



INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y
ELÉCTRICA

UNIDAD PROFESIONAL CULHUACAN

“TÉCNICAS DE MODULACIÓN DE VECTOR DE
ESPACIO Y SINCRONIZACIÓN PARA
CONVERTIDORES ELECTRÓNICOS DE ENERGÍA”

TESIS

QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMUNICACIONES Y ELECTRÓNICA

PRESENTA:

KEVIN CANO PULIDO



ASESORES:

Dr. ISMAEL ARAUJO VARGAS

Ing. ALEJANDRO TAPIA HERNÁNDEZ

OCTUBRE 2013

**INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y ELÉCTRICA
UNIDAD CULHUACAN**

TESIS INDIVIDUAL

Que como prueba escrita de su Examen Profesional para obtener el Título de Ingeniero en Comunicaciones y Electrónica, deberán desarrollar el C:

KEVIN CANO PULIDO

“TECNICAS DE MODULACION DE VECTOR DE ESPACIO Y SINCRONIZACION PARA CONVERTIDORES ELECTRONICOS DE ENERGIA”

En esta tesis se muestra el análisis, diseño y construcción de un PLL para un convertidor inversor controlado con Modulación de vector de espacio. Describe la manera en la que se controla un inversor trifásico tradicional con esta modulación y se desarrolla un programa que se implementa en un microcontrolador. Además, se hace un recuento de las principales herramientas para sincronizar circuitos. De esta manera se explica el funcionamiento de un PLL y se desarrolla uno mismo de manera digital implementándose en un microcontrolador. Finalmente se muestran los resultados obtenidos de manera simulada y experimental. Este último apartado demuestra el correcto funcionamiento del PLL.

CAPITULADO

**CAPITULO 1 ESTADO DE ARTE
CAPITULO 2 MODULACION DE VECTOR DE ESPACIO DE ANCHO DE PULSO
CAPITULO 3 SINCRONIZADOR DE FASE CERRADA
CAPITULO 4 IMPLEMENTACION DIGITAL
CAPITULO 5 PRUEBAS Y RESULTADOS
CAPITULO 6 OBSERVACIONES Y CONCLUSIONES
ANEXO A CODIGO DEL PROGRAMA DEL SVPWM Y EL PLL DIGITAL**

México D. F., a 19 de agosto del 2013

PRIMER ASESOR:

SEGUNDO ASESOR:

ING. ISMAEL ARAUJO VARGAS

ING. ALEJANDRO TAPIA HERNANDEZ

Vo. Bo.

APROBADO

M. en C. ANTONIO ROMERO ROJANO
JEFE DE LA CARRERA DE I.C.E.

M. en C. HECTOR BECERRIL MENDOZA
SUBDIRECTOR ACADÉMICO

Índice General

Índice General	I
Índice de Figuras	V
Índice de Tablas.....	VII
AGRADECIMIENTOS	VIII
DEDICATORIA	IX
RESUMEN.....	X
CAPÍTULO 1.....	1
1. Estado del Arte	1
1.1. Introducción.....	1
1.2. Estado del arte relativo a Convertidores electrónicos de energía que operan con suministro de CA.	2
1.2.1. Rectificadores	2
1.2.1.1. Rectificadores naturalmente controlados.....	2
1.2.1.2. Rectificadores controlados.....	3
1.2.1.3. Rectificadores activos.....	3
1.2.1.4. Resumen.....	4
1.2.2. Ciclo-convertidores.	4
1.2.2.1. Ciclo-convertidores a base de inversores.	4
1.2.2.2. Ciclo-convertidores en base a tiristores.....	5
1.2.2.3. Convertidores Matriciales.....	6
1.2.2.4. Resumen.....	6
1.2.3. Inversores.....	7
1.2.3.1. Inversores.....	7
1.2.3.2. Resumen.....	7
1.2.4. Filtros Activos.	8
1.2.4.1. Esquema General.	8
1.2.4.2. Clasificación.....	8
1.2.4.3. Resumen.....	8
1.3. Estado del arte relativo a Técnicas de sincronización de circuitos.	9
1.3.1. Detectores de cruce por cero.....	9

1.3.2.	El PLL.....	10
1.3.2.1.	Detector de Fase.	10
1.3.2.2.	EL VCO.	13
1.3.2.3.	Compensador.	14
1.3.3.	Herramientas modernas para manipular y procesar señales trifásicas en convertidores de energía.....	14
1.3.3.1.	Transformación de Clarke.	14
1.3.3.2.	Transformación de Park.	16
1.4.	Resumen del estado del arte.	16
1.5.	Objetivo.....	16
1.6.	Bosquejo.....	17
1.7.	Referencias.....	18
CAPÍTULO 2.....		19
2.	Modulación de vector de espacio de ancho de pulso.....	19
2.1.	Introducción.....	19
2.2.	PWM de Vector de Espacio Tradicional.....	20
2.2.1.	Principio de Operación.....	20
2.2.2.	Método de Implementación.....	24
2.2.3.	Realización Práctica.....	25
2.2.3.1.	Generación de Voltajes en un ciclo de Conmutación.....	25
2.2.3.2.	Principio de Operación del Algoritmo de SVM para un Inversor Trifásico.....	26
2.2.3.3.	Generador de Portadora.....	27
2.2.3.4.	Localizador de Sector.....	27
2.2.3.5.	Generador de Tiempos Activos.....	27
2.2.3.6.	Generador de Interrupciones.....	28
2.3.	Modulación de Espacio Vectorial de Ancho de pulso para otras topologías.....	28
2.3.1.	Inversores Multiniveles.....	28
2.3.2.	Inversores Asimétricos.....	28
2.3.3.	Ejemplificación de un inversor multiniveles.....	28
2.4.	Resumen.....	29
2.5.	Referencias.....	30
CAPÍTULO 3.....		31

3.	Sincronizador de fase cerrada.....	31
3.1.	Introducción.....	31
3.2.	Principio de Funcionamiento.....	32
3.2.1.	Marco de referencia rotante como Detector de Fase.....	32
3.2.2.	DCO Digital.....	34
3.2.3.	Compensador.....	34
3.3.	Diseño del PLL.....	37
3.4.	Metodología de Implementación.....	39
3.4.1.	Descripción del PLL.....	40
3.4.1.1.	Generación de tablas.....	40
3.4.1.2.	Lectura de Voltajes.....	40
3.4.1.3.	Transformaciones.....	40
3.4.1.4.	Cálculo de compensador.....	40
3.5.	Resumen.....	40
3.6.	Referencias.....	42
CAPÍTULO 4.....		43
4.	Implementación digital.....	43
4.1.	Introducción.....	43
4.2.	Breve descripción del microcontrolador.....	44
4.3.	Implementación de algoritmo de SVPWM.....	45
4.3.1.	Configuración del Timer.....	46
4.3.2.	Configuración de los Módulos de Comparación de Salida.....	47
4.4.	Verificación numérica del PLL.....	47
4.5.	Implementación Digital del PLL.....	50
4.5.1.	Configuración del ADC.....	52
4.5.2.	Configuración del DMA.....	52
4.6.	Resumen.....	52
4.7.	Referencias.....	53
CAPÍTULO 5.....		54
5.	Pruebas y resultados.....	54
5.1.	Introducción.....	54
5.2.	Verificación de las señales de control para el inversor tradicional.....	55

5.3.	Verificación experimental del inversor Trifásico y la modulación SVPWM.	56
5.4.	Desarrollo físico del PLL.....	58
5.5.	Verificación de transformación de Clarke en el dsPIC.	59
5.6.	Verificación experimental del PLL.	61
5.7.	Resumen.....	66
CAPÍTULO 6.....		67
6.	Observaciones y Conclusiones	67
6.1.	Introducción.	67
6.2.	Trabajo Futuro.....	68
6.3.	Conclusiones y observaciones.....	68
ANEXO A		69
7.	Código del programa del SVPWM y PLL digital.	69

Índice de Figuras

Figura 1.1 a) Diagrama rectificador monofásico no controlado b) Diagrama rectificador trifásico no controlado	2
Figura 1.2 Diagrama de un rectificador trifásico controlado.	3
Figura 1.3 Diagrama de un rectificador activo.	4
Figura 1.4 Diagrama general de un inversor de CA-CA utilizando un inversor.	5
Figura 1.5 Diagrama de un ciclo-convertidor monofásico.	5
Figura 1.6 Formas de onda del voltaje de salida del ciclo-convertidor con angulos de disparo de $\alpha=0^\circ$ y $\alpha=60^\circ$	6
Figura 1.7 Diagrama de un convertidor matricial.	6
Figura 1.8 Esquema general de un Inversor.	7
Figura 1.9 Diagrama de un Filtro activo.	8
Figura 1.10 Circuito de Interpolación de Doble Punto.	9
Figura 1.11 a) Circuito realimentado con histéresis. B) Comparador dinámico con histéresis.	10
Figura 1.12. a) Diagrama del PD b) Voltaje de entrada, c) Voltaje de salida d) Salida del PD.	12
Figura 1.13 Relación de cambio de la frecuencia ω_o , y el voltaje de control V_c	13
Figura 1.14 Ganancia del VCO de acuerdo a la relación de Voltaje de salida, v_o , y la frecuencia de este voltaje ω_o	14
Figura 1.15 Diagrama a bloques de un PLL.	14
Figura 1.16 Cambio de coordenadas por Transformación de Clarke.	15
Figura 1.17 Cambio de ejes en la transformada de Park.	16
Figura 2.1 Inversor Trifásico Tradicional.	20
Figura 2.2 Combinaciones válidas para un Inversor Trifásico Tradicional.	20
Figura 2.3 Posición de las 8 combinaciones válidas para los interruptores de un Inversor Trifásico Tradicional en el plano α - β	21
Figura 2.4 Cálculo de un vector arbitrario sumando los 2 vectores estacionarios más cercanos. ...	22
Figura 2.5 Formas de onda de voltaje a la salida de los puentes inversores para el sector 1.	25
Figura 2.6 Diagrama a Bloques del proceso de control SVM para un Inversor Trifásico Tradicional.	26
Figura 2.7 Esquema de señales de control y secuencia de interrupciones del Algoritmo para SVM.	27
Figura 2.8 Inversor híbrido multiniveles.	29
Figura 3.1 Diagrama a bloques del PLL Digital.	32
Figura 3.2 Transformaciones de Clark y Park con una Frecuencia de entrada de 65 Hz.	33
Figura 3.3 Transformaciones de Clark y Park con una Frecuencia de entrada de 60 Hz.	33
Figura 3.4 Ganancia del DCO de acuerdo a la relación del tiempo del Temporizador, y la frecuencia que genera a la salida.	34
Figura 3.5 Respuesta en frecuencia del compensador de [3-1].	34
Figura 3.6 Modelo Lineal del PLL.	35

Figura 3.7 Modelo Lineal del PLL con Laplace.....	35
Figura 3.8 Respuesta en Frecuencia de $G(s)$	36
Figura 3.9 Respuesta en frecuencia de Hs	36
Figura 3.10 Respuesta en frecuencia de lazo cerrado $F(s)$	37
Figura 3.11 Respuesta en frecuencia de $T(s)$	38
Figura 3.12 Respuesta en frecuencia de lazo cerrado Fs	38
Figura 3.13 Diagrama a Bloques del PLL.	39
Figura 4.1 Diagrama de Flujo que describe la forma de programación del algoritmo para un Inversor Trifásico Tradicional.	45
Figura 4.2 Diagrama a bloques en SIMULINK del PLL.....	48
Figura 4.3 Valor del Registro del <i>Timer</i> para una frecuencia fundamental del PLL de 62 Hz.	49
Figura 4.4 Valor del Registro del <i>Timer</i> para una frecuencia fundamental del PLL de 58 Hz.	49
Figura 4.5. Valor del Registro del <i>Timer</i> para una frecuencia fundamental del PLL de 60 Hz.	50
Figura 4.6 Diagrama a Bloques del programa.	51
Figura 5.5.1 a) Señales de control mostrando su frecuencia fundamental de 60 Hz b) Ampliación en tiempo de las señales de control.	55
Figura 5.2 Señales de control para el primer sextante de la SVPWM.....	56
Figura 5.3 Tiempos muertos en las señales de control.....	56
Figura 5.4 Compuerta AND como generador de combinación de tiempo muerto.....	56
Figura 5.5 Filtro activo de potencia.....	57
Figura 5.6 Inversor trifásico del filtro activo.	57
Figura 5.7 Diagrama a bloques del inversor.....	57
Figura 5.8 Voltajes V_{RN} , V_{SN} y V_{TN} obtenidos con SVPWM.....	58
Figura 5.9 Acercamiento de los voltajes V_{RN} , V_{SN} y V_{TN} obtenidos con SVPWM.	58
Figura 5.10. Tarjeta acondicionadora de voltaje.....	59
Figura 5.11. dsPIC en la tarjeta de desarrollo Explorer 16 con su conector de expansión.....	59
Figura 5.12. Comprobación de la transformación de Clarke.	60
Figura 5.13 Funcionamiento del PLL utilizando una fase y una señal cuadrada.....	61
Figura 5.14. Sincronización de la SVPWM con V_R a través del PLL digital.	62
Figura 5.15 Sincronización de la SVPWM.....	63
Figura 5.16 Acercamiento a las formas de onda sincronizadas por el PLL digital.....	63
Figura 5.17 Superposición de la señal del PLL y V_R	64
Figura 5.18. Sincronización de V_{RN} con la fase R de la fuente trifásica usando el PLL.....	65
Figura 5.19. Acercamiento de las señales sincronizadas.	65

Índice de Tablas

Tabla 2.1 Tabla de tiempos activos de un inversor trifásico.....	24
Tabla 4.1 Configuración del Timer 3 para el algoritmo de SVM para el inversor Trifásico.....	47
Tabla 4.2 Configuración del Modulo de comparación	47
Tabla 4.3 Frecuencia fundamental del PLL en relación al “Timer” del dsPIC.....	48
Tabla 4.4 Configuración del ADC para el algoritmo del PLL	52
Tabla 4.5 Configuración del DMA para el algoritmo del PLL.....	52

AGRADECIMIENTOS

Al Instituto Politécnico Nacional por apoyar en la construcción del presente trabajo, mi crecimiento profesional y por brindarme todos los medios para poder conseguir esta licenciatura. Esta institución es la madre de mis conocimientos.

