

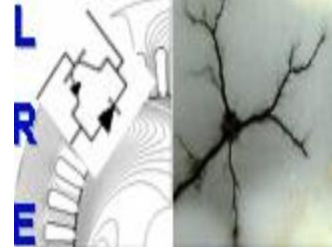
RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Ecole Nationale Polytechnique



المدرسة الوطنية المتعددة التقنيات  
Ecole Nationale Polytechnique



Département d'ELECTROTECHNIQUE

Laboratoire de Recherche en Electrotechnique

Mémoire de

**Master en Electrotechnique**

Intitulé

Implémentation de la MLI Vectorielle sur la  
carte DSP TMS320F28335 pour commander  
un onduleur triphasé

Présenté par

**MEFTAH Housseem eddine**

Sous la direction de

**Dr. T.ZEBBADJI et Pr. S.HADJI**

Présenté et soutenu publiquement le 18 juin 2017

**Composition du Jury :**

Président	H. SAHRAOUI, Docteur à l'ENP
Rapporteur/Promoteur	T. ZEBBADJI, Docteur à l'ENP S. HADJI, Professeur à l'EPST Alger
Examineur	MO. MAHMOUDI, Professeur à l'ENP

ENP 2017



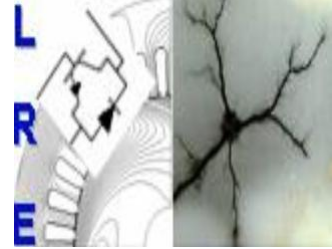
RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Ecole Nationale Polytechnique



المدرسة الوطنية المتعددة التقنيات  
Ecole Nationale Polytechnique



Département d'ELECTROTECHNIQUE

Laboratoire de Recherche en Electrotechnique

Mémoire de

**Master en Electrotechnique**

Intitulé

Implémentation de la MLI Vectorielle sur la  
carte DSP TMS320F28335 pour commander  
un onduleur triphasé

Présenté par

**MEFTAH Housseem eddine**

Sous la direction de

**Dr. T.ZEBBADJI et Pr. S.HADJI**

Présenté et soutenu publiquement le 18 juin 2017

**Composition du Jury :**

Président	H. SAHRAOUI, Docteur à l'ENP
Rapporteur/Promoteur	T. ZEBBADJI, Docteur à l'ENP S. HADJI, Professeur à l'EPST Alger
Examineur	MO. MAHMOUDI, Professeur à l'ENP

ENP 2017

## ملخص

الغرض من هذا العمل يكمن في تطبيق تقنية تضمين المتجه الفضائي (SVPWM) للتحكم علي مموج ثلاثي الأقطاب بواسطة بطاقة التحكم DSP TMS320F28335. يتم حساب الدوال المنطقية للقواطع باستخدام MATLAB/SIMULINK. تتم كتابة البرنامج المثبت في بطاقة التحكم DSP بلغة C. يتم إنشاء هذا البرنامج باستخدام MATLAB/SIMULINK وتشغيله عبر برنامج CCS IDE. إشارات التحكم التي تم الحصول عليها قريبة جدا من نتائج المحاكاة و بالتالي تأكد حسن سير العمل لهته التقني.

**الكلمات المفتاحية :** تقنية تضمين المتجه الفضائي , الدوال المنطقية للقواطع , إشارات التحكم.

## Abstract

The purpose of this work is the Implementation of the space vector pulse width modulation (SVPWM) strategy, for the control of a three phase inverter using DSP TMS320F28335 Experimenter kit. The logic functions of the switches are calculated using MATLAB/SIMULINK. The code implemented on the DSP is in C language; this code is generated employing MATLAB/SIMULINK software and then transferred to the DSP board (downloaded) for execution via the CCS IDE software. The control signals are very close to the simulation ones, they can therefore be termed as satisfactory.

**Key words:** SVPWM, DSP, control signals, logic functions.

## Résumé

Le but de ce travail est d'implémenter la MLI vectorielle sur la carte DSP TMS320F28335. Les fonctions logiques des interrupteurs sont déterminés à l'aide de logicielle MATLAB/SIMULINK. L'élaboration du programme implanté sur le DSP est effectuée en langage C. Ce programme est généré à partir du logiciel MATLAB/SIMULINK et transféré pour l'exécution via le logiciel CCS IDE. Les signaux de commande obtenus sont très proches des résultats de simulation, et peuvent êtres qualifiés de satisfaisants.

**Mots clés :** MLI vectorielle, carte DSP, signaux de commande, fonctions logiques.

# *Dédicaces*

*A mes parents, qui m'ont soutenu tout au long de  
mes études.*

*A tous les membres de ma famille.*

*À mes amis.*

*À tous mes professeurs*

*Houssein Eddine*

## **Remerciement**

*Je remercie, avant tout, notre Dieu de m'avoir donné la force et le courage pour réaliser ce travail.*

*Je remercie mes encadreurs : Mr T.ZEBBADJI et Mr S.HADJI, pour leurs précieuses directives, leurs fructueux conseils, leurs courtoisie si raffinée et leurs qualités humaines.*

*Je remercie vivement Melle S.Chouiref pour son aide, sa patience et sa disponibilité durant tout le long de notre travail.*

*J'adresse mes chaleureux remerciements aux membres du jury qui ont bien voulu juger mon travail et m'honorer avec leur participation.*

*Je souhaite également remercier tous les enseignants ayant assurés mes cours pendant mes années d'études.*

## **Table des matières :**

### **Liste des Tableaux**

### **Liste des Figures**

### **Liste des Abréviations**

<b>Introduction générale .....</b>	<b>10</b>
------------------------------------	-----------

## **Chapitre 1 : La MLI Vectorielle (SVM)**

1.1 Introduction .....	12
1.2 La structure d'un onduleur à deux niveaux .....	12
1.3 Etats d'un bras de l'onduleur .....	13
1.4 Tensions de sortie d'un onduleur .....	14
1.5 Principe du vecteur spatial (Space Vector) .....	14
1.6 Vecteurs de tension de référence.....	16
1.7 Période de commutation.....	18
1.8 Séquences des interrupteurs de l'onduleur suivant la stratégie.....	18
1.9 Résultats de simulation.....	21
1.9.1 Variation de taux de modulation r.....	21
1.9.2 Variation de la fréquence de commutation $F_e$ .....	22
1.10 Conclusion .....	24

## **Chapitre 2 : Implémentation de la MLI vectorielle sur DSP TMS320F28335**

2.1 Introduction .....	26
2.1.2 Nomenclature de la famille TMS320 .....	28
2.1.3 Les périphériques .....	28
2.2 Convertisseur analogique numérique (the Analog-to-digital converter ADC) .....	30
2.3 Direct Memory Access Controller (DMA) .....	31
2.4 Les Interruptions .....	32
2.4.1 Hiérarchie des interruptions .....	32
2.4.2 Niveaux de priorités des interruptions du DSP .....	33
2.4.3 Interrupt Flag Register (IFR).....	34
2.4.4 Interrupt Mask Register (IMR) .....	34
2.5 GP timers.....	34
2.5.1 Les entrée/sorties de GP timer .....	34
2.5.2 Les modes de comptage .....	35
2.6 Les sorties MLI .....	35
2.6.1 Calcul des temps actif et inactif de la sortie TxPWM.....	37
2.6.2 Génération de temps morts.....	37
2.6.3 L'unité de comparaison.....	39
2.7 Les outils de développement logiciel .....	39
2.7.1 Code Composer Studio (IDE) .....	39
2.7.2 Matlab/Simulink.....	39
2.7.3 Exemple de configurations des périphériques de la DSP sur Simulink .....	40
2.8 Implémentation sur DSP .....	42
2.9 Génération des signaux de commande par la technique MLI vectorielle .....	44
2.10 Conclusion.....	45

<b>Conclusion générale .....</b>	<b>47</b>
<b>Bibliographie.....</b>	<b>49</b>

## Liste des Tableaux :

<b>Tableau</b>		<b>Page</b>
Tableau 1.1	Etat des Interrupteurs et vecteurs de commutation de l'étage Onduleur	16
Tableau 2.1	Les interruptions suivant leur ordre de priorité et leurs descriptions	33
Tableau 2.2	Le temps mort généré par les différentes combinaisons de registre DTCONx en ( $\mu$ s)	38



## Liste des Figures :

Figure	Page
<b>Figure 1.1</b> Structure d'un onduleur triphasé à deux niveaux .....	12
<b>Figure 1.2</b> Fonctionnement d'un bras d'onduleur .....	13
<b>Figure 1.3</b> les huit états (configurations) d'un onduleur triphasé à deux niveaux. ....	13
<b>Figure 1.4</b> Diagramme représentant les positions vectorielles de l'onduleur à deux niveaux, relatives à la stratégie SVM.....	15
<b>Figure 1.5</b> Diagramme illustrant les états X, Y, Z, pour chacun des six secteurs de la SVM .....	18
<b>Figure 1.6</b> Formes des fonctions logiques $F_{ij}$ des interrupteurs pour les six secteurs et durant une période de commutation $T_s$ .....	20
<b>Figure 1.7</b> le spectre de la tension de sortie $V_{IN}$ pour $r=0.2$ et $F_e = 1000$ Hz.....	21
<b>Figure 1.8</b> le spectre de la tension de sortie $V_{IN}$ pour $r=0.5$ et $F_e = 1000$ Hz.....	21
<b>Figure 1.9</b> le spectre de la tension de sortie $V_{IN}$ pour $r=1$ et $F_e = 1000$ Hz.....	21
<b>Figure 1.10</b> courbe de réglage .....	22
<b>Figure 1.11</b> le spectre de la tension de sortie $V_{IN}$ pour $r=1$ et $F_e = 1000$ Hz.....	23
<b>Figure 1.12</b> le spectre de la tension de sortie $V_{IN}$ pour $r=1$ et $F_e = 1500$ Hz.....	23
<b>Figure 1.13</b> le spectre de la tension de sortie $V_{IN}$ pour $r=1$ et $F_e = 3000$ Hz.....	23
<b>Figure 2.1</b> Schéma fonctionnel du TMS320F28335 .....	27
<b>Figure 2.2</b> Exemple de nomenclature d'un dispositif F28335 .....	28
<b>Figure 2.3</b> Schéma bloc de convertisseur analogique numérique (ADC) .....	31
<b>Figure 2.4</b> Schéma bloc de DMA .....	31
<b>Figure 2.5</b> Multiplexage des sources d'interruption en utilisant le bloc PIE.....	33
<b>Figure 2.6</b> Compare match et les changements sur la broche TxPWM .....	35
<b>Figure 2.7</b> Génération d'une porteuse asymétrique (dent de scie) par un temporisateur en mode de comptage continu .....	36
<b>Figure 2.8</b> Génération d'une porteuse symétrique (triangulaire) par un temporisateur en mode comptage/décomptage continu .....	37
<b>Figure 2.9</b> Temps mort avec une porteuse asymétrique.....	38
<b>Figure 2.10</b> Temps mort avec une porteuse symétrique.....	38
<b>Figure 2.11</b> Capture d'écran relative à l'IDE Code Composer Studio.....	39
<b>Figure 2.12</b> Les différentes couches de développement logiciel.....	40
<b>Figure 2.13</b> Bloc de paramètres ePWM .....	40
<b>Figure 2.14</b> Bloc de paramètres ADC .....	41
<b>Figure 2.15</b> Bloc de paramètres Digital output .....	41
<b>Figure 2.16</b> La carte à DSP TMS320F28335 .....	42
<b>Figure 2.17</b> Organigramme de Programme .....	43
<b>Figure 2.18</b> les fonctions logiques des interrupteurs de l'onduleur et les signaux de commande correspondants (a) La fonction logique F11 (b) La fonction logique F21 (C) La fonction logique F31 .....	44

## Liste des Abréviations :

MLI : Modulation de Largeur d'Impulsion.

DC/AC: Direct Current /Alternative Current.

SVM : Space Vector Modulation.

DSP: Digital Signal Processor.

CPU: Central Processing Unit.

ADC: Analog Digital Converter.

EPWM: Enhanced Pulse Width Modulation.

HRPWM: High Resolution Pulse Width Modulation.

McBSP : Multichannel Buffered Serial Port.

PIE: Peripheral Interrupt Expansion.

NMI: Nonmaskable Interrupt.

INTM: The Global Interrupt Mask Bit.

GPIO: General Purpose Input Output.

---

# **Introduction générale**

---

## **Introduction générale :**

Les onduleurs de tension occupent de nos jours une place importante dans les chaînes industrielles et les tractions électriques mais aussi dans le domaine des énergies renouvelables qui nécessitent des convertisseurs statiques pour les exploiter. L'un des principaux problèmes liés aux onduleurs, et qui dépend de la stratégie de commande utilisée, est celui de la présence inévitable d'harmoniques dans la tension et le courant de sortie. Ces harmoniques ont des effets néfastes que ce soit sur les moteurs ou bien sur les réseaux électriques comme les pulsations du couple et la saturation du circuit magnétique, auxquels s'ajoutent les interférences avec les systèmes de communications/télécommunications.

Les stratégies de commande des onduleurs ont connu une évolution considérable en ce qui concerne la complexité, l'efficacité, la souplesse, la fiabilité de ces stratégies et la qualité du courant et de la tension de sortie, tout en gardant le contrôle simultané de la fréquence et de l'amplitude du fondamental de la tension de sortie. Cette évolution s'est appuyée, d'une part sur la transition énergétique vers les énergies renouvelables qui a poussé les travaux de recherche pour répondre aux problématiques dans ce sujet, d'autre part, sur le développement des calculateurs numériques qui a facilité les calculs complexes et l'implémentation de ces stratégies.

C'est dans ce contexte que s'inscrit notre travail dont le but est d'implémenter la MLI vectorielle (SVPWM) pour la commande d'un onduleur triphasé. Ainsi, nous avons organisé ce document en deux chapitres :

Dans le premier chapitre, nous allons décrire la modulation vectorielle et la détermination des instants de commutation ainsi que les fonctions logiques des interrupteurs.

Le deuxième chapitre concerne la description de la carte DSP TMS320F28335 et ses modules essentiels dédiée à la commande numérique ainsi que la liaison entre les outils de développement (Matlab/Simulink, Code Composer Studio) pour la génération des codes et l'implémentation de la MLI vectorielle sur cette carte.

Enfin, notre mémoire se clôture avec une conclusion générale où nous résumons le travail effectué et les résultats obtenus.

---

# **Chapitre 1 : La MLI Vectorielle**

---

## 1.1 Introduction

La forme de la tension de sortie d'un onduleur de tension dépend essentiellement de la méthode utilisée pour la commande de l'onduleur. Généralement on utilise des commandes par modulation de largeurs d'impulsions. On rencontre plusieurs stratégies de ce type de commande, telles que la modulation triangulo-sinusoidale, la modulation par hystérésis, et la modulation vectorielle.

La modulation vectorielle est reconnue comme étant la plus performante et la plus utilisée.

Tout d'abord, elle génère moins d'harmoniques avec une utilisation efficace du bus d'alimentation continue.

De plus, Elle facilite l'implémentation sur un circuit numérique, surtout avec le développement récent des microprocesseurs et l'accroissement de leurs puissances de calcul.

La modulation vectorielle (SVM) est, aussi, une méthode très flexible. Elle peut être adaptée aux convertisseurs DC/AC de structure multiniveaux. En revanche, L'algorithme de cette modulation devient de plus en plus compliqué lors de son application sur les onduleurs multiniveaux.

Dans ce chapitre, nous allons décrire et simuler la technique de modulation vectorielle pour commander un onduleur triphasé à deux niveaux.

## 1.2 La structure d'un onduleur à deux niveaux

L'onduleur de tension triphasé à deux niveaux (Figure 1.1) est composé de trois bras. Chaque bras a deux interrupteurs bidirectionnels, qui fonctionnent de manière complémentaire pour éviter le court circuit de la source de tension continue à l'entrée de l'onduleur.

L'interrupteur bidirectionnel peut être réalisé par un transistor de puissance en antiparallèle avec une diode de puissance.

