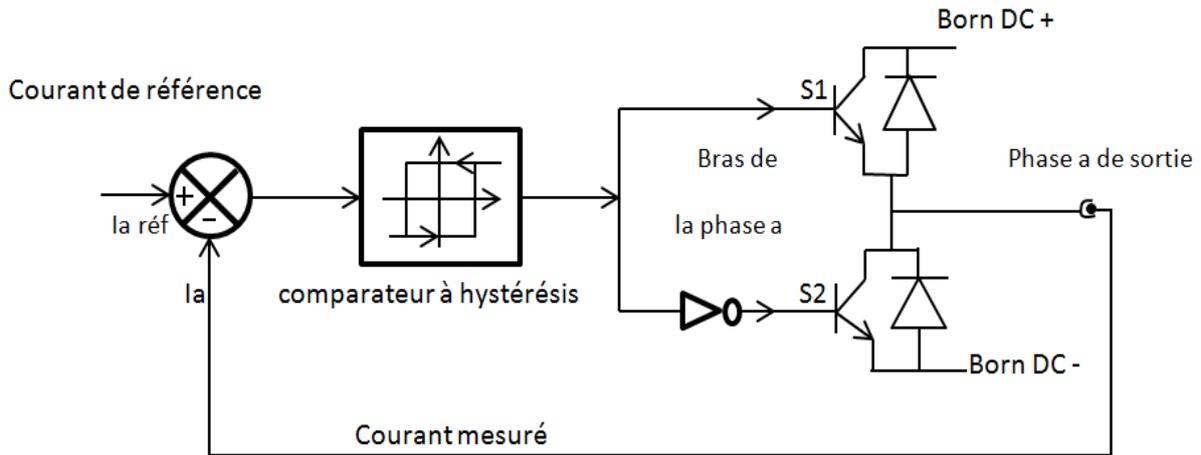


Figure II-3 : Schéma de principe du contrôle par hystérésis d'un bras de l'onduleur.

En conséquence, les conditions de commutation des interrupteurs statiques «  $S_i$  » ( $i=1, 2, 3, 4, 5, 6$ ) de l'onduleur sont déterminées en fonction des états logiques correspondants de la manière suivante :

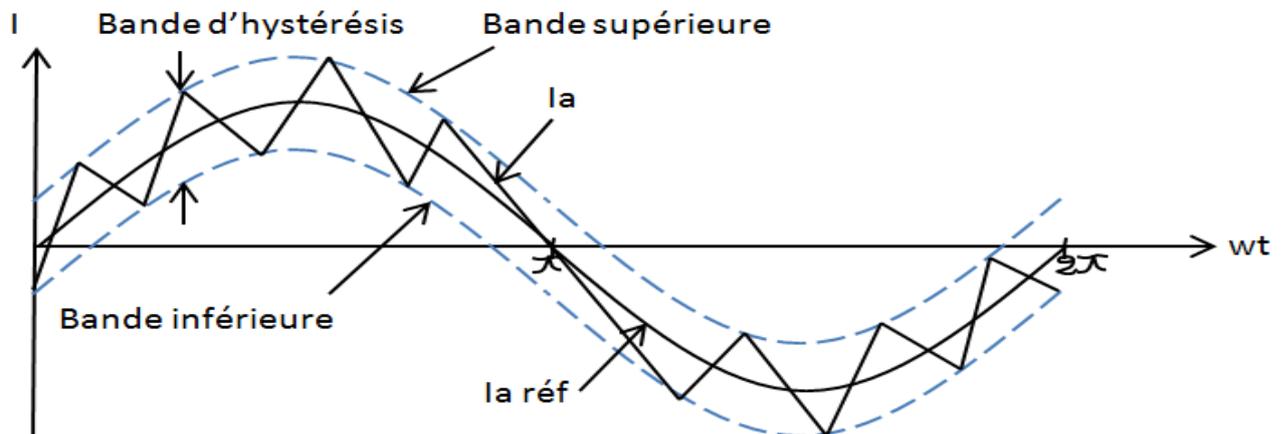


Figure II-4 : Bande d'Hystérésis.

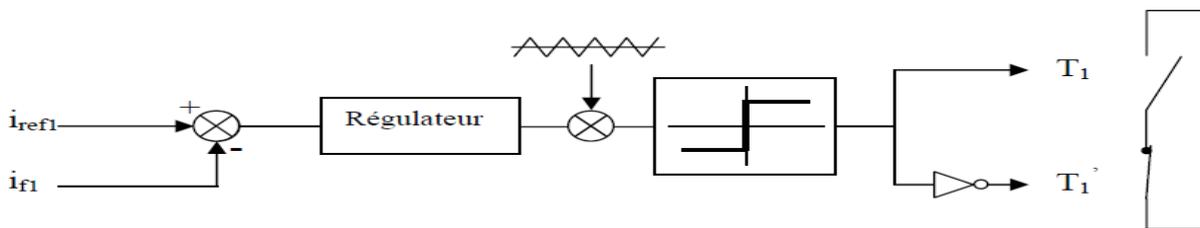
Les variables suivantes sont définies :

- $I_i$  ( $i = a, b, c$ ) représentent les courants des phases statoriques ( $I_a, I_b, I_c$ ).
- $I_{i\text{ref}}$  ( $i = a, b, c$ ) correspondent aux courants de référence provenant des circuits de commande des trois bras.

- La quantité  $\Delta i$  : appelée bande d'hystérésis, est choisie de manière à respecter la fréquence de commutation maximale des semi-conducteurs contrôlés et à minimiser les harmoniques de courant de manière adéquate [7].

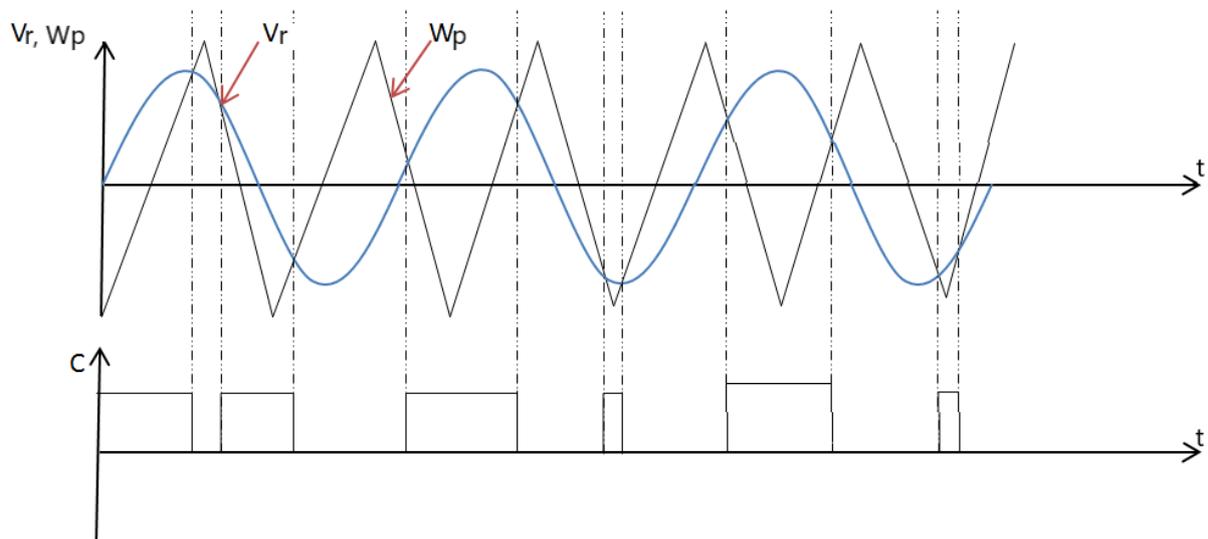
#### **II-4-Commande par modulation de largeur d'impulsion (MLI) :**

La technologie de contrôle PWM résout le problème de contrôle de la fréquence de commutation en fonctionnant à une fréquence fixe qui peut être facilement filtrée après l'onduleur. La modulation de largeur d'impulsion la plus simple et la plus connue est sans doute la MLI à échantillonnage naturel, également connu sous le nom de MLI intersective. Dans cette technique de commande, un régulateur est d'abord mis en place et la différence entre le courant mesuré et sa valeur de référence détermine la tension de référence de l'onduleur (modulateur). Cette dernière est alors comparée à une onde triangulaire, une porteuse haute fréquence qui fixe la fréquence de découpage. La sortie du comparateur fournit des ordres de commande aux interrupteurs. Un schéma fonctionnel est présenté à la Figure II-3.



*Figure II-4 : Principe du contrôle des courants par PWM.*

Les signaux logiques C qui en résultent donnent l'ordre de commutation des interrupteurs du filtre [8] :



*Figure II-5 : Répartition des commutations sur un bras d'un onduleur.*

Cette MLI est conçue pour résoudre ces problèmes et présente les avantages suivants :

- ✓ Fluctuations de la fréquence de la tension de sortie.
- ✓ Rejet des harmoniques de tension spécifiques.
- ✓ Déplacez les harmoniques vers des fréquences plus élevées [9].

**II-4-1-Commande MLI Triangulo-Sinusoidale (MLI-TS) :**

La commande MLI Triangulo-sinusoidale implique la comparaison d'une valeur de tension de référence, représentant le signal souhaité en sortie et de fréquence  $F_r$ , avec une porteuse en dent de scie ou de forme triangulaire de fréquence  $F_p$ . Les points d'intersection entre la modulation et la porteuse engendrent des transitions entre l'état d'allumage et d'extinction, formant ainsi des impulsions de durée variable. Ces impulsions se recombinent pour reconstituer le fondamental de la sinusoïde de référence. L'étude de la commande MLI triangulo-sinusoidale repose sur deux paramètres caractéristiques de cette méthode :

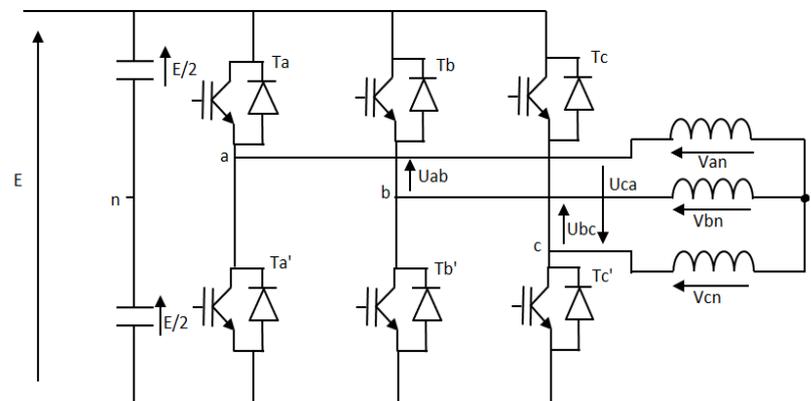
- L'indice de modulation  $m$  qui est l'image du rapport des fréquences de la porteuse  $F_p$  sur celle de la référence  $F_r$ . Si  $m$  est un entier, la modulation est synchrone. Sinon c'est asynchrone:

$$m = \frac{F_p}{F_r} \tag{II.1}$$

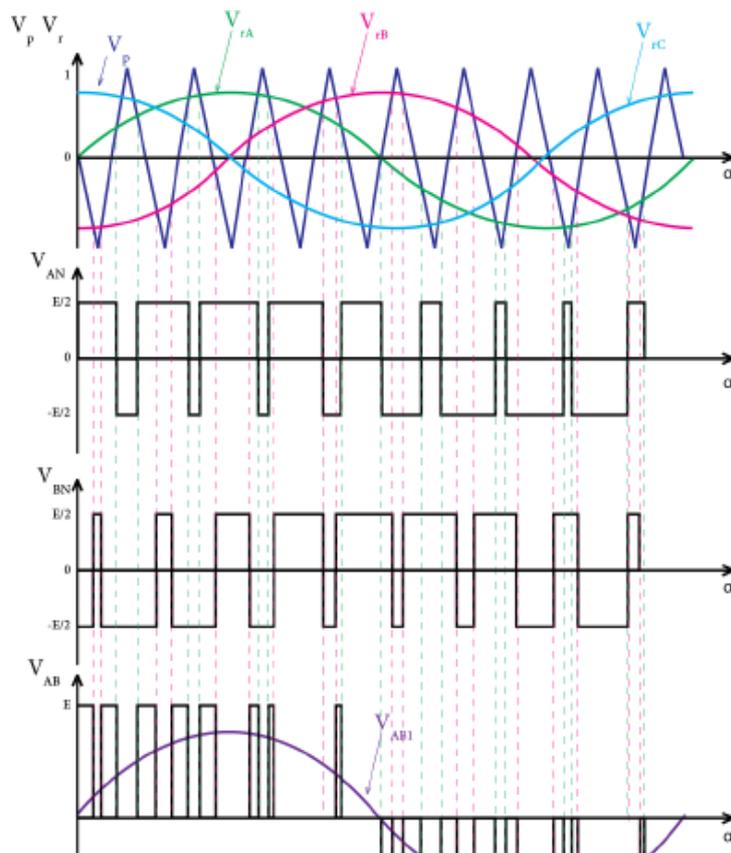
- Le facteur de modulation  $r$  est le rapport de l'amplitude de la tension de référence  $V_{rm}$  et de l'amplitude de la tension porteuse  $V_{pm}$ .

$$r = \frac{V_{rm}}{V_{pm}} \tag{II.2}$$

La porteuse la plus appropriée pour les convertisseurs à 2 niveaux est la triangulaire bipolaire. La figure ci-dessous montre le principe de génération d'impulsions de commande d'un onduleur de tension à deux niveaux [10].



*Figure II-6 : Onduleur triphasé de tension en pont.*



*Figure II-7 : Formes d'onde de tension porteuse, de référence et de sortie d'un onduleur triphasé contrôlé par PWM Triangulo- sinusoïdale.*

#### **II-4-2-Commande MLI Vectorielle :**

La modulation vectorielle est une modulation en temps réel. Elle utilise le fait qu'un vecteur peut présenter les trois tensions d'un système triphasé de somme nulle. Cette méthode de commande fera l'objet de notre partie spéciale qui sera étudié au chapitre suivant.

#### **II-5-Conclusion :**

La compréhension de ces différentes stratégies de commande est essentielle pour optimiser les performances des onduleurs dans diverses applications. Chaque technique présente des avantages et des inconvénients qui doivent être pris en compte en fonction des exigences spécifiques du système. Dans le chapitre suivant, nous allons s'intéressé à la commande vectorielle (SVM), formules et caractéristiques appropriées à cette stratégie.

# **Chapitre III: Description de la commande SVM d'un onduleur**

### III- 1-Introduction :

Afin de générer une tension sinusoïdale avec le moins d'harmoniques possible en sortie de l'onduleur, différentes méthodes ont été développées. SVM est la méthode la plus récente de contrôle des onduleurs de tension triphasés. Le principe de cette méthode est de convertir les états de commutation de l'onduleur sous forme de vecteurs, chaque vecteur représentant une situation bien définie de la commutation de ce dernier, ceci La commande (SVM) détermine les séquences d'allumage et d'extinction des composants du convertisseur et minimise les harmoniques de la tension de sortie de l'onduleur.

Dans ce chapitre nous allons étudier la MLI vectorielle et ses dérivées appliquées aux onduleurs triphasés de tension.

### III- 2-Principe de la MLI vectorielle :

Le principe de la modélisation vectorielle (SVM) est de recomposer le vecteur de tension de référence  $V_{ref}$  à partir de huit vecteurs de tension. Tous ces vecteurs correspondent à la combinaison d'états de commutation de l'onduleur de tension triphasé, Chaque vecteur est représenté par un nombre binaire, comprenant trois bits. Le premier bit représente le premier bras, le second est pour le second bras et ainsi de suite, puis on transforme le plan triphasé ( $v_a, v_b, v_c$ ) en un plan biphasé ( $v_\alpha, v_\beta$ ) par la transformation de Clarke, ce qui est pratique pour le calcul et le positionnement sur la base du vecteur de référence, La représentation dans le plan  $\alpha\beta$  de ces vecteurs équilibrèrent répartis tous les  $60^\circ$ , permet de déterminer un hexagone, les vecteurs tensions des six états actifs, peuvent s'écrire en fonction des portions de l'hexagone appelés secteur  $k$ , où  $k = 1, \dots, 6$  [11].

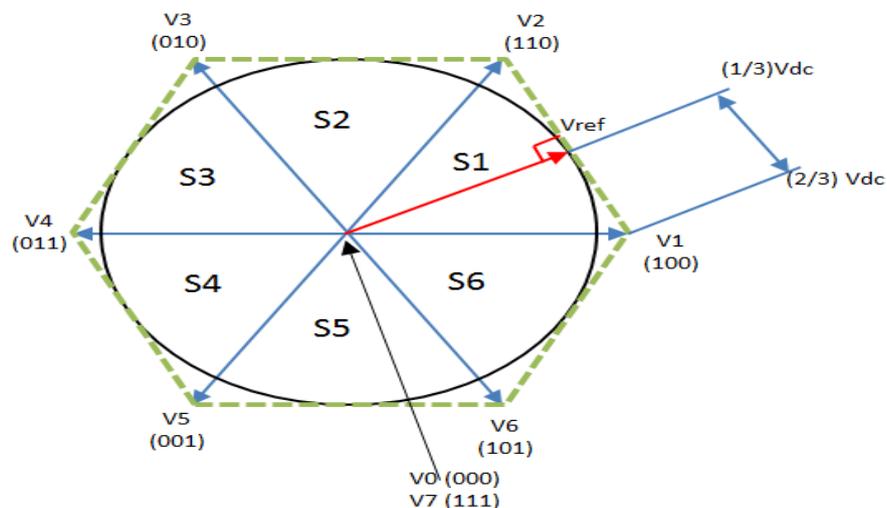
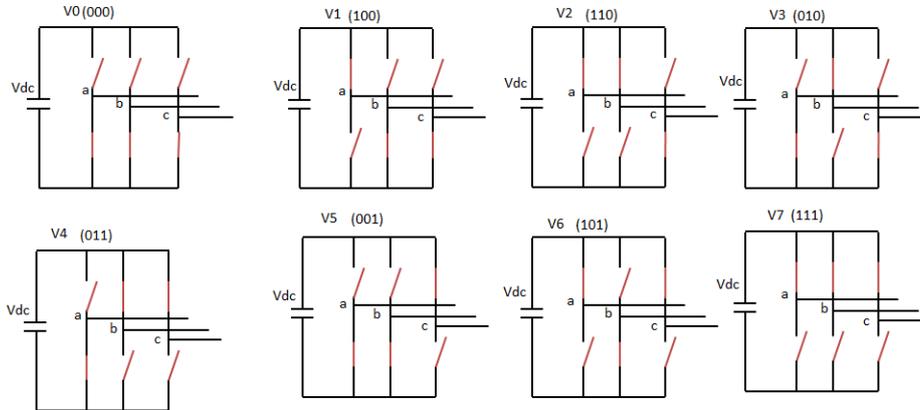


Figure III-1 : Les vecteurs de tension de l'onduleur dans le plan  $(\alpha, \beta)$ .



**Figure III-2** : Emplacement des huit vecteurs stationnaires pour un onduleur de tension à deux niveaux

### **III- 3-La transformation de Clark :**

Pour simplifier les calculs en modulation vectorielle, on utilise la transformation de Clark qui permet de remplacer le système triphasé à somme nulle de trois tensions  $V_a, V_b, V_c$  par un système biphasé considérant le transfert de puissance [12].

$$V_{ref} = V_\alpha + jV_\beta = \sqrt{\frac{2}{3}}(V_a + aV_b + a^2V_c) \quad (III.1)$$

Avec :  $a = e^{j\frac{2\pi}{3}}$

On aura donc :

$$V_{ref} = \sqrt{\frac{2}{3}}(V_a + e^{j\frac{2\pi}{3}}V_b + e^{j\frac{4\pi}{3}}V_c) \quad (III.2)$$

$$\begin{bmatrix} V_\alpha \\ V_\beta \end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} \quad (III.3)$$

$$|\vec{V}_{ref}| = \sqrt{V_\alpha^2 + V_\beta^2} \quad (III.4)$$

$$\varphi = \tan^{-1}\left(\frac{V_\beta}{V_\alpha}\right) \quad (III.5)$$

- ✚ Une analyse combinatoire de tous les états possibles des interrupteurs permet de calculer les vecteurs de tensions  $\mathbf{V}_\alpha$ ,  $\mathbf{V}_\beta$  correspondantes [13] :

**Tableau III-1** : Table de vérité pour un onduleur de tension triphasé à deux niveaux.

Vecteur	S1	S3	S5	Va	Vb	Vc	V $_\alpha$	V $_\beta$	V $_{ref}e^{j\varphi}$
<b>V0</b>	0	0	0	0	0	0	0	0	0
<b>V5</b>	0	0	1	-E/3	-E/3	2E/3	$-\sqrt{\frac{1}{6}}E$	$-\sqrt{\frac{1}{2}}E$	$\sqrt{\frac{2}{3}}Ee^{j\frac{4\pi}{3}}$
<b>V3</b>	0	1	0	-E/3	2E/3	-E/3	$-\sqrt{\frac{1}{6}}E$	$\sqrt{\frac{1}{2}}E$	$\sqrt{\frac{2}{3}}Ee^{j\frac{2\pi}{3}}$
<b>V4</b>	0	1	1	-2E/3	E/3	E/3	$-\sqrt{\frac{2}{3}}E$	0	$\sqrt{\frac{2}{3}}Ee^{j\frac{3\pi}{3}}$
<b>V1</b>	1	0	0	2E/3	-E/3	-E/3	$\sqrt{\frac{2}{3}}E$	0	$\sqrt{\frac{2}{3}}Ee^{j0}$
<b>V6</b>	1	0	1	E/3	-2E/3	E/3	$\sqrt{\frac{1}{6}}E$	$-\sqrt{\frac{1}{2}}E$	$\sqrt{\frac{2}{3}}Ee^{j\frac{5\pi}{3}}$
<b>V2</b>	1	1	0	E/3	E/3	-2E/3	$\sqrt{\frac{1}{6}}E$	$\sqrt{\frac{1}{2}}E$	$\sqrt{\frac{2}{3}}Ee^{j\frac{\pi}{3}}$
<b>V7</b>	1	1	1	0	0	0	0	0	0

### III-4-Les séquences de commutation de l'onduleur:

La symétrie du système triphasé permet de ramener l'étude au cas général Secteur à 60 degrés. On se place alors dans la situation où le vecteur de référence  $\overrightarrow{\mathbf{V}_{ref}}$  est dans le secteur 1. Dans ce cas, la tension de référence  $\overrightarrow{\mathbf{V}_{ref}}$  dans le repère est reconstituée en faisant une moyenne temporelle des tensions  $\overrightarrow{\mathbf{V}_1}$ ,  $\overrightarrow{\mathbf{V}_2}$ ,  $\overrightarrow{\mathbf{V}_0}$  et  $\overrightarrow{\mathbf{V}_7}$ .

Après, il suffit de déterminer la position du vecteur de référence  $\overrightarrow{\mathbf{V}_{ref}}$  dans le repère  $\alpha$ ,  $\beta$  et son secteur  $i$ . Pour une fréquence de commutation  $f_m$  suffisamment élevée, le vecteur spatial de référence  $\overrightarrow{\mathbf{V}_{ref}}$  est considéré comme constant pendant la période de commutation. En considérant que  $\overrightarrow{\mathbf{V}_1}$  et  $\overrightarrow{\mathbf{V}_2}$  sont des constantes et  $\overrightarrow{\mathbf{V}_0} = \overrightarrow{\mathbf{V}_7} = 0$ , il s'en suit pour un cycle de commutation.