A la Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Culhuacan, por apoyar mi superación con sus instalaciones y permitir mi estancia durante este tiempo.

Al CONACYT y al ICyTDF por recibirme para formar parte de este proyecto y apoyar económicamente el desarrollo de esta tesis.

Al Dr. Ismael Araujo Vargas por dirigir este proyecto, por haber compartido su conocimiento y por brindarme su amistad. Es un privilegio seguir en trabajando a su lado.

Al M. en C. Alejandro Tapia Hernández por apoyarme en el desarrollo de este trabajo, porque a lo largo de este tiempo pudo resolver muchas de mis dudas compartiendo conmigo su conocimiento y amistad a pesar de sus actividades.

DEDICATORIA

A mi Dios.

A mis padres Ariel Cano Gutiérrez y Mayra Pulido Apanco quienes siempre me brindaron todo su apoyo y amor. Gracias por el regalo de darme educación. Los amo.

A mi hermana Eli por ser mi mejor amiga. Por estar junto a mí durante toda su vida.

A mis abuelos Aarón, Libia y Clemencia, que en paz descansen y que desearon verme terminar esta etapa. A mi viejo querido Delfino Cano quien siempre me dio su cariño y me llena de consejos.

A mis amigos, Atoany, Jordan, Oso, Lalo, Chava, Jerry, Josh, Afro; quienes compartieron conmigo tantos momentos con muchos sinsabores y alegrías a lo largo de mi carrera.

A Flor que ha estado conmigo estos últimos años dándome su apoyo y cariño.

A mis amigos y compañeros en esta nueva etapa de mi vida, Niño, Onesimo, Panamá, Pancho, Jaz, Nancy, Alex, Alberto, Cid y Jose Luis.

RESUMEN

En esta tesis se muestra el análisis, diseño y construcción de un PLL para un convertidor inversor controlado con Modulación de Vector de espacio. Describe la manera en la que se controla un inversor trifásico tradicional con esta modulación y se desarrolla un programa que se implementa en un microcontrolador. Además, se hace un recuento de las principales herramientas para sincronizar circuitos. De esta manera se explica el funcionamiento de un PLL y se desarrolla uno mismo de manera digital implementado en un microcontrolador. Finalmente se muestran los resultados obtenidos de manera simulada y experimental. Este último apartado demuestra el correcto funcionamiento del PLL.

CAPÍTULO 1



ESIME

1. Estado del Arte

1.1.Introducción.

En este capítulo se describe un breve marco teórico de los tipos de convertidores que trabajan alimentados en CA (*Corriente Alterna*). También se hace una revisión de algunas de las técnicas y circuitos de sincronización de los convertidores con fuente de suministro en CA. Además, se describen las herramientas básicas para controlar estos circuitos, como por ejemplo las transformaciones de Clarke y Park.

1.2. Estado del arte relativo a Convertidores electrónicos de energía que operan con suministro de CA.

Los convertidores que operan con CA pueden clasificarse según diferentes criterios. El criterio más comúnmente utilizado es agruparlos dependiendo del tipo de forma de onda de entrada o salida a operar. Básicamente, según este criterio pueden establecerse cuatro grandes grupos:

- Rectificadores o convertidores CA-CD (Corriente Directa).
- Ciclo-convertidores o convertidores CA-CA.
- Inversores o convertidores CD-CA.
- Filtros Activos.

1.2.1. Rectificadores .

En este apartado se inicia con una descripción general de los rectificadores. La distribución de la energía eléctrica se hace, esencialmente, en corriente alterna, debido principalmente a la facilidad de adaptación a diferentes niveles de voltaje por medio de transformadores. Sin embargo, en muchas aplicaciones, especialmente en aparatos electrodomésticos, la carga requiere una corriente continua. La conversión CA-CD es realizada por convertidores comúnmente denominados rectificadores. Por tanto, un rectificador es un convertidor electrónico cuya función es convertir la corriente alterna a continua.

1.2.1.1. Rectificadores naturalmente controlados.

En el momento de llevar a cabo la rectificación, se utilizan elementos electrónicos que permiten el paso de la corriente en un sentido, permaneciendo bloqueado cuando se le aplica una tensión con polaridad opuesta. Para ello en los rectificadores no controlados, el componente más adecuado y utilizado es el diodo.

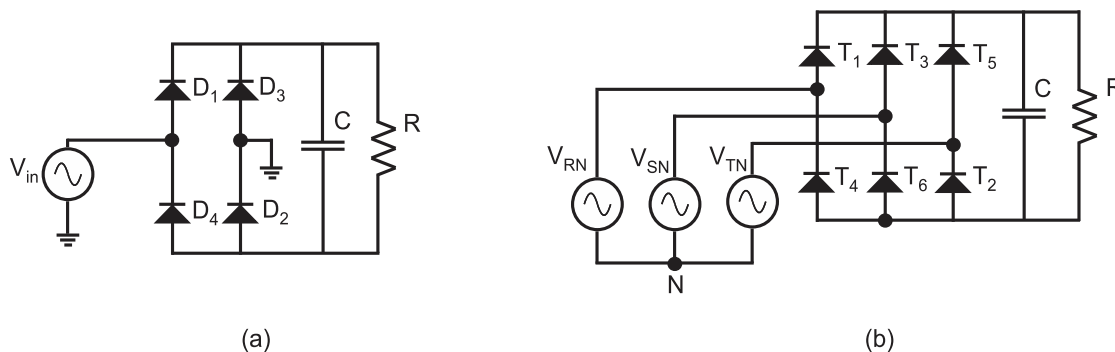


Figura 1.1 a) Diagrama rectificador monofásico no controlado b) Diagrama rectificador trifásico no controlado.

La Figura 1.1a muestra el diagrama del rectificador de onda completa. Por otro lado, la Figura 1.1b muestra un rectificador trifásico. Ambos cumplen con la misma función con la diferencia de que éstos últimos son alimentados por fuentes trifásicas, por lo que son más eficientes y pueden manejar grandes potencias, ya que en su salida presentan menor rizado de la señal. Son utilizados principalmente en la industria para producir voltajes y corrientes continuos que generalmente impulsan cargas de gran potencia, como motores CD [1-2].

1.2.1.2. Rectificadores controlados.

El rectificador trifásico controlado, que se observa en la Figura 1.2, permite variar el voltaje promedio de salida, empleando para su funcionamiento tiristores de potencia con los cuales se puede variar el ángulo de disparo y la potencia entregada a la carga. Los tiristores se disparan a un intervalo de $\frac{\pi}{3}$. La frecuencia del rizo de voltaje en la salida es seis veces la frecuencia de la fuente de entrada ($6f_s$), y el filtrado resulta más complejo en comparación con el de los convertidores naturalmente controlados[1-2,3].

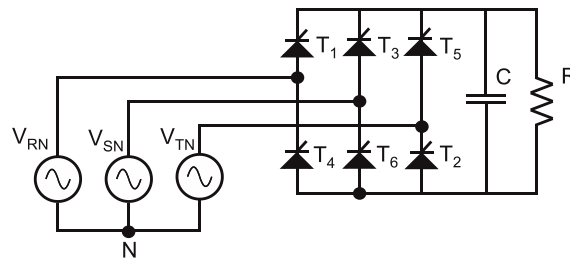


Figura 1.2 Diagrama de un rectificador trifásico controlado.

Controlando el ángulo de disparo de los tiristores se puede controlar el valor medio de la tensión en la carga. El ángulo de disparo deseado se obtendrá generando los pulsos de disparo, que se aplican en la puerta de los tiristores, en el momento adecuado.

1.2.1.3. Rectificadores activos.

La estrategia de funcionamiento de estos filtros activos se basa en la conexión en la barra de un transformador el cual es alimentado por un puente inversor a partir de un bus de CD. El inversor de compensación es controlado por un microprocesador y requiere de la medición de la corriente inyectada por el inversor y la corriente armónica a compensar. En los últimos años, los puentes rectificadores con control por ancho de pulso (PWM), han sido introducidos con la finalidad de realizar procesos de conversión AC-DC de alta calidad. Estos circuitos sólo permiten reducir el contenido de armónicos al sistema de CA de tal manera que el controlado del factor de

potencia en la barra de alimentación del puente y el voltaje de salida pueden ser controlados. En la Figura 1.3 se muestra un ejemplo de la topología básica de un rectificador activo.

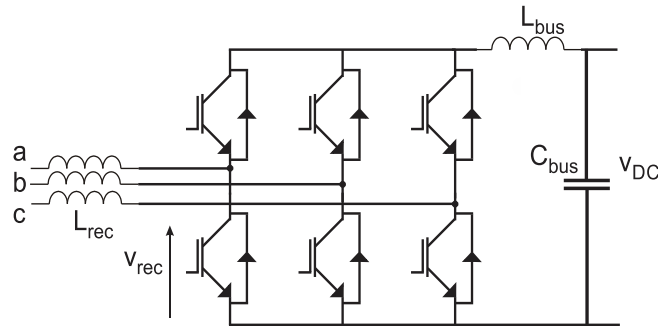


Figura 1.3 Diagrama de un rectificador activo.

1.2.1.4. Resumen.

De lo anterior se puede concluir que las últimas 2 categorías requieren de señales de control en sincronía y fase determinadas con la frecuencia del suministro para que estos circuitos funcionen apropiadamente.

1.2.2. Ciclo-convertidores.

En contraste con los convertidores de voltaje que operan con corriente alterna a una frecuencia constante, los ciclo-convertidores realizan una transformación CA-CA cambiando la frecuencia de la tensión de salida. La aplicación típica de este circuito es en el control de velocidad de motores síncronos y de inducción por medio de variación de frecuencia. A continuación se mencionan algunos de los principales usos para estos convertidores.

- Impulsores de molinos de cemento
- Control de motores de inducción
- Propulsión de vehículos.
- Calentamiento por alta frecuencia.
- Molinos para triturar minerales.
- Excavadoras de minas.

1.2.2.1. Ciclo-convertidores a base de inversores.

Tradicionalmente la conversión CA-CA se lleva a cabo en dos etapas: primero se realiza una conversión CA-CD con un rectificador y luego se realiza una conversión CD-CA mediante un inversor. La Figura 1.4

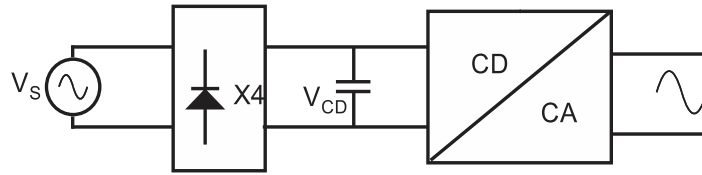


Figura 1.4 Diagrama general de un inversor de CA-CA utilizando un inversor.

1.2.2.2. Ciclo-convertidores en base a tiristores.

Estos ciclo-convertidores son diseñados principalmente para evitar las dos etapas de conversión mencionadas en el apartado anterior, lo que eleva el rendimiento y reduce la cantidad de componentes necesarios. Dependiendo del número de fases del suministro se tienen diferentes topologías de este convertidor, es decir, existen convertidores monofásico y trifásicos.

La Figura 1.5 muestra un ciclo-convertidor en base a tiristores monofásico. El circuito consiste en dos rectificadores conectados al mismo suministro.

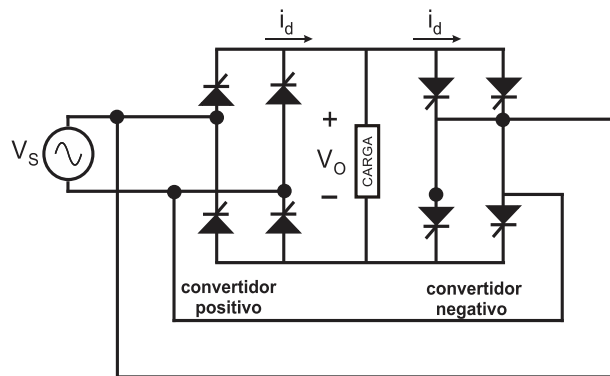


Figura 1.5 Diagrama de un ciclo-convertidor monofásico.

En el primer semiciclo se disparan los tiristores del puente llamado convertidor + quedando encendidos por una cierta cantidad de ciclos de la señal de entrada. En el segundo semiciclo funciona el convertidor - durante la misma cantidad de ciclos que el convertidor+. Así la carga recibe una tensión alterna compuesta por semiciclos de la tensión de entrada con una frecuencia que es submúltiplo de la original. Cabe destacar que para ajustar el valor RMS de la tensión de salida se varía el ángulo de disparo de los tiristores, como se observa en la Figura 1.6, con ángulos de 0° y 60° .

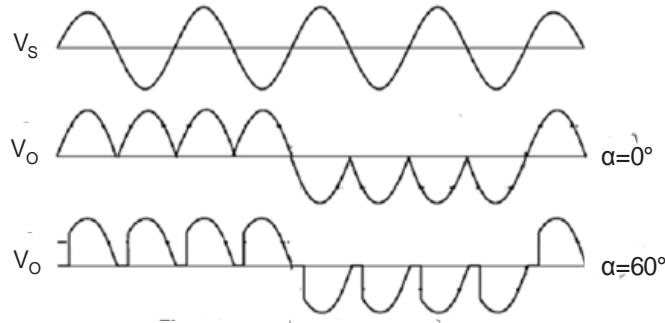


Figura 1.6 Formas de onda del voltaje de salida del ciclo-convertidor con ángulos de disparo de $\alpha=0^\circ$ y $\alpha=60^\circ$.

1.2.2.3. Convertidores Matriciales.

Un convertidor matricial (CM) es un convertidor CA-CA trifásico que consiste en un arreglo de interruptores bidireccionales que conectan una carga trifásica e la línea de alimentación. El elemento clave en el CM es el control de los interruptores bidireccionales, mostrado en la Figura 1.7, que operan a alta frecuencia. Estos son controlados de tal manera que el CM pueda suministrar a la carga un voltaje de amplitud y frecuencia variables.