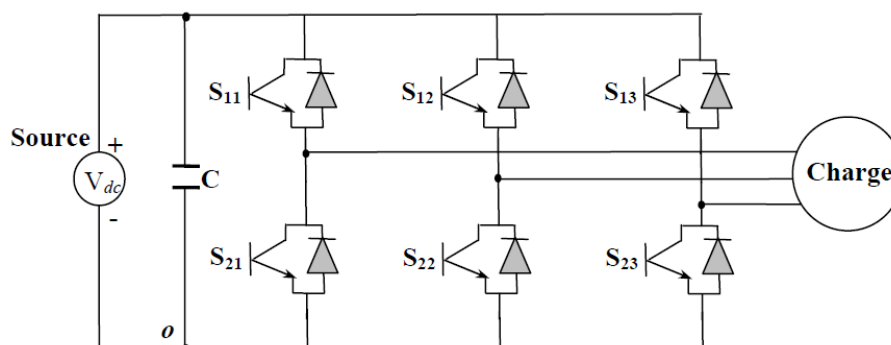


Figure 1.1 Structure d'un onduleur triphasé à deux niveaux [1].

### 1.3 Etats d'un bras de l'onduleur

Chaque bras de l'onduleur à deux états possibles

**Etat P** : L'interrupteur du haut  $S_{x1}(x = 1,2,3)$  est fermé, tandis que l'interrupteur du bas  $S_{x2}(x = 1,2,3)$  est ouvert.

**Etat N** : L'interrupteur du bas  $S_{x2}(x = 1,2,3)$  est fermé, tandis que l'interrupteur du haut  $S_{x1}(x = 1,2,3)$  est ouvert.

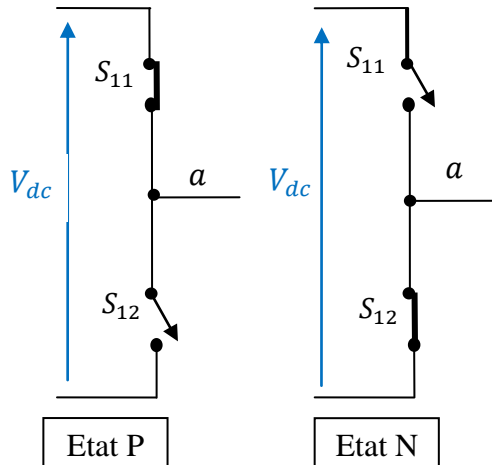


Figure 1. 2 Fonctionnement d'un bras d'onduleur

Cet étage d'onduleur à deux entrées et trois sorties possède  $2^3 = 8$  combinaisons possibles des états des interrupteurs qui sont (PPP, PPN, PNN, PNP, NNN, NNP, NPP, NPN) et cela tout en évitant le court-circuit de la tension continue fictive  $V_{dc}$ .

La configuration de l'onduleur pour chacune de ces états est représentée dans la (Figure 1.3) :

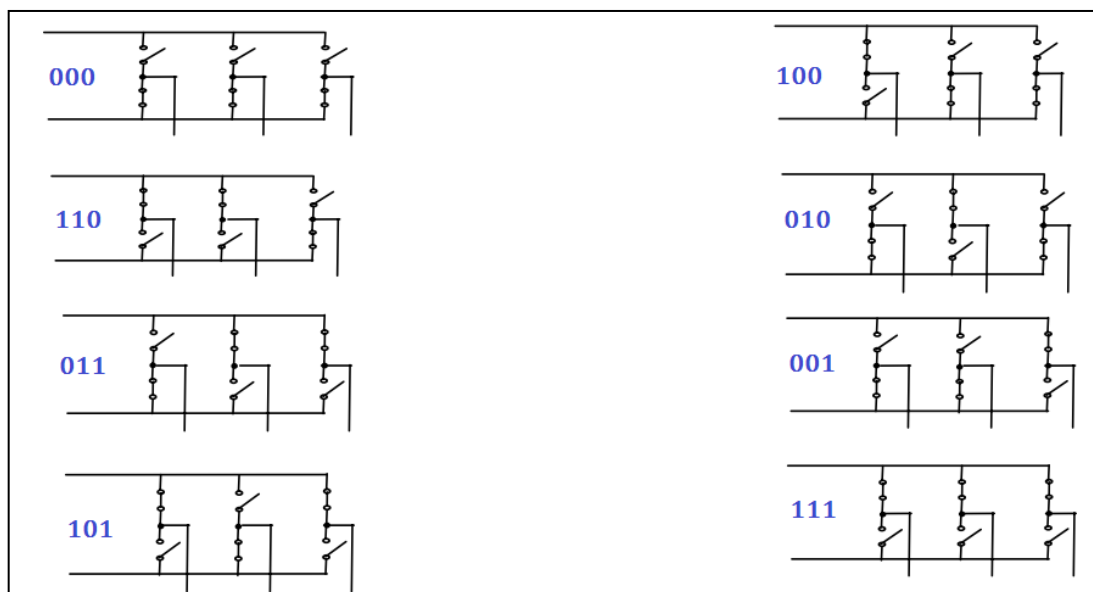


Figure 1.3 les huit états (configurations) d'un onduleur triphasé à deux niveaux.

### 1.4 Tensions de sortie d'un onduleur

En utilisant le schéma de la Figure 1.1, qui illustre la structure d'un onduleur triphasé de tension à deux niveaux. Les tensions triphasées de sorties par rapport à la référence de la source continue (o) peuvent être exprimées par :

$$\begin{pmatrix} v_{10} \\ v_{20} \\ v_{30} \end{pmatrix} = \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \end{pmatrix} \cdot V_{dc} \quad (1.1)$$

$V_{dc}$  étant la tension continue d'alimentation de l'onduleur.

Les tensions composées entre phases sont données par :

$$\begin{pmatrix} v_{12} \\ v_{23} \\ v_{31} \end{pmatrix} = \begin{pmatrix} v_{10} - v_{20} \\ v_{20} - v_{30} \\ v_{30} - v_{10} \end{pmatrix} \quad (1.2)$$

En supposant que la charge triphasée est équilibrée, on déduit les tensions triphasées de sortie par rapport au point neutre de la charge triphasée :

$$\begin{pmatrix} v_{12} \\ v_{23} \\ v_{31} \end{pmatrix} = \begin{pmatrix} v_{10} - v_{20} \\ v_{20} - v_{30} \\ v_{30} - v_{10} \end{pmatrix} = \begin{pmatrix} F_{11} - F_{12} \\ F_{12} - F_{13} \\ F_{13} - F_{11} \end{pmatrix} \cdot V_{dc} \quad (1.3)$$

En supposant que la charge triphasée est équilibrée, on déduit que les tensions triphasées par rapport au point neutre de la charge triphasée :

$$\begin{pmatrix} V_{1N} \\ V_{2N} \\ V_{3N} \end{pmatrix} = \frac{1}{3} \cdot \begin{pmatrix} v_{12} - v_{31} \\ v_{23} - v_{12} \\ v_{31} - v_{23} \end{pmatrix} = \frac{1}{3} \cdot \begin{pmatrix} 2F_{11} - F_{12} - F_{13} \\ -F_{11} + 2F_{12} - F_{13} \\ -F_{11} - F_{12} + 2F_{13} \end{pmatrix} \cdot V_{dc} \quad (1.4)$$

Les tensions triphasées par rapport au neutre de la source  $v_{10}, v_{20}, v_{30}$  peuvent prendre uniquement deux valeurs :  $V_{dc}$  et 0, c'est à l'origine de l'appellation d'onduleur à deux niveaux. [14]

Les tensions composées  $v_{12}, v_{23}, v_{31}$  peuvent prendre trois valeurs :  $V_{dc}, 0, -V_{dc}$ .

Les tensions triphasées par rapport au neutre de la charge peuvent prendre cinq valeurs :  $(-2/3) \cdot V_{dc}, (-1/3) \cdot V_{dc}, 0, (1/3) \cdot V_{dc}, (2/3) \cdot V_{dc}$ .

### 1.5 Principe du vecteur spatial (Space Vector)

On définit le vecteur de tension spatiale (Space Vector)  $v_s$ , ce vecteur représente les trois tensions triphasées  $v_{10}, v_{20}, v_{30}$ . Ce vecteur est défini ainsi :

$$v_s = v_{10} \cdot e^{j0} + v_{20} \cdot e^{-j \cdot 2\frac{\pi}{3}} + v_{30} \cdot e^{j \cdot 2\frac{\pi}{3}}$$

$$v_s = v_{1N} \cdot e^{j0} + v_{N0} \cdot e^{j0} + v_{2N} \cdot e^{-j \cdot 2\frac{\pi}{3}} + v_{N0} \cdot e^{-j \cdot 2\frac{\pi}{3}} + v_{3N} \cdot e^{j \cdot 2\frac{\pi}{3}} + v_{N0} \cdot e^{j \cdot 2\frac{\pi}{3}}$$



$$v_{N0} \cdot e^{j0} + v_{N0} \cdot e^{-j \cdot \frac{2\pi}{3}} + v_{N0} \cdot e^{j \cdot \frac{2\pi}{3}} = 0$$

D'où on a :

$$v_s = v_{1N} \cdot e^{j0} + v_{2N} \cdot e^{-j \cdot \frac{2\pi}{3}} + v_{3N} \cdot e^{j \cdot \frac{2\pi}{3}} \quad (1.5)$$

Afin d'obtenir les composantes de ce vecteur dans le repère biphasé d-q, on effectue la transformation (passage) du plan triphasé au plan biphasé d-q, en utilisant la matrice de Clark (conservation de l'amplitude), ce choix judicieux est effectué afin que le vecteur spatial  $v_s$  ait la même amplitude que le vecteur représentant la tension sinusoïdale d'une phase  $V_{1N}$ ,  $V_{2N}$  ou  $V_{3N}$ .

On effectue donc le passage du plan triphasé au plan d-q :

$$\begin{pmatrix} v_d \\ v_q \end{pmatrix} = \frac{2}{3} \cdot \begin{pmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{pmatrix} \times \begin{pmatrix} V_{1N} \\ V_{2N} \\ V_{3N} \end{pmatrix} \quad (1.6)$$

Le vecteur  $v_s$  s'écrit dans le plan d-q ainsi :

$$v_s = v_d + jv_q \quad (1.7)$$

Ce vecteur  $v_s$  peut prendre huit positions discrètes dans le plan d-q, chacune des positions est relative à une des huit configurations possibles des interrupteurs de l'onduleur, six configurations actives (PNN, PPN, NPN, NPP, NNP et PNP), ces six positions actives divisent le plan d-q en six secteurs triangulaires égaux d'un angle égal à ( $60^\circ$  ou  $\frac{\pi}{3}$ ), et elles définissent aussi les six sommets de l'hexagone, les deux positions restantes correspondent aux configurations nulles PPP et NNN, situées au centre de l'hexagone. Les différentes positions du vecteur  $v_s$  sont représentées dans la Figure suivante 1.4 :

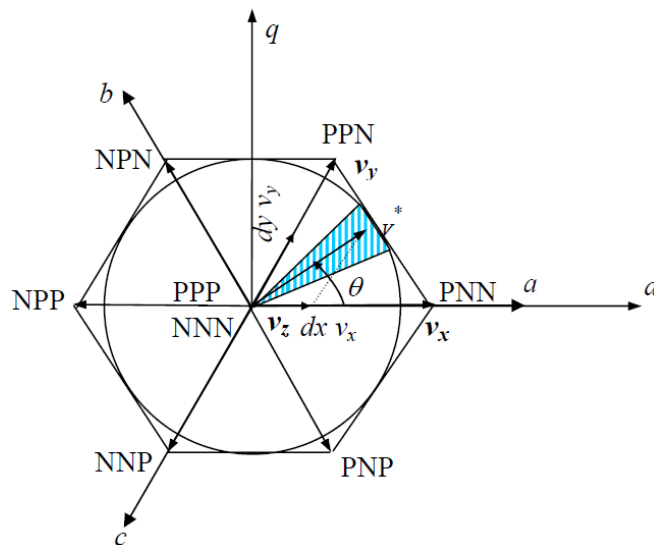


Figure 1.4 Diagramme représentant les positions vectorielles de l'onduleur à deux niveaux, relatives à la stratégie SVM [1]

Tableau 1.1 Etat des Interrupteurs et vecteurs de commutation de l'étage Onduleur

Vecteurs	$\begin{bmatrix} S_{11} & S_{21} & S_{31} \\ S_{12} & S_{22} & S_{32} \end{bmatrix}^t$	$V_{1N}$	$V_{2N}$	$V_{3N}$	$V_{sd}$	$V_{sq}$	$\theta_s$	$ V_s $
$V_1$ PNN	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}^t$	$\frac{2}{3}V_{dc}$	$-\frac{1}{3}V_{dc}$	$-\frac{1}{3}V_{dc}$	$\frac{2}{3}V_{dc}$	0	0	$\frac{2}{3}V_{dc}$
$V_2$ PPN	$\begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}^t$	$\frac{1}{3}V_{dc}$	$\frac{1}{3}V_{dc}$	$-\frac{2}{3}V_{dc}$	$\frac{1}{3}V_{dc}$	$\frac{1}{\sqrt{3}}V_{dc}$	$\frac{\pi}{3}$	$\frac{2}{3}V_{dc}$
$V_3$ NPN	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}^t$	$-\frac{1}{3}V_{dc}$	$\frac{2}{3}V_{dc}$	$-\frac{1}{3}V_{dc}$	$-\frac{1}{3}V_{dc}$	$\frac{1}{\sqrt{3}}V_{dc}$	$\frac{2\pi}{3}$	$\frac{2}{3}V_{dc}$
$V_4$ NPP	$\begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}^t$	$-\frac{2}{3}V_{dc}$	$\frac{1}{3}V_{dc}$	$\frac{1}{3}V_{dc}$	$-\frac{2}{3}V_{dc}$	0	$\pi$	$\frac{2}{3}V_{dc}$
$V_5$ NNP	$\begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}^t$	$-\frac{1}{3}V_{dc}$	$-\frac{1}{3}V_{dc}$	$\frac{2}{3}V_{dc}$	$-\frac{1}{3}V_{dc}$	$-\frac{1}{\sqrt{3}}V_{dc}$	$\frac{4\pi}{3}$	$\frac{2}{3}V_{dc}$
$V_6$ PNP	$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}^t$	$\frac{1}{3}V_{dc}$	$-\frac{2}{3}V_{dc}$	$\frac{1}{3}V_{dc}$	$\frac{1}{3}V_{dc}$	$-\frac{1}{\sqrt{3}}V_{dc}$	$\frac{5\pi}{3}$	$\frac{2}{3}V_{dc}$
$V_0$ NNN	$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}^t$	0	0	0	0	0	0	0
$V_7$ PPP	$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}^t$	0	0	0	0	0	0	0

On constate que :

- Le module du vecteur spatial de tension de sortie est constant pour tous les vecteurs  $V_1$  a  $V_6$  et il vaut  $\frac{2}{3}V_{dc}$ ,
- Le module du vecteur spatial de tension est nul pour les vecteurs zéro  $V_0$  et  $V_7$ .

## 1.6 Vecteurs de tension de référence

L'objectif de la commande SVM de l'onduleur est de rendre les tensions de sortie  $v_{1N}, v_{2N}$  et  $v_{3N}$  les plus proches possibles des tensions sinusoïdales de référence  $v_{1ref}, v_{2ref}$  et  $v_{3ref}$ .

Afin d'atteindre cet objectif, on définit le vecteur de tension  $v_{sref}$  à partir des vecteurs de tensions de tension de référence  $v_{1ref}, v_{2ref}$  et  $v_{3ref}$  :

$$v_{sref} = v_{1ref} \cdot e^{j0} + v_{2ref} \cdot e^{-j \cdot \frac{2\pi}{3}} + v_{3ref} \cdot e^{j \cdot \frac{2\pi}{3}} \quad (1.8)$$

$$v_{sref} = v_{dref} + j \cdot v_{qref} \quad (1.9)$$

$$v_{sref} = r \cdot V_{max} \cdot e^{j\theta} \quad (1.10)$$

Avec :

$v_{d\ ref}, v_{q\ ref}$  : Composantes du vecteur  $v_{s\ ref}$  dans le plan d-q.