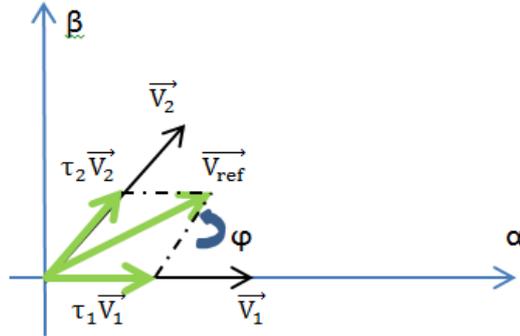
$$\vec{V}_{ref} T_m = \vec{V}_1 T_1 + \vec{V}_2 T_2 + \vec{V}_0 T_0 + \vec{V}_7 T_0 \quad (III.6)$$

Avec:

$T_1$  : Temps alloué au vecteur  $\vec{V}_1$

$T_2$  : Temps alloué au vecteur  $\vec{V}_2$

$T_0$  : Temps partagé entre les 2 vecteurs nuls  $\vec{V}_0$  et  $\vec{V}_7$



**Figure III-3** : Principe de construction du vecteur de tension  $\vec{V}_{ref}$

La résolution de cette équation, après décomposition sur les deux axes du plan Complexe ( $\alpha$ ,  $\beta$ ) donne [14]:

Avec :

- $k$  : le nombre de secteur.
- $\tau_1 = \frac{T_1}{T_m}$  Et  $\tau_2 = \frac{T_2}{T_m}$

$$T_1 = \frac{\sqrt{2}}{E} V_{ref} T_m \sin\left(\frac{k\pi}{3} - \varphi\right) \quad (III.7)$$

$$T_2 = \frac{\sqrt{2}}{E} V_{ref} T_m \sin\left(\varphi - \frac{(k-1)\pi}{3}\right) \quad (III.8)$$

$$T_0 = T_m - T_1 - T_2 \quad (III.9)$$

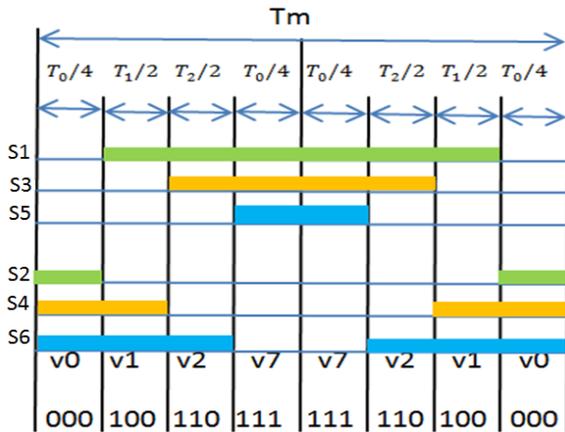


Figure III-4 : Formes des largeurs d'impulsion du secteur 1.

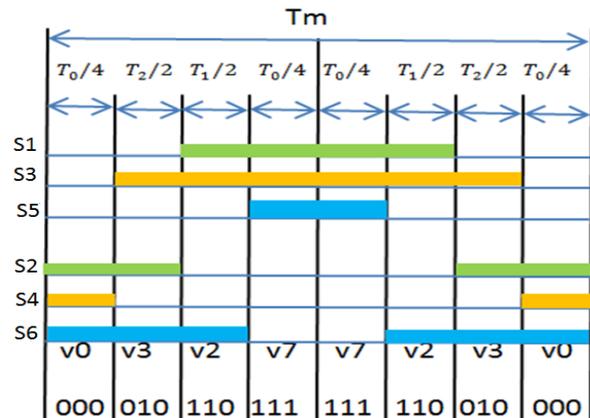


Figure III-5 : Formes des largeurs d'impulsions pour secteur 2.

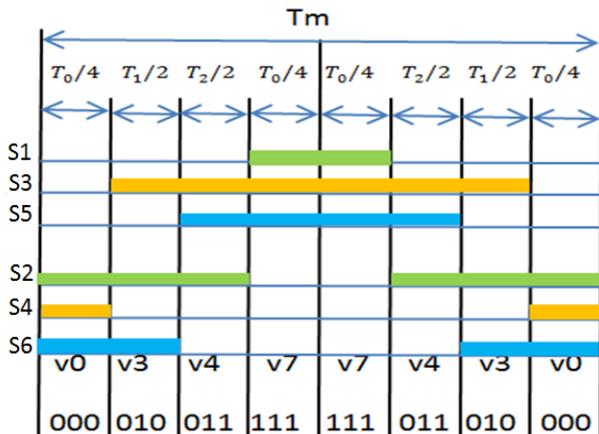


Figure III-6 : Formes des largeurs d'impulsion du secteur 3.

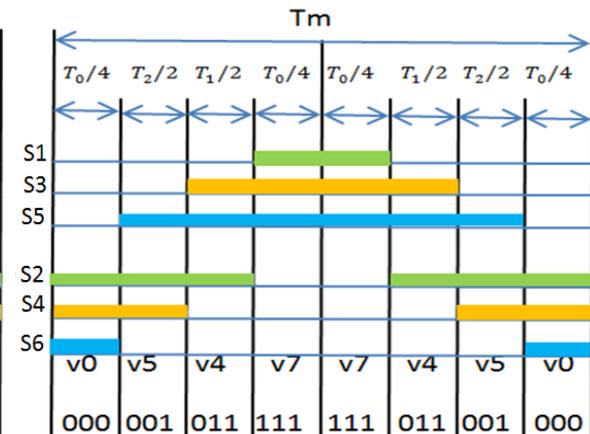


Figure III-7 : Formes des largeurs d'impulsions pour secteur 4.

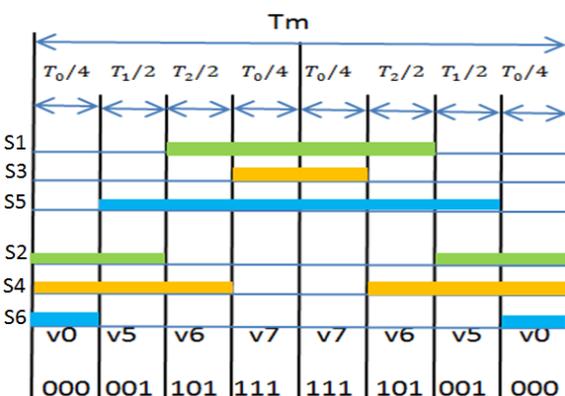


Figure III-8 : Formes des largeurs d'impulsion du secteur 5.

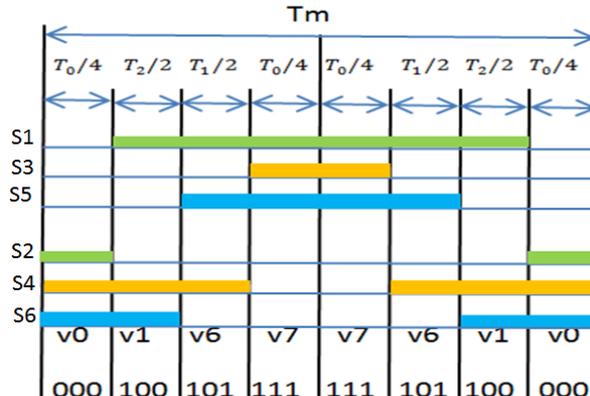


Figure III-9 : Formes des d'impulsion du secteur 6.

### III- 5-Algorithmes de modulation de la largeur d'impulsion (MLI) vectorielle :

Quatre algorithmes de modulation de largeur d'impulsion vectorielle (SVM) sont pris en compte :

- ✓ La séquence alignée à droite (SVM1) (Right aligned sequence).
- ✓ La séquence symétrique (SVM2) (Symmetric Sequence ).
- ✓ La Séquence vectorielle nulle alternée (SVM3) (Alternating Zero Vector Sequence).
- ✓ La Séquence sans commutation de courant le plus élevé (SVM4) (Highest Current Not-Switched Sequence).

**N.B:** Les schémas de modulation décrivent le cas quand le vecteur de référence est dans le secteur n°1, tous les autres cas sont symétriques.

#### III- 5-1-Schémas de modulation :

##### III- 5-1-1-La séquence alignée à droite (SVM1):

L'algorithme synthétise le vecteur de tension de sortie de manière simple, au début de la période de modulation  $T_m$  tous les interrupteurs inférieurs (ou supérieurs) sont activés Et s'arrêter à son tour, pour que le temps d'utilisation  $T_0$  du vecteur de tension zéro soit conçu entre  $V_7$  (111) et  $V_0$  (000).

Le schéma de commutation est illustré sur la figure pour deux périodes de modulation, il représente le signal de déverrouillage du bras supérieur de l'onduleur. Dans une période de modulation  $T_m$ , l'interrupteur supérieur à trois temps de commutation, et l'interrupteur inférieur à trois temps de commutation. Exécute une séquence justifiée à gauche, où la séquence vectorielle est exactement l'opposé de la séquence justifiée à droite.

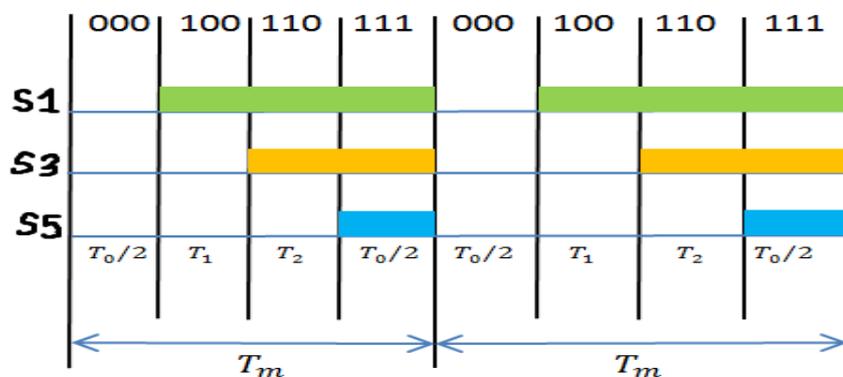


Figure III-10 : La séquence alignée à droite (SVM1).

**III- 5-1-2-Séquence symétrique (SVM2) :**

Rendez la forme d'onde de commutation symétrique, comme indiqué sur la figure Le nombre de temps de commutation dans un cycle de modulation est de six.

Ce schéma a le même nombre de commutateurs que SVM1, trois commutateurs ouverts et trois commutateurs fermés, les pertes de commutation est similaire à SVM1.

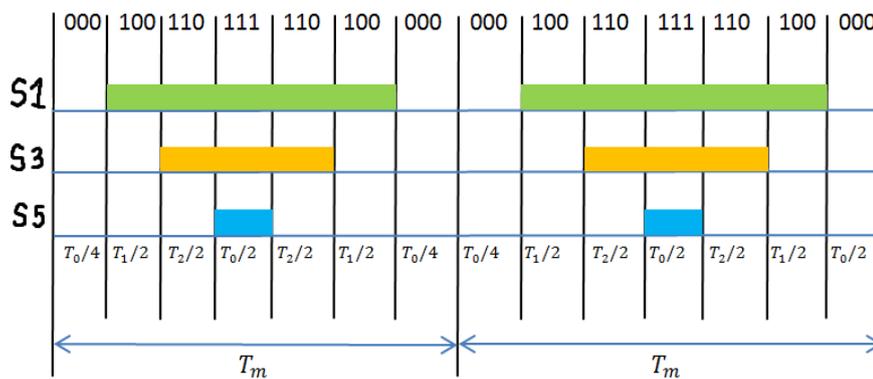


Figure III-11 : Séquence symétrique (SVM2).

**III- 5-1-3-Séquence vectorielle nulle alternée (SVM3) :**

Dans ce schéma, les vecteurs nuls  $V_7(111)$  et  $V_0(000)$  sont utilisés alternativement Dans des périodes adjacentes, la fréquence de commutation est divisée par tel divisé en deux, comme la montre la Figure suivante :

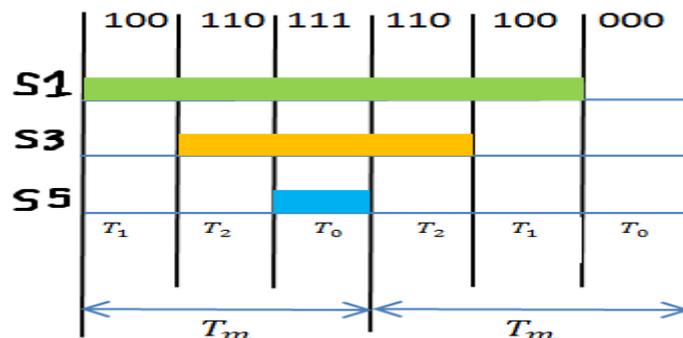


Figure III-12 : Séquence vectorielle nulle alternée (SVM3).

**III- 5-1-4-Séquence sans commutation de courant le plus élevé (SVM4) :**

Ce schéma est basé sur des pertes de commutation d'environ est proportionnel à l'amplitude du courant de commutation, il serait donc avantageux d'éviter de commuter le bras de l'onduleur transportant le courant instantané le plus élevé. Cela est possible dans la plupart des cas car tous

les vecteurs d'état de commutation (SSV) voisins dans le même bras ont des états de commutation différents. Donc, on n'utilise que des vecteurs nul  $V_7(111)$  ou  $V_0(000)$  dans un secteur donné, un des bras ne bascule pas comme indiqué [7].

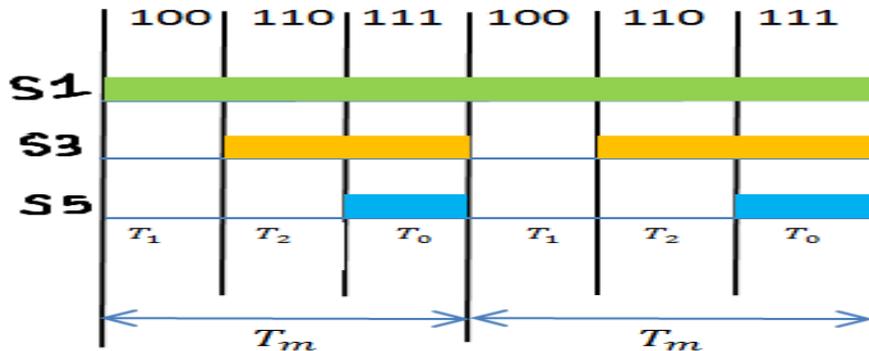


Figure III-13 : Séquence sans commutation de courant le plus élevé (SVM4).

### III- 6-La simulation de la commande SVM sous Matlab Simulink:

Dans cette partie, nous allons implémenter la commande SVM d'un onduleur triphasé de tension à deux niveaux sous logiciel Matlab, et cela on aura besoin des schémas de blocs qui présente les calculs appropriés à cette commande, tout d'abord on commence par schématiser notre onduleur qui débitera avec une charge RL en premier lieu et avec une MAS en deuxième.

#### III- 6-1-Débit sur charge RL :

Dans ce cas, on fixe la charge RL aux paramètres suivants :  $R=10$  ohms,  $L= 1$  mH.

Premièrement, on utilise des tensions de références ( $V_a, V_b, V_c$ ) qui représentent un système triphasé déphasées de  $(2*\pi/3)$  entre elles et d'amplitude équivalent à  $311V \ll 220*\sqrt{2} V \gg$ , ensuite on calcul l'angle du vecteur de référence ( $\phi$ ) en fonction des tensions dans le repère  $(\alpha, \beta)$  ( $V_{\alpha}, V_{\beta}$ ), après on passe au détection des secteurs qui nécessite de calculer les durées de conduction ( $T_1, T_2, T_0$ ) et cela on fixant la période de modulation à  $(1/3000\text{Hz})$ , enfin on termine par déterminer les signaux de conduction des interrupteurs en fonction des durées de conduction ( $T_1, T_2, T_0$ ) et numéro de secteur. La figure suivante représente le schéma global de la commande MLI vectorielle appliquée à un onduleur triphasé de tension à deux niveaux :

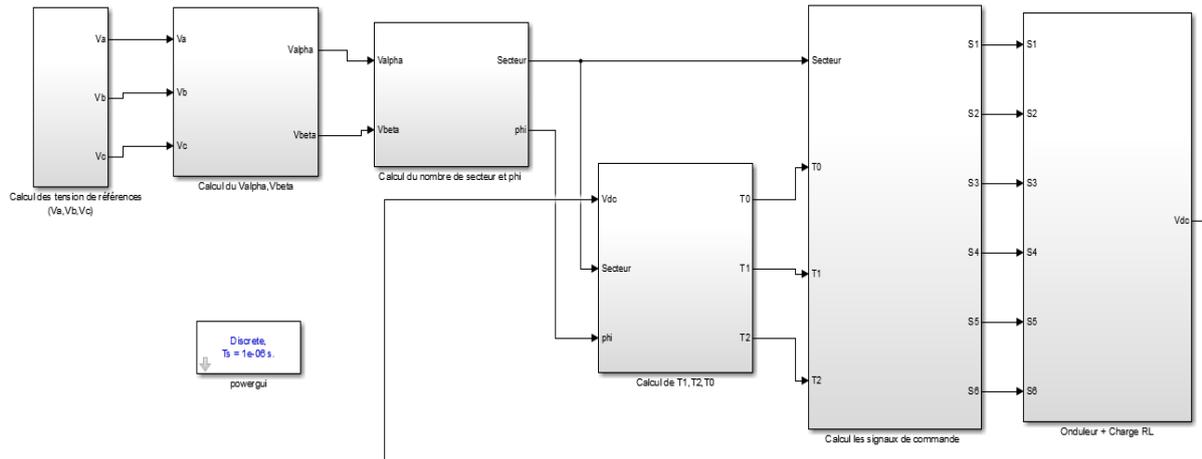


Figure III-14 : Schéma global de la commande SVM d'un onduleur sous Matlab Simulink

### III- 6-1-1 Représentation des tensions de référence :

De plus, afin d'éviter tout décalage indésirable entre les secteurs, les tensions de référence seront calculées en fonction de la loi du cosinus plutôt que du sinus. Cela permettra d'obtenir des résultats précis et cohérents dans nos calculs.

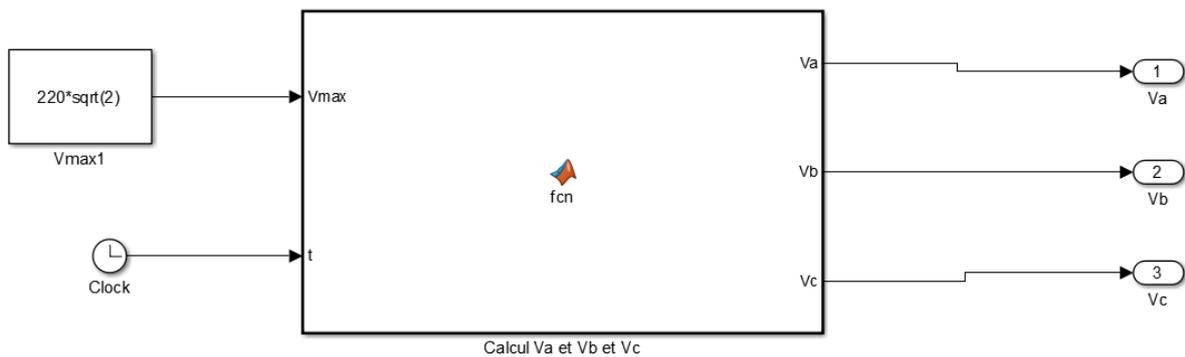


Figure III-15 : Schéma de bloc des tensions de référence.

```

Calcul des tension de références (Va,Vb,Vc)/Calcul Va et Vb et Vc*
1  function [Va,Vb,Vc] = fcn(Vmax,t)
2  %calcul des tensions de référence (Va, Vb et Vc) avec cos.
3  Va=Vmax*cos(100*pi*t+pi);
4  Vb=Vmax*cos(100*pi*t+(2*pi/3)+pi);
5  Vc=Vmax*cos(100*pi*t+(4*pi/3)+pi);
6  end
    
```

Figure III-16 : Programme pour calcul des tensions de référence.

### III- 6-1-1-1-Résultats de simulation :

Les figures suivantes représentent les tensions de référence appliquées à la commande du l'onduleur :

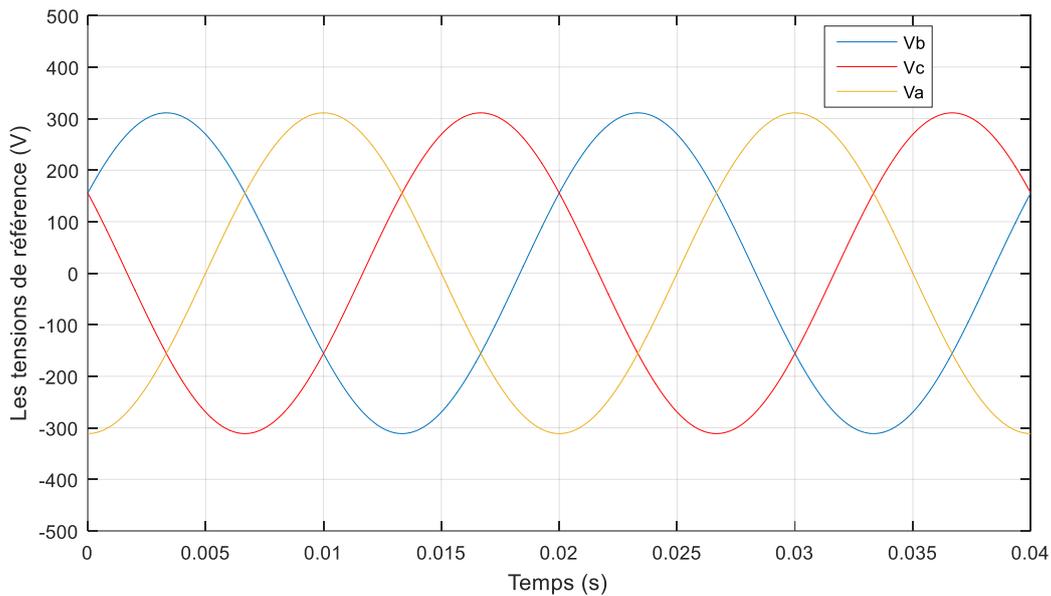


Figure III-17 : Allure des tensions de référence.

**III- 6-1-2--Détermination des tensions dans le repère (α, β) :**

Pour calculer Valpha et Vbeta, on utilise la transformation de Clark via l'équation (III.3) :

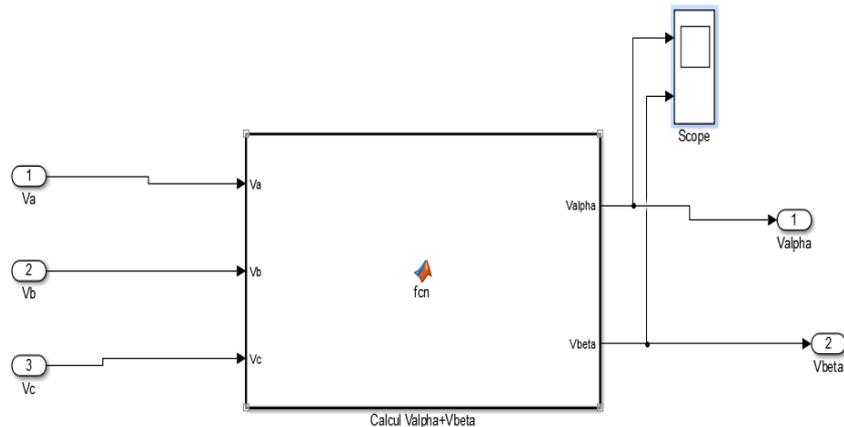


Figure III-18 : Bloc du Valpha, Vbeta.

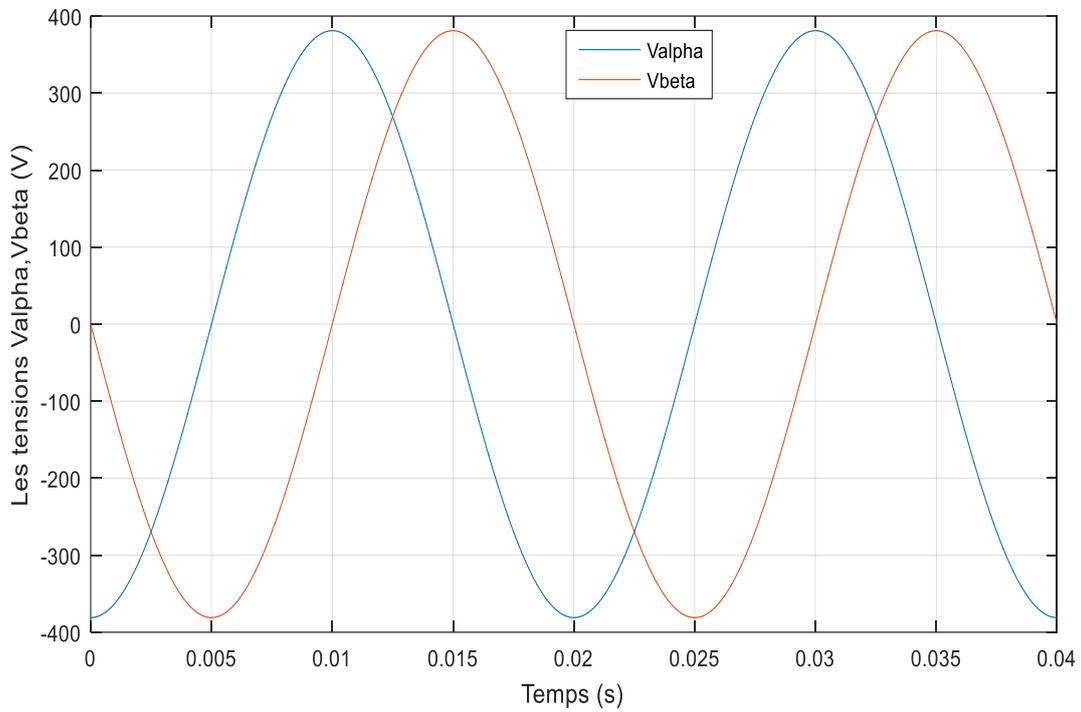
```

Calcul du Valpha,Vbeta/Calcul Valpha+Vbeta  x +
1  function [Valpha,Vbeta] = fcn(Va,Vb,Vc)
2      %calcul du Valpha et Vbeta
3      a=[1 -(1/2) -(1/2)];
4      b=[0 (-sqrt(3)/2) (sqrt(3)/2)];
5      c=[Va;Vb;Vc];
6      Valpha=sqrt(2/3)*a*c;
7      Vbeta=sqrt(2/3)*b*c;
8      end
    
```

Figure III-19 : Programme pour calcul de Valpha et Vbeta..