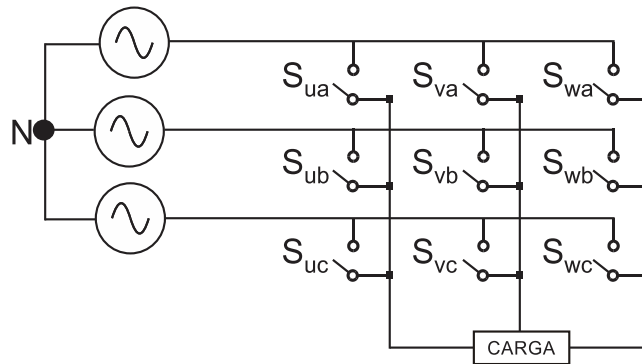


Figura 1.7 Diagrama de un convertidor matricial.

Los voltajes de salida son generados a través de patrones de modulación PWM (por sus siglas en inglés Pulse Width Modulation, Modulación de Ancho de Pulso), similares a los utilizados en inversores convencionales, excepto porque la entrada es una fuente de alimentación trifásica en lugar de un voltaje de CD. Una cualidad de estos circuitos es que se emplean arreglos complejos de transistores por cada interruptor.

1.2.2.4. Resumen.

Se concluye que las señales de control que los ciclo-convertidores también requieren una sincronía con el suministro para generar las formas de onda apropiadas como por ejemplo los

convertidores matriciales. Sin embargo para el caso de convertidores de dos etapas, no es de alta importancia sincronizar el inversor con la línea a menos que la aplicación requiera que la salida tenga sincronía con el suministro.

1.2.3. Inversores.

1.2.3.1. Inversores.

Los convertidores de CD-CA se conocen como inversores. Se trata de un circuito encargado de generar señales de CA (preferentemente sinusoidales) que sean capaces de proporcionar suficiente potencia a la carga a partir de un suministro de CD mediante la utilización de diversas técnicas de modulación. La Figura 1.8 muestra el diagrama a bloques de un inversor [1-1].

Existen diferentes tipos de control de inversores así como de topologías, por ejemplo, resonantes, PWM, multiniveles, multipulsos, entre otros.

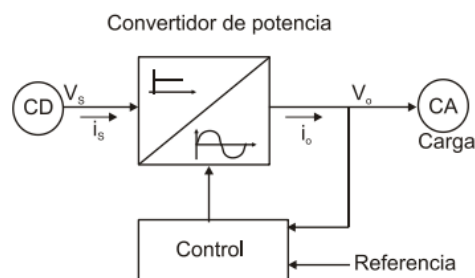


Figura 1.8 Esquema general de un Inversor.

En ciertas otras aplicaciones es imprescindible que la frecuencia fundamental del inversor esté sincronizada a la frecuencia de suministro, para esto se utilizan técnicas de sincronización. Algunas de las aplicaciones son los sistemas ininterrumpibles de energía, control de motores de CA, control de motores de inducción en generadores eólicos, distribución de energía eléctrica para sistemas de transporte, aeronaves entre otras. Dentro de las características que se desea tener en este tipo de convertidores es una alta eficiencia, peso ligero, tamaño reducido y bajo costo.

1.2.3.2. Resumen.

Se concluye que aunque un inversor puede determinar la frecuencia de salida, a veces es necesario sincronizarlo con la red eléctrica para aplicaciones específicas, por esta razón se desarrollan herramientas de sincronización que no solo se refieran a la frecuencia sino también a la fase.

1.2.4. Filtros Activos.

Debido al alto problema que causan los armónicos en los sistemas de potencia, se han desarrollado soluciones como lo son los filtros activos.

1.2.4.1. Esquema General.

Generalmente, un filtro activo se considera como una fuente de corriente conectada en paralelo con la carga. El enfoque se basa en el principio de la inyección de armónicos de corriente de magnitud y de fase inversa a la de los armónicos de la corriente de carga. La Figura 1.9 muestra el diagrama a bloques de un filtro activo paralelo [1-4].

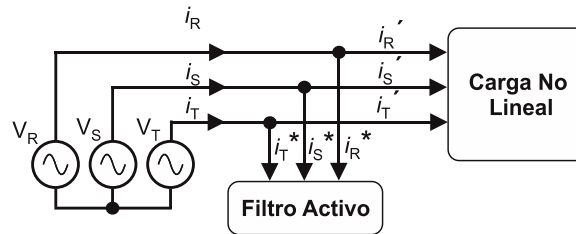


Figura 1.9 Diagrama de un Filtro activo.

La carga no lineal hace que la corriente suministrada por el sistema trifásico se distorsione, en señales con un gran número de armónicos, el filtro inyecta armónicos de corriente, de magnitud y de fase inversa a los armónicos de la carga, haciendo que las señales de corriente sean sinusoidales.

1.2.4.2. Clasificación.

Los filtros activos pueden clasificarse principalmente dependiendo de cómo se conecta el filtro respecto a la carga. Se pueden distinguir entonces filtros en serie, paralelo o mixto. Desde un punto de vista práctico, cada una de estas topologías actúa de forma distinta [1.-9]. Para la reducción de distorsión de la corriente en el punto donde se conecta el filtro se acostumbra a usar un filtro del tipo paralelo; mientras que para la reducción de la distorsión de la tensión en la carga se utilizan filtros en serie.

1.2.4.3. Resumen.

Se concluye que al tener que estar trabajando con la corriente de la línea, el filtro activo debe estar sincronizado con la misma. Esto es imprescindible ya que al tener que inyectar armónicos que cancelen a los no deseados, es necesario conocer la fase y frecuencia de la línea.

1.3.Estado del arte relativo a Técnicas de sincronización de circuitos.

Existen dos técnicas generales de sincronización, éstas son el detector de cruce por cero o ZCD (por sus siglas en inglés Zero Crossing Detector) y el PLL (por sus siglas en Inglés Phase Locked Loop). La primera por ejemplo, es muy usada en aplicaciones a sistemas de voltaje monofásicos mientras que las últimas utilizan técnicas basadas en espacios vectoriales que hacen que el sistema se sincronice con voltajes trifásicos.

1.3.1. Detectores de cruce por cero.

El ZCD, es el método más común para la medición de la frecuencia en una señal periodica. Este circuito produce un cambio de estado en su salida cuando la señal de entrada cruza por cero volts [1-6].

El circuito mostrado en la Figura 1.10, llamado de Interpolación de doble punto, usa dos optoacopladores para compensar variaciones en niveles de sensibilidad y tiempos de retraso en la conmutación. La implementación de este diseño identifica dos puntos en una onda sinusoidal, el primero, justo antes del cruce por cero y el segundo justo después del mismo cruce por cero.

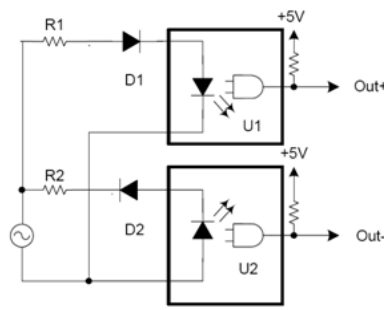


Figura 1.10 Circuito de Interpolación de Doble Punto.

De igual manera es típico usar circuitos similares al mostrado en la Figura 1.11a para transformar los niveles de voltaje y prever inmunidad al ruido usando histéresis. Los fabricantes de estos circuitos especifican un voltaje máximo negativo en ambas entradas, positiva y negativa de 0.3 volts. Los diodos Schottky D1 y D2, mantiene el voltaje de entrada al límite del comparador de entrada por debajo de los niveles positivo y negativo. Es importante notar que este circuito es usado tanto con voltajes monofásicos y trifásicos bajo la desventaja de posibilidad de operación incorrecta en caso de distorsión, desbalance y/o desfase.

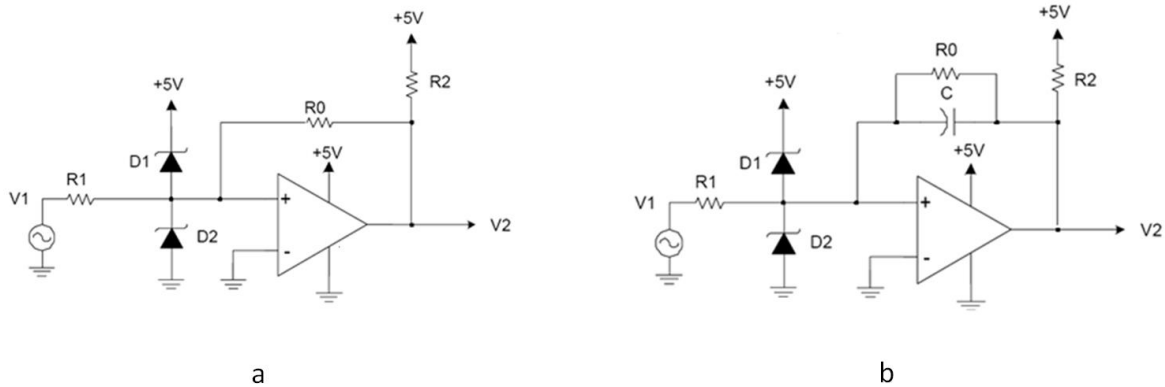


Figura 1.11 a) Circuito realimentado con histéresis. B) Comparador dinámico con histéresis.

El circuito mostrado en la Figura 1.11b es sugerido para asegurar rápidas transiciones y prevenir múltiples pulsos de salida.

1.3.2. El PLL.

Este circuito, también llamado Sincronizador de Fase Cerrada, es un sistema retroalimentado cuyo objetivo principal consiste en la generación de una señal de salida con amplitud fija y frecuencia coincidente con la de entrada, dentro de un margen determinado. El circuito PLL comprende las siguientes tres etapas fundamentales:

- Detector de Fase (PD, por sus siglas en inglés Phase Detector). Genera una señal de salida que depende del valor absoluto del desfase entre las señales de salida y de entrada. En algunos casos, esta etapa está constituida por un multiplicador.
- Compensador (Filtro de Lazo). Ayuda a converger al sistema filtrando a la componente de baja frecuencia de la salida del PD y proporcionando la señal de control para el oscilador.
- Oscilador controlado por tensión (VCO, por sus siglas Voltage Controlled Oscillator). Genera la frecuencia de salida, con tensión dependiente de la tensión de salida del filtro PLL.

1.3.2.1. Detector de Fase.

De acuerdo con [1-1], el modelo Lineal para un Detector de Fase es:

$$v_d = K_d \theta_e + V_{do} \quad (1.5)$$

donde K_d es la ganancia del PD, θ_e es el error de fase de la salida del VCO respecto a la señal de entrada, y V_{do} es el nivel de CD del voltaje. El valor de θ_e para el cual el Modelo Lineal es válido es llamado el rango del PD.

Un multiplicador actúa como un PD por la identidad trigonométrica

$$\sin(A) \cos(B) = 0.5 \sin(A - B) + 0.5 \sin(A + B) \quad (1.6)$$

Si las señales de entrada son:

$$v_i = V_i \sin(\omega_i t) \quad (1.7a)$$

$$v_o = V_o \cos(\omega_i t - \theta_e) \quad (1.7b)$$

La salida del multiplicador será:

$$v_d = K_m v_i v_o \quad (1.8)$$

donde K_m es una constante asociada con el multiplicador. Esto puede verse en el diagrama de la Figura 1.11a. Las unidades de K_m son volts^{-1} puesto que v_d está dada en volts. De acuerdo a las ecuaciones 1.6, 1.7a, 1.7b y 1.8 se obtiene que:

$$v_d = K_m V_i V_o \sin(\theta_e) + 0.5 K_m V_i V_o \sin(2\omega_i t - \theta_e) \quad (1.9)$$

Los dos términos de la ecuación 1.8 son evidentemente dos componentes sinusoidales. Para una θ_e constante la salida del PD deberá ser constante de acuerdo a la ecuación 1.8, y el segundo término de esta ecuación mantiene una frecuencia $2\omega_i t$. En la mayoría de aplicaciones de PLLs, esta frecuencia es lo suficientemente alta por lo que el segundo término no tiene efecto, en otros casos este término es removido por un filtro. En cualquier caso, solo el primer término se considera como la salida del PD:

$$v_d = 0.5 K_m V_i V_o \sin(\theta_e) \quad (1.10)$$

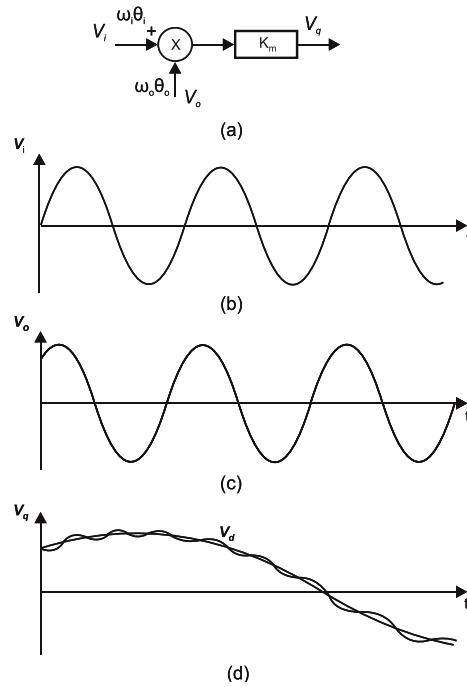


Figura 1.12. a) Diagrama del PD b) Voltaje de entrada, c) Voltaje de salida d) Salida del PD.

De esta manera la salida del PD es en realidad el promedio de la salida completa de v_d y puede ser escrita como

$$v_d = V_{dm} \sin(\theta_e) \quad (1.11)$$

Donde el máximo valor de v_d es:

$$V_{dm} = 0.5K_m V_i V_o \quad (1.12)$$

Esta característica sinusoidal es mostrada en la Figura 1.12d. Para valores pequeños de θ_e , $\sin(\theta_e) \approx \theta_e$ y por tanto:

$$v_d = 0.5K_m V_i V_o \theta_e \quad (1.13a)$$

Comparando las ecuaciones 1.8 y 1.13a, se observa que la ganancia del PD para pequeños valores de θ_e es:

$$K_d = 0.5K_m V_i V_o \quad (1.13b)$$

Se puede notar que la ganancia del PD depende de la amplitud de las señales de entrada.