$r$  : le taux de modulation ( $0 \leq r \leq 1$ ).

$\theta$  : position angulaire du vecteur  $v_{s\ ref}$ .

Le vecteur  $v_{s\ ref}$  tourne dans le plan d-q à une vitesse angulaire égale à la pulsation électrique des tensions de référence  $v_{1\ ref}, v_{2\ ref}$  et  $v_{3\ ref}$ , comme illustré dans la figure 1.4, dans laquelle on a représenté une position arbitraire  $\theta$  du vecteur  $v_{s\ ref}$ , et à partir de cette angle  $\theta$ , on déduit le secteur contenant le vecteur  $v_{s\ ref}$  de la manière suivante :

$$S = \begin{cases} 1 & \text{si } 0 \leq \theta \leq \pi/3 \\ 2 & \text{si } \pi/3 \leq \theta \leq 2.\pi/3 \\ 3 & \text{si } 2.\pi/3 \leq \theta \leq \pi \\ 4 & \text{si } \pi \leq \theta \leq 4.\pi/3 \\ 5 & \text{si } 4.\pi/3 \leq \theta \leq 5.\pi/3 \\ 6 & \text{si } 5.\pi/3 \leq \theta \leq 2.\pi \end{cases} \quad (1.11)$$

Les interrupteurs de l'onduleur doivent être commandés de telle façon que le vecteur de tension de référence  $v_{s\ ref}$  coïncide avec la moyenne sur une période de commutation  $T_s$ , du vecteur tension de sortie, généré par les trois vecteurs  $v_x, v_y, v_z$ , représentant les sommets du secteur triangulaire contenant le vecteur de tension de référence  $v_{s\ ref}$  [1] :

$$\bar{v}_s = v_{s\ ref} = r \cdot V_{max} \cdot e^{i\alpha} = d_x \cdot v_x + d_y \cdot v_y + d_z \cdot v_z \quad (1.12)$$

Avec ( $\alpha$ ) Position angulaire du vecteur  $v_{s\ ref}$  à l'intérieur du secteur :  $\Leftrightarrow \alpha = \theta \cdot [\pi/3]$

$$d_x = \frac{T_x}{T_s}, d_y = \frac{T_y}{T_s}, d_z = \frac{T_z}{T_s} \quad (1.13)$$

$T_s$  étant la période de commutation de la stratégie SVM.

$T_x$  : temps d'application du vecteur actif  $v_x$  pendant une période de commutation  $T_s$ .

$T_y$  : temps d'application du vecteur actif  $v_y$  pendant une période de commutation  $T_s$ .

$T_z$  : temps d'application des deux vecteurs nulles  $v_z = (V_0, V_7)$  pendant une période de commutation  $T_s$ .

$$\text{Tel que : } \begin{cases} d_x + d_y + d_z = 1 \\ T_x + T_y + T_z = T_s \end{cases} \quad (1.14)$$

Comme illustré sur la Figure 1.4, l'ensemble des lieux possibles pour le vecteur de tension de référence  $v_{s\ ref}$  est constitué par la forme hexagonale délimitée par les six vecteurs d'états discrets de la tension de sortie de l'onduleur  $v_s$ . Afin d'éviter les harmoniques de tension qui résultent de la forme non circulaire de cette hexagone, on impose donc au vecteur de tension de référence  $v_{s\ ref}$  de se positionner à l'intérieur du cercle délimitée par l'hexagone comme ainsi indiqué sur la Figure 1.4.

l'amplitude d'un de vecteurs discrets ( $V_1, \dots, V_6$ ) est de  $(2/3 \cdot V_{dc})$  (voir tableau 1.1). Et donc le vecteur de référence  $v_{s\ ref}$  ne doit pas excéder :

$$v_{s\ ref(max)} = \cos \frac{\pi}{6} \cdot (2/3 \cdot V_{dc}) = \frac{\sqrt{3}}{2} \cdot \left(\frac{2}{3} \cdot V_{dc}\right) = 1/\sqrt{3} \cdot V_{dc}.$$

L'amplitude de la tension simple maximale de sortie est ainsi déduite, et est égale à  $1/\sqrt{3} \cdot V_{dc}$ , ainsi Le taux de modulation est définie par:  $r = \frac{\sqrt{3} \cdot v_s}{V_{dc}}$ .

On en déduit ainsi que la tension composée maximale de sortie (entre lignes) est égale à  $V_{dc}$ , et sa valeur efficace est égale à  $\frac{V_{dc}}{\sqrt{2}} = 0.707 V_{dc}$ .

### 1.7 Période de commutation

La période de commutation  $T_s$  doit représenter une fraction très minime de la période T des tensions de référence  $v_{1\ ref}, v_{2\ ref}, v_{3\ ref}$ . La fréquence de commutation  $(1/T_s) = f_s$  doit être beaucoup plus élevée par rapport à la fréquence de la tension et du courant de sortie de l'onduleur  $f = (1/T)$ . La qualité de la tension de sortie de l'onduleur dépend en grande partie de la fréquence de commutation de l'onduleur. Cette tension de sortie s'améliore proportionnellement avec l'augmentation de la fréquence de commutation de la stratégie, et ainsi avec l'augmentation de la fréquence de commutation des interrupteurs. Cependant, une augmentation exagérée de la fréquence de commutation peut conduire à un échauffement des interrupteurs de l'onduleur, et peut conduire donc à des pertes excessives appelées « pertes de commutation » [1].

### 1.8 Séquences des interrupteurs de l'onduleur suivant la stratégie

Les vecteurs  $v_x, v_y, v_z$ , utilisés pour décomposer instantanément le vecteur de tensions de référence  $v_{s\ ref}$ , représentent des séquences d'états de l'onduleur. Ces vecteurs dépendent de la position du vecteur de tension  $v_{s\ ref}$  dans le plan d-q, c'est-à-dire le secteur dans lequel est contenu le vecteur  $v_{s\ ref}$ . Si on prend comme référence le cas illustré dans la Figure 1.4, on a le vecteur  $v_{s\ ref}$  qui est dans le secteur 1 (S=1, selon l'équation (1.11)), ainsi le vecteur  $v_x$  est généré par la séquence PNN (appelée alors état X), le vecteur  $v_y$  est généré donc par la séquence PPN (appelée état Y), aussi le vecteur  $v_z$  est généré par la séquence NNN (appelée état  $Z_1$ ) ou la séquence PPP (appelée état  $Z_2$ ). La Figure 1.5 indique chacun des états X, Y, Z pour chacun des six secteurs de la stratégie SVM [1].

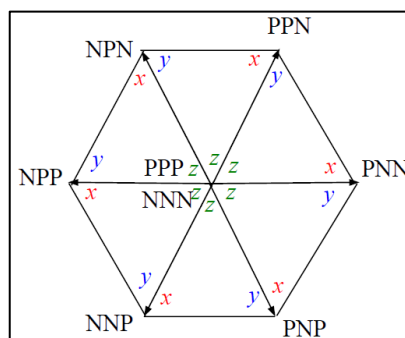


Figure 1.5 : Diagramme illustrant les états X, Y, Z, pour chacun des six secteurs de la SVM [1].

Afin de déduire les rapports  $d_x, d_y, d_z$ , on décompose l'équation 1.14 (projection sur les axes d-q), après simplifications, on obtient les équations suivantes :

$$\begin{cases} \frac{\sqrt{3}}{2} \cdot r \cdot \cos \alpha = d_x + \frac{1}{2} \cdot d_y \\ \frac{\sqrt{3}}{2} \cdot r \cdot \sin \alpha = \frac{\sqrt{3}}{2} \cdot d_y \end{cases} \quad (1.15)$$

Ainsi la résolution des équations (1.14) et (1.15) nous conduit aux résultats suivants :

$$\begin{cases} d_x = r \cdot \sin\left(\frac{\pi}{3} - \alpha\right) \\ d_y = r \cdot \sin \alpha \\ d_z = 1 - d_x - d_y \end{cases} \quad (1.16)$$

Les équations (1.16) peuvent s'appliquer à n'importe quel secteur, les résultats restent similaires puisque le repère stationnaire d-q peut être orienté sur n'importe quelle position discrète du vecteur  $v_s$ .

Dans le but d'approximer au maximum possible les tensions de sortie de l'onduleur ( $v_1, v_2, v_3$ ) aux tensions de référence ( $v_{s\ ref\ 1}, v_{s\ ref\ 2}, v_{s\ ref\ 3}$ ), durant chaque période de commutation  $T_s$ , on applique l'état X pendant une durée de temps  $d_x \cdot T_s$  ainsi que l'état Y pendant une durée de temps  $d_y \cdot T_s$ .

Cependant l'état Z peut prendre deux séquences :  $Z_1 = \text{NNN}$  et  $Z_2 = \text{PPP}$ , les deux séquences génèrent des tensions de sortie nulles, il est possible donc de choisir l'une ou l'autre (pendant la durée  $d_z \cdot T_s$ ) afin de générer la tension de sortie voulue. Ce choix va donc obéir à plusieurs critères, dont l'amélioration de la forme de la tension de sortie et la minimisation du nombre de commutation des interrupteurs pendant une période de commutation, et donc la minimisation de puissance par commutation, ce qui exige l'application de la séquence suivante pendant chaque période de commutation  $T_s$  [1] :

$$\begin{cases} [Z_1 - X - Y - Z_2 | Z_2 - Y - X - Z_1] \text{ Pour } S = 1 \text{ ou } 3 \text{ ou } 5 \\ [Z_1 - Y - X - Z_2 | Z_2 - X - Y - Z_1] \text{ Pour } S = 2 \text{ ou } 4 \text{ ou } 6 \end{cases} \quad (1.17)$$

Cet enchaînement d'états spécifiques consiste à [1] :

- Appliquer les trois quatre états X, Y,  $Z_1, Z_2$  dans un certain ordre dans la première demi-période de commutation, et les appliquer dans un ordre inverse dans la seconde demi-période.
- Il faut appliquer les états  $Z_1$  et  $Z_2$  par alternance.
- D'utiliser le même état zéro ( $Z_1$  ou  $Z_2$ ) au début et à la fin de chaque période de commutation  $T_s$ .
- Inverser l'ordre des états X et Y par alternance entre les six secteurs.

A partir de cette succession d'états, on peut déduire les fonctions logiques associées aux interrupteurs de l'onduleur  $F_{ij}$  ainsi que leurs formes durant toute une période de commutation  $T_s$  et dans les différents secteurs constituant le diagramme fonctionnel illustré

dans la Figure 1.4. On constate alors que chaque interrupteur ne change son état que deux fois (2 commutations) durant chaque période de commutation.

La Figure 1.6 illustre les formes des fonctions logiques des interrupteurs dans les six secteurs et durant une période complète de commutation  $T_s$ .

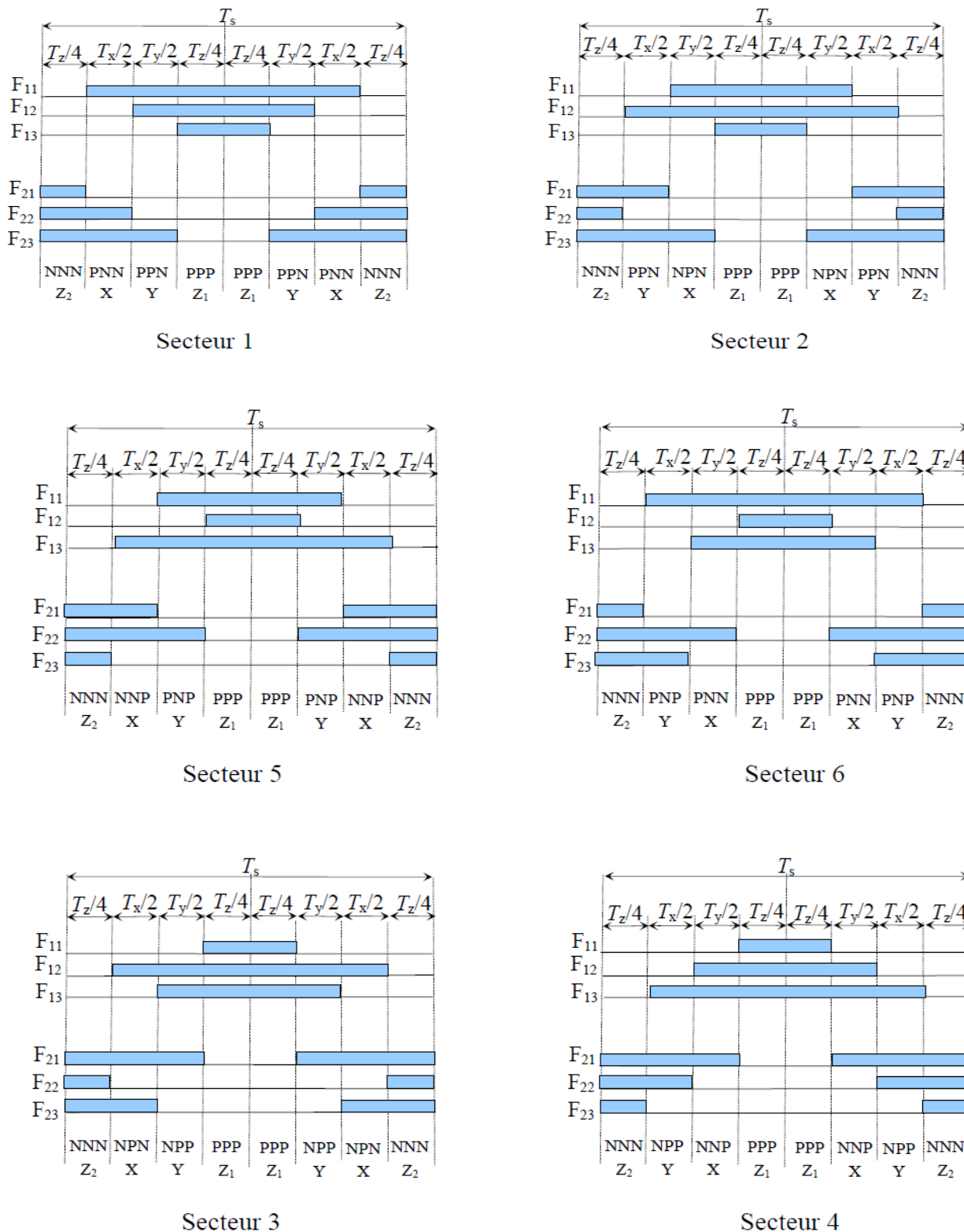


Figure 1.6 Formes des fonctions logiques  $F_{ij}$  des interrupteurs pour les six secteurs et durant une période de commutation  $T_s$  [1].

## 1.9 Résultats de simulation

### 1.9.1 Variation de taux de modulation $r$

Dans les résultats de simulation des Figures (1.7, 1.8, 1.9) nous avons fixé la fréquence de commutation à  $F_e = 1000$  Hz et nous avons varié le taux de modulation  $r$  pour voir l'effet de ce dernier sur la tension simple de sortie  $V_{IN}$  et son spectres d'harmoniques.