**III- 6-1-2-1-Résultats de simulation :**

Les figures suivantes représentent les tensions dans le plan ( $\alpha, \beta$ ) en fonction du temps :



*Figure III-20 : Allure de Valpha et Vbeta.*

**III- 6-1-3-Détermination du numéro de secteur et phi :**

Pour mesurer l'angle du vecteur de référence, on utilise l'équation (III.5), et pour la détection du nombre de secteur, on exploite la figure(III.21).

```

Calcul du nombre de secteur et phi /Calcul des Secteurs
1  function [Secteur,phi] = fcn(Valpha,Vbeta)
2  %calcul de theta et phi et secteur
3  phi= atan2(Vbeta,Valpha)+pi;
4  if ((phi >= 0) && (phi < pi/3))
5      theta=phi;
6      Secteur=1;
7  elseif ((phi >= pi/3) && (phi < 2*pi/3))
8      theta=phi-(pi/3);
9      Secteur=2;
10 elseif ((phi >= 2*pi/3) && (phi < pi))
11     theta=phi-(2*pi/3);
12     Secteur=3;
13 elseif ((phi >= pi) && (phi < 4*pi/3))
14     theta=phi-(3*pi/3);
15     Secteur=4;
16 elseif ((phi >= 4*pi/3) && (phi < 5*pi/3))
17     theta=phi-(4*pi/3);
18     Secteur=5;
19 else
20     theta=phi-(5*pi/3);
21     Secteur=6;
22 end
23 end
    
```

Figure III-21 : Code pour calcule de phi et nombre de secteur.

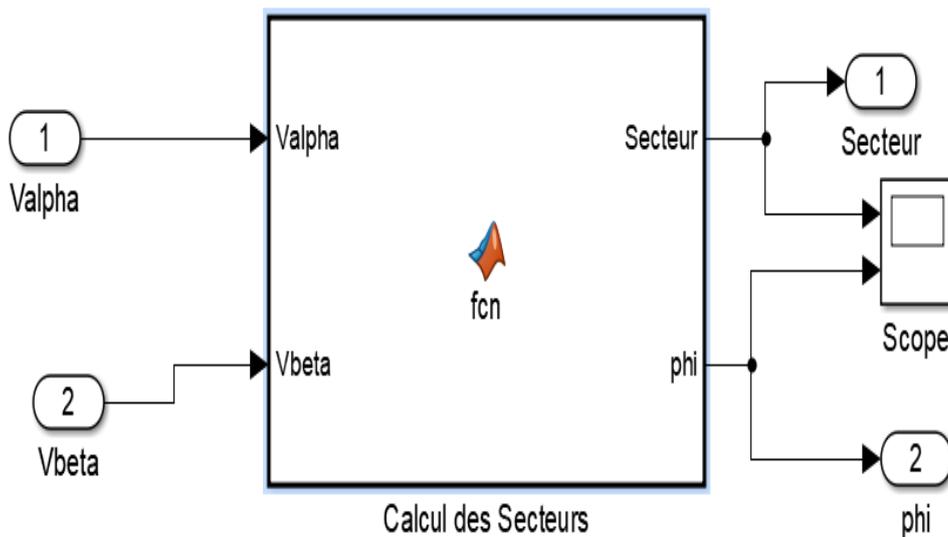


Figure III-22 : Schéma de bloc pour numéro de secteur et phi.

**III- 6-1-3-1-Résultats de simulation :**

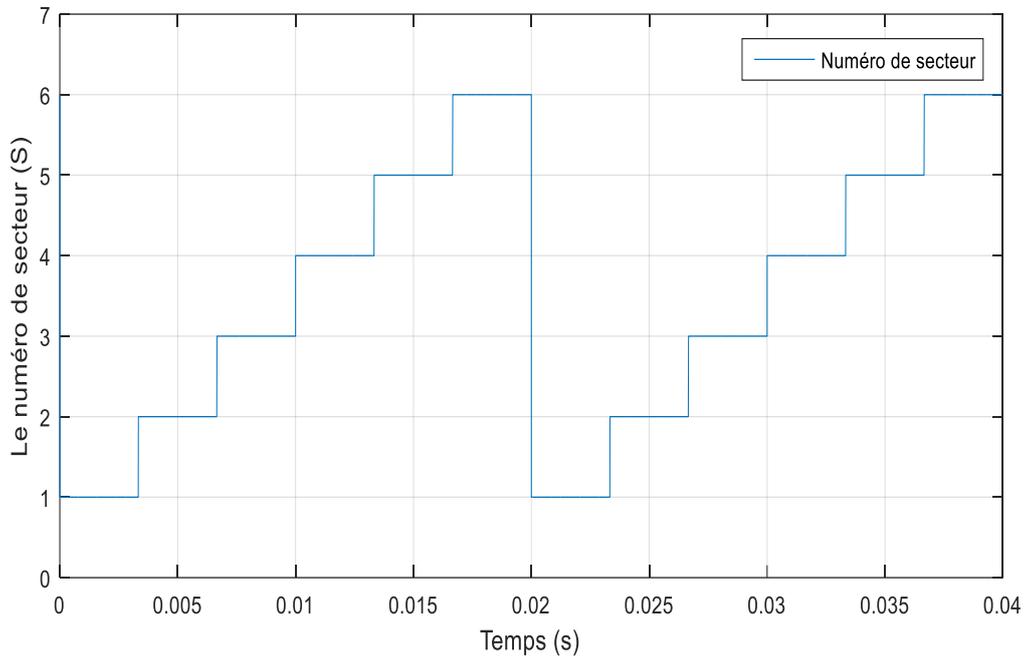


Figure III-23 : Le déplacement du vecteur de référence entre les secteurs en fonction du temps.

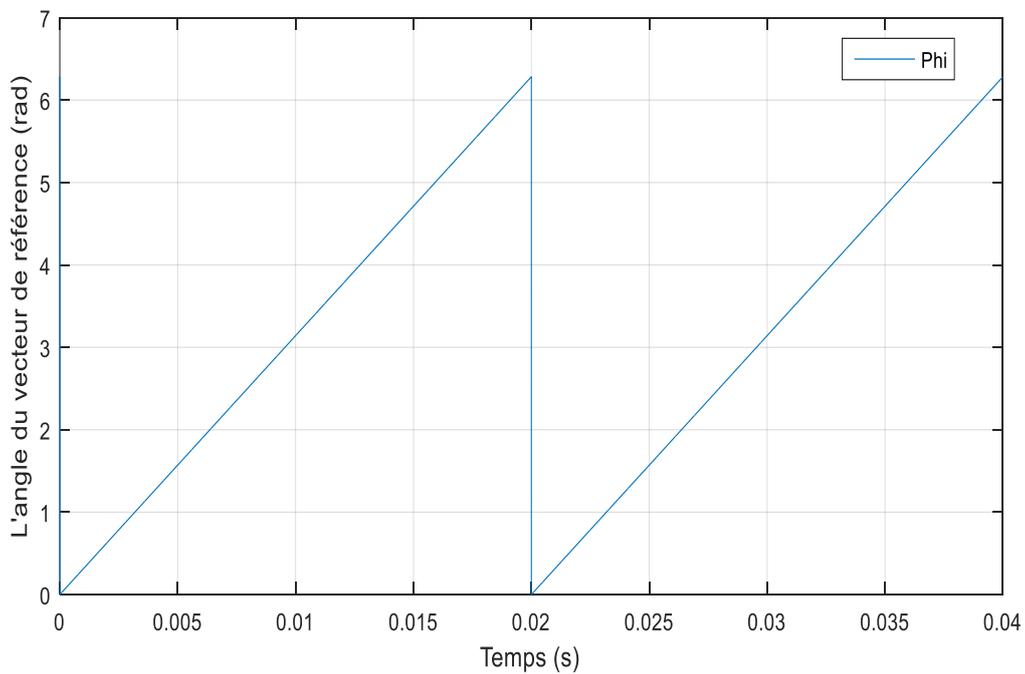
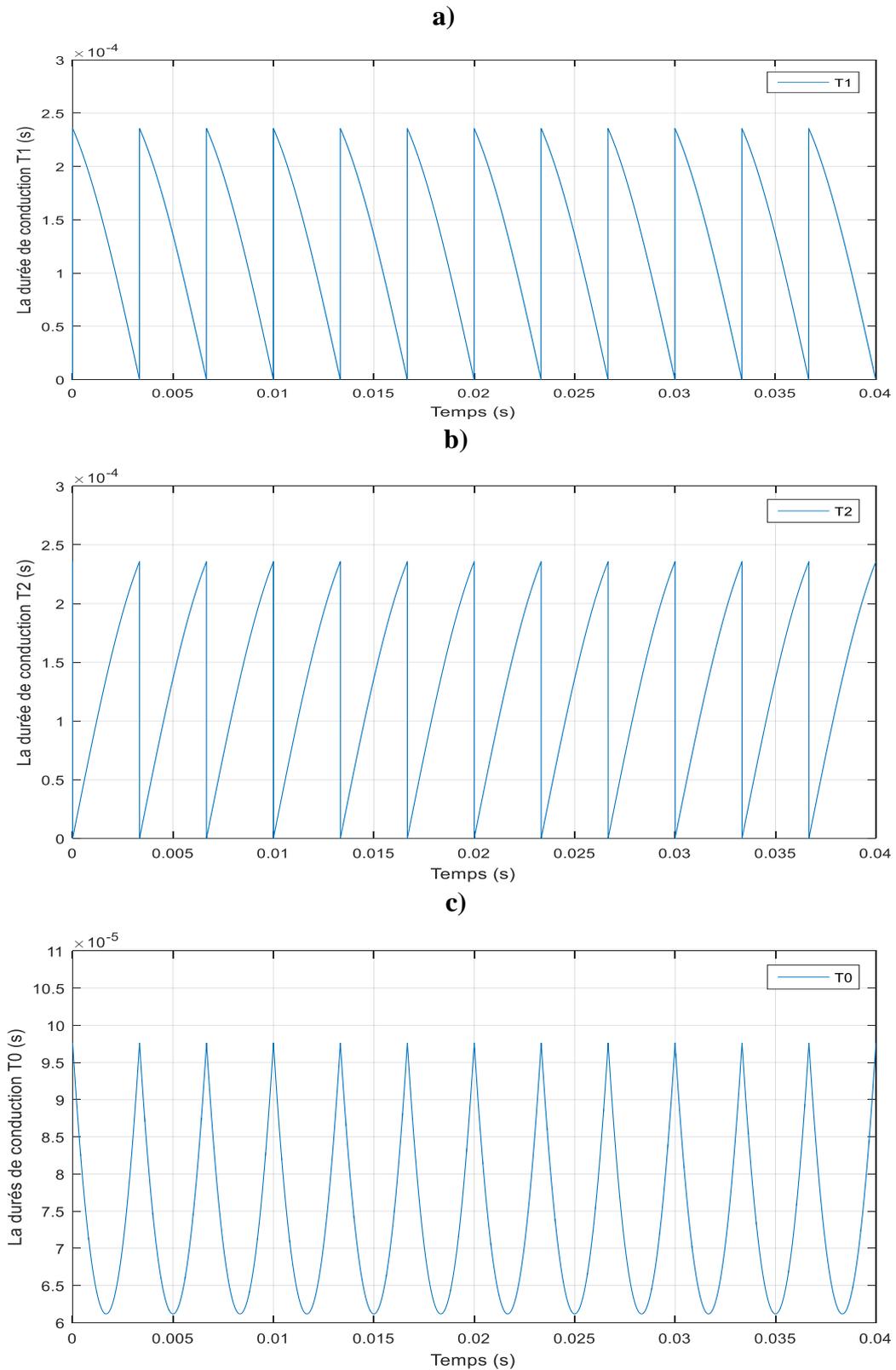


Figure III-24 : L'angle du vecteur de référence.

**III- 6-1-4-Détermination des durées de conduction (T1, T2, T0) :**

Pour mesurer ces durées, on va utiliser les équations (III.7), (III.8), (III.9) :

**III- 6-1-4-1-Résultats de simulation :**



**Figure III-25** : Les allures des durés de conduction (**a** : T1, **b** : T2, **c** : T0).

**III- 6-1-5-Détermination des signaux de commande des interrupteurs :**

Selon la figure (III-26), les signaux de commande des interrupteurs dépendent principalement des durées de conduction ( $T_1$ ,  $T_2$ ,  $T_0$ ).

```

1  function [S1,S3,S5] = fcn(Secteur,t,T0,T1,T2)
2  %Calcul les signaux de commande
3  if(Secteur==1)
4      if (t<(T0/4))
5          S1=0;S3=0;S5=0;%V0(000)
6          elseif ((t>=(T0/4)) && (t<((T0/4)+T1/2)))
7              S1=1;S3=0;S5=0;%V1(100)
8              elseif((t>=((T0/4)+T1/2)) && (t<((T0/4)+T1/2+T2/2)))
9                  S1=1;S3=1;S5=0;%V2(110)
10             elseif ((t>=((T0/4)+T1/2+T2/2)) && (t<(3*T0/4)+T1/2+T2/2))
11                 S1=1;S3=1;S5=1;%V7(111)
12             elseif ((t>=(3*T0/4)+T1/2+T2/2) && (t<(3*T0/4)+T1/2+T2))
13                 S1=1;S3=1;S5=0;%V2(110)
14             elseif((t>=(3*T0/4)+T1/2+T2) && (t<(3*T0/4)+T1+T2))
15                 S1=1;S3=0;S5=0;%V1(100)
16             else
17                 S1=0;S3=0;S5=0;%V0(000)
18             end
    
```

*Figure III-26 : Code pour détecter le secteur1.*

**III- 6-1-5-1-Résultats de simulation :**

A titre d'exemple, on prend le code d'un seul secteur pour expliquer le séquençage des vecteurs et leurs configurations car les autres secteurs ont une même configuration mais de vecteur d'états différents :

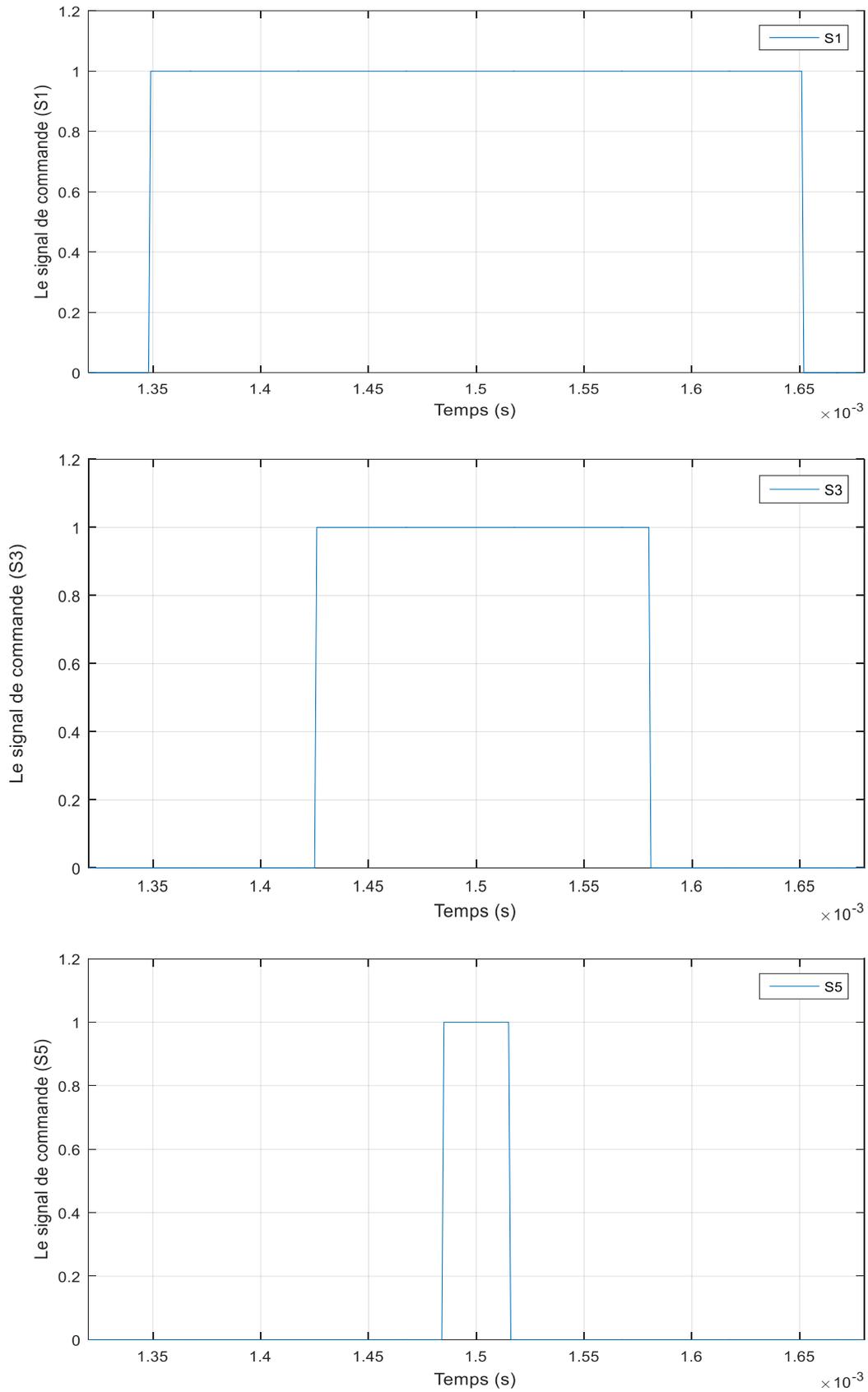


Figure III-27 : Les signaux de commande des interrupteurs (S1, S3, S5) de secteur 1.

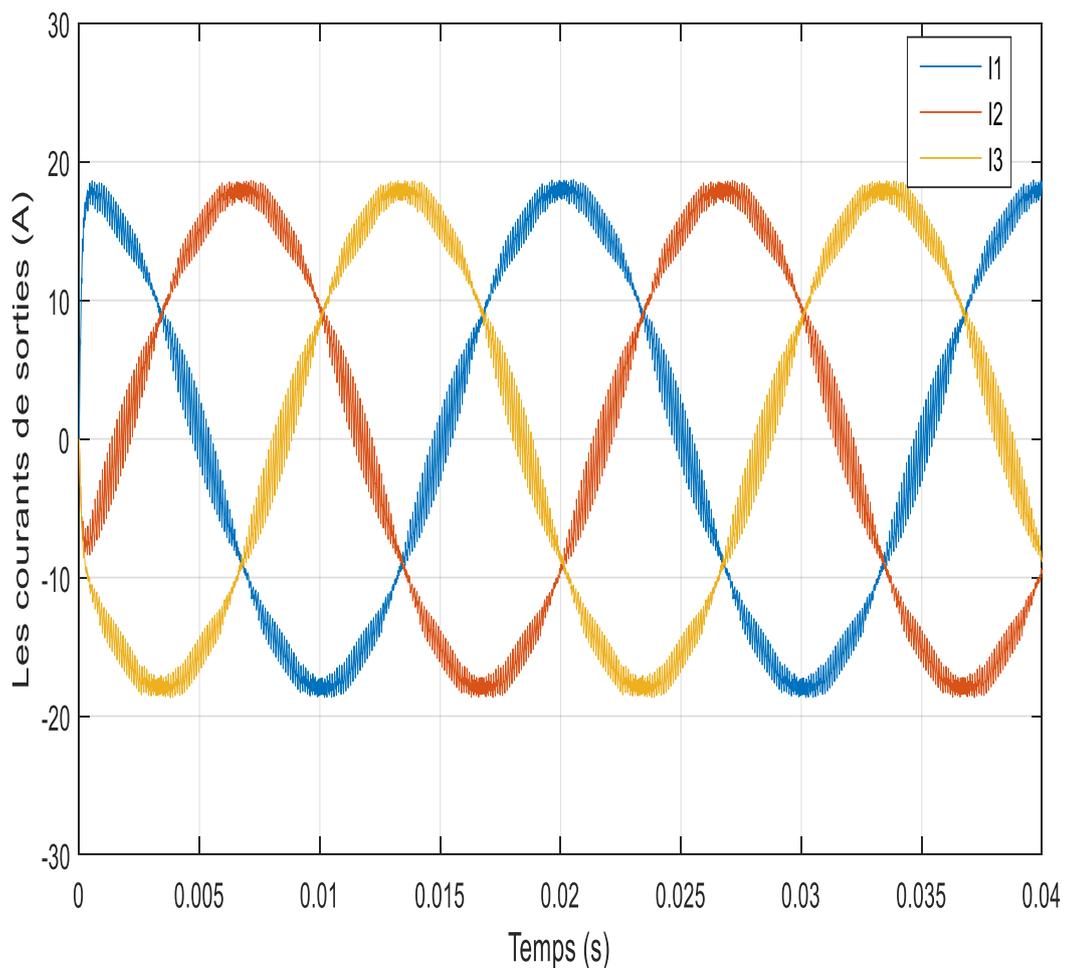
**N.B:** les signaux de commande (S1, S3, S5) pour d'autres secteurs ainsi leurs code sous Matlab Simulink seront dans l'annexe A.