1.3.2.2. EL VCO.

El VCO es un oscilador electrónico diseñado para ser controlado a alguna frecuencia de oscilación en función del voltaje de entrada. El VCO oscila libremente a una frecuencia, determinada por una red RC o LC, llamada frecuencia central ω_c . Esta frecuencia es comparada con la frecuencia ω_i de la señal de entrada del detector de fase, el cual entrega una mezcla de ambas $\omega_i - \omega_o$ o $\omega_o - \omega_i$ dependiendo de cual es mayor. Los productos de altas frecuencias como $\omega_i + \omega_o$, $2\omega_i$, $2\omega_o$, etc. son eliminados por el compensador $H(s)$.

Una característica típica de un VCO es que su frecuencia ω_o es una función lineal del voltaje de control v_c como se observa en la Figura 1.13.

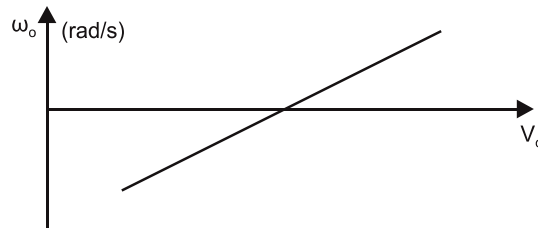


Figura 1.13 Relación de cambio de la frecuencia ω_o , y el voltaje de control V_c .

Si la frecuencia de la señal V_q es lo suficientemente baja para que el filtro pasabajas no lo atenúe ni la desfase en exceso, v_c controlará el VCO, tendiendo a reducir la diferencia de frecuencias hasta que se igualen.

Una vez que se sincronizan v_o y v_i , es decir $\omega_i = \omega_o$, el detector de fase entrega una tensión V_c , con una componente continua estable necesaria para que el VCO iguale la frecuencia de la señal de referencia. Esto es, como se observa en la Figura 1.13, si el voltaje, V_c , varía, entonces la frecuencia, ω_o , también varía de acuerdo a la relación que muestra la pendiente. En este caso se establece una diferencia de fase θ_e para producir la tensión V_e antedicha.

El rango de captura de un PLL depende de las características del VCO, de acuerdo al voltaje V_c y la frecuencia de operación del VCO es posible observar que existe una relación como se muestra en la ecuación 1.14.

$$\Delta\omega_o = K_o\Delta v_o \quad (1.14)$$

En donde $\Delta\omega_o$ es el rango de captura del PLL, Δv_o es el rango de voltaje en que opera el VCO de acuerdo a $\Delta\omega_o$, mientras que K_o es la pendiente que representa la ganancia del VCO.

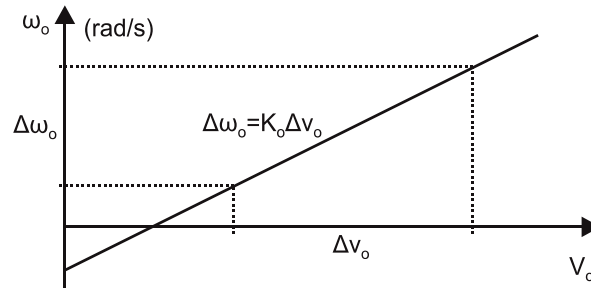


Figura 1.14 Ganancia del VCO de acuerdo a la relación de Voltaje de salida, v_o , y la frecuencia de este voltaje ω_o .

1.3.2.3. Compensador.

El compensador tiene dos importantes funciones. Primero, elimina el ruido y cualquier componente de alta frecuencia de la salida del detector de fase, tales como $\omega_i + \omega_o$, armónicas de ω_i y ω_o , etc., dejando pasar solo la componente de baja frecuencia $\omega_i - \omega_o$ y $\omega_o - \omega_i$ cuando se está adquiriendo el estado fijo, una señal continua, o pequeñas variaciones cuando el PLL ya está en estado fijo. Segundo, es el bloque más importante en la determinación de las características dinámicas del lazo, rango de captura y respuesta en frecuencia.

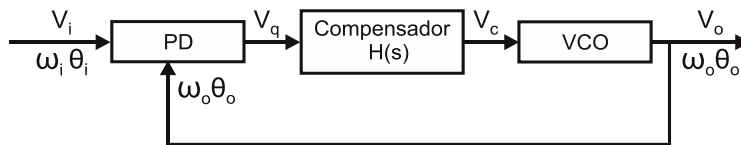


Figura 1.15 Diagrama a bloques de un PLL.

1.3.3. Herramientas modernas para manipular y procesar señales trifásicas en convertidores de energía.

1.3.3.1. Transformación de Clarke.

La transformación de Clarke transforma las variables eléctricas de un sistema trifásico con fases R-S-T en otras α - β que representan un sistema bidimensional de coordenadas fijo. Geométricamente supone la proyección de los ejes trifásico R-S-T sobre dos ejes α - β .

Para entender el principio de operación de esta transformación se debe considerar un sistema trifásico simétrico y balanceado, en él se puede demostrar que las corrientes y tensiones trifásicas del sistema se pueden representar mediante un vector espacial de amplitud y velocidad constantes en el plano α - β como el mostrado en la Figura 1.15. En primer lugar definimos una tensión trifásica de amplitud constante cuyas componentes sinusoidales, R, S y T, están desfasadas 120° con una amplitud máxima V_{max} , tal como se muestra en la ecuación 1.1

$$v(t) = \begin{bmatrix} v_R \\ v_S \\ v_T \end{bmatrix} = \begin{bmatrix} V_{max} \sin(\omega t) \\ V_{max} \sin(\omega t - 2\pi/3) \\ V_{max} \sin(\omega t + 2\pi/3) \end{bmatrix} \quad (1.1)$$

donde se cumple la siguiente condición:

$$V_R + V_S + V_T = 0 \quad (1.2)$$

Por convención [1-5], se asume que la proyección del eje de la fase R en el plano α - β forma el eje de referencia α . El segundo eje del plano α - β está definido de tal manera que es ortogonal al eje α , y el producto cruz $\alpha \times \beta$ forma un tercer eje, que es normal al plano α - β , conocido como nulo o de secuencia cero, tal como se ve en la Figura 1.16

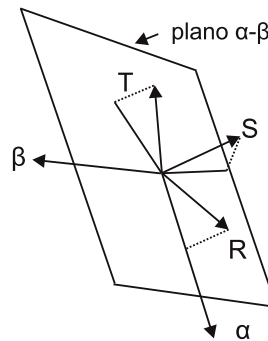


Figura 1.16 Cambio de coordenadas por Transformación de Clarke.

De esta manera, la transformación de Clarke está dada por la ecuación 1.2.

$$V(\alpha, \beta) = \frac{2}{3} \begin{bmatrix} 1 & -1/2 & -1/2 \\ 0 & \sqrt{3}/2 & -\sqrt{3}/2 \end{bmatrix} \begin{bmatrix} V_R \\ V_S \\ V_T \end{bmatrix} \quad (1.3)$$

1.3.3.2. Transformación de Park.

Una vez hecho el cambio de ejes, las tensiones girarán con velocidad angular constante ω sobre el plano definido por los ejes α - β . El principal interés en la transformación de Park es establecer los vectores espaciales de voltaje en una determinada posición, de modo que se busque identificar una diferencia entre dos diferentes velocidades angulares. La transformación de Park convierte dos fases de un plano estacionario (α - β) en dos fases de un plano rotante (d - q). La Figura 1.17 muestra cómo se realiza el cambio de ejes.

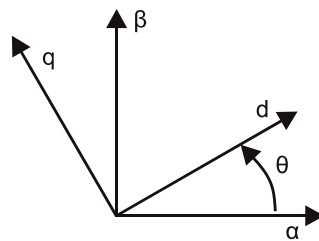


Figura 1.17 Cambio de ejes en la transformada de Park.

A partir del ángulo de rotación de los ejes d - q , se puede definir la ecuación 1.3 que permite el cambio del plano α - β al plano d - q .

$$V(d, q) = \begin{bmatrix} \cos \theta_o & \text{sen } \theta_o \\ -\text{sen } \theta_o & \cos \theta_o \end{bmatrix} \begin{bmatrix} V_\alpha \\ V_\beta \end{bmatrix} \quad (1.4)$$

1.4. Resumen del estado del arte.

Existen diferentes tipos de convertidores los cuales son necesariamente sincronizados a la red eléctrica para generar voltajes síncronos a la misma. De esta manera, es necesario conocer y entender las estrategias de modulación de cada uno de estos convertidores. Además, es de suma importancia conocer las técnicas de sincronización existentes.

Las dos principales técnicas de sincronización son el Detector de cruce por cero (ZCD) y el PLL. El ZCD suele ser empleado para sincronización en sistemas monofásicos, pero es muy sensible a las variaciones causadas por ruido eléctrico, por otra parte el PLL se utiliza en sistemas trifásicos empleando técnicas como las transformaciones de Clarke y de Park las cuales crean componentes virtuales para sincronizarse a los tres voltajes.

1.5. Objetivo.

Analizar, diseñar y construir un PLL Digital implementándolo en un microcontrolador que sincronice un inversor con modulación de vector de espacio, a la red eléctrica trifásica

convencional. Este circuito puede ser utilizado en otro tipo de convertidores con diferentes topologías y cambiando la estrategia de modulación. Cabe destacar que la lógica de éste, será diseñada para poder ser implementada en algún otro microcontrolador con mejores características y velocidades similares.

1.6. Bosquejo.

El desarrollo de esta tesis se basa primeramente en la revisión literaria , capítulo 1, de los tipos de convertidores que utilizan voltajes de CA y que por necesidad deben ser sincronizados a alguna frecuencia externa, siendo la frecuencia de la red eléctrica la más comúnmente utilizada. Además se hace hincapié en los principales métodos de sincronización existentes y se realiza una comparación de estos. Se presentan además algunas técnicas numéricas modernas utilizadas en los métodos de sincronización.

El capítulo 2 describe el principio de operación de la Modulación de vector de espacio. Se hace referencia a un inversor tradicional y se describe paso a paso la forma de obtener los tiempos activos de cada vector estacionario , además se muestra a detalle el procedimiento a seguir para implementar esta modulación en base a un algoritmo de control.

El capítulo 3 explica el funcionamiento de un PLL y hace una descripción de sus componentes, Además se muestra paso a paso el procedimiento para calcular los valores de los componentes a una frecuencia necesaria de operación que estará sincronizada y dependerá de la Modulación de vector de espacio. Se desarrolla también un algoritmo de implementación en base a un diagrama a bloques.

En el capítulo 4 se desarrolla la implementación digital de los algoritmos descritos en los capítulos 1 y 2. Se hace también una comprobación numérica de los valores obtenidos en los cálculos previos.

El capítulo 5 muestra las conclusiones generales de la tesis. En este apartado se pueden observar todas las formas de onda utilizadas en el desarrollo del PLL y la Modulación. Se pueden observar fotografías de los circuitos desarrollados y la formas de onda que comprueban los resultados experimentales de este trabajo.

1.7.Referencias.

- [1-1] N. Mohan, T.M. Undeland and W.P. Robbins, "Power Electronics", 2nd Edition, Wiley, USA, 1989.
- [1-2] M.h: Rashid, "Electrónica de Potencia", 3rd Edition, Pearson-Prentice Hall, México, 2004.
- [1-3] J.G. Kassakian, M.F. Schlecht G.C. Verghese, "Principles of Power Electronics", Addison-Wesley, 1991.
- [1-4] Benslimane T., Aliouane K., "A new optimized SVPWM technique control for autonomous parallel active filter.", Power electronics Laboratory, E. M. P.
- [1-5] D. Grahame Holmes, Thomas A. Lipo, "Pulse width modulation for power converters Principles and Practice", IEEE series on Power Engineering, Mohamed E. El-Hawary, series editor.
- [1-6] Wall R.W., "Simple methods for detecting zero crossing", Senior Member, IEEE.
- [1-7] Pindado R., "Phase Locked-Loop (PLL): Fundamento y aplicaciones", Universidad Politécnica de Cataluña.
- [1-8] Wolaver, Dan H. "Phase-Looocked Loop Circuit design". Prentince Hall.
- [1-9] Petit Suárez, J. Farith, "Topologías y algoritmos de control para filtros activos aplicados a la mejora de la calidad de suministro eléctrico", Universidad Carlos III de Madrid.

CAPÍTULO 2

2. Modulación de vector de espacio de ancho de pulso.

2.1.Introducción.

En este capítulo se describe el principio de operación de la Modulación de Vector de Espacio de Ancho de Pulso (SVPWM, por sus siglas en Inglés Space Vector Pulse Width Modulation). Esta descripción parte del principio de operación del Inversor Trifásico Tradicional, el cual se utiliza para el control de dispositivos semiconductores. La principal característica del SVPWM es el uso de un índice de modulación para controlar la amplitud del voltaje a la salida del inversor.

2.2.PWM de Vector de Espacio Tradicional.

2.2.1. Principio de Operación.

El principio de la Modulación de Vector de Espacio (SVM) se basa en un inversor tradicional como el mostrado en la Figura 2.1, en donde existen ocho posibles combinaciones, las cuáles se muestran en la Figura 2.2. Dos de estos estados (\overline{SV}_0 y \overline{SV}_7) corresponden a un voltaje nulo de salida, también conocido como Vector Neutro; las seis combinaciones restantes (\overline{SV}_1 a \overline{SV}_6) se conocen como Vectores Estacionarios [2-1].

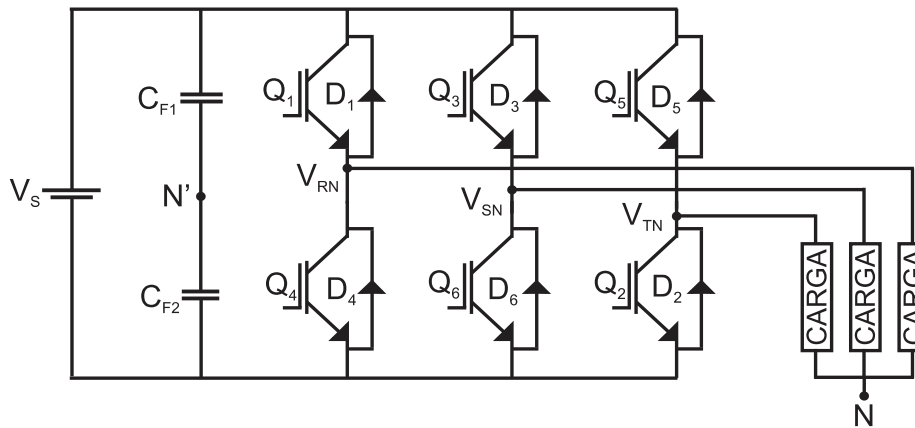


Figura 2.1 Inversor Trifásico Tradicional.