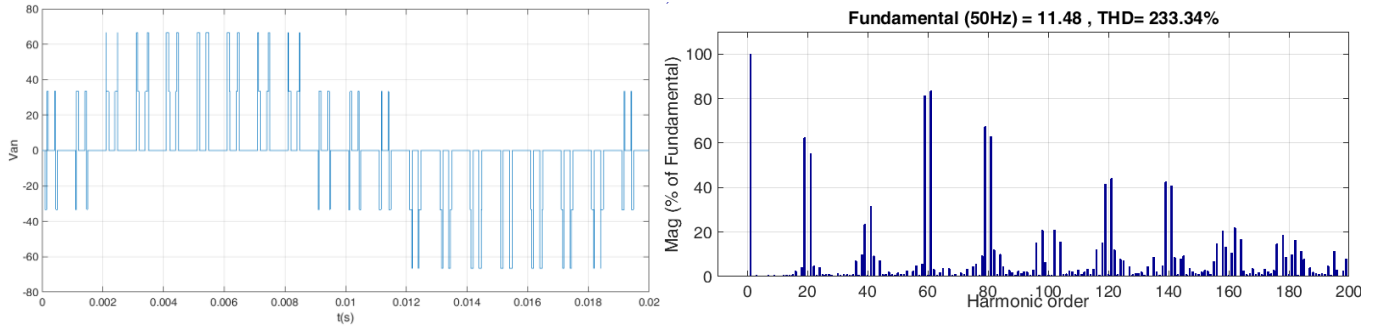


Figure 1.7 le spectre de la tension de sortie  $V_{IN}$  pour  $r=0.2$  et  $F_e = 1000$  Hz.

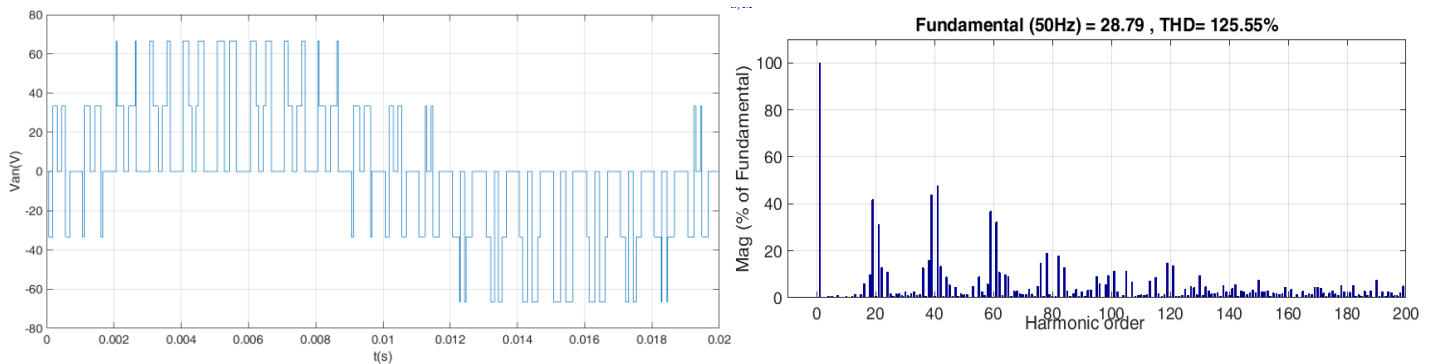


Figure 1.8 le spectre de la tension de sortie  $V_{IN}$  pour  $r=0.5$  et  $F_e = 1000$  Hz.

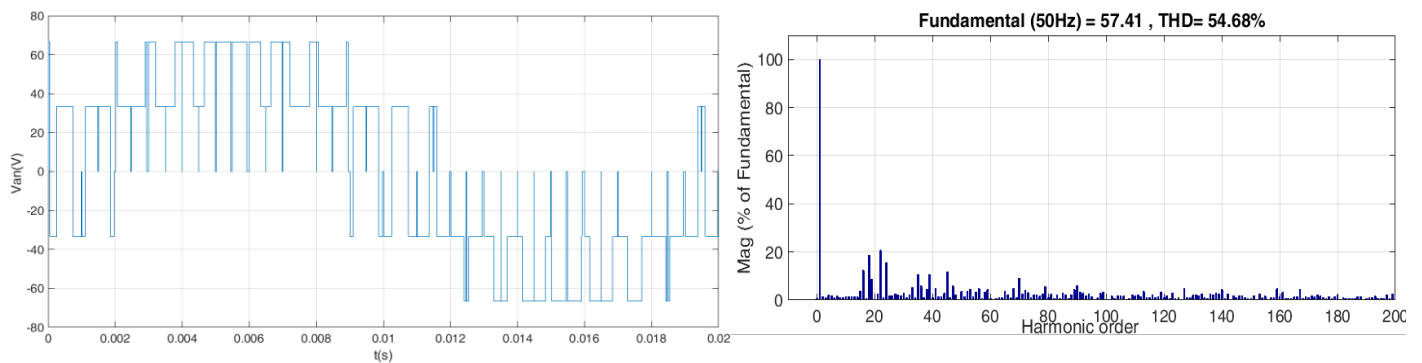


Figure 1.9 le spectre de la tension de sortie  $V_{IN}$  pour  $r=1$  et  $F_e = 1000$  Hz.

Pour voir l'effet de taux de modulation  $r$  sur le fondamental de la tension simple  $V_{1N}$ , nous avons tracé la courbe de réglage (voir Figure 1.10).

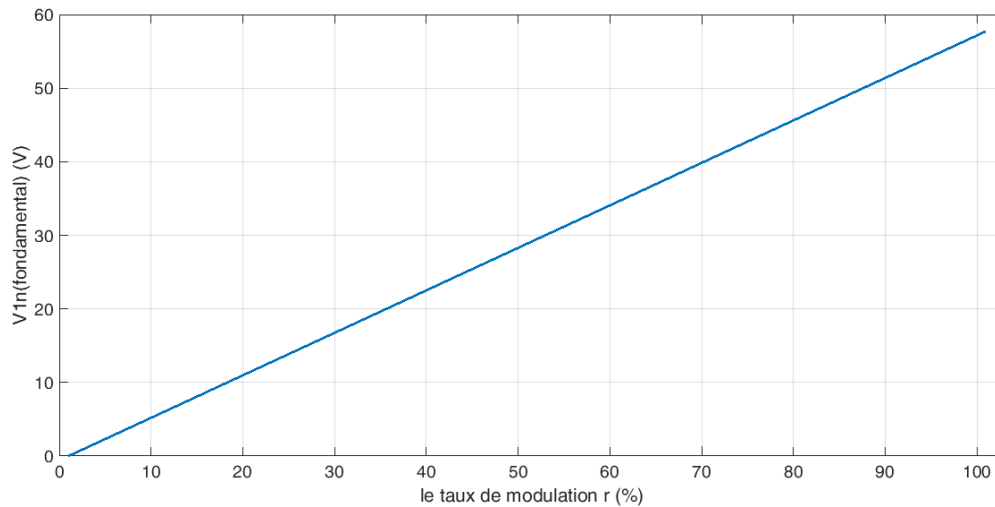


Figure 1.10 courbe de réglage

Les Figures 1.7, 1.8, 1.9, 1.10 montrent que :

- Lorsque on change le taux de modulation  $r$ , on remarque que la valeur de tension du fondamentale atteint la valeur désirée suivant l'équation de réglage  $V_{1n} = \frac{r \times V_{dc}}{\sqrt{3}}$ , ainsi la courbe de réglage a une forme linéaire.
- le THD de la tension de sortie est inversement proportionnel au taux de modulation  $r$ .
- Les harmoniques non négligeables se regroupent autour des harmoniques de rang multiple de 20 car la fréquence de commutation  $F_e = 20 \times 50 = 1000$  Hz.

### 1.9.2 Variation de la fréquence de commutation $F_e$

Dans les résultats de simulation des Figures (1.11, 1.12, 1.13) nous avons fixé le taux de modulation à  $r=1$  et nous avons varié la fréquence de commutation pour voir l'effet de cette dernière sur la tension simple de sortie  $V_{1N}$  et son spectres d'harmoniques.



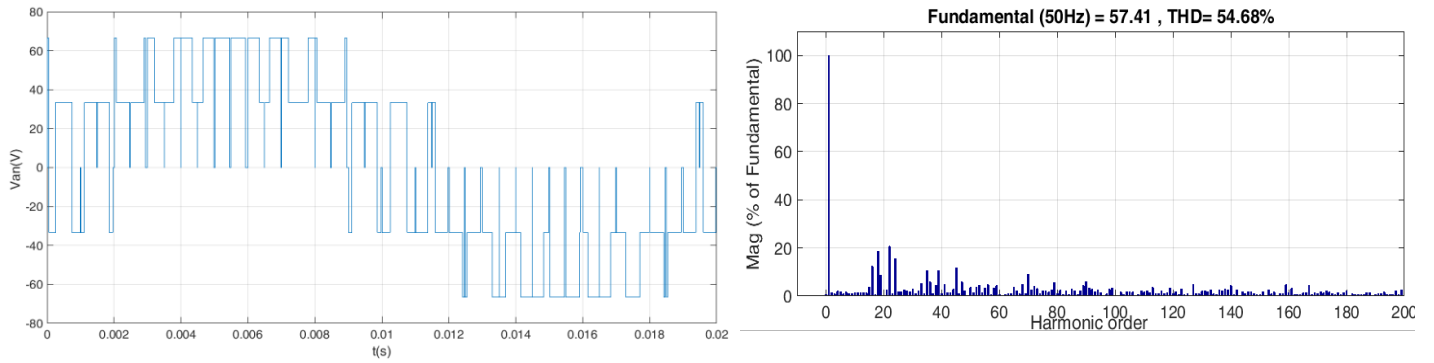


Figure 1.11 le spectre de la tension de sortie  $V_{1N}$  pour  $r=1$  et  $F_e = 1000$  Hz.

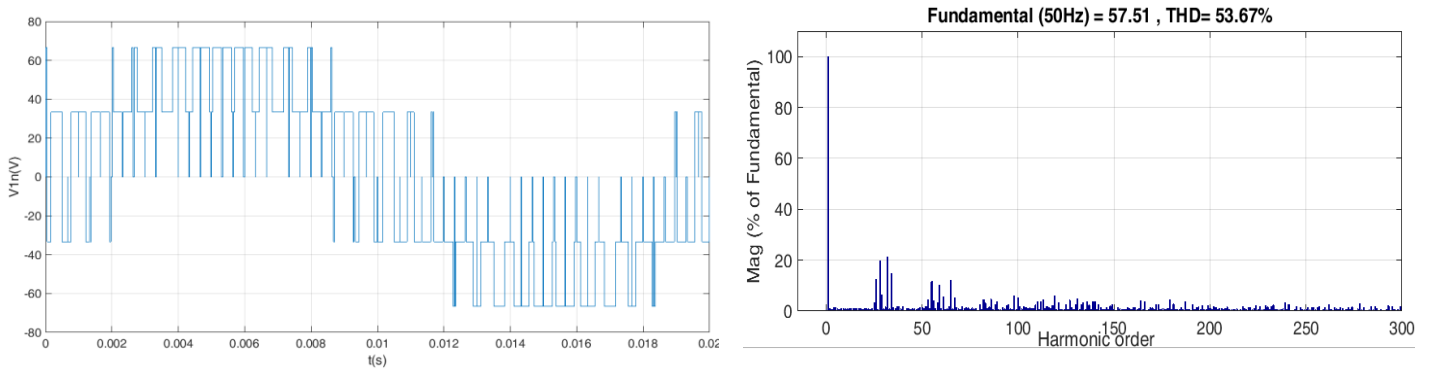


Figure 1.12 le spectre de la tension de sortie  $V_{1N}$  pour  $r=1$  et  $F_e = 1500$  Hz.

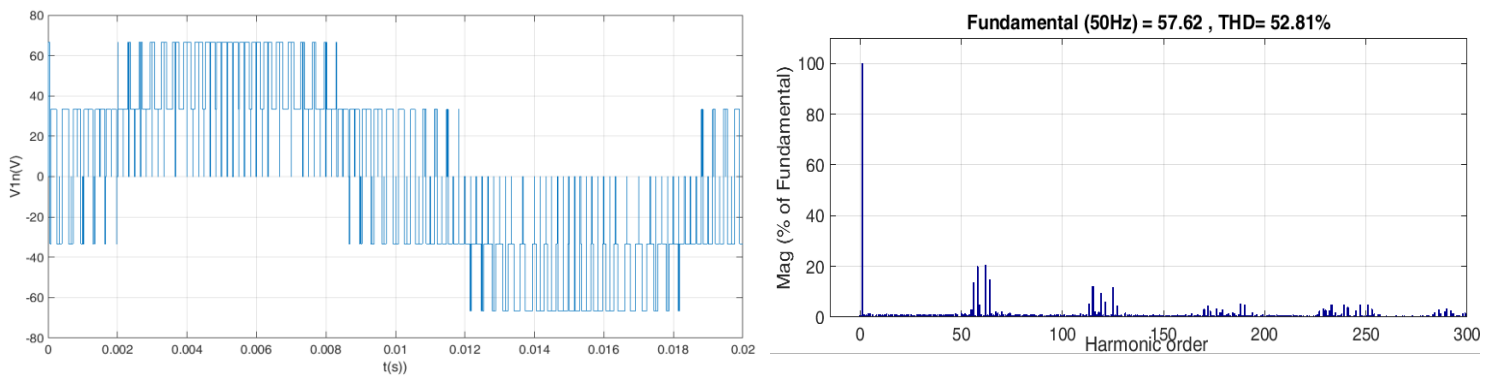


Figure 1.13 le spectre de la tension de sortie  $V_{1N}$  pour  $r=1$  et  $F_e = 6000$  Hz.

Les Figures 1.11, 1.12, 1.13 montrent que :

- La valeur maximale de la composante fondamentale désirée est atteinte et celle-ci ne dépend pas de *la fréquence de commutation* mais seulement de *r* (taux de modulation), tel

que  $V_{1n} = \frac{r \times V_{dc}}{\sqrt{3}}$ .

- Les harmoniques d'amplitude non négligeable peuvent être représentées par familles :

- La famille centrée sur l'harmonique  $(2n+1)m$  c'est-à-dire  $(m, 3m, 5m, \dots)$  et contenant :

L'harmonique de rang  $(2n+1)m$ .

Les harmoniques  $(2n+1)m-2$  et  $(2n+1)m+2$ .

Les harmoniques  $(2n+1)m-4$  et  $(2n+1)m+4$ .

- La famille centrée sur l'harmonique  $(2n)m$  c'est-à-dire  $(2m, 4m, 6m, \dots)$  et contenant :

Les harmoniques  $(2n)m-1$  et  $(2n)m+1$ .

Les harmoniques  $(2n)m-3$  et  $(2n)m+3$ .

- Le THD de la forme d'onde MLI tension de sortie est égal à 53% à degrés près et ne dépend pas de *la fréquence de commutation* (l'augmentation de *Fe* ne change pas la valeur de THD, mais permet de déplacer les harmoniques non négligeables vers les fréquences élevées et ainsi facilite leurs filtrage).

## 1.10 Conclusion

Dans ce chapitre, nous avons présenté la modulation vectorielle à deux niveaux (SVM à deux niveaux). Les simulations montrent que cette méthode génère une tension simple qui approxime une référence sinusoïdale par cinq valeurs de tension différentes. L'amplitude de son fondamental suit parfaitement la courbe de réglage.

---

## **Chapitre 2 : Implémentation de La MLI vectorielle sur DSP TMS320F28335**

---

## 2.1 Introduction

Le TMS320F28335 Delfino est un microcontrôleur numérique programmable de l'entreprise Texas Instruments avec un processeur central C2XXX DSP (CPU : Central Processing Unit) de 32-bit avec une précision à virgule fixe et flottante et des périphériques intégrées dans une seule pièce de silicone. Avec le processeur central et les périphériques de contrôle l'utilisateur peut concevoir un système de contrôle numérique compact et à faible coût ou bien élaborer les asservissements numériques en général.

La DSP F28335 offre 150 million d'instructions par seconde, Cette performance permet à l'utilisateur de calculer les paramètres en temps réel et est adaptée pour les algorithmes où une grande quantité de calculs rapides est exigée comme le cas de la commande d'une machine sans capteur.

Les périphériques de la DSP F28335 permettent de réaliser n'importe quelle commande numérique et leurs applications varient de la conversion analogique numérique jusqu'à la modulation à largeur d'impulsion(MLI) ainsi que d'autres applications assurées par les périphériques ci-dessous :

- Timer à usage général : 3 timers de 32 bit.
- Générateurs MLI.
- Convertisseur analogique numérique(ADC) : 16 entrées de 12 bits avec une conversion rapide.
- Interface Controller Area Network (CAN).
- Serial Peripheral Interface (SPI).
- Serial Communications Interface (SCI).
- Entrées-Sorties bidirectionnelles (GPIO).
- Watchdog timer (chien de garde).