### **III- 6-1-6-Visualisation des tensions simples et composées avec courants de sorties :**

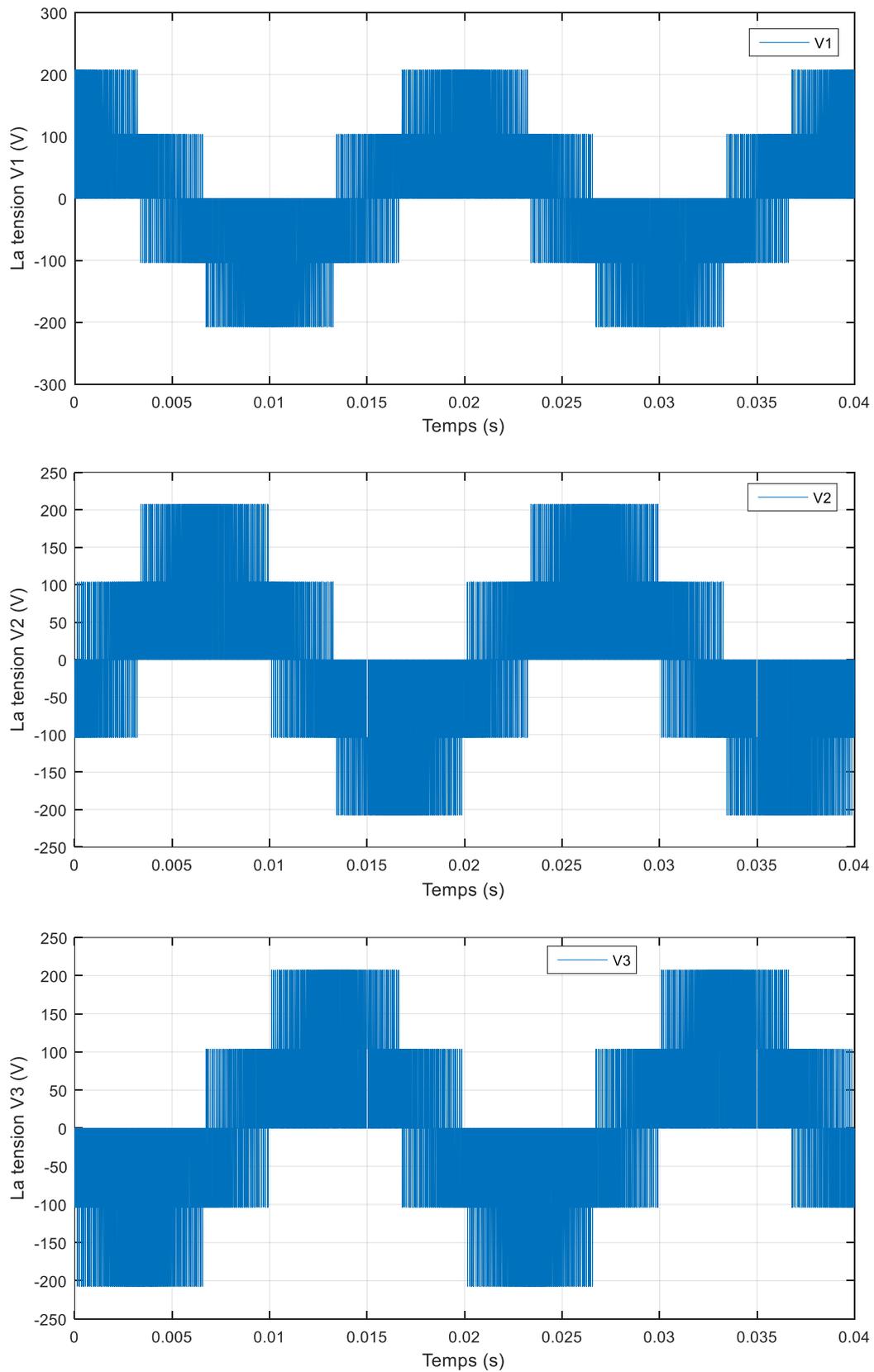
Pour visualisation les tensions simples et composées en plus les courants à la sortie de l'onduleur, on a utilisé des voltmètres et un bloc de mesure triphasé.

#### **III- 6-1-6-1-Résultats de simulation :**

Les figures suivantes représentent les courants ainsi les tensions composées et simples de sortie de l'onduleur.



**Figure III-28 :** Les courants ( $I_1$ ,  $I_2$  et  $I_3$ ) avec charge RL.



**Figure III- 29** : Allures des tensions simples avec charge RL.

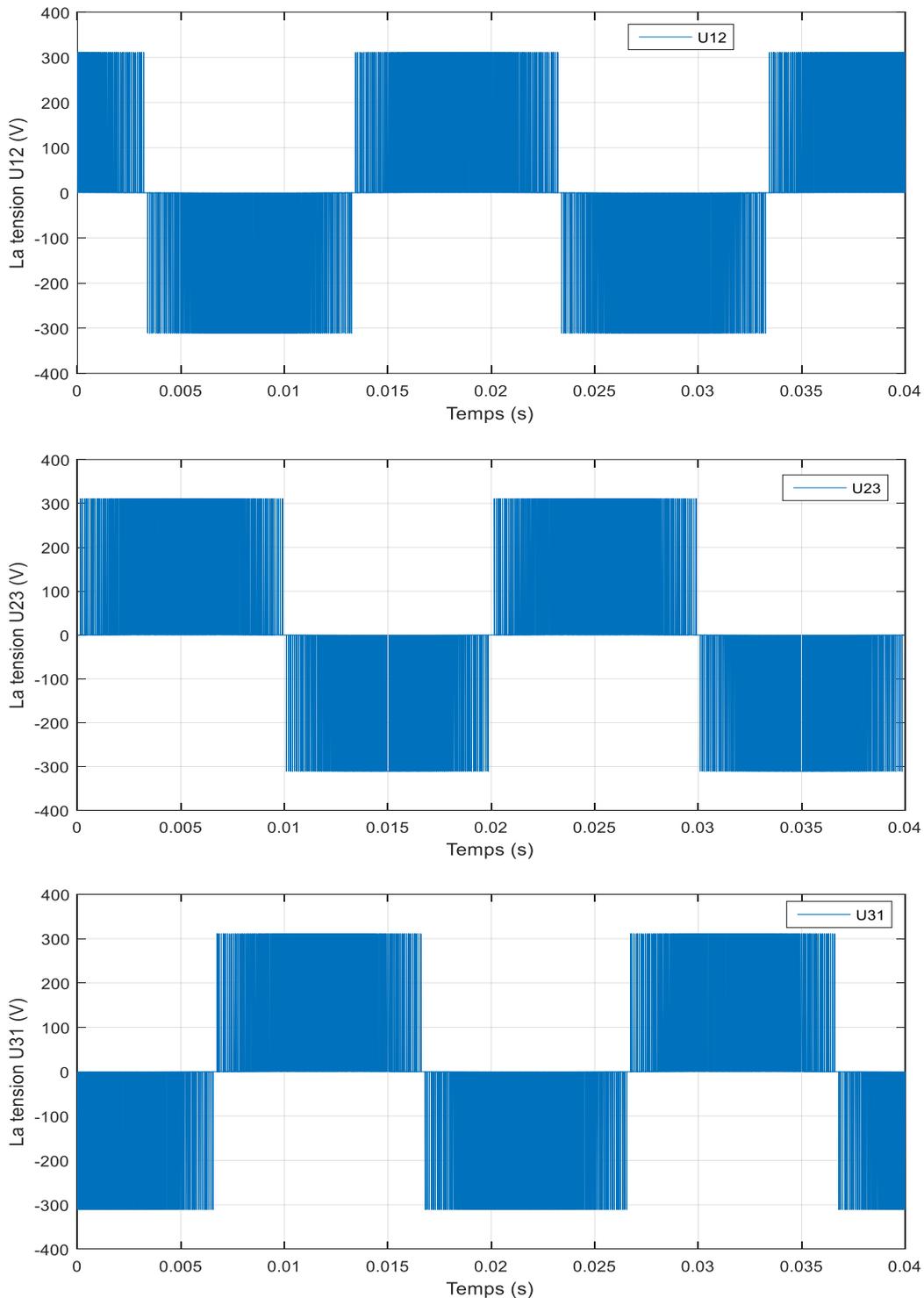


Figure III- 30 : Allure des tensions composées avec charge RL.

**III- 6-1-7-Interprétations des résultats :**

A partir des résultats obtenus on constate que :

- La tension simple possède trois niveaux (0,  $E/3$ ,  $2E/3$ ) et la tension composée possède deux niveaux de tension (+E et -E, avec :  $E=311V$ ).

- Les courants de sortie sont déphasés de  $120^\circ$  entre eux.
- La tension  $V_{\alpha}$  est décalée de  $90^\circ$  par rapport à la tension  $V_{\beta}$ .
- Nos signaux dédiés à la simulation correspondent aux critères de la commande SVM, et décrivent conformément le principe de cette commande.

### III- 6-2-Débit sur charge MAS :

En deuxième lieu, ça concerne la MAS associée à l'onduleur comme une charge, qui ensuite va jouer le rôle d'une génératrice des tensions de référence. et cela nous a conduit à l'emploi de la régulation de vitesse ( $\Omega$  [rad/s]) et de la commande scalaire pour avoir les tensions de références désirées.

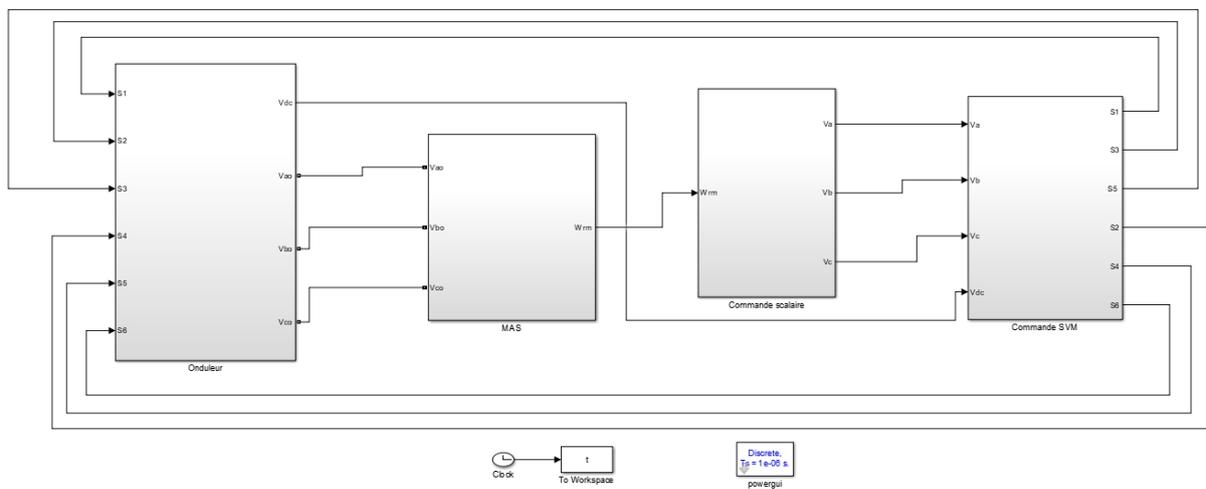


Figure III- 31 : Schéma global de simulation de la commande SVM avec une charge (MAS).

Tableau III-2 : Les paramètres de la MAS.

Les paramètres	La valeur
La puissance nominale	$P_n = 4.36 \text{ KW}$
Fréquence nominale	$F = 50 \text{ Hz}$
Nombre de paire de pole	$P = 2$
Tension d'alimentation	$U_n = 230\text{v}/400\text{v}$
Résistance statorique	$R_s = 1.273 \text{ ohm}$
Résistance rotorique	$R_r = 0.6358 \text{ ohm}$
Inductance statorique	$L_s = 0.0096 \text{ H}$
Inductance rotorique	$L_r = 0.0096 \text{ H}$
Inductance mutuelle	$M_{sr} = 0.255 \text{ H}$
Moment d'inertie	$J = 0.037 \text{ Kg.m}^2$
Coefficient de frottement	$f = 0.005 \text{ N.m.s}$

**III- 6-2-1-La commande scalaire de la charge (machine asynchrone):**

La commande des machines asynchrone se fait par plusieurs méthodes ,la plus simple à réaliser c'est la commande scalaire, la commande scalaire de la MAS c'est une stratégie basée sur l'ajustement sur les grandeurs principal qui contrôle la machine comme la tension imposé au circuit statorique et sur la fréquence de ses tensions ,sauf le courant car il est imposé par la charge entraîné par le moteur .la stratégie scalaire la plus connu c'est d'ajuster la fréquence  $f_s$  et la tension statorique  $V_s$  de la MAS pour Contrôler la vitesse de rotation de moteur à condition de maintenir le rapport  $(V_s/f_s)$  constant et ce rapport représente le flux magnétique de la machine  $\phi=k'*v_s /f_s$  (ou  $k'$  est un constant),et pour cela nous avons obligé de faire une boucle de régulation en utilisant le régulateur PI(proportionnel intégral)pour garder ce rapport constant donc le flux constant [8].

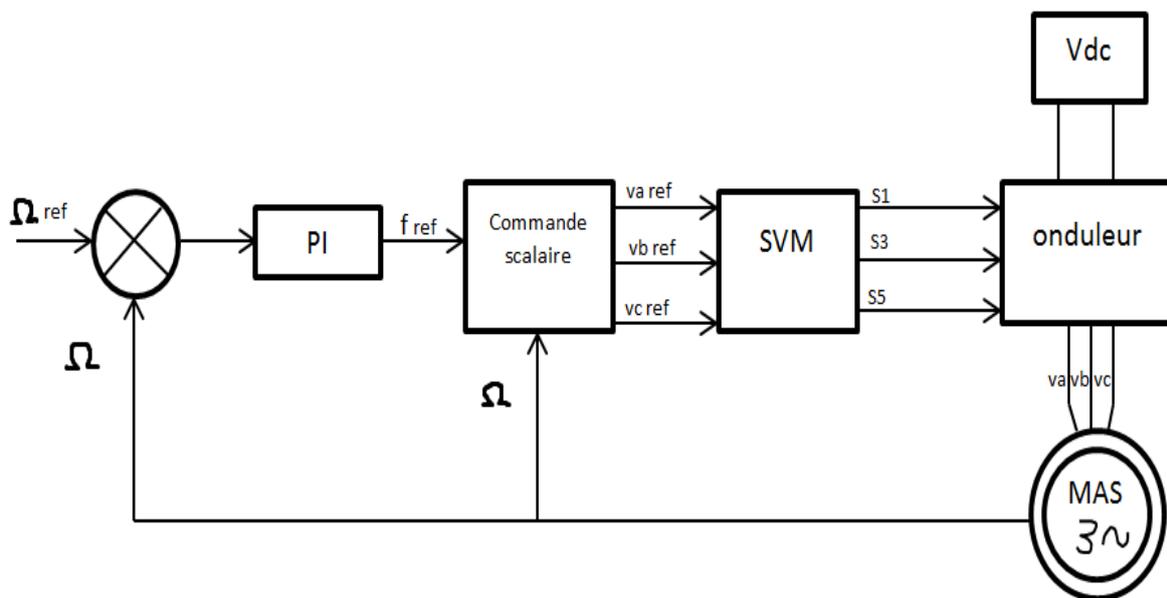


Figure III-32 : Schéma de principe de la commande scalaire de la MAS.

**III- 6-2-2-La simulation de la commande scalaire sous Matlab Simulink :**

Après la mesure de la vitesse de rotation de la MAS, on souhaite que la MAS tourne à une vitesse de référence égale à 149 rad/s entre [0 1s], et à 109 rad/s entre [1 5s].Pour les paramètres du régulateur PI sont les suivants :  $K_i=0,8$  ;  $K_p=9$ .

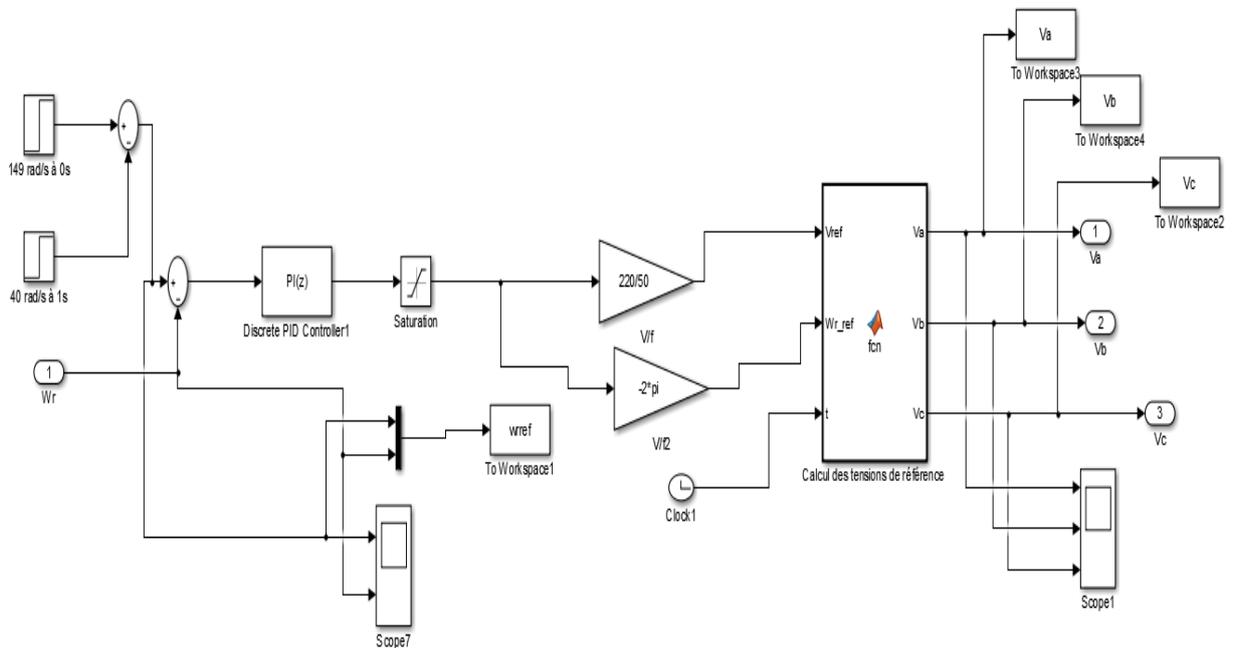


Figure III- 33 : Schéma du bloc de la commande scalaire sous Matlab Simulink.

```

Commande scalaire/Calcul des tensions de référence
1  function [Va ,Vb ,Vc ] = fcn(Vref,Wr_ref,t)
2  -   Va=Vref*sin(Wr_ref*t);
3  -   Vb=Vref*sin((Wr_ref*t)-(2*pi/3));
4  -   Vc=Vref*sin((Wr_ref*t)-(4*pi/3));
5  -   end
    
```

Figure III- 34 : Codes pour calcul des tensions de référence.

### III- 6-2-2-1-Résultats de simulation :

Les figures suivantes représentent les résultats de la régulation de vitesse du MAS ainsi les tensions composées et simples avec le courant statorique Ia.

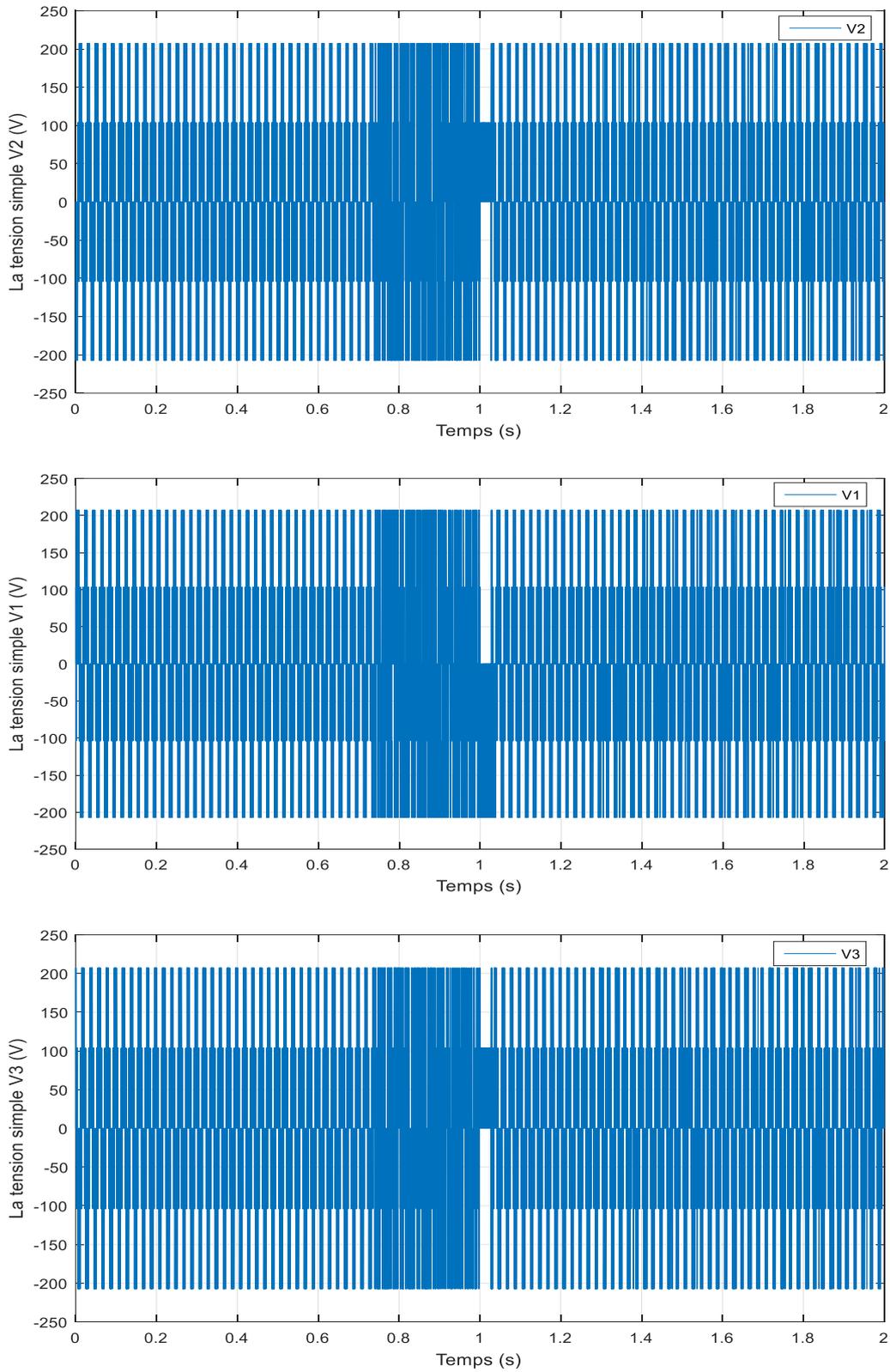


Figure III- 35 : Allure des tensions simples avec charge MAS

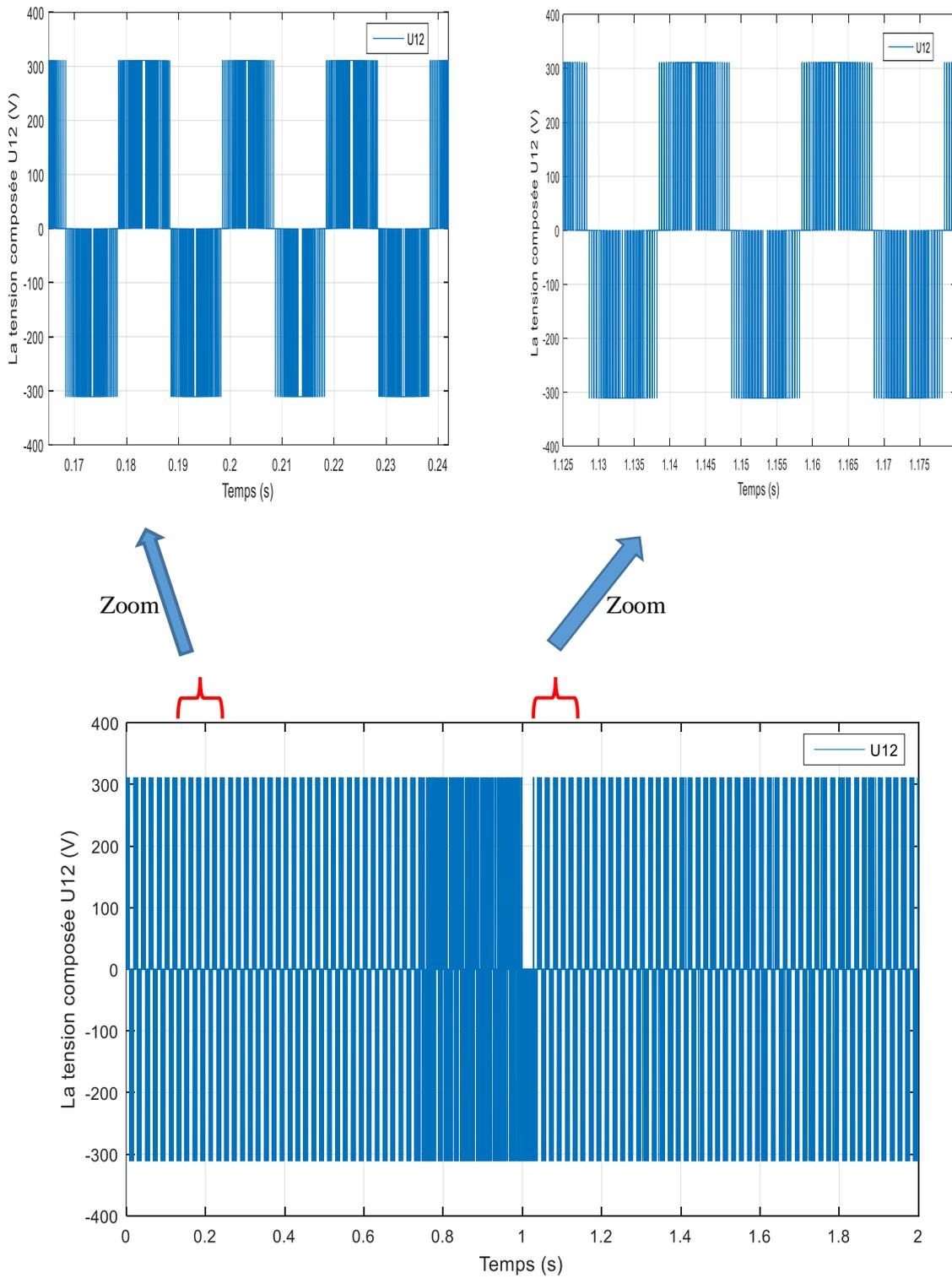


Figure III-36 : Allure de la tension composée  $U_{12}$  avec charge MAS.