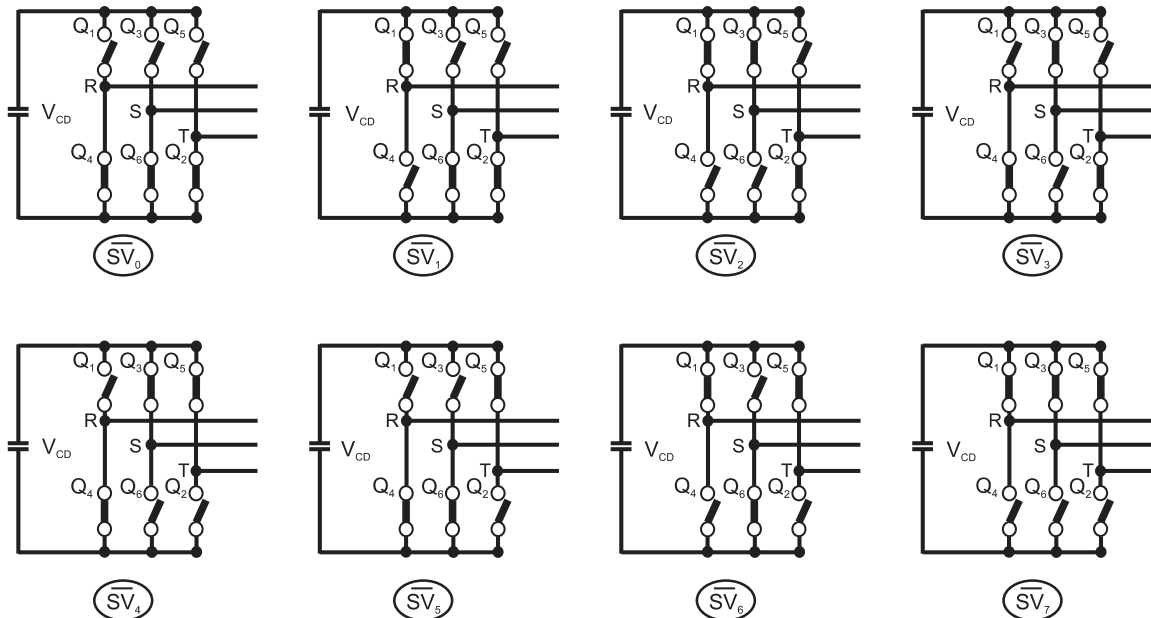


Figura 2.2 Combinaciones válidas para un Inversor Trifásico Tradicional.

Las ocho posibles combinaciones de la Figura 2.2 pueden formar un vector de Voltaje de Salida $\vec{V}_o = [v_{RN}, v_{SN}, v_{TN}]$, referido al nodo N' del circuito de la Figura 2.1, el cual puede ser transformado al plano α - β mediante la siguiente ecuación de transformación:

$$V(\alpha, \beta) = \frac{2}{3} \begin{bmatrix} 1 & -1/2 & -1/2 \\ 0 & \sqrt{3}/2 & -\sqrt{3}/2 \end{bmatrix} \begin{bmatrix} V_R \\ V_S \\ V_T \end{bmatrix} \quad (2.1)$$

donde $V(\alpha, \beta)$ es un espacio vectorial de dos dimensiones [2-2]. La Figura 2.3 muestra la transformación en el plano α - β en donde se presentan los 8 vectores de salida de espacio vectorial. En este mismo plano puede apreciarse que la magnitud de los Vectores Estacionarios es de $V_M = \frac{2}{3} V_s$, la cual está dada por la ecuación (2.1). Los vectores estacionarios están situados a 60° de separación a lo largo del plano α - β generando 6 sectores, con una dirección en sentido opuesto al de las manecillas del reloj.

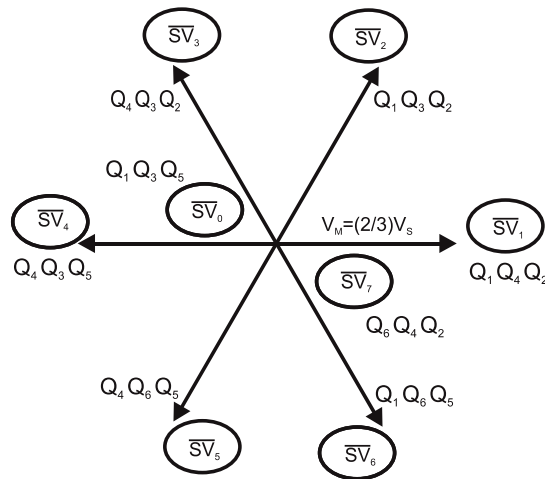


Figura 2.3 Posición de las 8 combinaciones válidas para los interruptores de un Inversor Trifásico Tradicional en el plano α - β .

Una vez identificados los 6 vectores estacionarios, se forma el vector arbitrario \vec{V}_o^* para cualquier sector, $S_1 - S_6$, con la capacidad de cambiar su amplitud y su ángulo. El vector arbitrario \vec{V}_o^* está compuesto de cuatro vectores locales dependiendo del sector en que se encuentre, es decir, se construye por proyecciones de los vectores estacionarios de voltaje y los vectores neutros de voltaje. Por ejemplo, en la Figura 2.4 se muestra el vector arbitrario \vec{V}_o^* para el Sector 1, el cual se construye con las proyecciones de los vectores estacionarios de voltaje más cercanos, \vec{SV}_1 y

\overline{SV}_2 . Los vectores de proyección que forman \overline{V}_0^* cambian dependiendo la posición angular en que se encuentre θ , donde $\theta = \omega_o t$, $\omega_o = 2\pi f_o$ y f_o es la frecuencia de salida del inversor. La amplitud máxima que puede tomar el vector \overline{V}_0^* es V_M , y corresponde a la distancia que existe desde el origen del plano α - β hasta el punto medio de la unión de los vértices del hexágono causado por los vectores de espacio estacionarios. Por lo tanto \overline{V}_0^* debe tener un valor menor a V_M .

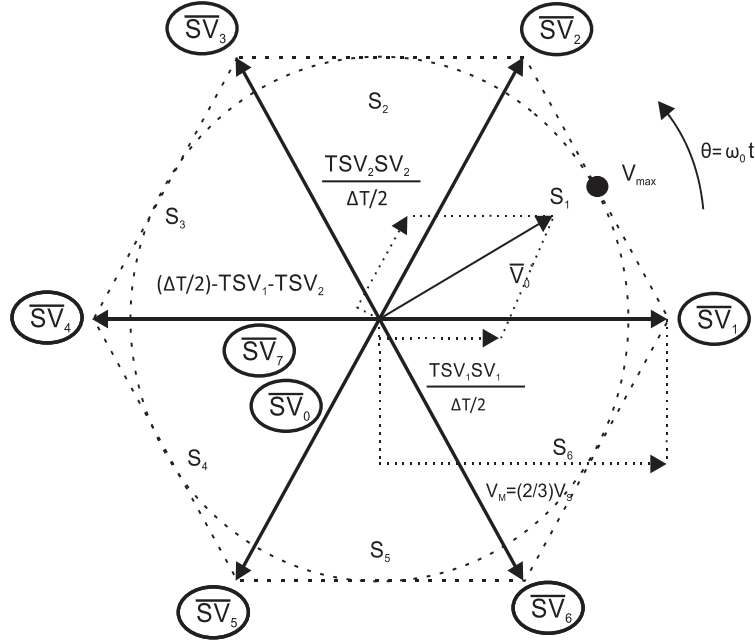


Figura 2.4 Cálculo de un vector arbitrario sumando los 2 vectores estacionarios más cercanos.

Promediando a \overline{V}_0^* para el periodo de conmutación $\Delta T/2$ de los vectores estacionarios, la suma geométrica mostrada en la Figura 2.3 se puede expresar matemáticamente como se muestra en la ecuación (2.2).

$$\overline{V}_0^* = \frac{T_{SV1}}{\Delta T/2} \overline{SV}_1 + \frac{T_{SV2}}{\Delta T/2} \overline{SV}_2 \quad (2.2)$$

Cuatro diferentes periodos de tiempo componen a $\Delta T/2$, estos tiempos se muestran en la ecuación (2.3).

$$\frac{\Delta T}{2} = T_{SV1} + T_{SV2} + T_{SV01} + T_{SV02} \quad (2.3)$$

donde $T_{SV_{01}}$ y $T_{SV_{02}}$ son la duración de los tiempos de los vectores neutros $\overline{SV_0}$ y $\overline{SV_7}$ respectivamente. T_{SV_1} es el tiempo para el cual el vector estacionario $\overline{SV_1}$ es seleccionado y T_{SV_2} es para el cual el vector estacionario $\overline{SV_2}$ es seleccionado.

Tomando en cuenta que el vector arbitrario $\overline{V_o^*}$ depende de la amplitud y posición angular dentro del plano α - β , $\overline{V_o^*} = V_o \angle \theta$, la ecuación (2.3) se puede expresar en forma polar de la siguiente manera.

$$\frac{\Delta T}{2} V_o \angle \theta = T_{SV_1} \overline{SV_1} + T_{SV_2} \overline{SV_2} \quad (2.4)$$

donde, $\theta = [0, \pi/3]$ para el Sector 1.

Sustituyendo $\overline{SV_1} = V_M \angle 0$ y $\overline{SV_2} = V_M \angle \pi/3$ en su forma polar dentro de la ecuación (2.4), se obtiene la ecuación (2.5).

$$\frac{\Delta T}{2} V_o (\cos \theta + j \sin \theta) = T_{SV_1} \overline{SV_1} (\cos 0 + j \sin 0) + T_{SV_2} \overline{SV_2} \left(\cos \frac{\pi}{3} + j \sin \frac{\pi}{3} \right) \quad (2.5)$$

Agrupando las partes reales y las partes imaginarias de la ecuación 2.5 y utilizando las propiedades trigonométricas adecuadas, se obtienen los tiempos activos T_{SV_1} y T_{SV_2} de los vectores estacionarios $\overline{SV_1}$ y $\overline{SV_2}$.

$$T_{SV_1} = \Delta T \frac{V_o}{V_M} \cos(\theta_o - \frac{\pi}{6}) ; T_{SV_2} = \Delta T \frac{V_o}{V_M} \cos(\theta_o - \frac{\pi}{2}) \quad (2.6)$$

El mismo procedimiento se puede aplicar para obtener los tiempos activos $T_{SV_3} - T_{SV_6}$, usando las ecuaciones (2.2) y (2.5).

La Tabla 2.1 muestra los componentes de los vectores estacionarios más cercanos usados para calcular cualquier vector arbitrario de cualquier ángulo junto con la expresión apropiada que define el intervalo de tiempo activo para cada vector estacionario.

Tabla 2.1 Tabla de tiempos activos de un inversor trifásico.

Sector	$\omega\theta_0 = t$	Vectores Estacionarios	Tiempos activos de los vectores estacionarios
1	$0 \leq \theta_0 < \frac{\pi}{3}$	$\overrightarrow{SV_1}$ $\overrightarrow{SV_2}$	$T_{SV_1} = \Delta T \frac{V_o}{V_M} \cos(\theta_0 - \frac{\pi}{6})$ $T_{SV_2} = \Delta T \frac{V_o}{V_M} \cos(\theta_0 - \frac{\pi}{2})$
2	$\frac{\pi}{3} \leq \theta_0 < \frac{2\pi}{3}$	$\overrightarrow{SV_2}$ $\overrightarrow{SV_3}$	$T_{SV_2} = \Delta T \frac{V_o}{V_M} \cos(\theta_0 - \frac{\pi}{6})$ $T_{SV_3} = \Delta T \frac{V_o}{V_M} \cos(\theta_0 - \frac{5\pi}{6})$
3	$\frac{2\pi}{3} \leq \theta_0 < \pi$	$\overrightarrow{SV_3}$ $\overrightarrow{SV_4}$	$T_{SV_3} = \Delta T \frac{V_o}{V_M} \cos(\theta_0 - \frac{\pi}{2})$ $T_{SV_4} = \Delta T \frac{V_o}{V_M} \cos(\theta_0 - \frac{7\pi}{6})$
4	$\pi \leq \theta_0 < \frac{4\pi}{3}$	$\overrightarrow{SV_4}$ $\overrightarrow{SV_5}$	$T_{SV_4} = \Delta T \frac{V_o}{V_M} \cos(\theta_0 - \frac{5\pi}{6})$ $T_{SV_5} = \Delta T \frac{V_o}{V_M} \cos(\theta_0 - \frac{3\pi}{2})$
5	$\frac{4\pi}{3} \leq \theta_0 < \frac{5\pi}{3}$	$\overrightarrow{SV_5}$ $\overrightarrow{SV_6}$	$T_{SV_5} = \Delta T \frac{V_o}{V_M} \cos(\theta_0 - \frac{7\pi}{6})$ $T_{SV_6} = \Delta T \frac{V_o}{V_M} \cos(\theta_0 - \frac{11\pi}{6})$
6	$\frac{5\pi}{3} \leq \theta_0 < 2\pi$	$\overrightarrow{SV_6}$ $\overrightarrow{SV_1}$	$T_{SV_6} = \Delta T \frac{V_o}{V_M} \cos(\theta_0 - \frac{3\pi}{2})$ $T_{SV_1} = \Delta T \frac{V_o}{V_M} \cos(\theta_0 - \frac{\pi}{6})$

2.2.2. Método de Implementación.

Una técnica de implementación de la SVM es centrando el vector de espacio activo en la mitad del periodo de conmutación y divide el tiempo restante del espacio del vector neutro en partes iguales entre $\overrightarrow{SV_0}$ y $\overrightarrow{SV_7}$. Esto crea una secuencia de vectores de espacio para $0 \leq \theta_0 \leq \pi/3$ en el siguiente orden:

$$\overrightarrow{SV_0} \rightarrow \overrightarrow{SV_1} \rightarrow \overrightarrow{SV_2} \rightarrow \overrightarrow{SV_7} \rightarrow \overrightarrow{SV_7} \rightarrow \overrightarrow{SV_2} \rightarrow \overrightarrow{SV_1} \rightarrow \overrightarrow{SV_0}$$

$\xleftarrow{\Delta T/2} \qquad \qquad \qquad \xleftarrow{\Delta T/2}$

la cual es similar a la secuencia producida por una PWM regularmente muestreada a excepción del espacio correspondiente al vector neutro.