La Figure 2.1 illustre le Schéma fonctionnel (Architecture) du TMS320F28335.

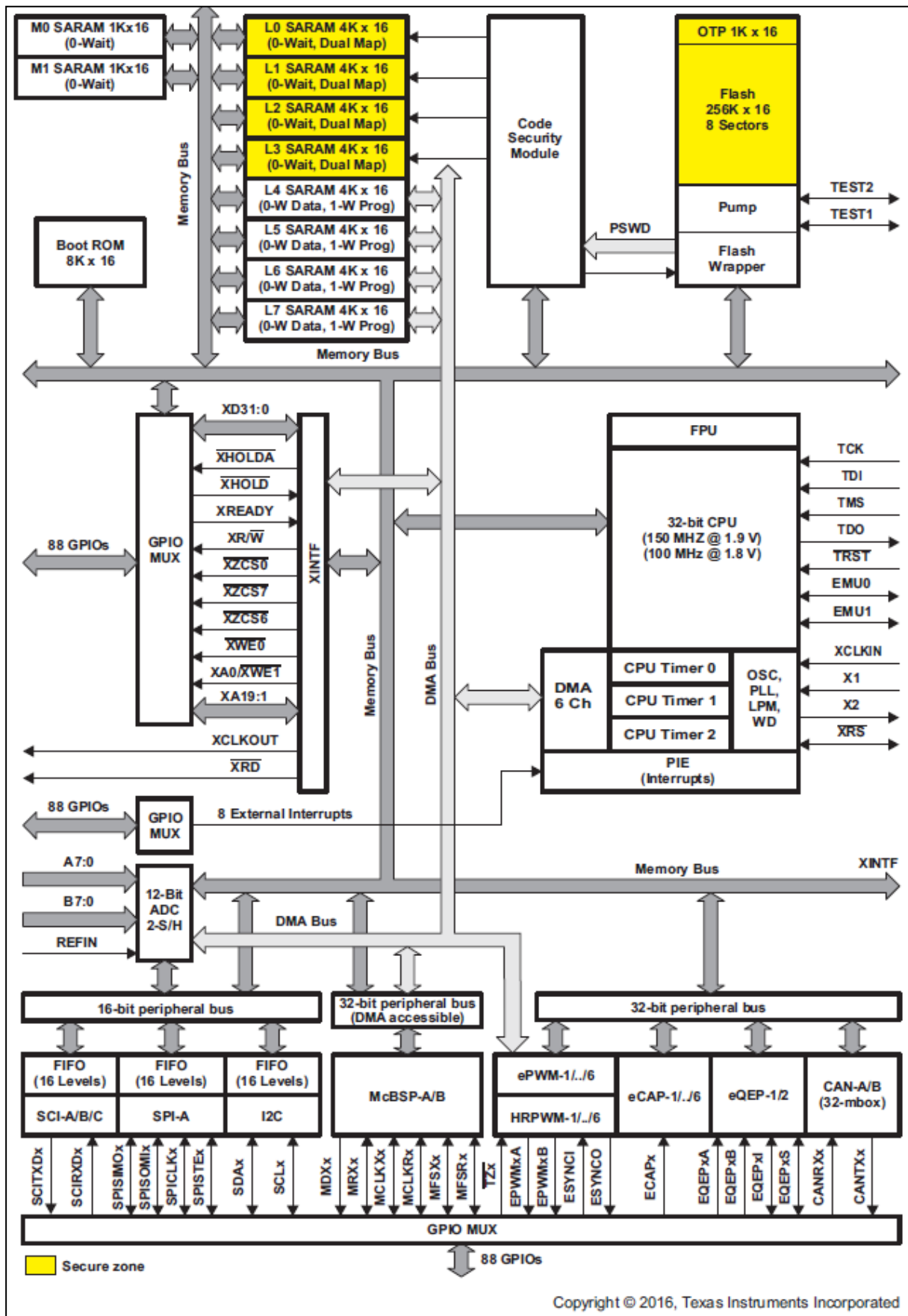


Figure 2.1 Schéma fonctionnel du TMS320F28335 [2].

### 2.1.2 Nomenclature de la famille TMS320

Pour désigner la phase de conception, TI assigne un préfixe pour le nombre de chaque dispositif, il existe 3 préfixes qui représentent l'évolution de la conception de produit à partir de l'« engineering prototype » (TMX/TMDX) jusqu'au produit final hautement qualifié (TMS/TMDS).

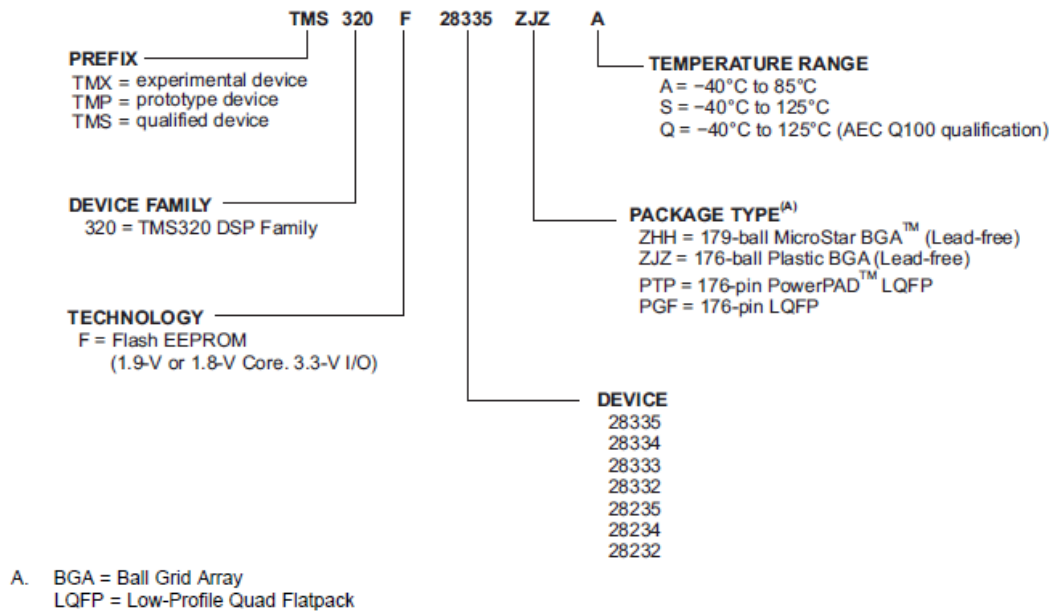


Figure 2.2 Exemple de nomenclature d'un dispositif F28335 [2].

### 2.1.3 Les périphériques

#### Convertisseur analogique numérique (The Analog-to-Digital Converter (ADC)) :

Le convertisseur analogique numérique est utilisé pour échantillonner un signal analogique et le convertir au format numérique. Par exemple, dans le cas de la commande d'un moteur en boucle fermée, il permet de mesurer le courant avec une résistance shunt au lieu d'utiliser un capteur de courant.

#### L'interface Control Area Network (CAN):

Ce circuit est utilisé pour la communication série avec des composants matériels externes. Ce composant (CAN) assure une grande intégrité de donnée et est destiné à l'utilisation dans un environnement qui présente des forts bruits comme l'automobile ou bien l'industrie.

#### Serial peripheral interface (SPI) and Serial communications interface (SCI):

L'interface SPI est un port de communication synchrone très rapide. Ce port est utilisé pour la communication avec les registres à décalage et les pilotes d'affichage.

La SCI est utilisée pour la communication asynchrone avec le CPU et autres périphériques, nous ne l'utiliserons pas dans le reste de notre travail.

#### Chien de garde (Watch dog) :

Le chien de garde (WD) est un périphérique qui assure la remise à zéro ou réinitialisation (reset) du system en cas d'un débordement. Par exemple, si le CPU subit des perturbations si le programme rentre dans une boucle infinie, le DSP est réinitialisé et le programme est remis

à son point de départ. Le WD assure donc le bon fonctionnement du système et augmente le rendement du CPU.

### **Les Entrées/Sorties Bidirectionnelles (I/O) :**

Les broches (pins) d'entrées/sorties peuvent fonctionner en deux modes différents et sont contrôlés par neuf registres 16-bit et il existe deux types de ces registres :

- Les registres de multiplexage (I/O MUX Control registers (MCRx)) : ils choisissent le mode de fonctionnement des pins.
- Les registres de contrôle de direction et de donnée (Data and direction control registers (PxDATDIR)) : il permet de contrôler les données et la direction des données.

### **Joint test action group (JTAG) port:**

Le port JTAG est une interface entre l'ordinateur et le DSP. Un émulateur est nécessaire pour la connexion entre le DSP et l'ordinateur à travers le port JTAG et ainsi permet à l'ordinateur de contrôler toute le DSP quand le Code Composer Studio (CCS) est lancé.

### **Phase locked loop (PLL) Clock Module:**

La PLL est un multiplieur de l'horloge qui permet ainsi à l'utilisateur de contrôler la fréquence de l'horloge.

### **L'allocation de l'espace mémoire:**

Le DSP F28335 a trois zones d'espace mémoire différentes : mémoire de données, mémoire de programme (instructions) et mémoire d'entrée/sortie.

L'espace mémoire de données est utilisé pour les calculs de programme et toutes les mémoires utilisées par l'algorithme, la mémoire de données se présente sous forme intégrée (on-chip) RAM ou externe (external) RAM.

L'espace mémoire de programme est une zone mémoire réservée pour le programme de l'utilisateur, elle peut être localisée soit hors composant (off-chip RAM) ou bien intégrée (on-chip RAM), selon le réglage logique de la broche (MP/MC).

L'espace mémoire d'entrée/sortie représente une mémoire virtuelle utilisée pour le transfert de données au périphérique externe.

### **Random Access Memory (RAM):**

Le DSP F28335 a huit blocks de mémoire à accès aléatoire (single Access Radom Access Memory) de L0 jusqu'à L1, chacun a 4K mots de 16 bits.

Les adresses associées à la mémoire SARAM peuvent être utilisées comme des mémoires de données ou des mémoires de programme.

### **Mémoire flash non volatile (Non-Volatile Flash Memory):**

La DSP F28335 contient 256K mots de mémoire intégrée (on-chip flash Memory). La mémoire flash permet de stocker les lignes de code même si la source est éteinte. Cette mémoire est utilisée dans le cas où il n'y a pas de connexion avec l'ordinateur, c'est-à-dire le DSP seul, sinon la RAM externe du module d'évaluation (Evaluation Module (EVM)) est utilisée comme une mémoire flash. A noter que les deux mémoires sont programmables et effaçables électroniquement.

**Logiciel :**

Pour la programmation, on utilise le Code Composer Studio (CCS) de Texas Instrument (TI) c'est facile à utiliser et permet d'écrire et développer soit en C soit en langage assembleur de TI.

**2.2 Convertisseur analogique numérique (the Analog-to-digital converter ADC)**

Le convertisseur analogique-numérique (Figure 2.3) est un échantillonneur de signal analogique, d'où la sortie est un entier qui représente le niveau de tension qui sera utilisé dans les calculs de l'algorithme. Sa résolution est de 12 bits et la mesure est assurée par un échantillonneur-bloqueur.

Il existe 16 lignes d'entrées multiplexées pour la seule entrée de l'ADC contrôlées par un auto-séquenceur qui fixe l'ordre des Canaux.

L'ADC peut être configuré avec un seul séquenceur 16-conversion ou bien deux séquenceurs 8-conversion (SEQ1 et SEQ2).

Il existe 16 registres pour stocker les valeurs converties (RESULT0-RESULT15).

Le lancement de la séquence de conversion (start of conversion SOC) peut être déclenché à partir de multiple sources :

- a. Software : à partir du programme.
- b. le module MLI A (EPWMA).
- c. le module MLI B (EPWMB).
- d. externe : a partir de pin ADCSOC

Le control des interruptions permet de générer une interruption à chaque fin de séquence (end-of-sequence (EOS)).

La valeur numérique de signal analogique d'entrée est calculée comme suit :

$$\text{La valeur numérique} = \begin{cases} 0 & \text{si l'entrée} \leq 0 \text{ V} \\ 4096 \times \frac{\text{la tension analogique d'entré} - \text{ADCLOC}}{3} & \text{si } 0\text{V} < \text{l'entrée} < 3.3\text{V} \\ 4095 & \text{si l'entrée} \geq 3.3\text{V} \end{cases}$$

Les registres de l'ADC sont accessibles avec l'horloge SYSCLKOUT, la séquence à l'échelle du temps interne de périphérique est contrôlé par l'horloge à haute cadence : " High Speed Peripheral Clock (HSPCLK) ".





Les sources/destinations des données :

- L4-L7 SARAM.
- Toute la zone des interruptions XINTF
- Les registres des résultats de l'ADC.
- Les tampons (buffers) récepteurs, transmetteurs des McBSP-A et McBSP-B.
- Trois registres des périphériques EPWM 1-6/HRPWM1-6.

## 2.4 Les Interruptions

L'interruption, qui est une rupture de séquence asynchrone, est générée suite à un événement interne ou externe (le plus probable). Lors d'une interruption, le traitement courant du processeur est arrêté de manière à exécuter la tâche associée, il s'agit d'une routine d'interruption surveillée par le système contrôleur des interruptions qui autorise le service des routines d'interruptions (ISR) selon la priorité attribuée à chacune d'elles, et cela tout en sauvegardant les conditions et les contenus des registres du programme pour être restaurés lors du retour (d'interruption).

Le CPU 28x supporte une interruption non-maskable (NMI) et 16 interruptions masquables. Ces interruptions sont vectorisées selon des priorités et sont centralisées autour du module d'extension des interruptions périphériques PIE (peripheral interrupt expansion) qui contrôle les interruptions (96 interruptions) provenant des différents périphériques et d'autres broches externes. Le module PIE prend en compte les interruptions et les classe en 12 groupes selon la priorité (INT1 à INT12). Dans le Peripheral Vector Register (PIVR), on charge un code qui identifie chaque interruption afin que le ISR puisse différencier entre les interruptions se passant au même niveau. Le module PIE assure un contrôle complémentaire avant qu'une interruption n'atteigne le CPU.

Les interruptions externes peuvent également être forcées par programme grâce à l'instruction INTR. Par ailleurs, des interruptions internes peuvent être générées par des périphériques intégrés (le timer génère INTT, les ports série génèrent RINT0, XINT0, RINT1 et XINT1).

### 2.4.1 Hiérarchie des interruptions

#### a) Le niveau de périphérique :

Il existe deux niveaux: Le bit d'interruption flag (drapeau) et le bit d'autorisation (ou enable) disponibles dans les registres de configuration des périphériques afin de faire correspondre à chaque événement une interruption.

Le bit drapeau (flag) est le premier bit à s'enclencher quand une interruption survient, alors que le bit d'autorisation (enable bit) opère comme une porte. Si ce bit est mis à 1 (Set), un signal d'autorisation d'interruption est généré qui passe à un niveau suivant et qui est ensuite multiplexé à travers le PIE (Peripheral Interrupt Expansion). Sinon aucun signal autorisant les interruptions n'est généré.

#### b) Le niveau PIE :

Le bloc PIE regroupe 12 interruptions, chaque groupe génère une seule interruption représentant huit (08) interruptions périphériques multiplexées pour le CPU. Par exemple, le groupe PIE 1 produit une interruption CPU 1 (INT1), ils existent pour chaque groupe un registre drapeau (PIEIFRx) et un registre autorisation (PIEIERx) dont (x=PIE groupe 1-PIE groupe 12) et y=1-8 correspond aux interruptions multiplexées par chaque groupe. Lorsque le bloc PIE reçoit une demande d'interruption, c'est-à-dire le bit PIEFIRx.y s'enclenche, il

vérifie si le CPU est prêt à recevoir une interruption à travers le bit PIEACKx envoie ensuite une demande au CPU.