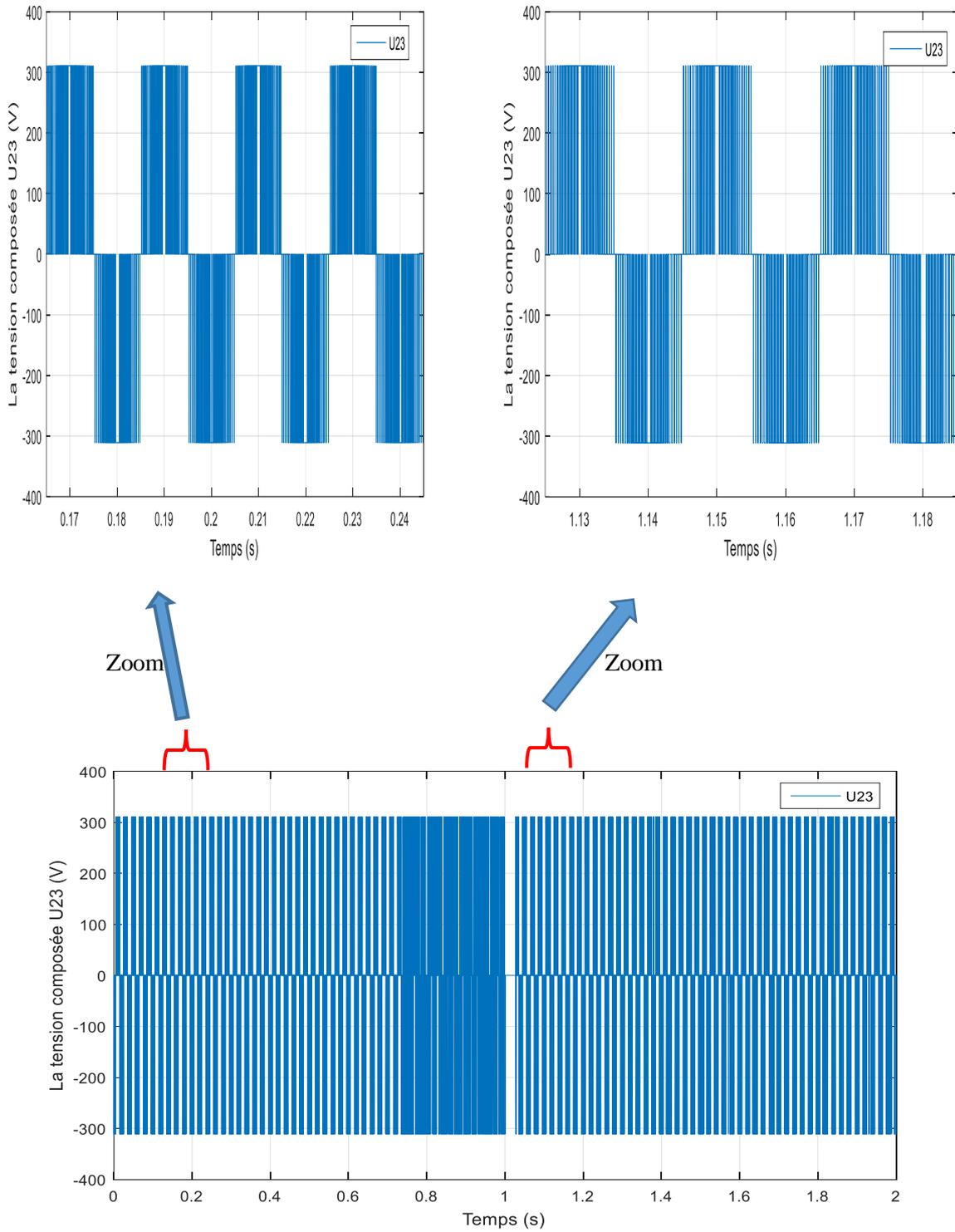


Figure III- 37 : Allure da la tension composée  $U_{23}$  avec charge MAS.

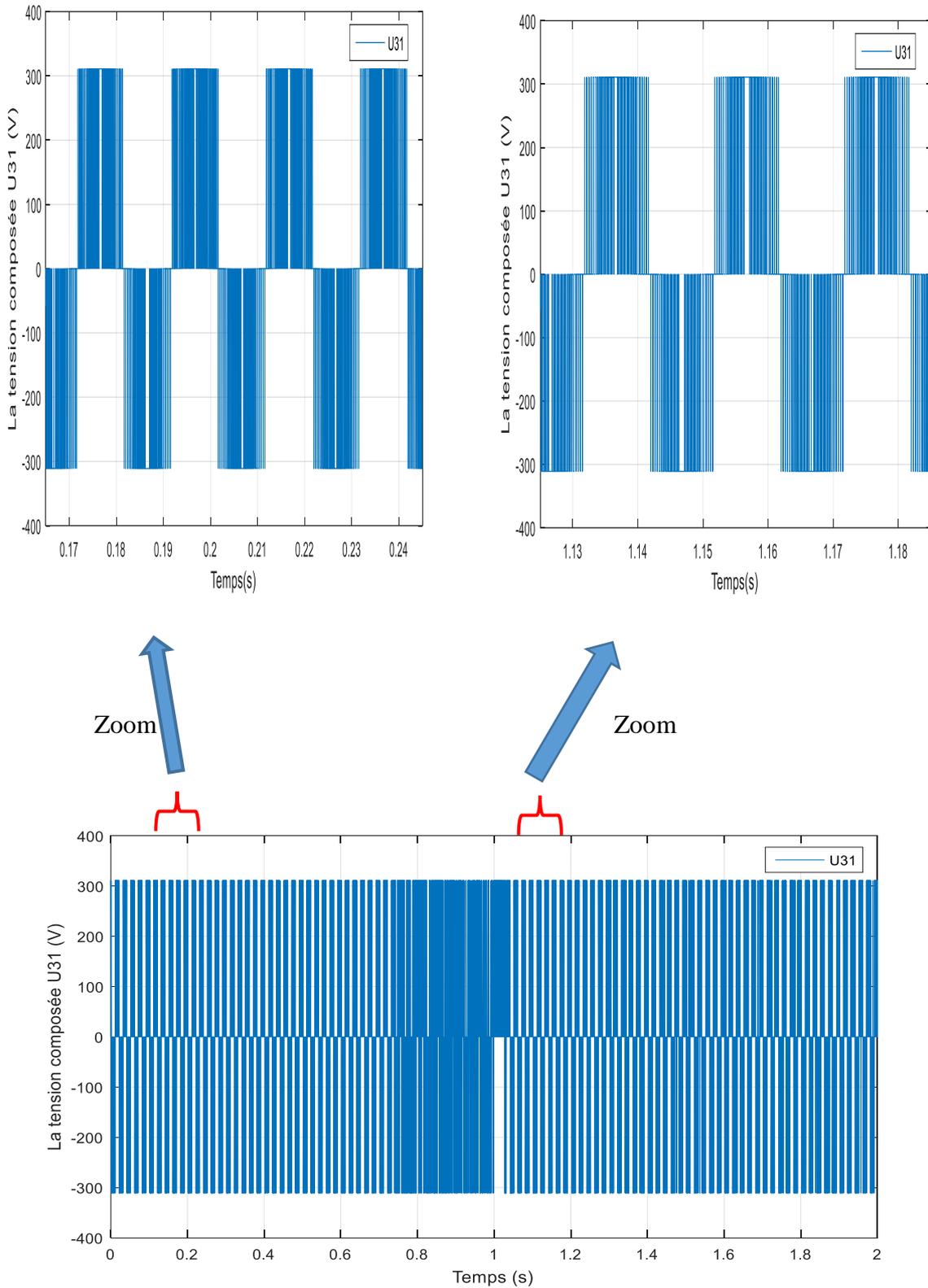


Figure III- 38 : Allure da la tension composée  $U_{31}$  avec charge MAS.

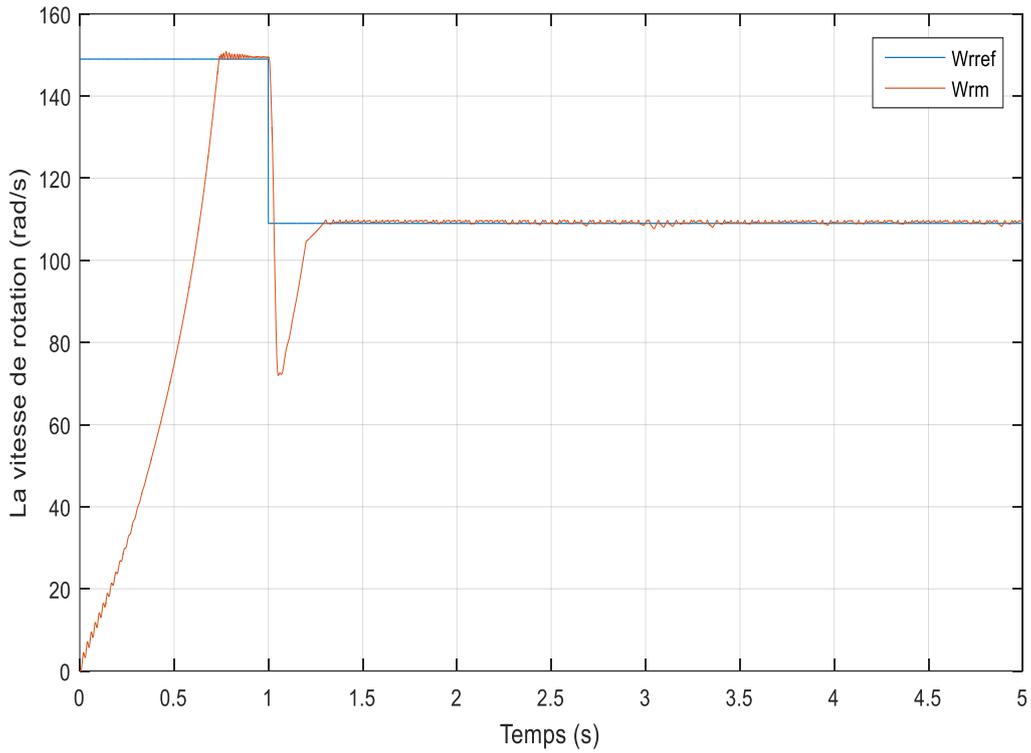


Figure III- 39 : Allure de la vitesse de rotation de la MAS.

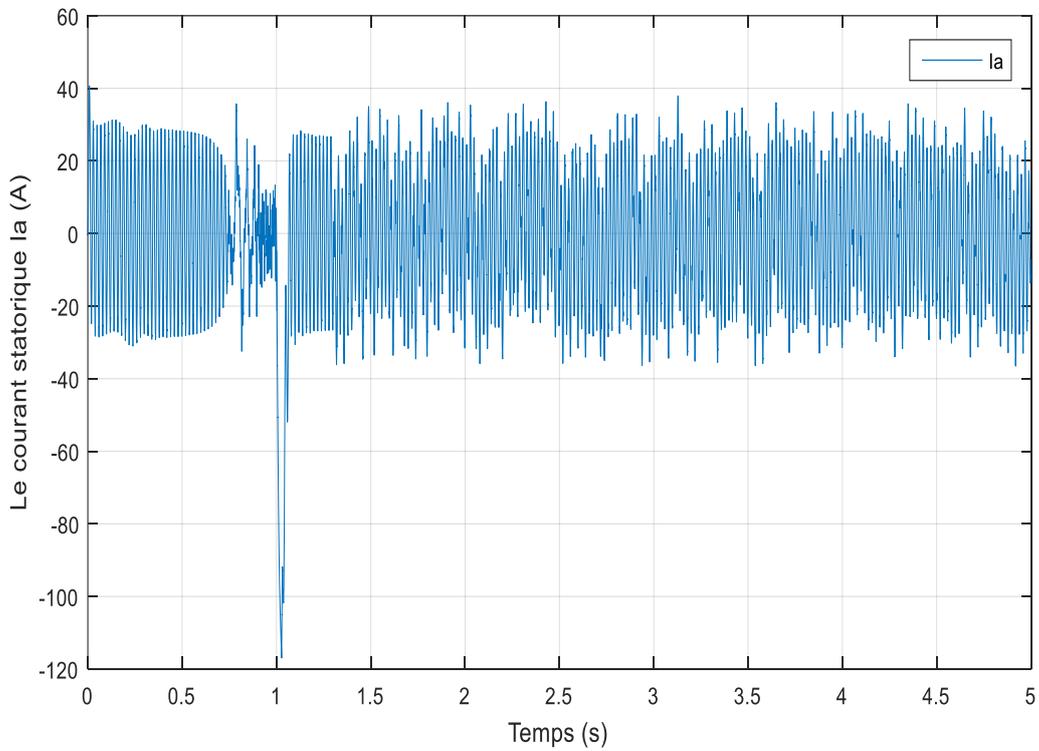


Figure III- 40 : Allure du courant statorique Ia.

### **III- 6-2-3-Interprétation des résultats:**

A partir des résultats obtenus on constate que :

- La tension composée possède deux niveaux de tension (+E et -E, avec :  $E=311V$ ).
- La vitesse de rotation de moteur suit la référence imposé malgré le couple résistant ( $C_r=15$  N.m) appliques au moteur (à  $t=1.2s$ ), grâce au régulateur PI, mais présente des perturbations qui influent sur le système.

### **III- 7-Conclusion :**

Ce chapitre a fourni une description complète de la commande SVM d'un onduleur. La commande SVM offre une excellente qualité de sortie avec une faible distorsion harmonique et une meilleure efficacité. Les résultats de simulation ont démontré l'efficacité de la commande SVM dans la génération de signaux de sortie de qualité pour différentes charges.

# **Chapitre IV:**

# **Implémentation sur Arduino**

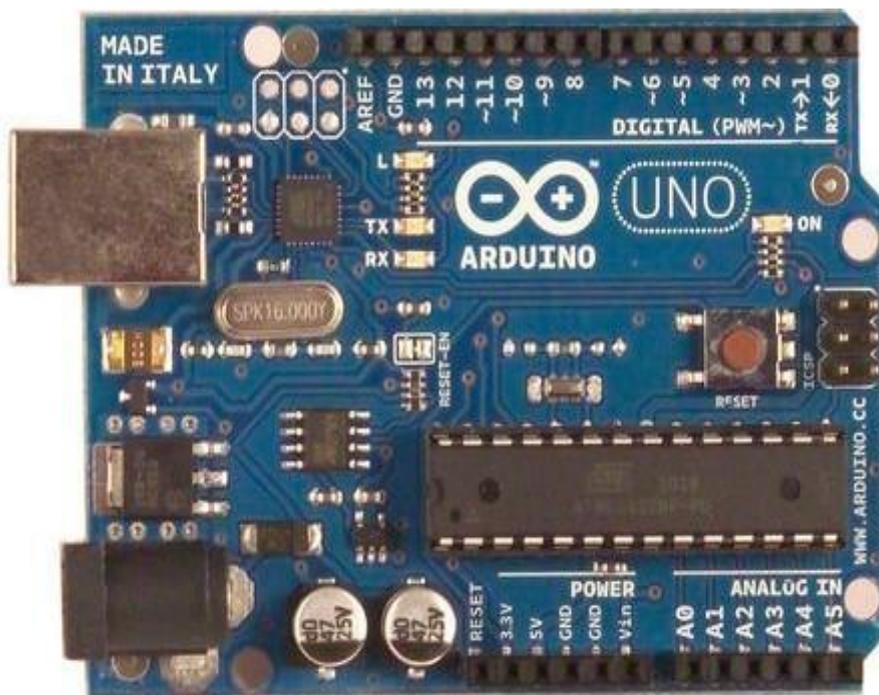
### IV-1- Introduction :

ARDUINO est une carte de circuit imprimé matérielle libre avec un microcontrôleur dessus qui peut être programmé pour analyser et générer des signaux électriques pour programmer des systèmes électroniques.

Dans ce chapitre, on présentera de manière brève des généralités sur la carte Arduino et le dispositif expérimental de notre système étudié, ensuite en implémente notre programme de la commande SVM dans la carte Arduino pour commander notre onduleur triphasé et générer des tensions sinusoïdal à la sortie.

### IV-2- La carte ARDUINO Uno :

La carte Arduino Uno est basée sur un ATmega328 fonctionnant à 16 MHz. C'est la carte microcontrôleur la plus récente et la plus abordable pour Arduino. Des connecteurs sur le bord extérieur de la carte permettent la connexion de divers modules complémentaires.



*Figure IV- 1 : Carte Arduino Uno.*

Elle comprend :

- 14 broches d'E/S numériques, dont 6 peuvent être utilisées comme sorties PWM (MLI).
- 6 entrées analogiques qui peuvent même être utilisées broches d'entrée/sortie numériques.
- un quartz 16Mhz.
- une connexion USB.

- une Prise de connexion d'alimentation.
- un connecteur ICSP (programmation en circuit).
- un bouton de remise à zéro (reset) [15].

### IV -2-1- Caractéristiques de la carte Arduino Uno:

- ✓ Microcontrôleur: ATmega328.
- ✓ Tension d'alimentation interne = 5V.
- ✓ 13 entrées /sorties numériques numérotées de 0 à 13 (digital pins).
- ✓ Tension d'alimentation (recommandée) = 7 à 12V, limites = 6 à 20 V.
- ✓ Entrées/sorties numériques : 14 dont 6 sorties PWM.
- ✓ Entrées analogiques = 6.
- ✓ Courant max par broches E/S = 40 mA.
- ✓ Courant max sur sortie 3,3V = 50mA.
- ✓ 32 Ko de mémoire flash, dont 0,5 Ko est utilisé par le boot loader (chargeur d'amorçage).
- ✓ Mémoire SRAM 2 KB.
- ✓ Mémoire EEPROM 1 KB.
- ✓ Fréquence horloge = 16 MHz.
- ✓ Dimensions = 68.6mm x 53.3mm [16].

### IV -3- Implémentation de la commande SVM sur Proteus :

Au niveau de Proteus, on a utilisé :

- une carte Arduino Uno pour l'exécution de notre programme Arduino (langage c),
- un visuel pour afficher les valeurs mesurées par la carte dont le **secteur** et **theta** et **phi** ...etc.
- un potentiomètre pour varier la fréquence de référence (**Fref**).
- Un oscilloscope pour visualiser les signaux de commandes des interrupteurs (**S1**, **S3**, **S5**).

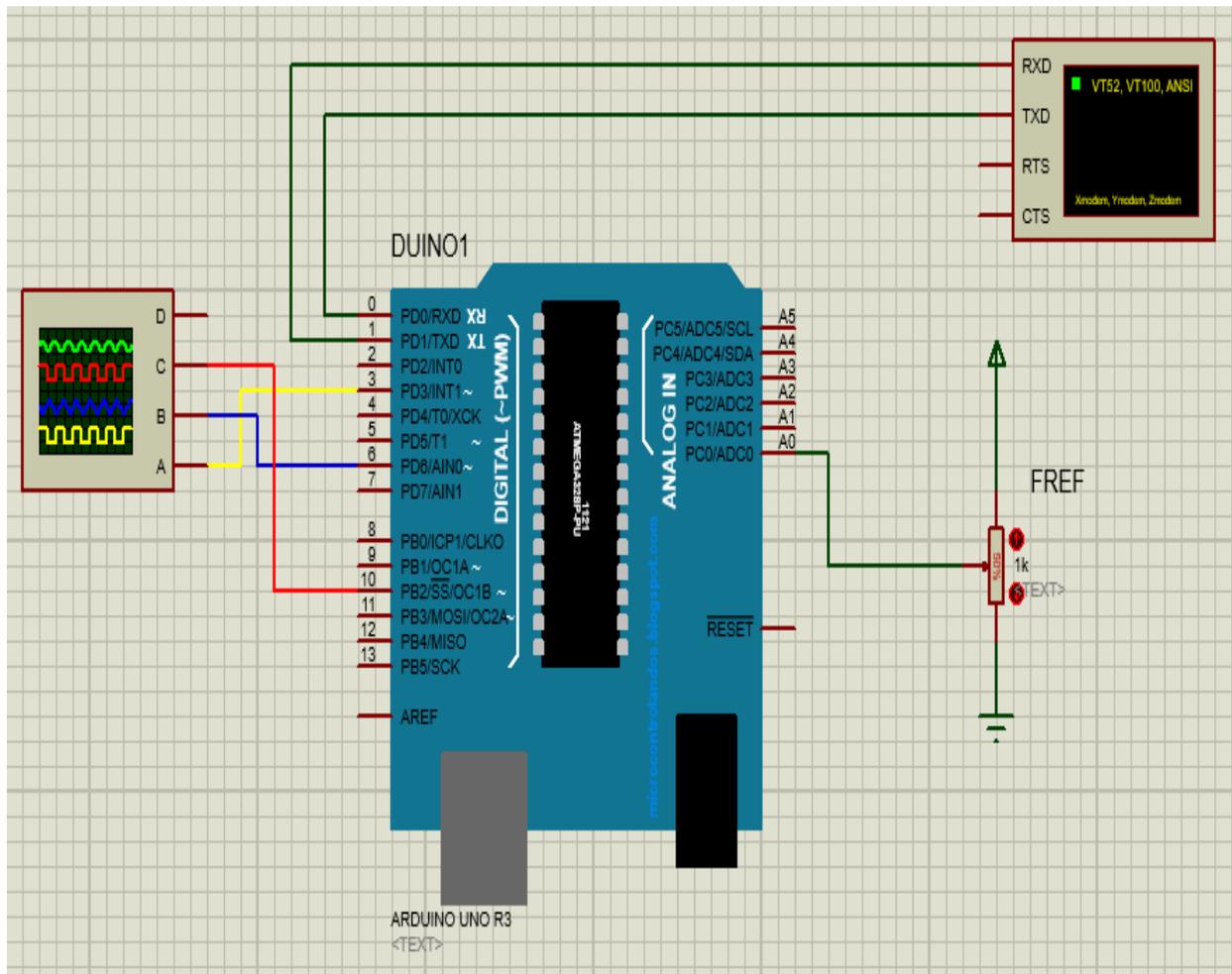


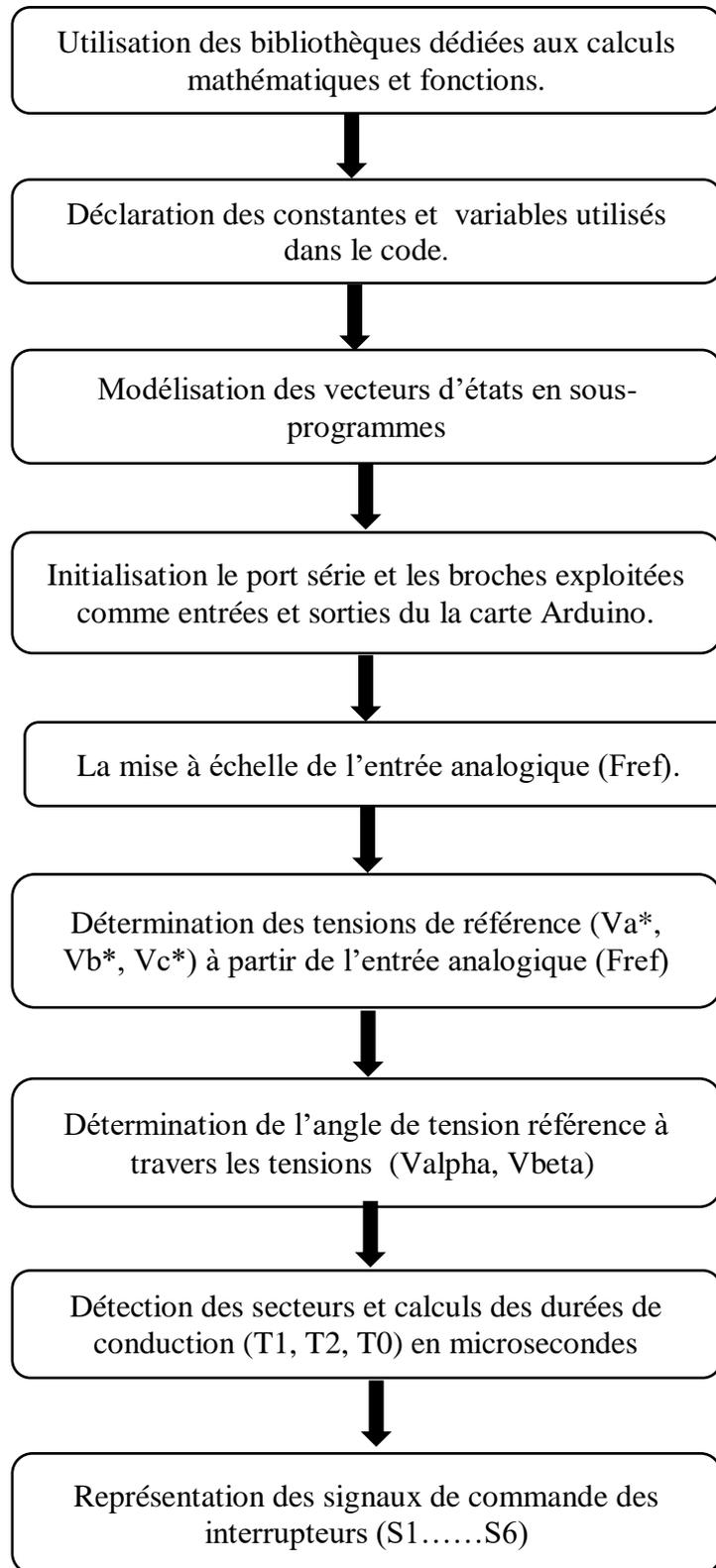
Figure IV- 2 : Schéma global de simulation au niveau de Proteus.