Una secuencia similar se establece para el resto de los sectores, usando los dos vectores de espacio más cercanos para cada sextante. La Figura 2.5 muestra el patrón de los pulsos en el primer sextante para una SVM convencional con los vectores neutros \vec{SV}_0 y \vec{SV}_7 espaciados igualmente.

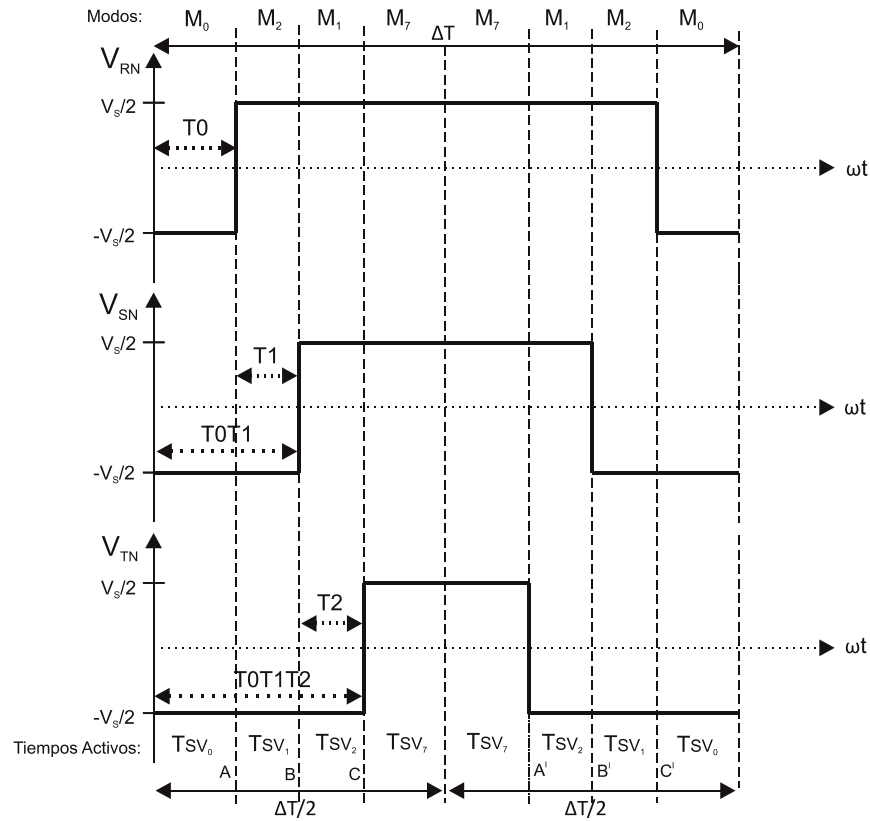


Figura 2.5 Formas de onda de voltaje a la salida de los puentes inversores para el sector 1.

Esta figura también presenta en la parte inferior los tiempos activos T_{SV_0} , T_{SV_2} , T_{SV_1} y T_{SV_7} , para la generación de los voltajes neutros, mientras que en la parte superior se observan los modos correspondientes para el primer sector.

2.2.3. Realización Práctica.

2.2.3.1. Generación de Voltajes en un ciclo de Conmutación.

Para generar las formas de onda de voltaje v_{RN} , v_{SN} y v_{TN} en un periodo ΔT , para el i -ésimo ciclo de conmutación centrado en un ángulo θ_i como el que se muestra en la Figura 2.5, es necesario tomar en cuenta los modos M_{n+1} y M_n del sector en el que se encuentra el vector arbitrario, ya que de esta manera se producirán los vectores de espacio SV_n y SV_{n+1} . Las formas de

onda de voltaje de la Figura 2.5 son generadas de acuerdo a los vectores de estado de los transistores. Al inicio del periodo se observa que está activado el modo M_0 , provocando que las salidas de voltaje del inversor sean de $-V_s/2$ durante el tiempo activo T_{SV_0} , al llegar a la transición de los modos M_0 y M_2 , los vectores de estado cambian de posición de acuerdo a M_2 provocando que el voltaje V_{RN} conmute de $-V_s/2$ a $V_s/2$, siendo T_{SV_2} la duración de M_2 . Cuando T_{SV_2} llega a su fin, se genera otro cambio de modo de M_2 a M_1 , provocando que el voltaje V_{SN} cambie de $-V_s/2$ a $V_s/2$, quedando activo M_1 durante el tiempo de T_{SV_1} . Al llegar M_1 a su fin, el modo M_7 es activado durante el tiempo T_{SV_7} creando una transición de voltaje en la salida V_{TN} de $-V_s/2$ a $V_s/2$. Al terminar el tiempo activo T_{SV_7} , también finaliza $\Delta T/2$ y por lo tanto el proceso de conmutación se repite pero en un orden inverso hasta completar ΔT . Siguiendo este mismo análisis, se generan las formas de onda de voltaje para cada uno de los 6 sectores del plano α - β .

2.2.3.2. Principio de Operación del Algoritmo de SVM para un Inversor Trifásico.

La Figura 2.6 muestra en un diagrama a bloques el proceso del algoritmo de control SVM para el Inversor Trifásico Tradicional. Este diagrama consta de 4 etapas: Generador de Portadora, Localizador de Sector, Generador de Tiempos Activos y Generador de interrupciones. A continuación se describe con detalle cada una de estas etapas.

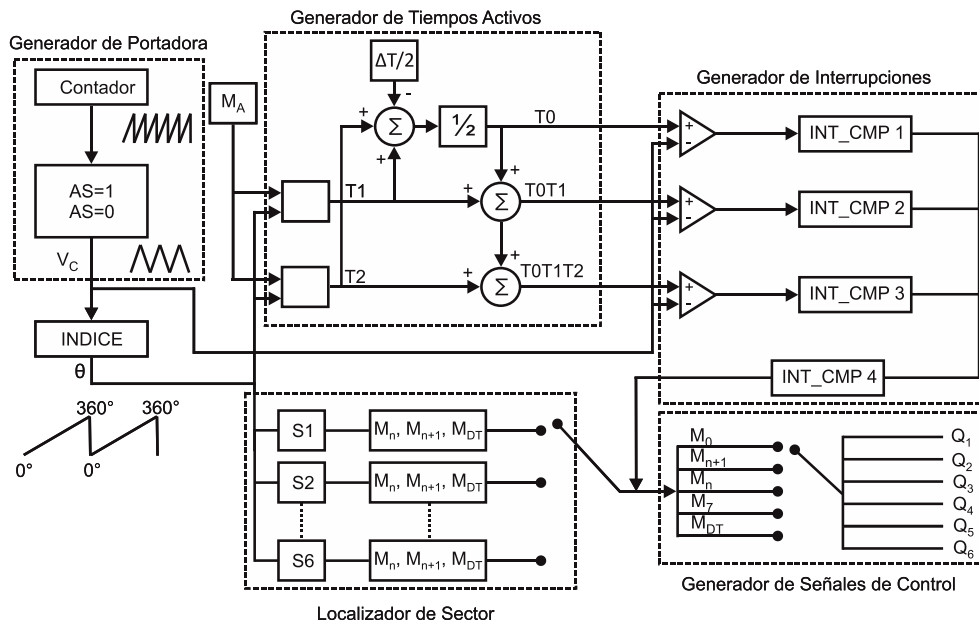


Figura 2.6 Diagrama a Bloques del proceso de control SVM para un Inversor Trifásico Tradicional.

2.2.3.3. Generador de Portadora.

El generador de portadora crea una señal que define el orden de los pulsos de conmutación para cada sector de la SVM. La forma simétrica de esta señal triangular es creada a partir de la adjunción de dos periodos de una señal tipo “diente de sierra”, la cual es generada con un contador, y definen la subida y bajada de una nueva señal triangular. Estas 2 señales son mostradas en la Figura 2.7. La frecuencia de la señal triangular está definida por el número total de pulsos por sector P_i y la frecuencia de salida $f_{sw} = 6 f_o P$.

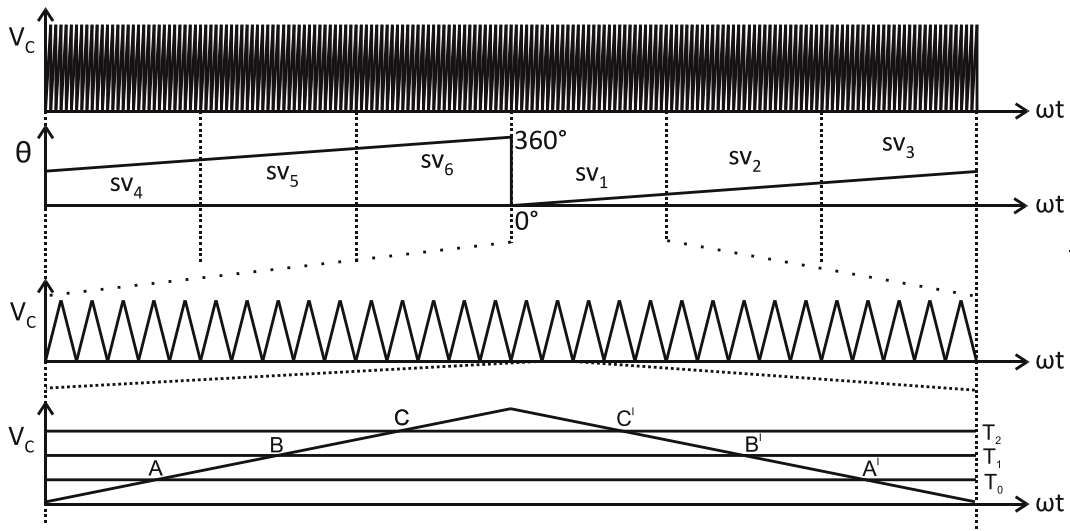


Figura 2.7 Esquema de señales de control y secuencia de interrupciones del Algoritmo para SVM.

2.2.3.4. Localizador de Sector.

Al cumplirse un periodo en la portadora se dispara un contador, el cual indica el sector de operación, y es representador por θ . Este contador es equivalente a una señal escalonada con un rango de generación de 0° a 360° , el cual está dividido en 6 sectores de 60° cada uno. Dependiendo de la posición angular del Vector arbitrario \vec{V}_o , se determina el sector de operación y los modos de estado de los transistores M_0, M_7, M_{n+1} y M_n .

2.2.3.5. Generador de Tiempos Activos.

El generador de tiempos Activos calcula los tiempos de duración T_0, T_1 y T_2 de los modos M_0, M_7, M_{n+1} y M_n , ocurridos en cada sector. Los valores de T_0, T_1 y T_2 cambian de acuerdo al Sector y al pulso p_i en el que se encuentre el vector arbitrario \vec{V}_o .

2.2.3.6. Generador de Interrupciones.

Este bloque programa la ocurrencia de los eventos A, B, C, A', B', C' en el inicio del ciclo de la portadora. Esta se logra utilizando módulos de comparación-interrupción como se muestra en la Figura 2.7. Cuando V_C es igual al ángulo de los elementos A, B, C se ejecuta el cambio de modo en los transistores tomando en cuenta el ciclo de subida o bajada de la señal triangular. La Figura 2.5 muestra el resultado de los comparadores, V_{MOD1} , V_{MOD2} y V_{MOD3} , que registran los eventos A, B, C y A', B' y C', como filos de subida y filos de bajada respectivamente. Los filos de estas señales indican el cambio de modo de los transistores.

2.3. Modulación de Espacio Vectorial de Ancho de pulso para otras topologías.

Con el propósito de mejorar el desempeño de los inversores se han mejorado las topologías de inversores existentes. Los inversores multiniveles por ejemplo, se han aplicado especialmente para el control de motores de CA, distribución de potencia en redes de CA y fuentes de energía ininterrumpible por mencionar algunas [2-3].

2.3.1. Inversores Multiniveles.

Los inversores multiniveles están básicamente compuestos de un arreglo de capacitores en cascada como fuentes de voltaje y semiconductores de potencia. El voltaje de salida en el inversor tiene una forma escalonada dependiente de la cantidad de interruptores, lo que determina el número de niveles en la forma de onda de salida.

2.3.2. Inversores Asimétricos.

El inversor asimétrico tiene como característica que los valores de sus voltajes de CD son distintos entre sí. En función de la razón de sus voltajes CD existen inversores asimétricos binarios, trinarios, etc. Estos últimos tienen la cualidad de proveer una forma de onda con 3^n niveles, donde n es el número de celdas conectadas en serie. Para el caso de $n=3$, se obtiene un voltaje de salida de 27 niveles, asegurando un voltaje con bajo contenido armónico [2-3].

2.3.3. Ejemplificación de un inversor multiniveles.

Un ejemplo de un inversor multiniveles modulado con Vector de Espacio es el inversor híbrido multiniveles mostrado en la Figura 2.8. La topología de este inversor está basada en un rectificador, el cuál fue desarrollado para aplicaciones aeroespaciales. Una descripción del

principio de operación de esta topología se encuentra en la Sección 2.2, Capítulo 2 de la referencia [2-2].

El inversor híbrido multiniveles está formado principalmente por un suministro de voltaje de corriente directa, un filtro de entrada, un arreglo de dos puentes inversores trifásicos conectados en serie, un transformador en conexión delta estrella con razón de transformación $\sqrt{3}: 1$ y un filtro LC para reducir el contenido armónico de las formas de onda de salida.