**c) le niveau CPU :** Une fois la demande est reçue par le CPU, le bit (IFR) correspondant à INTx est activée, l'interruption n'est mise en service que si le bit correspondant dans le registre (IER) est activé.

La figure 2.5 montre comment s'opère le multiplexage des différentes sources d'interruptions pour les deux familles F28x and C28x.

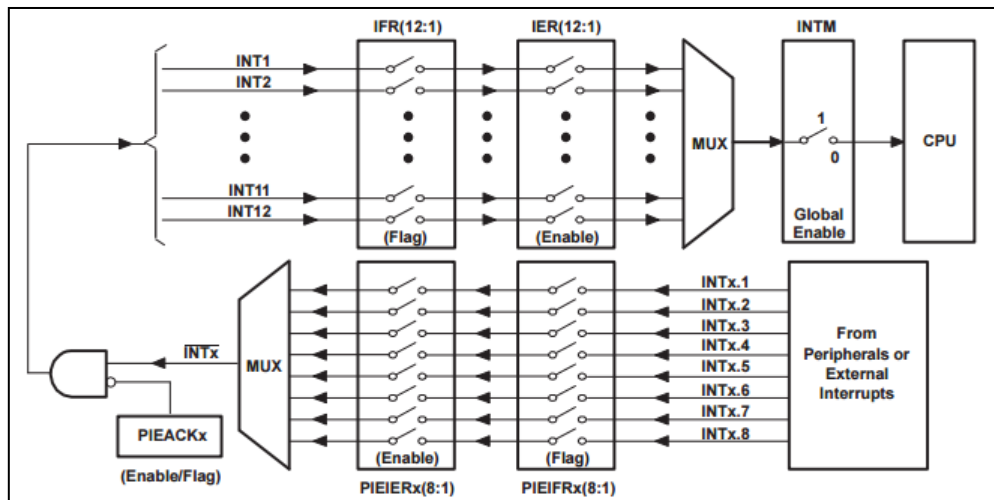


Figure 2.5 Multiplexage des sources d'interruption en utilisant le bloc PIE [4].

Après cela, la requête d'interruption doit être approuvée. Si l'interruption est masquable, un certain nombre de conditions doivent être respectées pour qu'elle soit approuvée, tandis que pour les interruptions matérielles non-masquables et les interruptions logicielles, l'approbation est immédiate.

### 2.4.2 Niveaux de priorités des interruptions du DSP

Tableau 2.1 les interruptions suivant leur ordre de priorité et leurs descriptions

INTERRUPTION	Description	PRIORITE
RESET	RESET tous les registres	1
EMUINT	CPU Emulation interrupt	2
NMI	External non-maskable interrupt	3
RTOSINT	CPU real time OS interrupt	4
INT1	PIE groupe 1	5
INT2	PIE groupe 2	6
INT3	PIE groupe 3	7
INT4	PIE groupe 4	8
INT5	PIE groupe 5	9
INT6	PIE groupe 6	10
INT7	PIE groupe 7	11
INT8	PIE groupe 8	12
INT9	PIE groupe 9	13
INT10	PIE groupe 10	14
INT11	PIE groupe 11	15
INT12	PIE groupe 12	16
INT13	Interruption externe (XINT13) ou timer 1	17
INT14	Timer 2	18

### 2.4.3 Interrupt Flag Register (IFR)

IFR est un registre 16-bit et chaque bit correspond à une interruption. Ce registre est adressé en mémoire à 0006h. Il permet d'identifier et de valider l'interruption en cours au niveau du CPU et contient des bits drapeau pour interruptions masquables.

La mise à 1 de l'IFR se fait lorsqu'un seul signal d'interruption survient et parvient au niveau du CPU. Le bit drapeau particulier dépend de la priorité de l'interruption. Après cela, le bit correspondant à l'interruption dans l'IFR sera remis à 0.

### 2.4.4 Interrupt Mask Register (IMR)

L'IMR est similaire bit drapeau IFR à la différence que sa localisation dans la mémoire est à l'adresse 0004h. On y trouve aussi un bit de masque pour les 6 priorités d'interruption.

Quand un des bits de l'IMR est à "0", l'interruption qui lui correspond est masquée c'est-à-dire qu'elle sera interrompue au niveau du CPU, le processeur ne recevra pas le signal demandé indépendamment du statut du bit INTM. Par contre, si le bit est mise à "1" et avec la condition que le bit INTM est à "0", l'interruption sera reconnue et pourra être lue car l'IMR différencie entre celles qui sont masquées et celles qui ne le sont pas.

## 2.5 GP timers

Il existe deux temporisateurs à usage général (GP timers), chacun d'entre eux étant synchronisé à la fréquence du CPU ou à une fréquence présélectionnée.

Les temporisateurs sont utilisés pour sélectionner la période d'échantillonnage de l'ADC (amorcer le début de conversion) ou bien pour sélectionner la période de la commutation pour la génération des signaux MLI.

Chaque GP timer est constitué de :

- un registre compteur 16-bit TxCNT contenant la valeur instantanée du compteur.
- un registre comparateur 16-bit TxCMPR.
- un registre de période 16-bit TxPR.
- un registre de control 16-bit TxCON.
- horloge programmable présélectionnée.
- logique d'interruption.

Le temporisateur peut aussi être configuré pour la génération des interruptions ou bien l'amorçage des autres périphériques comme l'ADC.

### 2.5.1 Les entrée/sorties de GP timer

- Les entrées :

-l'horloge référence : elle peut être l'horloge interne du CPU ou bien déterminée par le registre TCLKINA/B.

-la direction de comptage : TDIRA/B quand le timer est actif en mode comptage/décomptage.

- signal Reset.

- Les sorties :

- sortie de comparaison TxCMP , x=1,2,3,4 .

- signal lancement de la séquence de conversion (connecté au module ADC).

- les signaux : dépassement (Overflow), atteint la valeur zéro (underflow), égalité (compare match).

-la mode de comptage.

Le registre (GPTCONA/B) fixe les actions de timer à chaque instant et spécifie le mode de comptage.

### 2.5.2 Les modes de comptage

1. échantillonnage/blocage (le temporisateur garde son état).

2. le mode de comptage continu (ce mode est utilisé pour la génération des signaux MLI asymétrique).

3. le mode de comptage/décomptage commandé (l'entrée TDIRA/B détermine le sens de comptage, quand TDIR A/B est au niveau haut (high) le comptage est mis en marche, en revanche si TDIRA/B est au niveau bas (low) le décomptage est mis en marche).

4. le mode de comptage /décomptage continu (ce mode est utilisé pour la génération des signaux MLI symétrique).

### 2.6 Les sorties MLI

le mode de comptage décrit une forme d'onde pour le temporisateur dans le sens où la valeur de TxCNT détermine la fréquence d'une onde de forme triangulaire asymétrique « dent de scie » ou bien triangulaire symétrique et qui définit la porteuse pour la génération des signaux MLI.

A chaque cycle d'horloge une comparaison est faite entre la valeur de comparaison enregistrée dans le registre TxCMPR (modulante) avec la valeur instantanée du compteur (porteuse). Quand ils auront la même valeur, la sortie (TxPWM) bascule du niveau haut vers le niveau bas ou bien le contraire (voir Figure 2.6). En variant la valeur du registre TxCMPR qui prend les valeurs d'un signal sinusoïdal, on génère la modulante sinusoïdale.

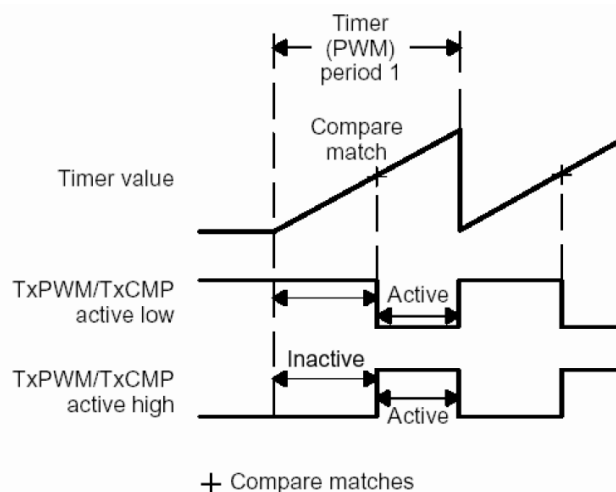


Figure 2.6 Compare match et les changements sur la broche TxPWM [5].

La polarité de la sortie TxPWM est configurée par le registre GPTCON A/B, si la sortie est configurée **active à l'état bas** alors elle va basculer du niveau haut vers le niveau bas lorsque le match est détecté, par contre si la sortie est configurée **active à l'état haut** alors elle va basculer du niveau bas vers le niveau haut.

*a) Génération d'une porteuse asymétrique (dent de scie) :*

On génère une porteuse asymétrique (voir Figure 2.7) lorsque le temporisateur est en mode de comptage continu, la sortie TxPWM change de niveau comme suit :

- 1. La sortie est à l'état inactif avant le début de comptage.
- 2. La sortie reste à l'état inactif jusqu'à la détection de match.
- 3. La sortie bascule à l'état actif après la détection de match.
- 4. La sortie garde l'état actif jusqu'à la fin de la période du temporisateur.
- 5. A la fin de période, la sortie est réinitialisée vers l'état inactif si la nouvelle valeur de registre comparateur n'est pas nulle.

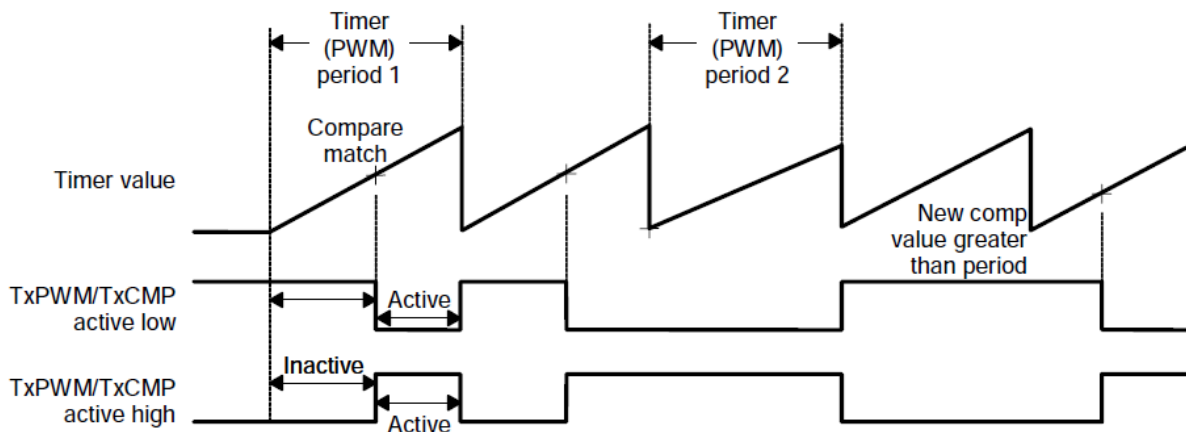


Figure 2.7 Génération d'une porteuse asymétrique (dent de scie) par un temporisateur en mode de comptage continu [5].

La valeur de registre TxPR =  $\left[ \frac{\text{la période de la porteuse}}{\text{la période présélectionné de l'horloge}} - 1 \right]$

*b) Génération d'une porteuse symétrique (triangulaire) :*

On génère une porteuse symétrique (voir Figure 2.8) lorsque le temporisateur est en mode comptage/décomptage continu, la sortie TxPWM change de niveau comme suit :

1. La sortie est à l'état inactif avant le début de comptage.
2. La sortie reste à l'état inactif jusqu'à la détection de match.
3. La sortie bascule à l'état actif après la détection de premier match.
4. La sortie garde l'état actif jusqu'à la fin de la détection de second match.
5. La sortie bascule à l'état inactif après la détection de second match.
6. la sorite garde l'état inactif jusqu'à le prochain match.

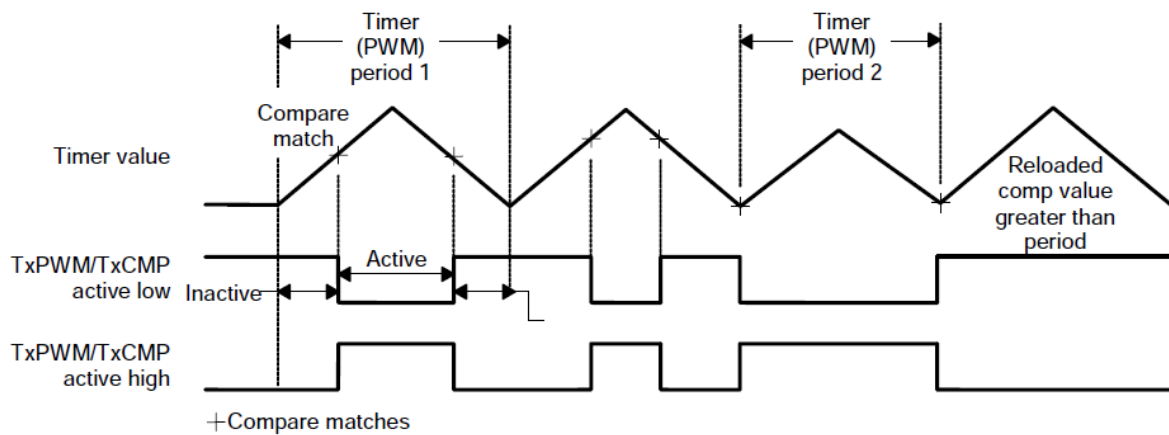


Figure 2.8 Génération d'une porteuse symétrique (triangulaire) par un temporisateur en mode comptage/décomptage continu [5].

La valeur de registre TxPR =  $\left[ \frac{\text{la période de la porteuse}}{2 * \text{la période présélectionné de l'horloge}} \right]$

### 2.6.1 Calcul des temps actif et inactif de la sortie TxPWM

Il serait nécessaire de programmer les angles de commutation et avoir le contrôle sur les temps de l'état actif et inactif pour l'implémentation des différentes techniques de MLI comme la MLI pré-calculée qui définit les largeurs d'impulsion sur une période.

Le principe de calcul est présenté ci-dessous pour les 2 types de porteuse symétrique et asymétrique :

a) *Porteuse asymétrique :*

Le temps d'impulsion active =  $[(TxPR) - (TxCMPR) + 1] \times (\text{la période d'un cycle d'horloge d'entrée présélectionnée})$

Le temps d'impulsion inactive =  $TxCMPR \times (\text{la période d'un cycle d'horloge d'entrée présélectionnée})$

-si la valeur de registre comparateur TxCMPR est nulle, alors la sortie reste à l'état actif toute la période.

- si la valeur de registre comparateur est supérieur à la valeur de registre de période ( $TxCMPR > TxPR$ ) alors la sortie reste à l'état inactif toute la période.

b) *Porteuse symétrique :*

-Le temps d'impulsion active =  $[(TxPR)-(TxCMPR)_{up} + (TxPR)-(TxCMPR)_{dn}] \times (\text{la période d'un cycle d'horloge d'entrée présélectionnée})$

### 2.6.2 Génération de temps morts

Le DSP F28335 génère un temps mort entre les sorties PWM dépendante et indépendante. Ce temps mort peut éliminer le problème de shoot through ou court circuit dans un bras d'onduleur. Les figures qui suivent illustrent l'addition d'un temps mort pour les deux sorties avec porteuse symétrique (Figure 2.9) et asymétrique (Figure 2.10). Les deux sorties basculent comme suit : la 1<sup>ère</sup> sortie bascule en premier PWM<sub>x+1</sub>, après un certain temps mort, puis la 2<sup>ème</sup> sortie PWM<sub>x</sub> bascule.