### IV -3-1-Implémentation de la commande scalaire sous Proteus :

La commande scalaire a été simplifiée en un potentiomètre dont son rôle est de varier la fréquence de référence. Dans notre cas, on a imposé une fréquence équivalente d'ordre (2.70V sous Arduino).

**IV-4-Le séquençement du programme Arduino :**

Pour expliquer le séquençement de notre programme dédié à la commande SVM, voici un organigramme explicatif qui résume les étapes de la réalisation de ce code :



*Figure IV-3: Organigramme décrivant les démarches suivies pour la construction du code Arduino.*

**IV -4-1-Exemples des démarches suivies :**

- **Initialisation du port série et les broches utilisées dans le code :**

```
void setup()
{
  Serial.begin(115200);
  pinMode(3, OUTPUT); //S1
  pinMode(5, OUTPUT); //S3
  pinMode(6, OUTPUT); //S5
  pinMode(A0, INPUT); //Fref
}
```

- **La mise à l'échelle de l'entrée analogique (A0) :**

```
Fref = analogRead(A0);
Frefn = (Fref * AREF / 1024);
```

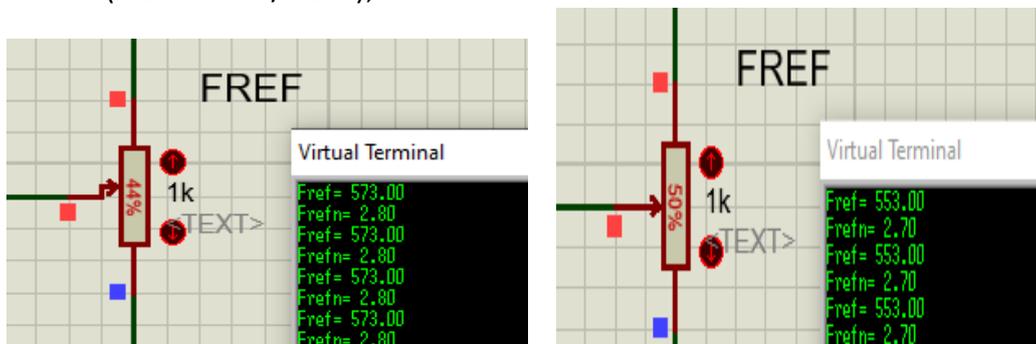


Figure IV- 4 : la mise à échelle de la fréquence de référence.

**IV -5-Résultats de simulation :**

**IV -5-1- Détermination de Va\*,Vb \*et Vc\* en fonction de Fref :**

- **Programme :**

```
Fref = analogRead(A0);
Frefn = (Fref * AREF / 1024);
float amplitude = Frefn * (Vs / f);
float t = millis() / 1000.0;
float w = 2 * PI * Frefn;
float Va = amplitude * sin(w * t);
float Vb = amplitude * sin(w * t - (2 * PI / 3));
float Vc = amplitude * sin(w * t - (4 * PI / 3));
```

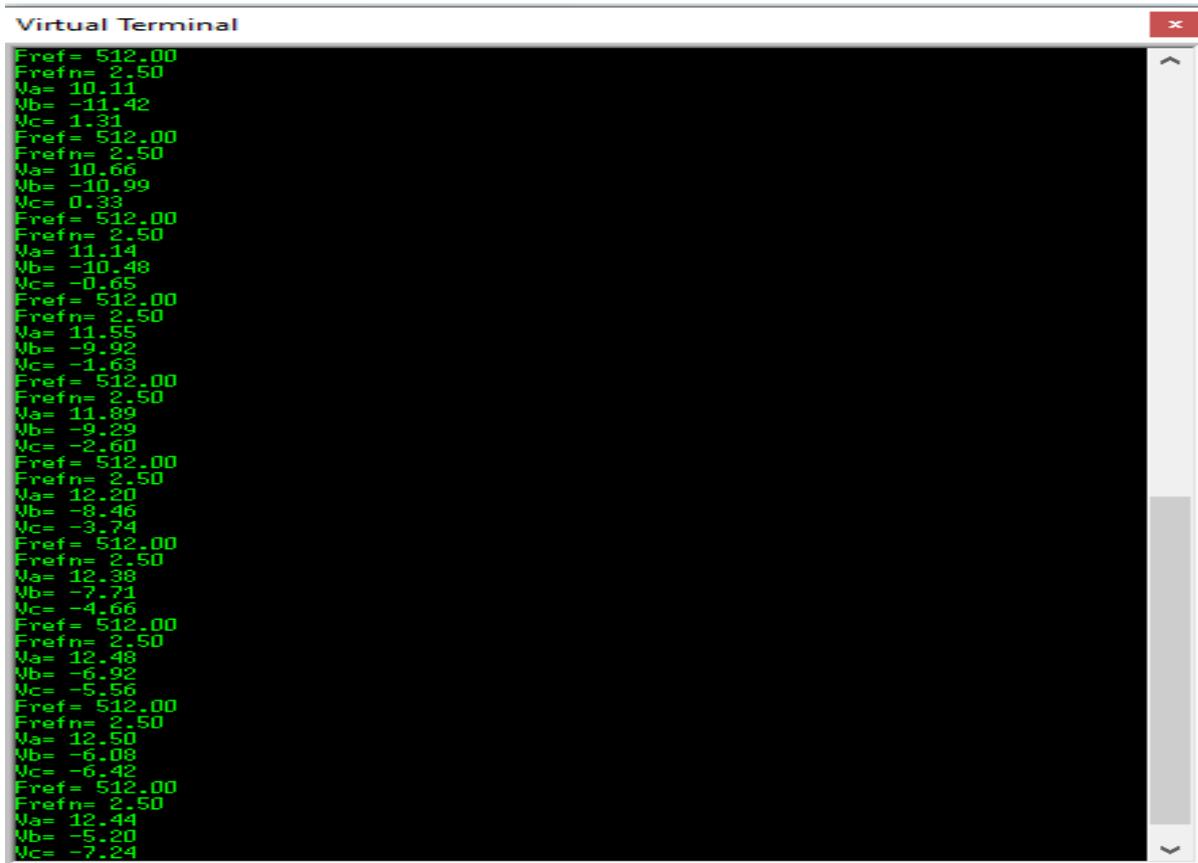


Figure IV- 5: Valeurs de Va, Vb et Vc en fonction de Fref et Frefn.

#### IV -5-2-Détermination de Phi en fonction de Valpha et Vbeta :

➤ **Programme :**

```

float a[3] = {1, -1.0 / 2, -1.0 / 2};
float b[3] = {0, sqrt(3.0) / 2, -sqrt(3.0) / 2};
Valpha = sqrt(2.0 / 3.0) * (a[0] * Va + a[1] * Vb + a[2] * Vc);
Vbeta = sqrt(2.0 / 3.0) * (b[0] * Va + b[1] * Vb + b[2] * Vc);
phi = atan2(Vbeta, Valpha) + PI;
    
```

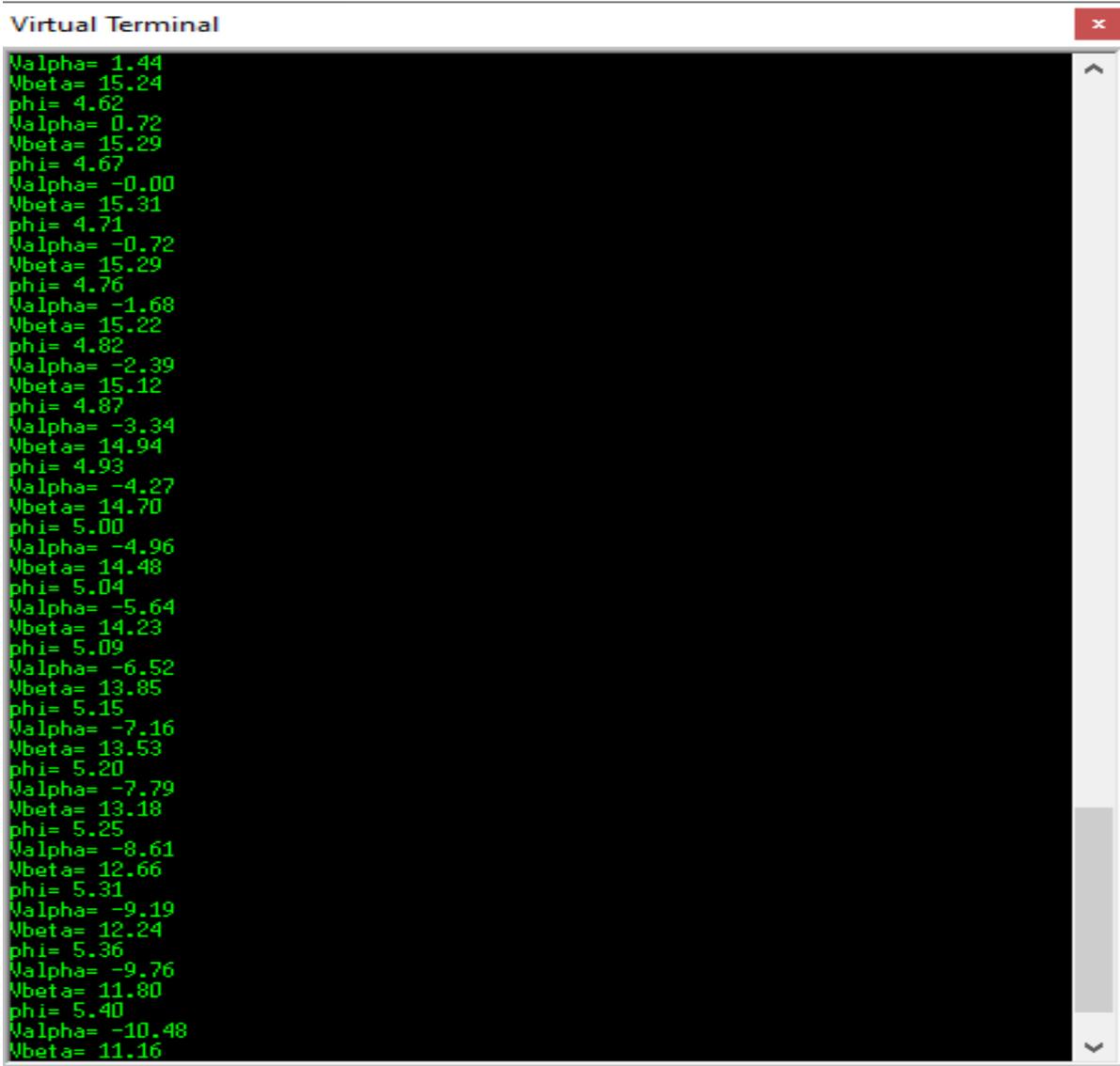


Figure IV- 6 : Valeurs de phi en fonction de Valpha et Vbeta.

#### IV -6-Visualisation des signaux de commande des interrupteurs :

- Exemple du calcul des signaux de commande des interrupteurs pour le premier secteur :

```

if (Secteur == 1)
{ cas0(); // V0(000)
  delayMicroseconds(T0 / 4);
  cas1(); // V1(100)
  delayMicroseconds(T1 / 2);
  cas2(); // V2(110)
  delayMicroseconds(T2 / 2);
  cas7(); // V7(111)
  delayMicroseconds((T0 / 2) + 22);
  cas2(); // V2(110)
  delayMicroseconds((T2 / 2) + 25);
  cas1(); // V1(100)
    
```

```

delayMicroseconds((T1 / 2) + 25);
cas0(); // VO(000)
delayMicroseconds(T0 / 4);
}
    
```

❖ **Secteur 1 :**

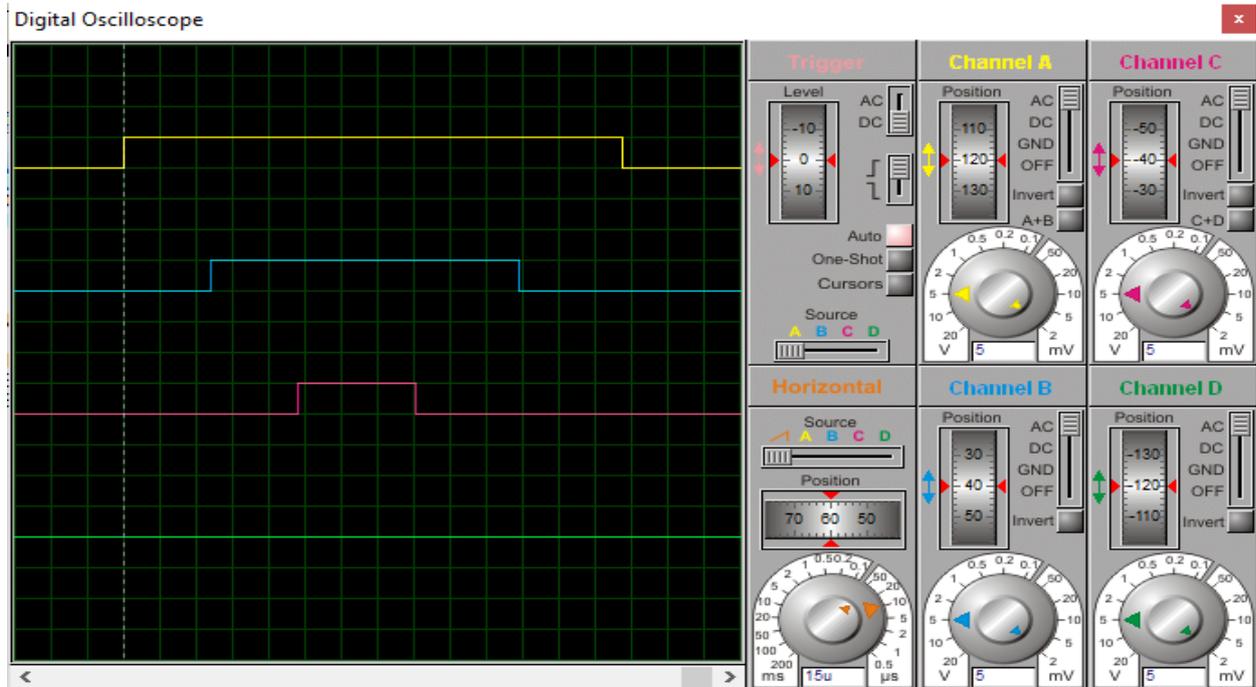


Figure IV- 7: Signaux (S1, S3 et S5) du secteur 1.

❖ **Secteur 2 :**

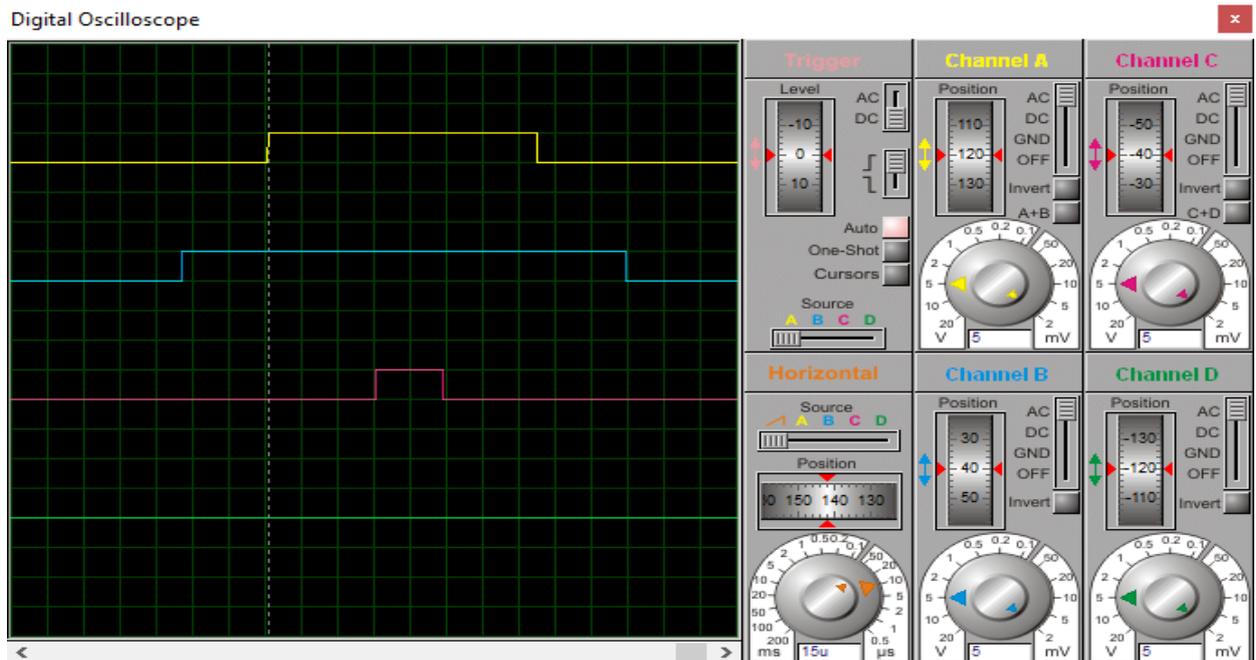


Figure IV- 8 : Signaux (S1, S3 et S5) du secteur 2.

❖ **Secteur 3 :**

Digital Oscilloscope

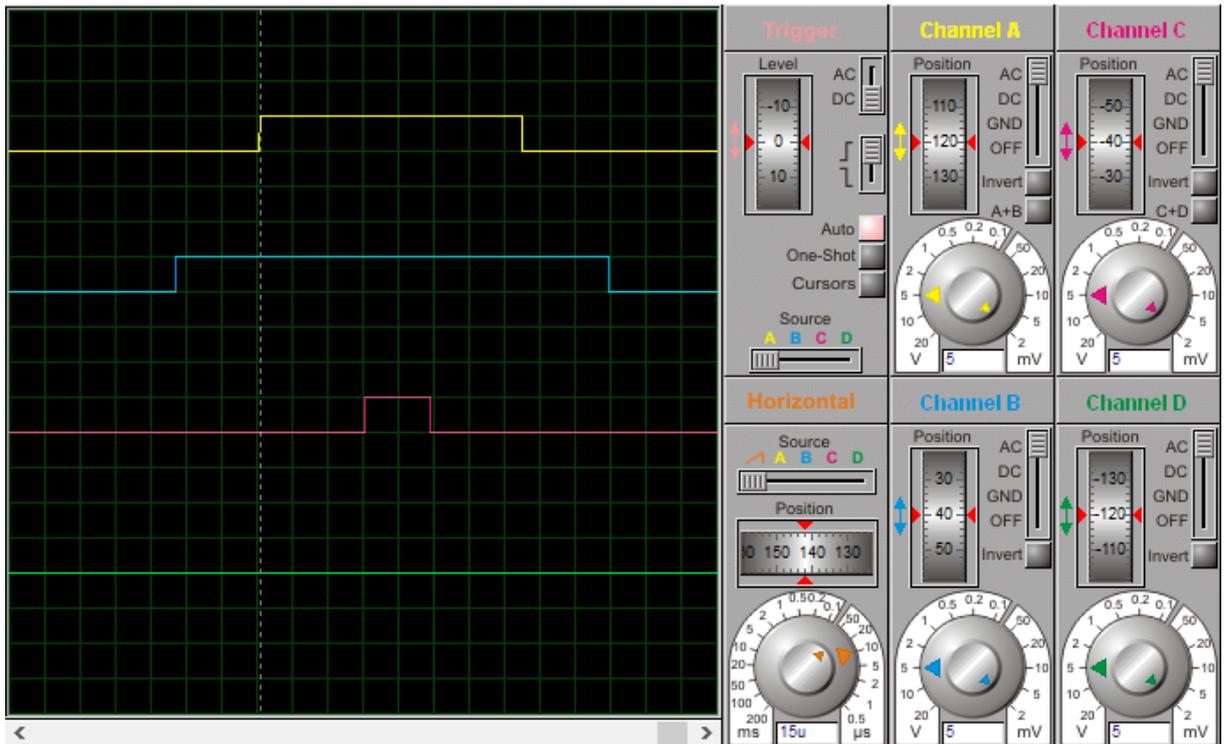


Figure IV- 9 : Signaux (S1, S3 et S5) du secteur 3.

❖ **Secteur 4 :**

Digital Oscilloscope

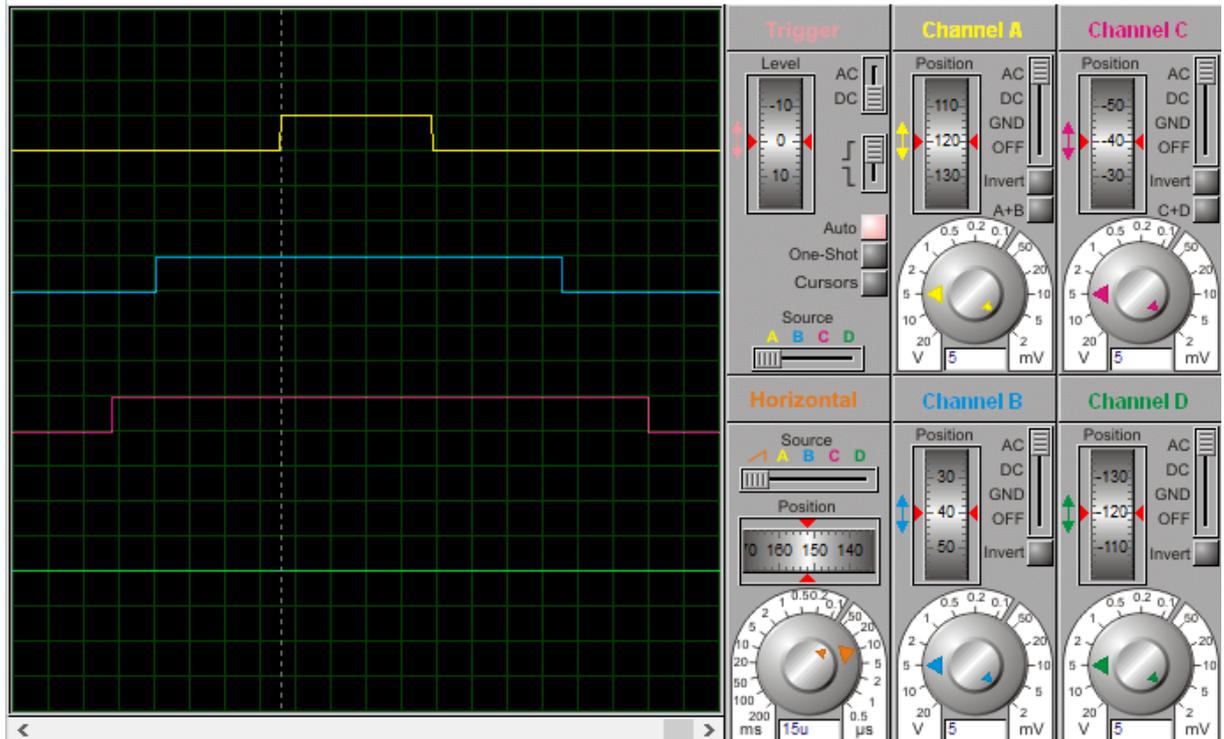


Figure IV- 10 : Signaux (S1, S3 et S5) du secteur 4.