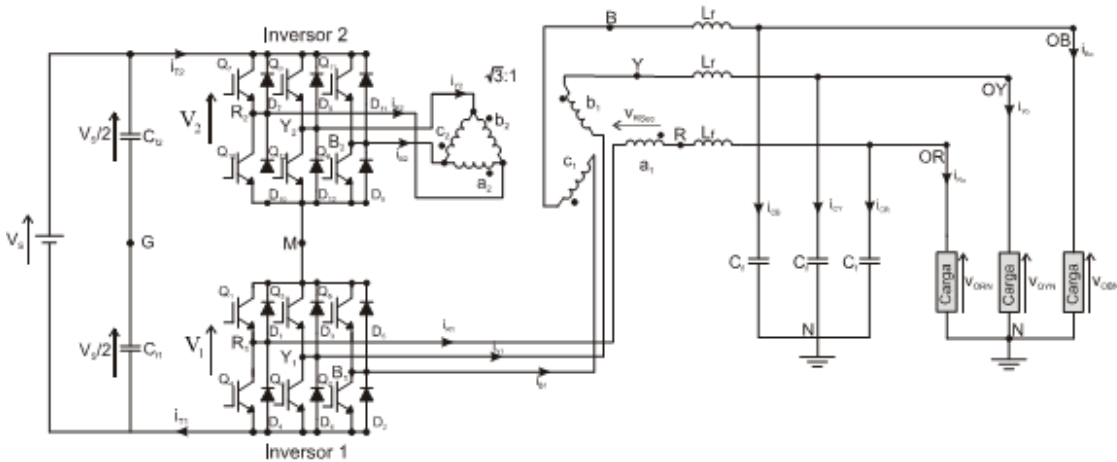


Figura 2.8 Inversor híbrido multiniveles.

La modulación de vector de espacio para esta topología se basa en el hecho que existen 12 configuraciones básicas del inversor y están definidas de acuerdo a los estados de los transistores de los inversores superior e inferior. La descripción de estos estados y su cálculo se describen en la referencia [2-4].

2.4. Resumen.

En un inversor tradicional controlado con SVM existen ocho posibles combinaciones de voltaje a la salida. Dos de estas combinaciones son voltajes nulos a la salida, conocidos como vectores neutros, y los otros seis son vectores estacionarios. Estos vectores pueden ser controlados de manera vectorial usando algoritmos PWM. Variando el tiempo de activación de cada uno de los vectores, se puede obtener un voltaje sinusoidal a la salida. La implementación de esta modulación se puede realizar en un microcontrolador o un DSP.

2.5.Referencias.

- [2-1] D. Grahame Holmes, Thomas A. Lipo, "Pulse width modulation for power converters Principles and Practice", IEEE series on Power Engineering, Mohamed E. El-Hawary, series editor.
- [2-2] Nicolas-Villalva Caren Ivét, "Modulación de Espacio Vectorial (SVM) como estrategia de control PWM para un Inversor Híbrido multiniveles", Sección de Estudios de Posgrado e Investigación, unidad Culhuacan.
- [2-3] Mondragon-Escamilla Nancy, "Análisis, diseño y construcción de un inversor multiniveles de 1kW para futuros medios de transporte", Sección de Estudios de Posgrado e Investigación, unidad Culhuacan.
- [2-4] Ramírez-Hernández Jazmín, "Modulación de espacio vectorial PWM para el control de un Inversor Híbrido multiniveles", Sección de Estudios de Posgrado e Investigación, unidad Culhuacan.
- [2-5] E. Hendawi, F. Khater, A. Shaltout, "Analysis, Simulation and Implementation of Space Vector Pulse Width Modulation Inverter", Electronics Research Institute, Cairo University.
- [2-6] K. Vinoth Kumar, Prawin Angel Michael, Joseph P. John, "Simulation ad comparison of SPWM and SVPWM control for three phase inverter". School of electrical Sciences, Karunya University, Tamilnadu, India.
- [2-7] Panda, Sunil. Mishra, Anupam. Srinivas, B. "Control of voltage inverters using PWM/SVPWM for adjustable speed drive applications". Department of Electrical Engineering National Institute of Rourkela. Rourkela.
- [2-8] Londoño N, C.Andrés. "Análisis y simulaciones de algunas estrategias de control SPWM y SVPWM en corriente para un Puente inversor" Universidad tecnológica de Pereira, Pereira. |

CAPÍTULO 3



3. Sincronizador de fase cerrada.

3.1.Introducción.

En este capítulo se describe el principio de operación del circuito Sincronizador Digital de Fase Cerrada o PLL Digital (DPLL, por sus siglas en inglés Digital Phase Locked Loop). Este circuito se basa en el principio de funcionamiento de un PLL tradicional, cuya tecnología es ampliamente utilizada en aplicaciones militares, aeroespaciales y algunos sistemas electrónicos que buscan sincronizar su funcionamiento con un evento externo. Los orígenes de esta tecnología se remontan a la revolución industrial siendo usada por los astrónomos para observar algunas características físicas de las estrellas. En convertidores de potencia son ampliamente utilizados para la sincronización con la red eléctrica [3-1].

3.2.Principio de Funcionamiento.

La estructura del PLL Digital está basada en el uso de un marco de referencia síncrono, un compensador y un VCO Digital como se observa en la Figura 3.1. Esta estructura está basada en la sincronización entre el vector rotante del marco de referencia y la señal de salida generada por un microcontrolador, por tanto, usando las transformaciones de Clarke y de Park, los tres valores instantáneos de voltaje son transformados a un marco de referencia rotante. Posteriormente, el compensador procesa la señal, filtrando el error y cualquier componente de alta frecuencia. Finalmente, el VCO Digital es modificado para cambiar la frecuencia fundamental de salida generada por el microcontrolador.

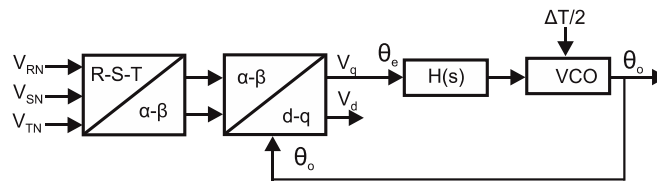


Figura 3.1 Diagrama a bloques del PLL Digital.

3.2.1. Marco de referencia rotante como Detector de Fase.

Empleado la fase de entrada de la señal generada por la transformación de Clarke, θ_i , y el ángulo de salida del DPLL, θ_o , y aplicando la transformación de Park se obtiene la diferencia de fase entre estos dos sistemas. Para poder entender este funcionamiento, se proponen tres señales que simulan el sistema Trifásico y dos señales, seno y coseno, que simulan el ángulo de una señal de referencia a 60 Hz, en este caso simulando el DCO, todas con una amplitud de 1 v y realizando la simulación en SIMULINK, se obtienen los resultados de las Figuras 3.2 y 3.3.

En las Figuras 3.2 y 3.3 se observa que la primer forma de onda corresponde a α (Clarke), la segunda corresponde a β (Clarke), la tercera a d(Park) y la última a q(Park). De la Figura 3.2 puede verse que si el sistema tiene frecuencias diferentes, entonces las señales d y q son oscilantes, mientras que en la Figura 3.3 cuyas frecuencias de entrada como de referencia son iguales, las señales d y q son constantes, d es uno y q es cero.

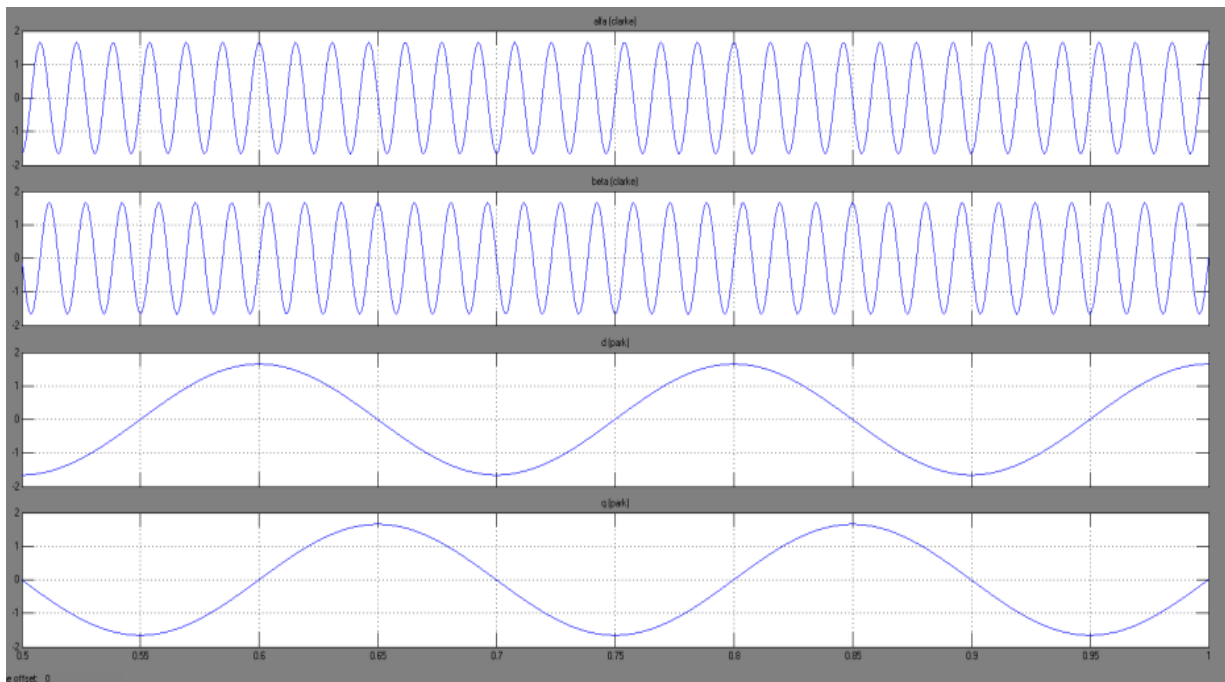


Figura 3.2 Transformaciones de Clark y Park con una Frecuencia de entrada de 65 Hz.

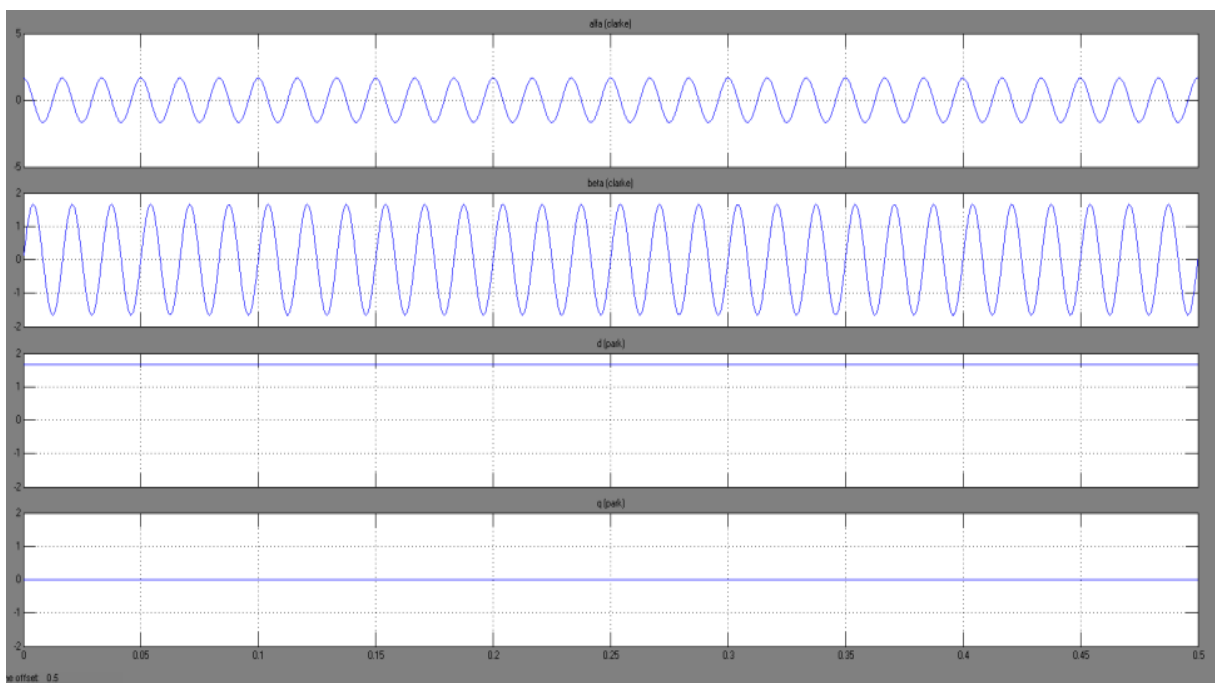


Figura 3.3 Transformaciones de Clark y Park con una Frecuencia de entrada de 60 Hz.

3.2.2. DCO Digital.

El DCO es un oscilador electrónico digital diseñado para trabajar con una frecuencia de oscilación en función de un temporizador, es decir, el periodo de la señal oscilante está dado por la duración del temporizador. Este tiempo está en función de la velocidad de respuesta del dispositivo electrónico utilizado ya que de esta depende el rango de frecuencias a las que el DCO es capaz de responder [3-2].

De la Figura 3.4 se observa la relación del temporizador y la frecuencia de salida del DCO.

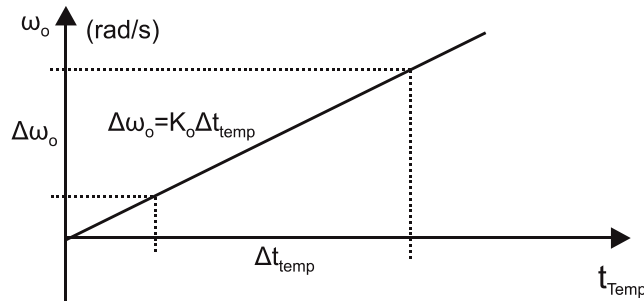


Figura 3.4 Ganancia del DCO de acuerdo a la relación del tiempo del Temporizador, y la frecuencia que genera a la salida.