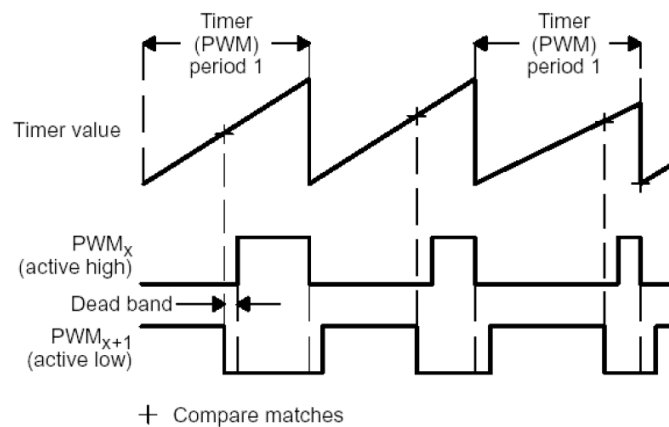


Figure 2.9 Temps mort avec une porteuse asymétrique [5].

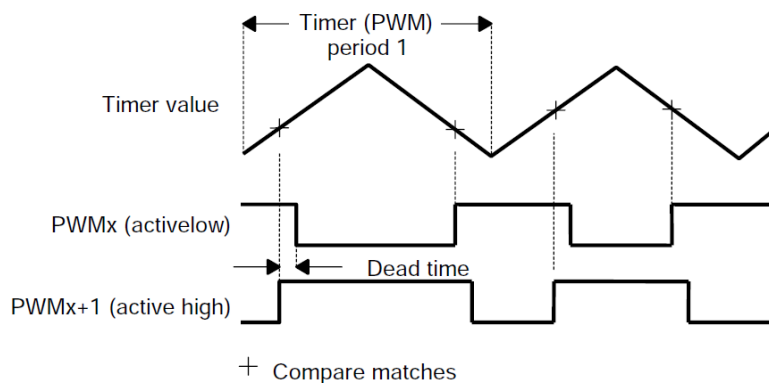


Figure 2.10 Temps mort avec une porteuse symétrique [5].

La valeur de temps mort est déterminée par la formule suivante :

$$(\text{Nombre de cycles de l'horloge de temps mort}) = \frac{\text{bit}[8 \rightarrow 11] \text{ de registre } DBTCONx}{\text{la valeur de l'horloge présélectionné}}$$

Le tableau ci-dessous montre les différents temps mort générés par les combinaisons de registre de contrôle DBTCONx :

Tableau 2.2 Le temps mort généré par les différentes combinaisons de registre DTCONx en (µs) [5]

DBTCONx	(DBTCONx bits [4-2])					
	110 and bits [11-8] 1x1 (P=32)	100 (P=16)	011 (P=8)	010 (P=4)	001 (P=2)	000 (P=1)
0	0	0	0	0	0	0
1	0.8	0.4	0.2	0.1	0.05	0.025
2	1.6	0.8	0.4	0.2	0.1	0.05
3	2.4	1.2	0.6	0.3	0.15	0.075
4	3.2	1.6	0.8	0.4	0.2	0.1
5	4	2	1	0.5	0.25	0.125
6	4.8	2.4	1.2	0.6	0.3	0.15
7	5.6	2.8	1.4	0.7	0.35	0.175
8	6.4	3.2	1.6	0.8	0.4	0.2
9	7.2	3.6	1.8	0.9	0.45	0.225
A	8	4	2	1	0.5	0.25
B	8.8	4.4	2.2	1.1	0.55	0.275
C	9.6	4.8	2.4	1.2	0.6	0.3
D	10.4	5.2	2.6	1.3	0.65	0.325
E	11.2	5.6	2.8	1.4	0.7	0.35
F	12	6	3	1.5	0.75	0.375



### 2.6.3 L'unité de comparaison

Le signal MLI peut être aussi généré de l'unité de comparaison. Contrairement au registre de comparaison de temporisateur, chaque unité de comparaison génère deux sorties qui basculent au même instant de match et qui sont complémentaires.

## 2.7 Les outils de développement logiciel

### 2.7.1 Code Composer Studio (IDE)

Le CCS (Figure 2.11) est un environnement de développement intégré (IDE), souple et facile à utiliser, qui incorpore une variété d'outils logiciels pour la génération du code exécutable à savoir : le compilateur, le linker, permet aussi l'émulation, le débogage du programme en temps réel et l'analyse des résultats produits. La programmation du DSP TMS 320F28335 est faite en langage C ou en langage assembleur.

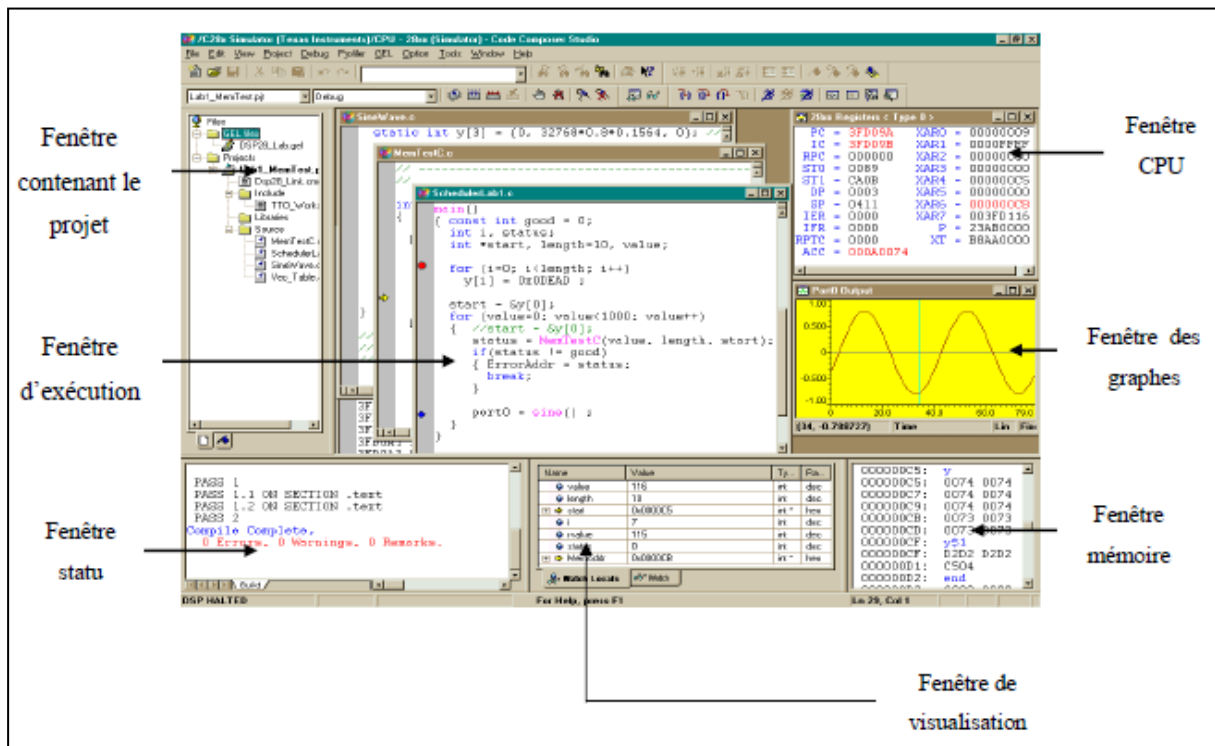


Figure 2.11 Capture d'écran relative à l'IDE Code Composer Studio [6].

### 2.7.2 Matlab/Simulink

La génération des codes programme peut être aussi effectuée avec le logiciel Matlab/Simulink. Il suffit d'installer le package «Support Package for Texas instrument C2000 Processors» pour spécifier le système cible (Target system) visé par les outils de développement, dans notre cas le système cible est la carte TMS320F28335. Ce package permet l'accès aux différents périphériques de la carte DSP : GPIO, CAN, ADC, PWM.

Pour n'importe quelle commande qui nécessite un calculateur, la combinaison entre les blocs Simulink classique et les blocs des périphériques de la carte DSP permettent de générer un code. Une fois le modèle de la commande désirée est simulé avec succès sur Simulink, le workshop génère un code pour « temps réel » avec langage C et crée un projet CCS. En suite, il faut ouvrir le CCS qui compile et effectue le lien (link) et le programme est transféré vers le système cible (F28335).

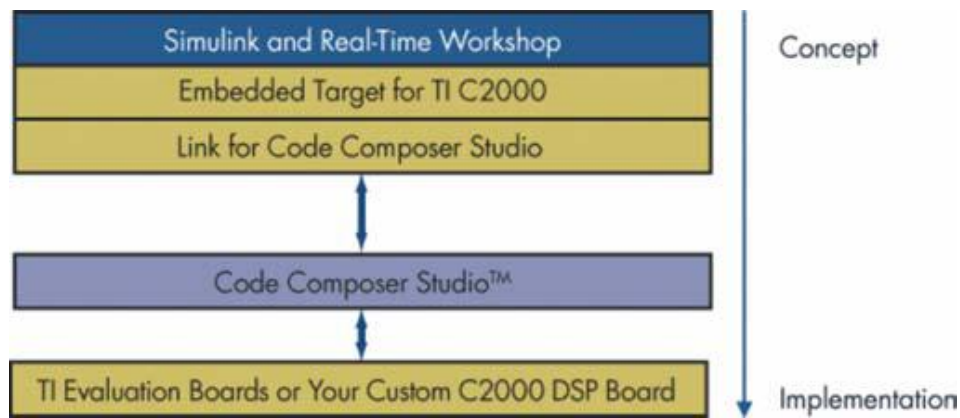
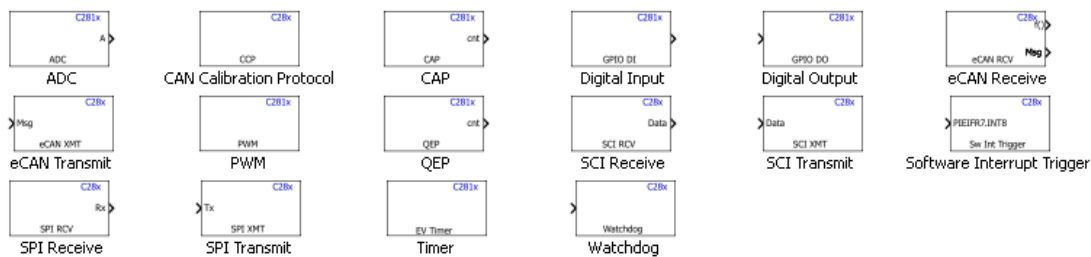


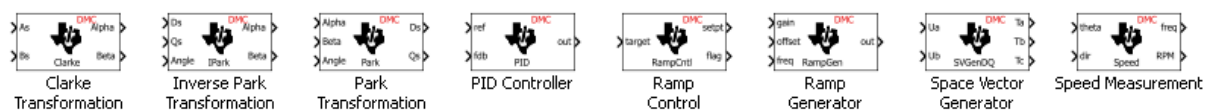
Figure 2.12 Les différentes couches de développement logiciel [7].

### 2.7.3 Exemple de configurations des périphériques de la DSP sur Simulink

L'ensemble de blocs que contient « Embedded Coder Support Package for Texas Instruments C2000 Processors » est présenté ci-dessous :



Il contient aussi un ensemble dédié à la commande numérique des moteurs (C28X DMC) Digital Motor Control :



#### a) Bloc ePWM :

Pour générer des signaux MLI, il faut configurer le bloc ePWM qui est plus simple que la programmation en langage C ou langage assembleur.

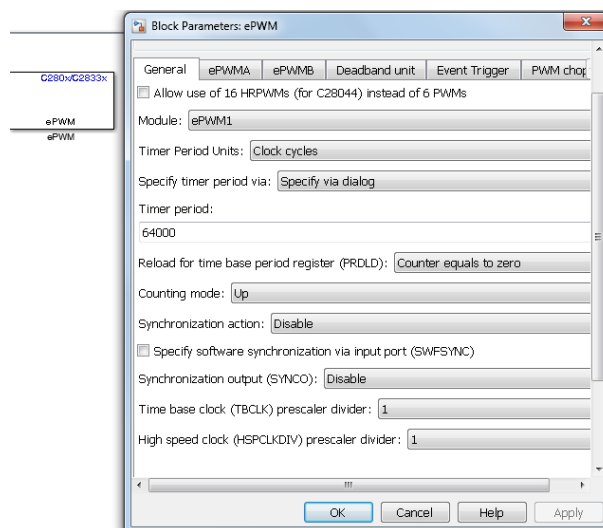


Figure 2.13 Bloc de paramètres ePWM.

Il faut configurer le temporisateur comme il a été décrit dans le sous-chapitre (1.5.1 GP timers) (la forme de la porteuse, l'horloge présélectionné...), spécifier quelle sortie utilisée (PWM1/PWM2) et le rapport cyclique, la logique utilisée (actif à l'état haut ou à l'état bas), spécifier la période de temps mort (deadband).

b) Bloc ADC :

Pour configurer le module ADC, il faut commencer par le choix de module A ou B ou bien les deux simultanément. Chaque module contient 8 canaux d'entrée selon le choix du module, ensuite il faut choisir le nombre de conversions et le canal d'entrée associé à chaque conversion.

-Le choix du module A affiche les canaux : ADCINA0, ADCINA1,...ADCINA7

-Le choix du module B affiche les canaux : ADCINB0, ADCINB1,...ADCINB7

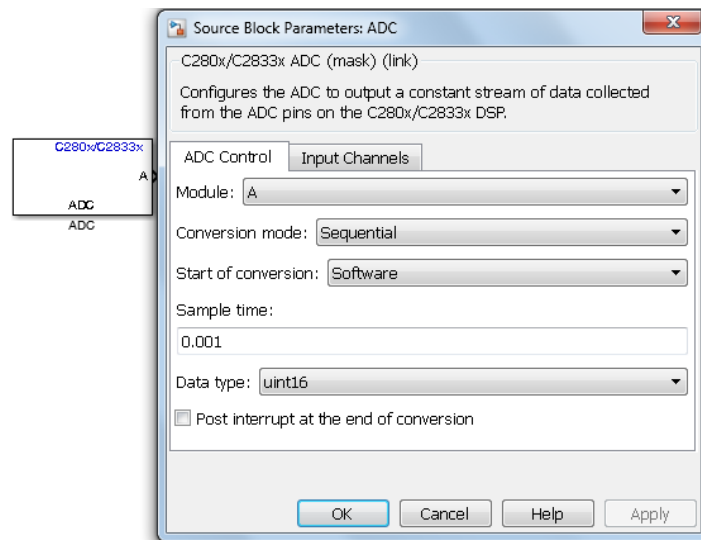


Figure 2.14 Bloc de paramètres ADC

c) Bloc GPIO :

Le bloc GPIO est divisé en deux sous blocs : entrée numérique (digital input), sortie numérique (digital output). Ce bloc facilite la génération de signal personnalisé (sinusoïdal, train d'impulsion..). Il suffit de choisir le port et le bit de la sortie où on veut récupérer le signal.