❖ **Secteur 5 :**

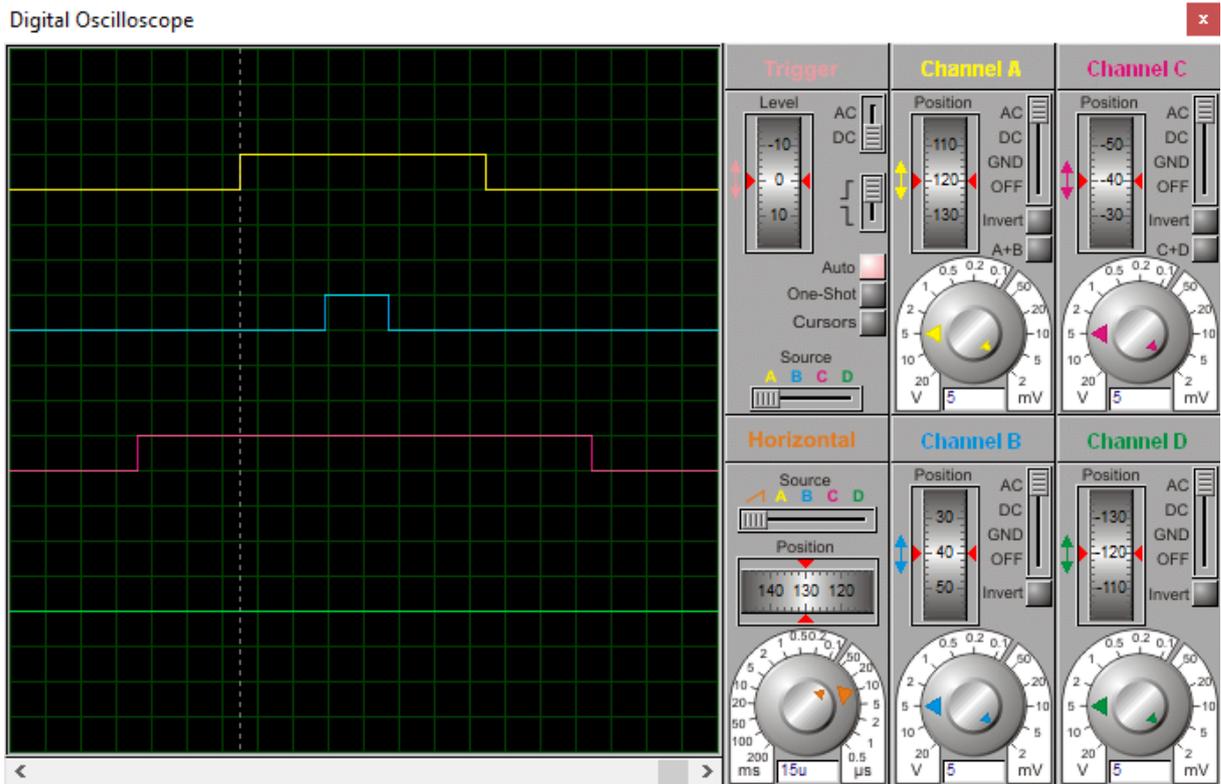


Figure IV- 11 : Signaux (S1, S3 et S5) du secteur 5.

❖ **Secteur 6 :**

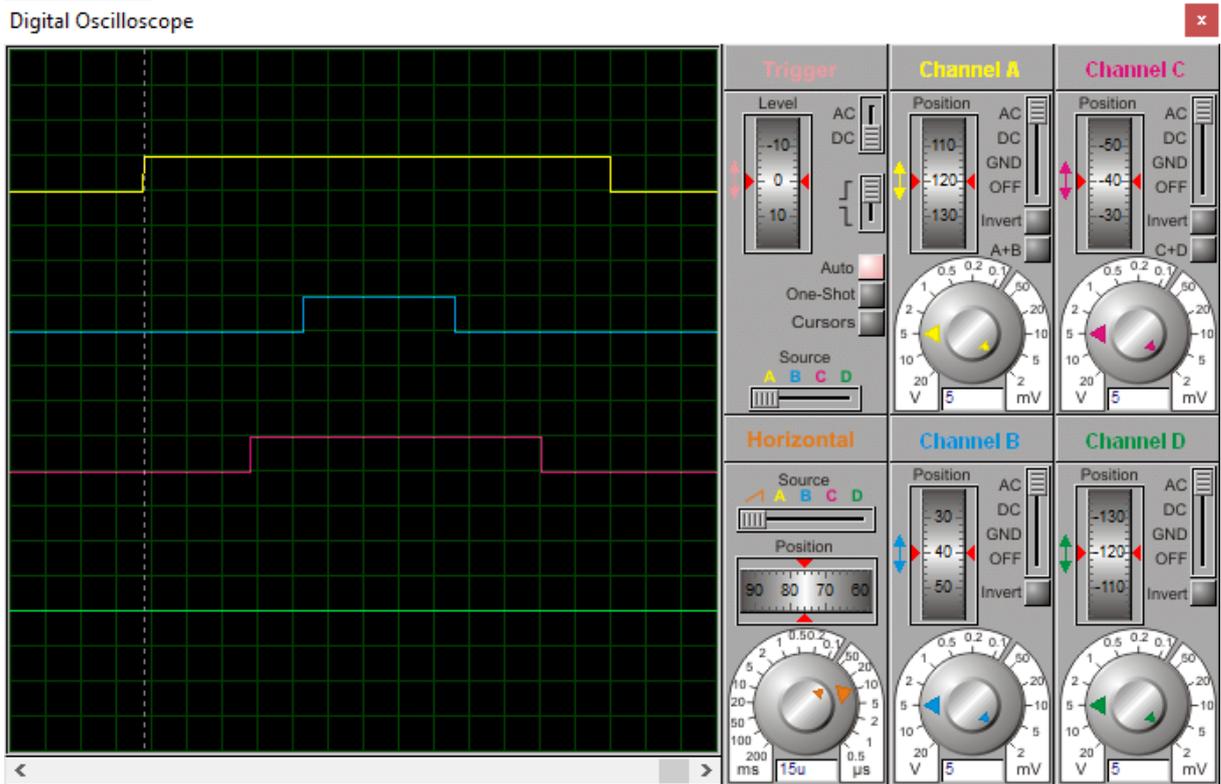


Figure IV- 12 : Signaux (S1, S3 et S5) du secteur 6.

**N.B:** la suite du code pour la représentation des signaux de commande des interrupteurs pour d'autres secteurs se situe dans l'Annexe B.

### **IV-6-1-Interprétation des résultats :**

A partir des résultats obtenus on constate que :

- Il est nécessaire d'effectuer une mise à l'échelle des valeurs avant de les introduire directement via les entrées analogiques de la carte Arduino.
- Il y'a un léger décalage au niveau des signaux de commande des interrupteurs, et cela est due aux temps de calcul et d'exécution du programme Arduino approprié à cette carte. Mais en général, les résultats obtenus à propos des signaux de commande des interrupteurs (S1, S3, S5) correspondent aux formes de largeur d'impulsion des six secteurs vues en troisième chapitre (voir les **Figures III- 4, 5, 6, 7, 8, 9**).

### **IV -7-Conclusion :**

En conclusion, ce chapitre a fourni une approche pratique pour l'implémentation de la commande SVM sur la carte Arduino Uno. Cette mise en œuvre permet aux ingénieurs et aux passionnés d'électronique de puissance d'expérimenter et de tester la commande SVM dans des applications réelles. La combinaison de la commande SVM et d'une plateforme telle qu'Arduino offre des possibilités intéressantes pour le développement de systèmes d'onduleurs avancés et la réalisation de projets dans le domaine de l'électrotechnique et de l'électronique de puissance.

## **Conclusion générale**

Ce mémoire se concentre sur l'implémentation de la commande SVM d'un onduleur triphasé de tension sur une carte Arduino Uno. L'étude met en évidence l'importance des onduleurs et des techniques de commande dans le processus de conversion de l'énergie électrique. Parmi les méthodes de commande étudiées, la modulation de largeur d'impulsion (MLI) vectorielle, cette commande s'est révélée particulièrement efficace pour générer des signaux de commande précis et de haute qualité pour les onduleurs triphasés. Les simulations réalisées sur Matlab de cette commande ont démontré les performances satisfaisantes de cette approche sous différents aspects.

L'implémentation concrète de la commande SVM sur la carte Arduino Uno a permis de valider la faisabilité de cette méthode dans des applications réelles. Les résultats obtenus ont confirmé la validité de la commande SVM et ont ouvert des perspectives intéressantes pour son utilisation dans divers domaines. Il est toutefois important de noter que cette étude présente certaines limitations, nécessitant ainsi des recherches futures pour les aborder.

Concernant les futurs travaux, par exemple s'ouvrir sur d'autres dérivées de la commande SVM tel que la commande des onduleurs multi niveaux pour obtenir un rendement amélioré et plus performant et réduire les harmoniques et les pertes par commutation, ainsi construire une maquette contenant tous les éléments réels de cette commande (onduleur, carte Arduino, machine asynchrone, circuit de commande scalaire, régulateur...etc.).

En conclusion, ce mémoire a contribué à une meilleure compréhension des onduleurs et des techniques de commande, en démontrant l'efficacité de la commande SVM. Les connaissances acquises peuvent servir de fondement pour des études ultérieures et des applications pratiques dans le domaine de la conversion de l'énergie électrique. Cependant, il est essentiel de souligner que des recherches supplémentaires sont nécessaires pour aborder les limitations identifiées et explorer davantage les possibilités offertes par la commande SVM dans d'autres domaines pertinents.

## Références bibliographiques

- [1] **BOUKAROURA Abdelkader** « Modélisation et Diagnostic d'un Onduleur Triphasé par l'Approche Bond-Graph », Magister en électrotechnique, université de Ferhat Abbas de Sétif, année 2009.
- [2] **Guy Séguier, Francis Labrique et Philippe Delarue** « Électronique de puissance Structures, commandes, applications » livre, 10e édition, année 2015.
- [3] **Moussa BOUSEHABA** « REALISATION D'UNE COMMANDE MLI A CHOIX MULTIPLE », mémoire fin d'étude, université de Abou Bekr Belkaïd de Tlemcen, année 2017.
- [4] **HAMADENE Somia et HADDOUCHE Khedidja** « Conception, réalisation et commande numérique d'un onduleur de tension triphasé », mémoire fin d'étude, université de
- [5] **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 fin d'étude, université de M'sila, année 2014.
- [6] **Ben Ahmed Bouzid et Chelbi Farid** « Implémentation sur DSP F28335 des Stratégies de Commande d'un Convertisseur DC/AC », mémoire fin d'étude, université A.Mira de Béjaïa.
- [7] **BENDAIKHA Abdelmalik** « Commande par MLI vectorielle d'un Onduleur multi-niveaux », thèse de doctorat, Université Badji Mokhtar- Annaba, année 2018.
- [8] **Tadjer Sid-Ahmed** « Etude d'un système de compensation d'harmonique en utilisant un générateur photovoltaïque "GPV" », mémoire de Magister, Université M'Hamed Bougara de Boumerdes, 2008.
- [9] **HAMMAMID Chaouki** « Amélioration de la compensation active du courant harmonique par la commande MLI vectorielle », mémoire fin d'étude, université de M'sila, année 2012.
- [10] **ZOGHBI Abderrezzaq** « Commande des onduleurs multi-niveaux » mémoire de master, école nationale polytechnique, année 2013.
- [11] **A. Sarinar Toled** « Commande Directe et Observation des convertisseurs de puissance : Application à l'Onduleur de tension Triphasé », These de Doctorat, Institut National Polytechnique de Grenoble, France, 2000.
- [12] **OKBAOUI Sidiiali , CHAIN Moncef** «Réalisation et commande d'un onduleur triphasé a deux niveaux», Mémoire de Master, UNIVERSITE Amar Telidji Laghoat, algérie, Année Universitaire 2018/2019.
- [13] **J. P. Caron, J.P. Hautier**, « Modélisation Et Commande de La Machine Asynchrone ». Edition TECHNIP, Paris 1995.
- [14] **Adjimi Nadia, Belaidi Wahiba** «Modélisation et commande d'un onduleur MLI ». Mémoire de Master, Universitaire Larbi Ben M'hidi Oum El-Bouaghi, Algérie, Année Universitaire : 2008/2009.
- [15] <https://www.tunisia-sat.com>.
- [16] Etude et programmation d'une carte ARDUINO UNO pour le comptage d'entrées/sorties d'un parking.

## Annexe A :

**A-1-** La suite du programme pour représenter les signaux de commande pour d'autres secteurs :

```
elseif(Secteur==2)
  if (t<(T0/4))
    S1=0;S3=0;S5=0;%V0(000)
  elseif ((t>=(T0/4)) && (t<((T0/4)+T2/2)))
    S1=0;S3=1;S5=0;%V3(010)
  elseif ((t>=(T0/4)+T2/2) && (t<((T0/4)+T1/2+T2/2)))
    S1=1;S3=1;S5=0;%V2(110)
  elseif ((t>=(T0/4)+T1/2+T2/2) && (t<(3*T0/4)+T1/2+T2/2))
    S1=1;S3=1;S5=1;%V7(111)
  elseif ((t>=(3*T0/4)+T1/2+T2/2) && (t<(3*T0/4)+T1+T2/2))
    S1=1;S3=1;S5=0;%V2(110)
  elseif ((t>=(3*T0/4)+T1+T2/2) && (t<(3*T0/4)+T1+T2))
    S1=0;S3=1;S5=0;%V3(010)
  else
    S1=0;S3=0;S5=0;%V0(000)
  End

elseif(Secteur==3)
  if (t<(T0/4))
    S1=0;S3=0;S5=0;%V0(000)
  elseif ((t>=(T0/4)) && (t<((T0/4)+T1/2))))
    S1=0;S3=1;S5=0;%V3(010)
  elseif ((t>=(T0/4)+T1/2) && (t<((T0/4)+T1/2+T2/2)))
    S1=0;S3=1;S5=1;%V4(011)
  elseif ((t>=(T0/4)+T1/2+T2/2) && (t<(3*T0/4)+T1/2+T2/2))
    S1=1;S3=1;S5=1;%V7(111)
  elseif ((t>=(3*T0/4)+T1/2+T2/2) && (t<(3*T0/4)+T1/2+T2))
    S1=0;S3=1;S5=1;%V4(011)
  elseif ((t>=(3*T0/4)+T1/2+T2) && (t<(3*T0/4)+T1+T2))
    S1=0;S3=1;S5=0;%V3(010)
  else
    S1=0;S3=0;S5=0;%V0(000)
  end

elseif(Secteur==4)
  if (t<(T0/4))
    S1=0;S3=0;S5=0;%V0(000)
  elseif ((t>=(T0/4)) && (t<((T0/4)+T2/2)))
    S1=0;S3=0;S5=1;%V5(001)
  elseif ((t>=(T0/4)+T2/2) && (t<((T0/4)+T1/2+T2/2)))
    S1=0;S3=1;S5=1;%V4(011)
  elseif ((t>=(T0/4)+T1/2+T2/2) && (t<(3*T0/4)+T1/2+T2/2))
    S1=1;S3=1;S5=1;%V7(111)
  elseif ((t>=(3*T0/4)+T1/2+T2/2) && (t<(3*T0/4)+T1+T2/2))
    S1=0;S3=1;S5=1;%V4(011)
  elseif ((t>=(3*T0/4)+T1+T2/2) && (t<(3*T0/4)+T1+T2))
    S1=0;S3=0;S5=1;%V5(001)
  else
    S1=0;S3=0;S5=0; %V0(000)
  end

elseif(Secteur==5)
  if (t<(T0/4))
```

```

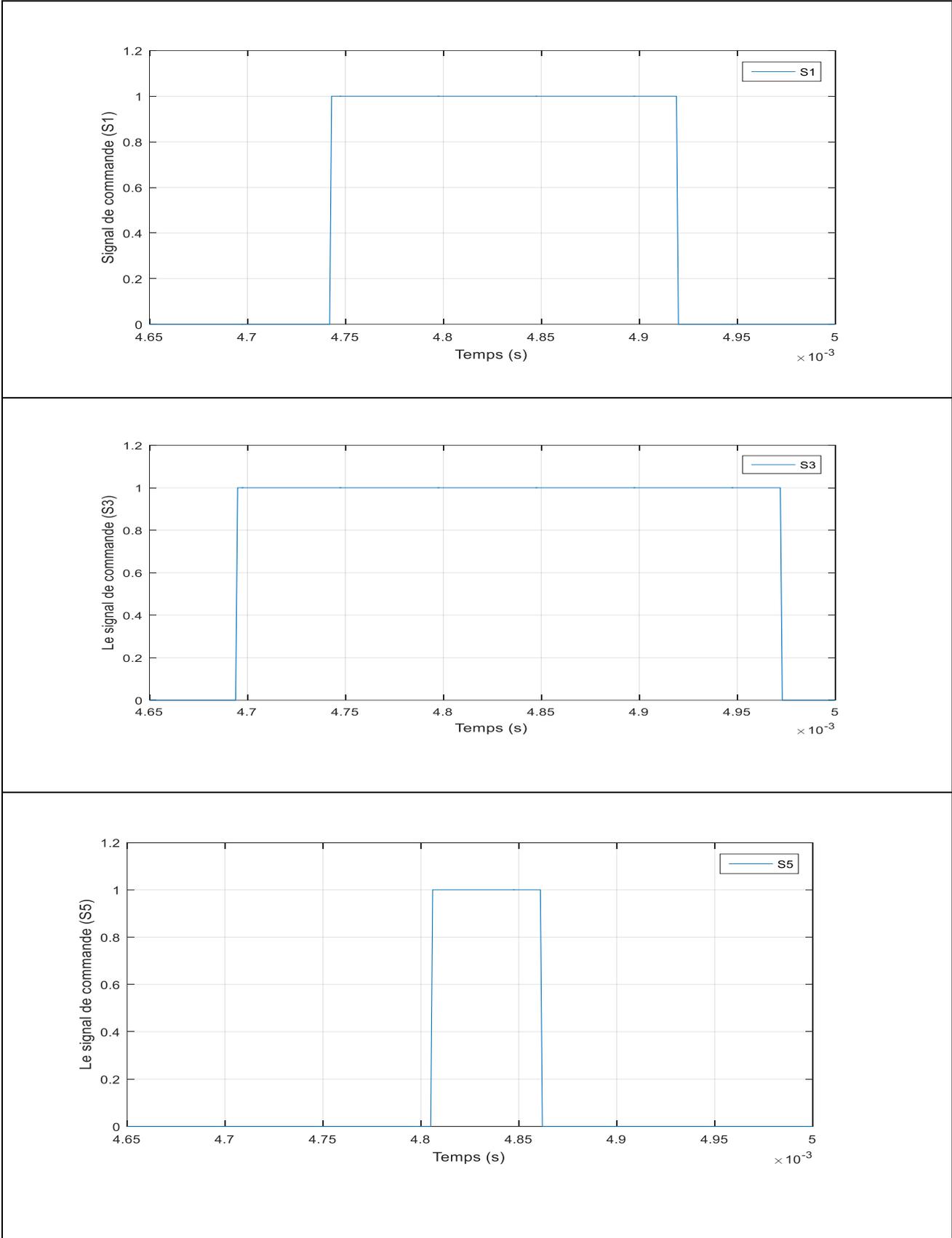
S1=0;S3=0;S5=0;%V0(000)
elseif ((t>=(T0/4)) && (t<(T0/4)+T1/2))
S1=0;S3=0;S5=1; %V5(001)
elseif((t>=(T0/4)+T1/2) && (t<((T0/4)+T1/2+T2/2)))
S1=1;S3=0;S5=1; %V6(101)
elseif ((t>=(T0/4)+T1/2+T2/2)) && (t<(3*T0/4)+T1/2+T2/2))
S1=1;S3=1;S5=1;%V7(111)
elseif ((t>=(3*T0/4)+T1/2+T2/2) && (t<(3*T0/4)+T1/2+T2))
S1=1;S3=0;S5=1; %V6(101)
elseif((t>=(3*T0/4)+T1/2+T2) && (t<(3*T0/4)+T1+T2))
S1=0;S3=0;S5=1;%V5(001)
else
S1=0;S3=0;S5=0;%V0(000)
End

else %Secteur=6
if (t<(T0/4))
S1=0;S3=0;S5=0;%V0(000)
elseif ((t>=(T0/4)) && (t<((T0/4)+T2/2)))
S1=1;S3=0;S5=0;%V1(100)
elseif((t>=(T0/4)+T2/2)) && (t<((T0/4)+T1/2+T2/2)))
S1=1;S3=0;S5=1;%V6(101)
elseif ((t>=(T0/4)+T1/2+T2/2)) && (t<(3*T0/4)+T1/2+T2/2))
S1=1;S3=1;S5=1;%V7(111)
elseif ((t>=(3*T0/4)+T1/2+T2/2)) && (t<(3*T0/4)+T1+T2/2))
S1=1;S3=0;S5=1;%V6(101)
elseif((t>=(3*T0/4)+T1+T2/2) && (t<(3*T0/4)+T1+T2))
S1=1;S3=0;S5=0;%V1(100)
else %V0(000)
S1=0;S3=0;S5=0;
end

end
end