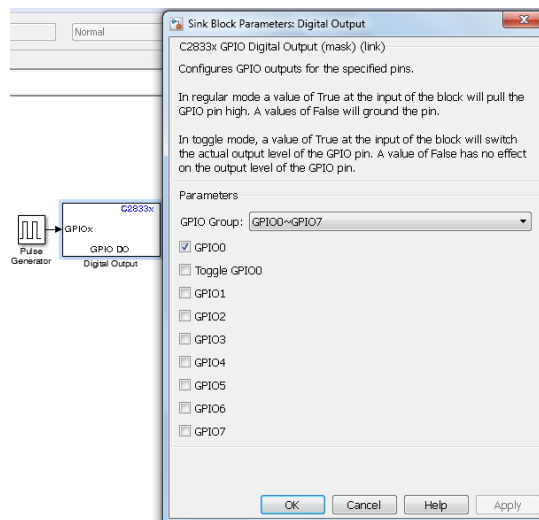


Figure 2.15 Bloc de paramètres Digital output

## 2.8 Implémentation sur DSP

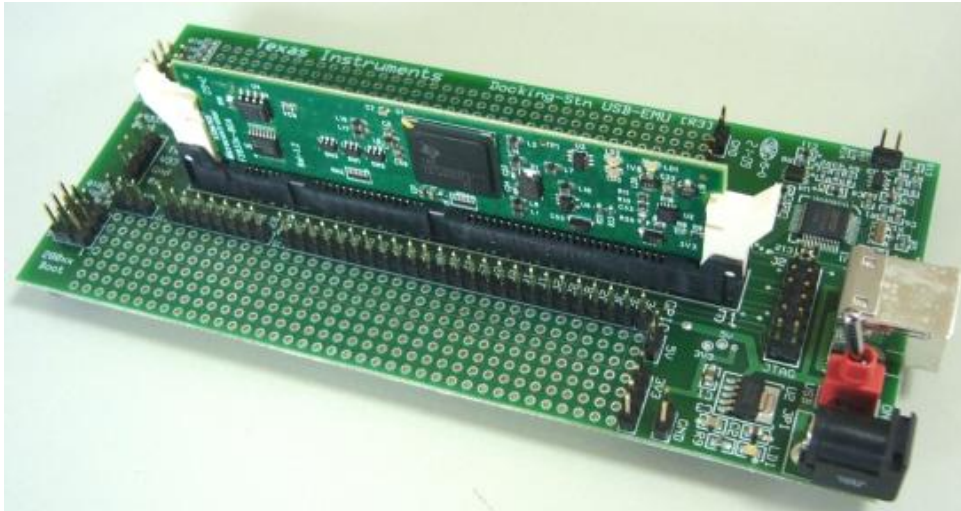


Figure 2.16 La carte à DSP TMS320F28335

Le projet implémenté dans la carte (Figure 2.16) est composé essentiellement des modules header, module d'initialisation et module routine de service d'interruption PWM (ISR).

Le programme d'initialisation :

La partie concernant l'initialisation est essentielle pour l'organisation intrinsèque et le bon fonctionnement de tous les périphériques et les modules de la carte. Elle permet d'initialiser les périphériques (ADC, DMA, CPU Timers, la table vecteur d'interruption PIE..) et configurer les broches du GPIO ainsi que de désigner la fonction de chacune d'elles.

Dans notre programme, nous avons choisi une période du temporisateur de 10 $\mu$ s.

Le programme d'interruption :

- La routine de service d'interruption (ISR) est exécutée après que le GP timer génère une interruption (qui doit être acceptée ou reconnue) par laquelle l'ISR est chargée d'empiler dans les registres CMPB les nouvelles valeurs qui sont spécifiées via une porte externe du bloc ePWM.
- Le changement des valeurs de registre de CMPB à chaque période nous permet de changer le rapport cyclique des impulsions et ainsi permet de faire une modulation de largeur d'impulsions MLI.
- La configuration des registres CMPB est faite à partir des pourcentages par rapport à la période du temporisateur. Par exemple, S'il est chargé avec un pourcentage de 100%, il génère une impulsion pleine avec une période de 10  $\mu$ s.

Les quatre signaux de gâchettes sont générés à partir de Matlab. Il suffit juste d'échantillonner les quatre signaux en respectant le théorème d'échantillonnage (théorème de Shannon) qui énonce que **“La représentation discrète d'un signal exige des échantillons régulièrement espacés à une fréquence d'échantillonnage supérieure au double de la fréquence maximale présente dans ce signal”**.

$$F_e > 2 \times F_{\max}$$

La période d'échantillonnage doit être égale à la période du temporisateur pour la synchronisation lors de la reconstitution de signal à partir de la carte et donc  $f_e=100$  KHz et la fréquence de commutation ne dépasse pas 6 KHz. Cela satisfait la condition de Shannon.

Lorsque le signal de commande est échantillonné, on obtient un tableau de valeurs (0,1). Ce tableau va définir le pourcentage de chaque impulsion par rapport à la période du temporisateur, ainsi il va attaquer la porte externe du module ePWM pour charger à chaque période la valeur de registre CMPB. La Figure 2.17 résume les étapes nécessaires pour la programmation de la carte DSP.

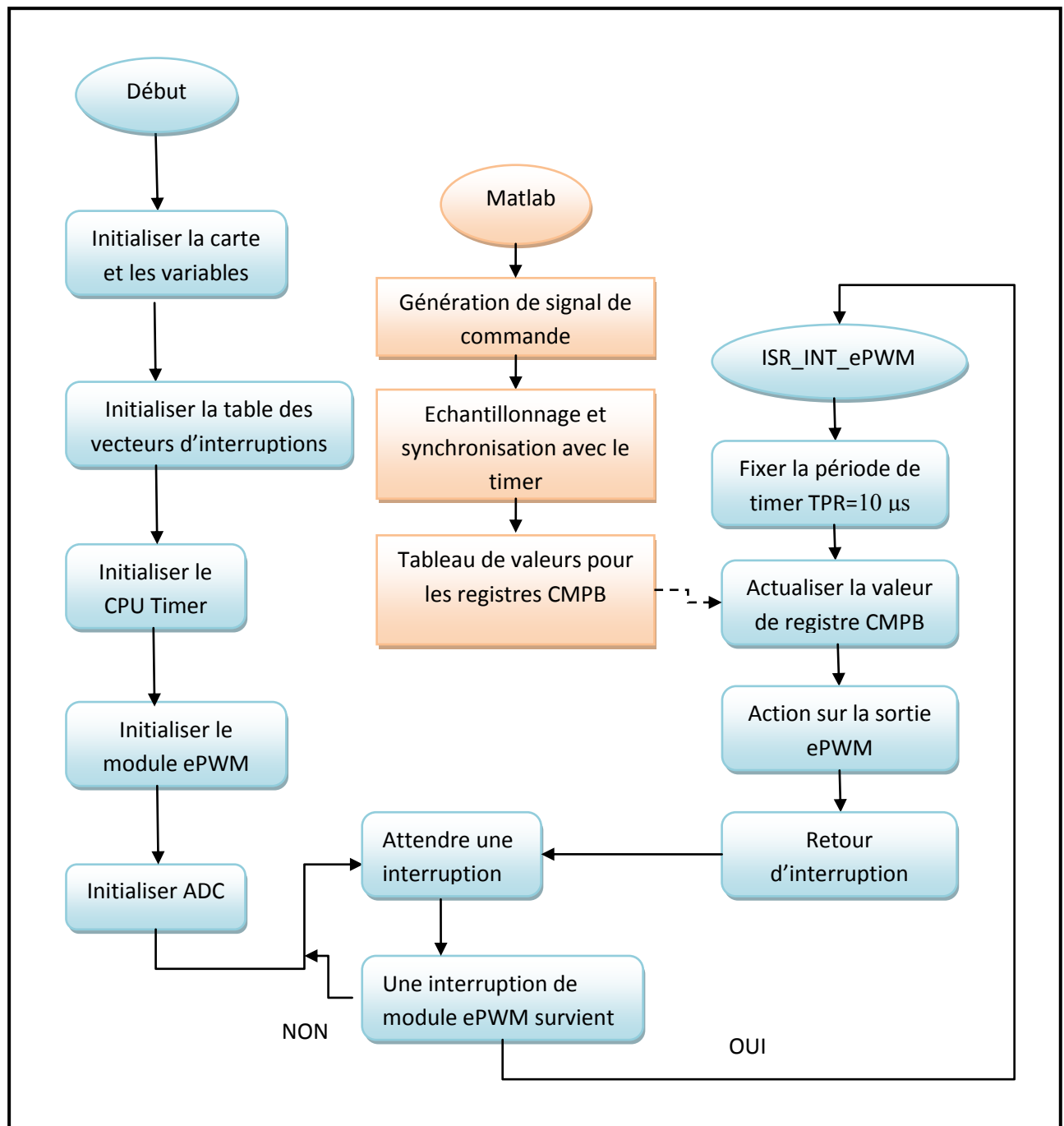
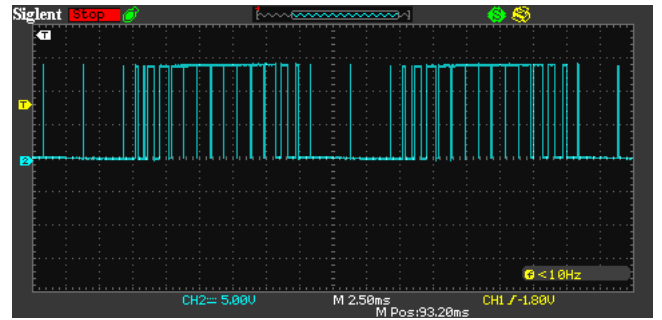
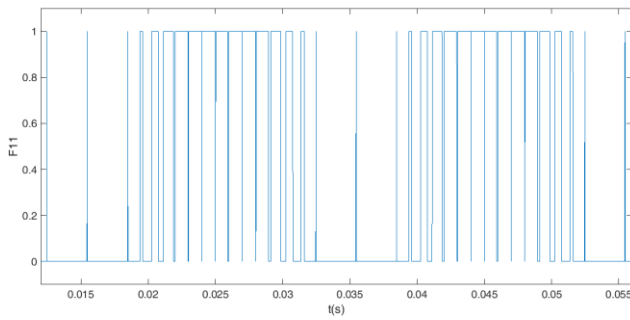


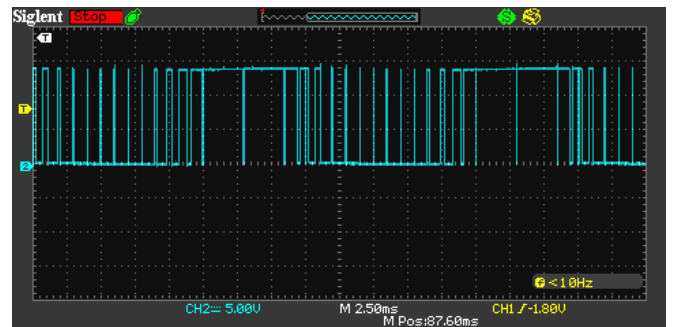
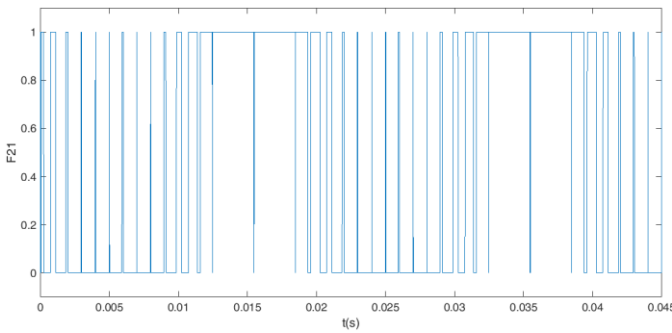
Figure 2.17 Organigramme de Programme

### 2.9 Génération des signaux de commande par la technique MLI vectorielle

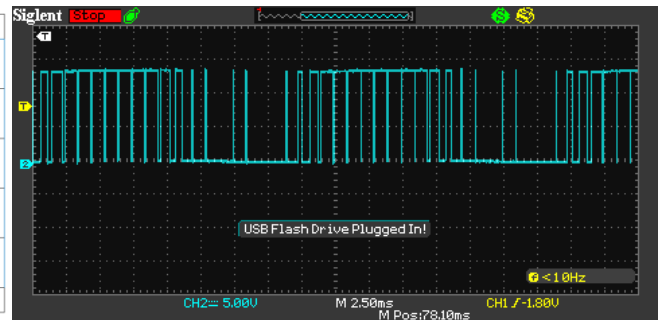
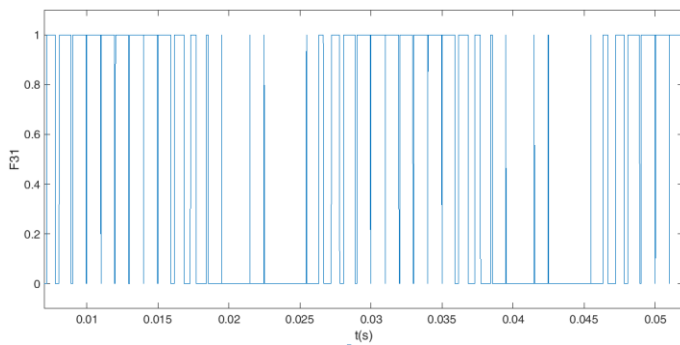
La Figure 2.18 montre les fonctions logiques des trois interrupteurs (niveaux haut) de l'onduleur et les signaux de commande correspondants générés à partir de la carte DSP par la technique MLI vectorielle pour un  $r=1$  et fréquence de commutation  $F_e=1Khz$ .



(a)



(b)



(c)

Figure 2.18 les fonctions logiques des interrupteurs de l'onduleur et les signaux de commande correspondants (a) La fonction logique  $F_{11}$  (b) La fonction logique  $F_{21}$  (c) La fonction logique  $F_{31}$

## **2.10 Conclusion**

L'implantation logicielle sur la carte DSP F28335 de la stratégie de commande basée sur l'algorithme de la modulation « MLI vectorielle » (SVPWM) est effectué à l'aide des outils de développement logiciel qui ont facilité la mise en œuvre du DSP F28335. Ainsi, le DSP F28335 peut être qualifié de bon calculateur pour la génération des signaux de commande grâce aux fonctionnalités et caractéristiques qu'ils possèdent.

---

## **Conclusion générale**

---



## Conclusion générale :

Au cours de ce travail nous avons étudié et implémenté la technique « MLI vectorielle » sur la carte DSP TMS320F28335.

Pour Cela nous avons utilisé le processus de la modulation vectorielle. D'autre part, nous avons exploité les outils numériques disponibles dans le logiciel MATLAB/SIMULINK pour générer les fonctions logiques des six interrupteurs par la technique MLI vectorielle. Ces fonctions logiques sont transférées vers la carte DSP F28335. Celle-ci est utilisée pour générer le train d'impulsions à appliquer aux gâchettes des interrupteurs d'électroniques de puissance (IGBT). Nous avons aussi donné un aperçu global sur les éléments et les fonctionnalités de base de cette carte que nous avons étudié afin de mieux exploiter ses caractéristiques dans la réalisation de notre projet.

L'élaboration de ce travail nous a permis d'améliorer nos connaissances dans la programmation et le monde des microcontrôleurs. Le long de ce projet, nous avons eu l'occasion de travailler avec plusieurs logiciels à savoir : le Code Composer Studio v6 (CCS IDE) qui permet la compilation et le débogage du programme en temps réel, l'outil Matlab/Simulink pour la simulation et la génération des codes.

Étant donné que notre carte DSP n'a pas été exploitée au maximum de ses capacités, Le travail effectué ouvre les perspectives suivantes:

- la commande en boucle fermée des moteurs alternatifs triphasés en utilisant les périphériques suivant : ADC, Quadrature Encoder Pulse (eQEP).
- La synchronisation avec le réseau électrique en utilisant le bloc PLL pour la connexion au réseau des alternateurs.

---

## **Bibliographie**

---

## **Bibliographie**

- [1] LALILI, Djafer. MLI vectorielle et commande non linéaire du bus continu des onduleurs multiniveaux: Application à la conduite de la machine asynchrone. 2008. Thèse de doctorat. Ecole nationale supérieure polytechnique.
- [2] Texas instruments “ TMS320F2833x, TMS320F2823x Digital Signal Controllers (DSCs) ” June 2007 revised October 2016.
- [3] Frank Bormann, Architecture of the TMS320F28335, OpenStax-CNX module: m36737.
- [4] TMS320x2833x, 2823x System Control and Interrupts, Reference Guide September 2007– Revised March 2010.
- [5] TOLIYAT, Hamid A. et CAMPBELL, Steven G. DSP-based electromechanical motion control. CRC press, 2003.
- [6] Kamel BOUDJIT, «Commande synchrone en vitesse de deux moteurs à courant continu à base de DSP », mémoire de magister, USTHB, 2008.
- [7] DOBRA, Petru, DUMA, Radu, MOGA, Daniel, et al. Digital control applications using TI digital signal controller. WSEAS Transactions on Systems and Control, 2008, vol. 3, no 6, p. 558-568.