```

**A-2- Les résultats de simulations :**



**Figure A-1 :** Les signaux de commande (S1, S3, S5) du secteur 2.

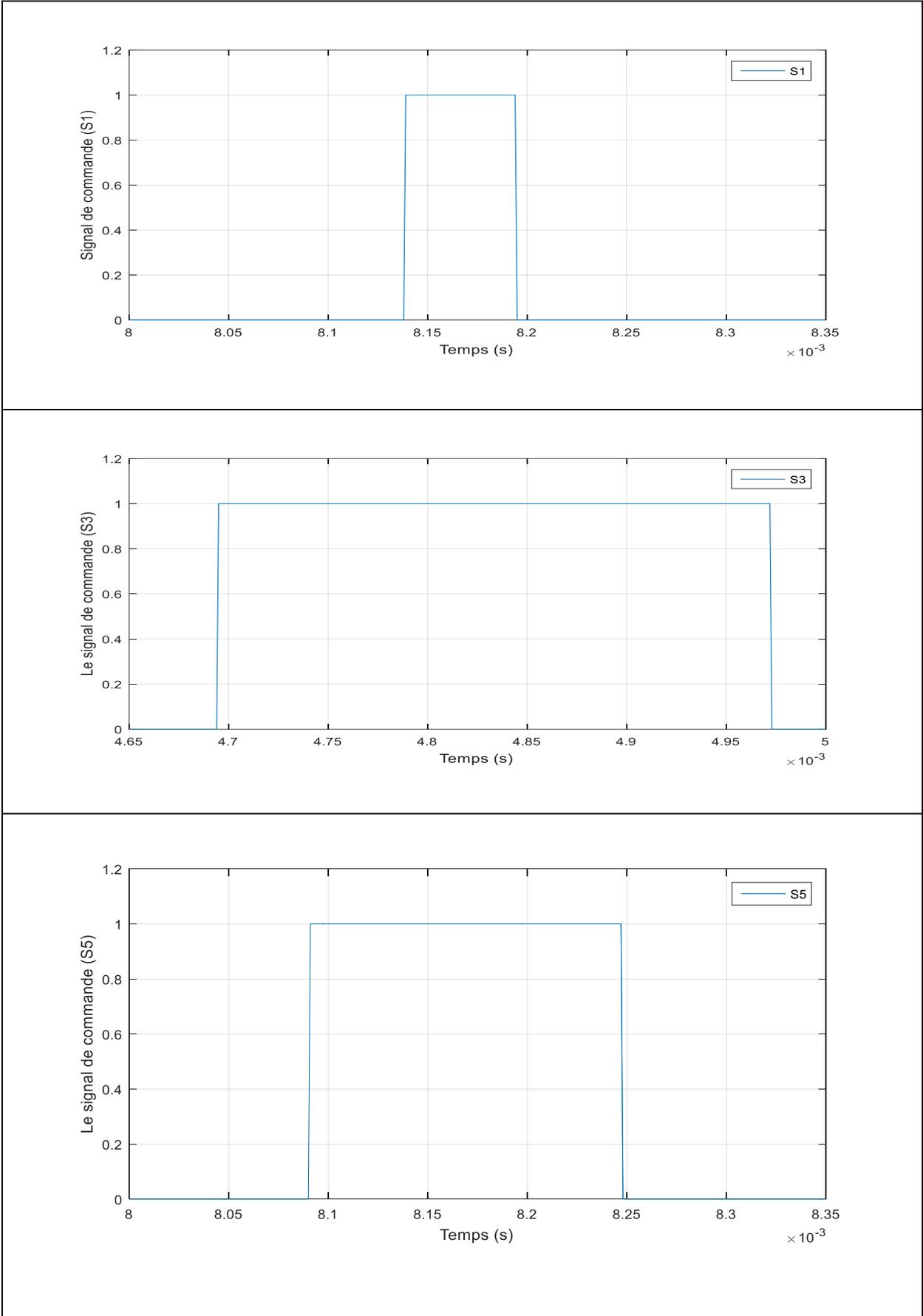
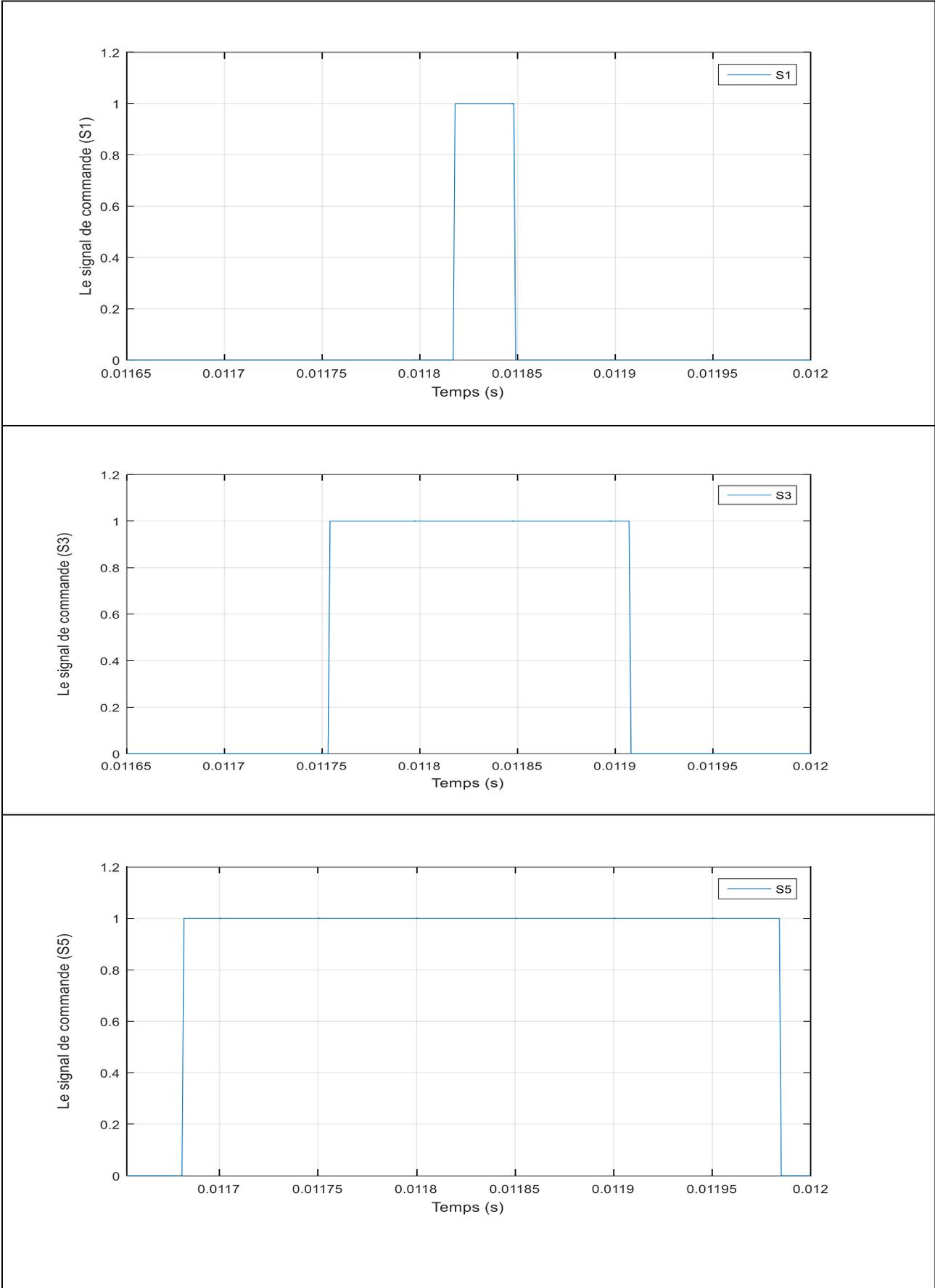
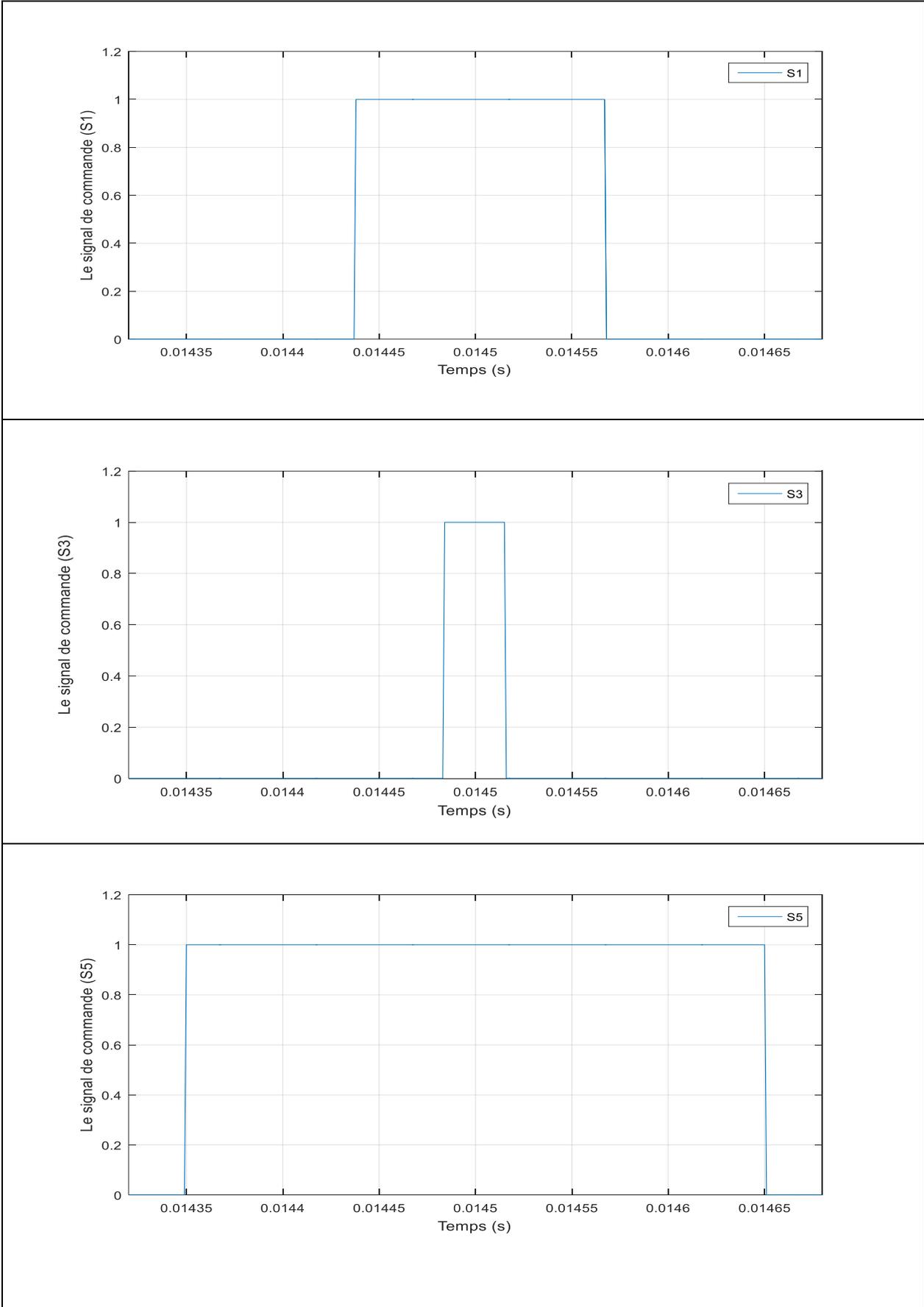


Figure A-2 : Les signaux de commande (S1, S3, S5) du secteur 3.



**Figure A-3 :** Les signaux de commande (S1, S3, S5) du secteur 4.



**Figure A-4 :** Les signaux de commande (S1, S3, S5) du secteur 5.

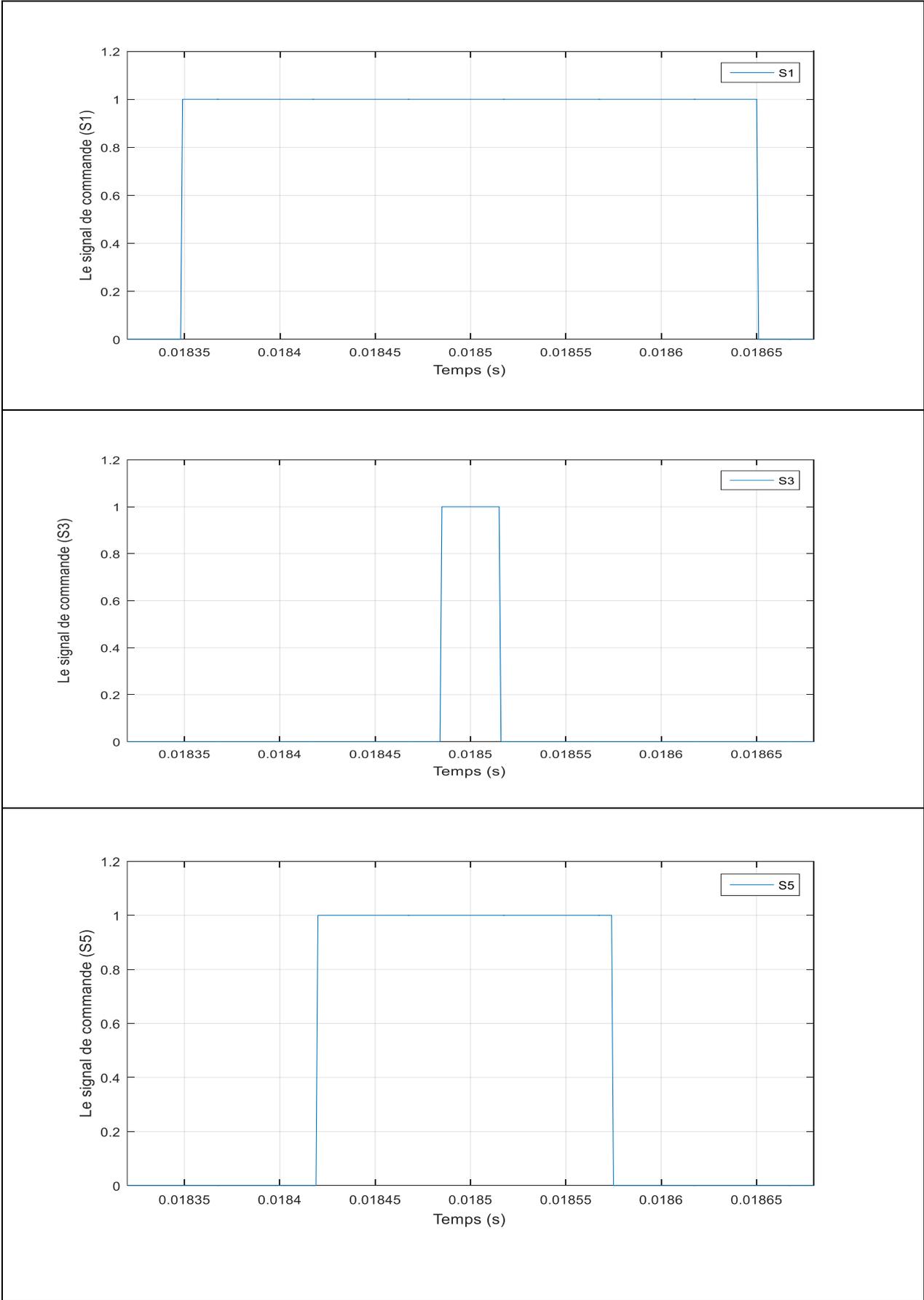


Figure A-5 : Les signaux de commande (S1, S3, S5) du secteur 6.

## Annexe B :

### **B-1-** La suite du code Arduino de la commande SVM :

```
#include <stdio.h>

#include <math.h>

#define PI 3.14159265

// Constantes

const int AREF = 5;

const int f = 50;

const int Vs = 400 / sqrt(2);

const float Vref = 2.886;

const float Tc = 1.0 / 10000.0;

const int Vdc = 5;

// Variables globales

float Fref = 0;

float Frefn = 0;

float Valpha = 0;

float Vbeta = 0;

float phi = 0;

float theta = 0;

float T1 = 0, T2 = 0, T0 = 0;

int Secteur = 1;

float Va = 0;

float Vb = 0;

float Vc = 0;

float amplitude = 0;

void cas0()/V0(000)

{ digitalWrite(3, LOW);
```

```
digitalWrite(5, HIGH);
digitalWrite(6, LOW);
digitalWrite(9, HIGH);
digitalWrite(10, LOW);
digitalWrite(11, HIGH);
}

void cas1()//V1(100)
{ digitalWrite(3, HIGH);
  digitalWrite(5, LOW);
  digitalWrite(6, LOW);
  digitalWrite(9, HIGH);
  digitalWrite(10, LOW);
  digitalWrite(11, HIGH);
}

void cas2()//V2(110)
{
  digitalWrite(3, HIGH);
  digitalWrite(5, LOW);
  digitalWrite(6, HIGH);
  digitalWrite(9, LOW);
  digitalWrite(10, LOW);
  digitalWrite(11, HIGH);
}

void cas3()//V3(010)
{
  digitalWrite(3, LOW);
  digitalWrite(5, HIGH);
```

```
digitalWrite(6, HIGH);  
digitalWrite(9, LOW);  
digitalWrite(10, LOW);  
digitalWrite(11, HIGH);  
}
```

```
void cas4()//V4(011)
```

```
{  
digitalWrite(3, LOW);  
digitalWrite(5, HIGH);  
digitalWrite(6, HIGH);  
digitalWrite(9, LOW);  
digitalWrite(10, HIGH);  
digitalWrite(11, LOW);  
}
```

```
void cas5()//V5(001)
```

```
{  
digitalWrite(3, LOW);  
digitalWrite(5, HIGH);  
digitalWrite(6, LOW);  
digitalWrite(9, HIGH);  
digitalWrite(10, HIGH);  
digitalWrite(11, LOW);  
}
```

```
void cas6()//V6(101)
```

```
{  
digitalWrite(3, HIGH);  
digitalWrite(5, LOW);  
}
```

```
digitalWrite(6, LOW);
digitalWrite(9, HIGH);
digitalWrite(10, HIGH);
digitalWrite(11, LOW);
}
void cas7()/V7(111)
{
digitalWrite(3, HIGH);
digitalWrite(5, LOW);
digitalWrite(6, HIGH);
digitalWrite(9, LOW);
digitalWrite(10, HIGH);
digitalWrite(11, LOW);
}
void setup() {
Serial.begin(115200);
pinMode(3, OUTPUT); //S1
pinMode(5, OUTPUT); //S2
pinMode(6, OUTPUT); //S3
pinMode(9, OUTPUT); //S4
pinMode(10, OUTPUT); //S5
pinMode(11, OUTPUT); //S6
pinMode(A0, INPUT); //Fref
}
void loop() {
Fref = analogRead(A0);
Frefn = (Fref * AREF / 1024);
```

```

float amplitude = Frefn * (Vs / f);

float t = millis() / 1000.0;

float w = 2 * PI * Frefn;

float Va = amplitude * sin(w * t);

float Vb = amplitude * sin(w * t - (2 * PI / 3));

float Vc = amplitude * sin(w * t - (4 * PI / 3));

float a[3] = {1, -1.0 / 2, -1.0 / 2};

float b[3] = {0, sqrt(3.0) / 2, -sqrt(3.0) / 2};

Valpha = sqrt(2.0 / 3.0) * (a[0] * Va + a[1] * Vb + a[2] * Vc);

Vbeta = sqrt(2.0 / 3.0) * (b[0] * Va + b[1] * Vb + b[2] * Vc);

phi = atan2(Vbeta, Valpha) + PI;

Serial.print("Secteur= ");

Serial.println(Secteur);

if (phi >= 0 && phi < (PI / 3)) {

    theta = phi;

    Secteur = 1;

} else if (phi >= (PI / 3) && phi < (2 * PI / 3)) {

    theta = phi - (PI / 3);

    Secteur = 2;

} else if (phi >= (2 * PI / 3) && phi < (PI)) {

    theta = phi - (2 * PI / 3);

    Secteur = 3;

} else if (phi >= (PI) && phi < (4 * PI / 3)) {

    theta = phi - (3 * PI / 3);

    Secteur = 4;

} else if (phi >= (4 * PI / 3) && phi < (5 * PI / 3)) {

    theta = phi - (4 * PI / 3);

```

```

Secteur = 5;

} else if (phi >= (5 * PI / 3) && phi < (2 * PI)) {

    theta = phi - (5 * PI / 3);

    Secteur = 6;

}

T1 = ((2 / (sqrt(3) * Vdc)) * Tc * Vref * sin((Secteur * PI / 3) - phi)) ;
T2 = ((2 / (sqrt(3) * Vdc)) * Tc * Vref * sin(phi - ((Secteur - 1) * PI / 3))) ;
T0 = (Tc - T1 - T2) ;

// Conversion des temps en microsecondes

unsigned long T1 = (unsigned long)(T1 * 1000000);
unsigned long T2 = (unsigned long)(T2 * 1000000);
unsigned long T0 = (unsigned long)(T0 * 1000000);

//secteur(1)

if (Secteur == 1)
{ cas0(); // V0(000)

    delayMicroseconds(T0 / 4);

    cas1(); // V1(100)

    delayMicroseconds(T1 / 2);

    cas2(); // V2(110)

    delayMicroseconds(T2 / 2);

    cas7(); // V7(111)

    delayMicroseconds((T0 / 2) + 22);

    cas2(); // V2(110)

    delayMicroseconds((T2 / 2) + 25);

    cas1(); // V1(100)

```

```

delayMicroseconds((T1 / 2) + 25);
cas0(); // V0(000)
delayMicroseconds(T0 / 4);
}
//secteur(2)
else if (Secteur == 2)
{ cas0(); //V0(000)
delayMicroseconds(T0 / 4);
cas3(); //V3(010)
delayMicroseconds((T2 / 2) + 18);
cas2(); //V2(110)
delayMicroseconds((T1 / 2));
cas7(); // V7(111)
delayMicroseconds(T0 / 2);
cas2(); //V2(110)
delayMicroseconds((T1 / 2) + 30);
cas3(); //V3(010)
delayMicroseconds((T2 / 2));
cas0(); // V0(000)
delayMicroseconds((T0 / 4));
}
//secteur(3)
else if (Secteur == 3)
{ cas0(); // V0(000)
delayMicroseconds(T0 / 4);
cas3(); //V3(010)
delayMicroseconds(T1 / 2);

```

```
cas4(); //V4(011)
delayMicroseconds((T2 / 2) + 20);
cas7(); // V7(111)
delayMicroseconds((T0 / 2) + 38);
cas4(); //V4(011)
delayMicroseconds(T2 / 2);
cas3(); //V4(010)
delayMicroseconds((T1 / 2)+15);
cas0(); // V0(000)
delayMicroseconds(T0 / 4);
}
//secteur(4)
else if (Secteur == 4)
{ cas0(); // V0(000)
delayMicroseconds(T0 / 4);
cas5(); //V5(001)
delayMicroseconds(T2 / 2);
cas4(); //V4(011)
delayMicroseconds((T1 / 2)+35);
cas7(); // V7(111)
delayMicroseconds((T0 / 2)+38);
cas4(); //V4(011)
delayMicroseconds((T1 / 2)+20);
cas5(); //V5(001)
delayMicroseconds(T2 / 2);
cas0(); // V0(000)
delayMicroseconds(T0 / 4);
```

```

}
//secteur(5)
else if (Secteur == 5)
{ cas0(); // V0(000)
  delayMicroseconds(T0 / 4);
  cas5(); //V5(001)
  delayMicroseconds((T1 / 2) + 20);
  cas6(); //V6(101)
  delayMicroseconds((T2 / 2));
  cas7(); // V7(111)
  delayMicroseconds(T0 / 2);
  cas6(); //V6(101)
  delayMicroseconds((T2 / 2) + 15);
  cas5(); //V5(001)
  delayMicroseconds((T1 / 2));
  cas0(); // V0(000)
  delayMicroseconds(T0 / 4);
}
else
{ cas0(); // V0(000)
  delayMicroseconds(T0 / 4);
  cas1(); //V1(100)
  delayMicroseconds((T2 / 2));
  cas6(); //V6(101)
  delayMicroseconds((T1 / 2) + 5);
  cas7(); // V7(111)
  delayMicroseconds((T0 / 2) + 38);
}

```

```
cas6(); //V6(101)
delayMicroseconds(T1 / 2);
cas1(); //V1(100)
delayMicroseconds((T2 / 2) + 20);
cas0(); // V0(000)
delayMicroseconds(T0 / 4);
}
}
```

## **Résumé :**

Notre travail consiste à étudier la commande SVM (Space Vector Modulation) d'un onduleur triphasé de tension implémentée sur la carte Arduino. Ce mémoire explore les différents types d'onduleurs et leur classification, ainsi que les avantages et les limites des différentes méthodes de commande. Une attention particulière est accordée à la commande par modulation de largeur d'impulsion (MLI) vectorielle, mettant en évidence la commande SVM comme une approche efficace. Des simulations ont été réalisées pour évaluer les performances de cette technique sous différents aspects. De plus, l'implémentation de la commande SVM à l'aide de la carte Arduino Uno a confirmé sa viabilité dans des applications concrètes. Ce mémoire souligne l'importance des onduleurs et des techniques de commande dans la conversion de l'énergie électrique, offrant ainsi des perspectives prometteuses pour des recherches futures et des applications pratiques dans ce domaine.

## **Abstract :**

Our work consists of studying the Space Vector Modulation (SVM) control of a three-phase voltage inverter implemented on the Arduino board. This thesis explores the different types of inverters and their classification, as well as the advantages and limitations of various control methods. Special attention is given to Vectorized Pulse Width Modulation (PWM) control, highlighting SVM control as an effective approach. Simulations have been conducted to evaluate the performance of this technique in various aspects. Furthermore, the implementation of SVM control using the Arduino Uno board has confirmed its viability in practical applications. This thesis underscores the importance of inverters and control techniques in electric power conversion, thereby offering promising prospects for future research and practical applications in this field.

### **ملخص:**

يتكون عملنا من دراسة تحكم SVM (تعديل ناقل الفضاء) لعاكس جهد ثلاثي الأطوار يتم تنفيذه على لوحة Arduino. تستكشف هذه الرسالة الأنواع المختلفة من العاكسات وتصنيفها، بالإضافة إلى مزايا وقيود طرق التحكم المختلفة. يتم إيلاء اهتمام خاص للتحكم في تعديل عرض النبض (PWM) المستند إلى المتجهات، مما يبرز التحكم في SVM كنهج فعال. تم إجراء عمليات المحاكاة لتقييم أداء هذه التقنية في مختلف الجوانب. علاوة على ذلك، أكد تنفيذ أمر SVM باستخدام لوحة Arduino Uno جدواها في تطبيقات العالم الحقيقي. تؤكد هذه الرسالة على أهمية المحولات وتقنيات التحكم في تحويل الطاقة الكهربائية، وبالتالي تقدم منظورات واعدة للبحث والتطبيقات العملية في المستقبل في هذا المجال.