3.2.3. Compensador.

De acuerdo a [3-3], la respuesta del compensador ideal tiene una ganancia K_h desde ω_z hasta ∞ . En la práctica, la respuesta del compensador decae hasta cierta frecuencia ω_p , como se muestra en la Figura 3.5. La expresión de la función de transferencia de este compensador es:

$$H(s) = K_h \frac{s + \omega_z}{s(\frac{s}{\omega_p} + 1)} \quad (3.1)$$

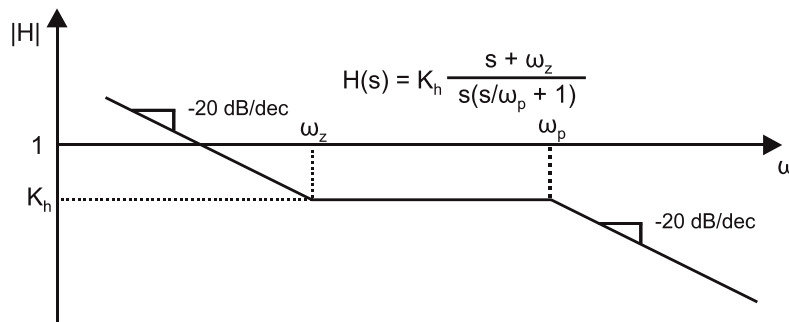


Figura 3.5 Respuesta en frecuencia del compensador de [3-1].

El modelo lineal del PLL se presenta en la Figura 3.6.

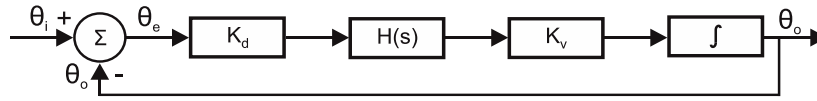


Figura 3.6 Modelo Lineal del PLL.

Aplicando la transformada de Laplace se obtiene el modelo de la Figura 3.7.

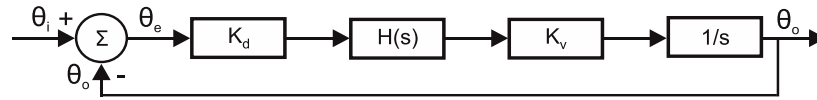


Figura 3.7 Modelo Lineal del PLL con Laplace.

Si se hace que:

$$\frac{K_d K_o H(s)}{s} = T(s) \quad (3.2)$$

Sabiendo que $\theta_e = \theta_i - \theta_o$, entonces:

$$\theta_o = T(s)\theta_e = T(s)(\theta_i - \theta_o) \quad (3.3a)$$

$$\frac{\theta_o}{\theta_i} = \frac{T(s)}{1 + T(s)} \quad (3.3b)$$

Sabiendo que $T(s) = \frac{K_d K_o H(s)}{s}$ podemos decir que:

$$T(s) = G(s)H(s) \quad (3.4)$$

Así, se obtiene que:

$$G(s) = \frac{K_d K_o}{s} \quad (3.5)$$

Y cuya respuesta en frecuencia puede verse en la Figura 3.8.

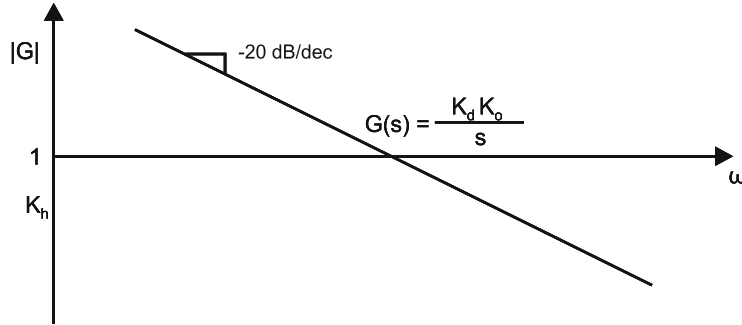


Figura 3.8 Respuesta en Frecuencia de $G(s)$.

Tomando como referencia [3-3], $H(s)$ se observa en la ecuación 3.4 y su respuesta en frecuencia se muestra en la Figura 3.9.

$$H(s) = K_h \frac{s + \omega_z}{s(\frac{s}{\omega_p} + 1)} \quad (3.6)$$

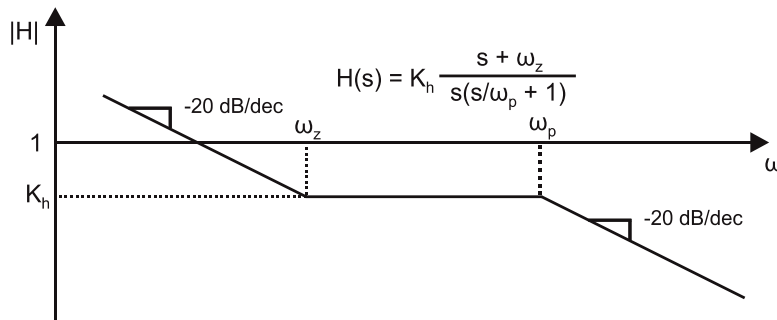


Figura 3.9 Respuesta en frecuencia de $H(s)$.

De las ecuaciones 3.4 y 3.6 se obtiene:

$$T(s) = K \frac{s + \omega_z}{s^2(\frac{s}{\omega_p} + 1)} \quad (3.7)$$

Y así, la función de transferencia del PLL viene descrita por $F(s)$ y su respuesta en frecuencia puede verse en la Figura 3.10.

$$F(s) = \frac{\theta_o}{\theta_i} = \frac{T(s)}{1 + T(s)} \quad (3.8)$$

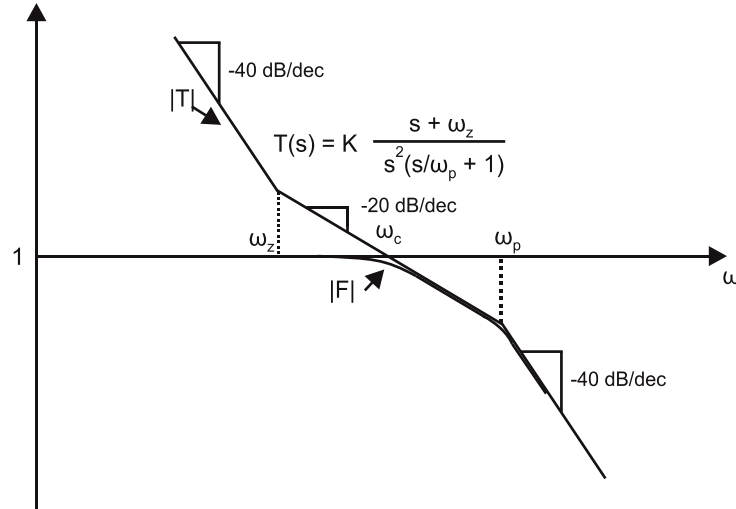


Figura 3.10 Respuesta en frecuencia de lazo cerrado $F(s)$

3.3. Diseño del PLL.

Proponiendo un ancho de banda de 10 Hz, la frecuencia de cruce ω_c , de acuerdo a [3-3], está en $B\omega = \omega_c = 2\pi(10\text{Hz}) = 62.83 \text{ rad/s}$.

Proponiendo $\omega_z = 10 \text{ rad/s}$, y de acuerdo a la media geométrica,

$$\omega_p = \frac{\omega_c^2}{\omega_z} = 394.78 \text{ rad/s} \quad (3.9)$$

Sustituyendo estos valores en la ecuación 1.10 se obtiene la expresión 3.10.

$$T(s) = K \frac{s + \omega_z}{s^2 \left(\frac{s}{\omega_p} + 1 \right)} = \frac{0.1s + 1}{1.28 \times 10^{-6}s^2 + 50.51 \times 10^{-3}s} \quad (3.10)$$

De esta manera, $K_h = 1979.81$, y calculando la respuesta en frecuencia de $T(s)$ en MATLAB se obtiene la gráfica mostrada en la Figura 3.11. Esta respuesta en frecuencia considera un margen de fase de 10 Hz lo cual se consideró propio para el presente trabajo.

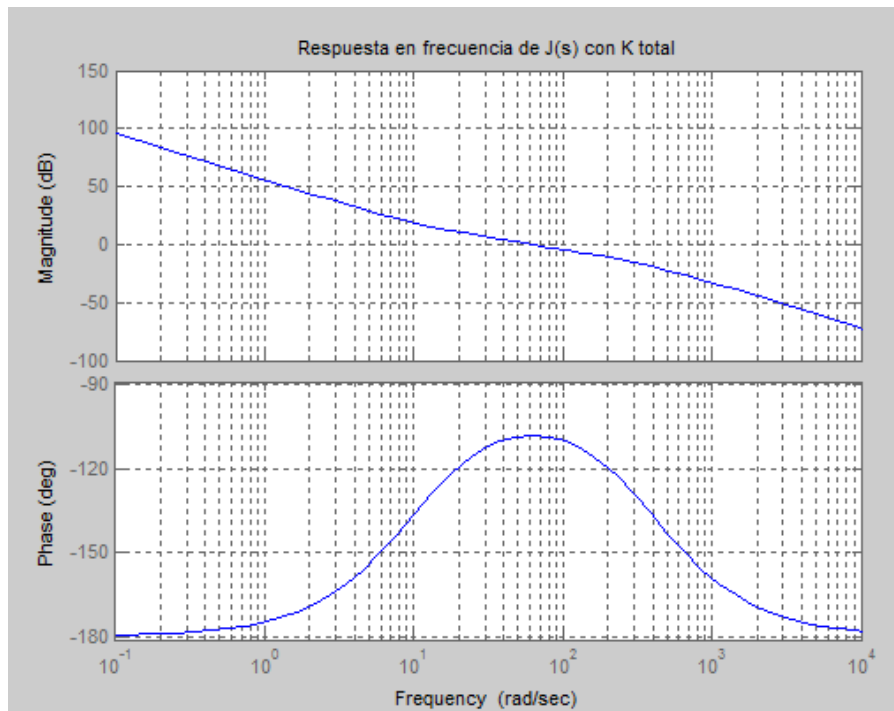


Figura 3.11 Respuesta en frecuencia de T(s).

La respuesta en frecuencia de lazo cerrado $F(s)$ simulado en MATLAB se muestra en la Figura 3.12.

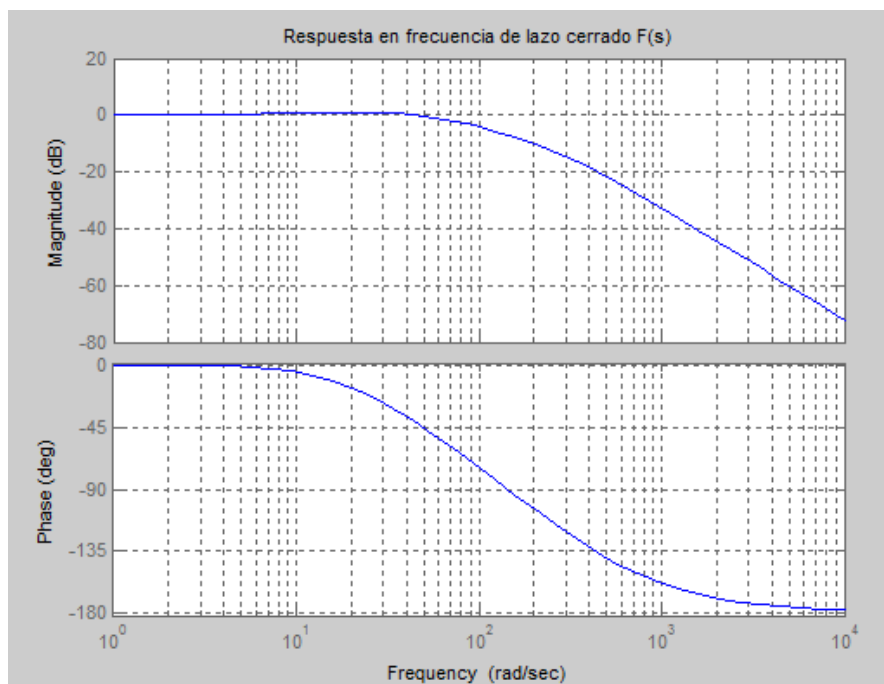


Figura 3.12 Respuesta en frecuencia de lazo cerrado $F(s)$.

Para implementar el compensador de manera digital, de acuerdo a la ecuación 3.11, es necesario obtener una versión discreta de $u(s)$, llamada $u(z)$, en función de la ecuación 3.10 y el error $e(s)$. Esta expresión se determina usando la transformación bilineal a una frecuencia de muestreo de 21.6 kHz y posteriormente se procede a obtener su ecuación diferencial.

$$H(s) = \frac{u(s)}{e(s)} \quad (3.11)$$

De esta manera, utilizando la transformada Bilineal con una frecuencia de muestreo de 21.6 kHz se procedió a calcular la ecuación diferencial que representa nuestra versión discreta de la ecuación 3.11 perteneciente al compensador, es la mostrada en la ecuación 3.12.

$$u(k) = 1.981896u[k - 1] - 0.981896u[k - 2] + 1.792494e[k] + 829.666 \times 10^{-6}e[k - 1] - 1.791664e[k - 2] \quad (3.12)$$

3.4. Metodología de Implementación.

En la Figura 3.13 se muestra el diagrama a bloques del PLL el cual consta principalmente de 4 etapas: Generación de Tablas, Lectura de Voltajes, Transformaciones, Cálculo de Compensador y VCO. La descripción detallada se describe en las siguientes secciones.

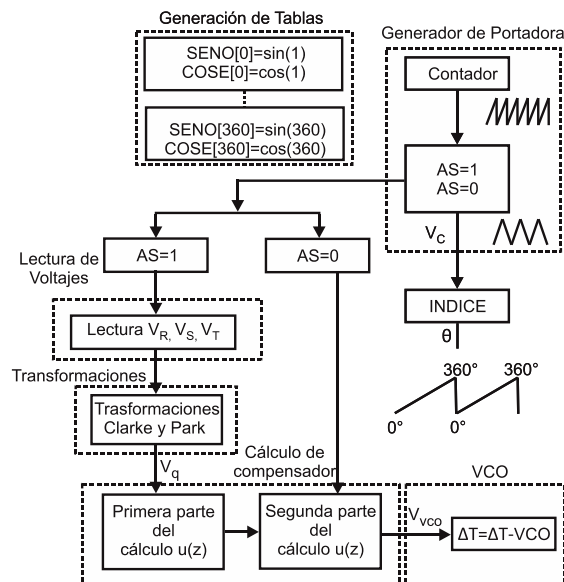


Figura 3.13 Diagrama a Bloques del PLL.

3.4.1. Descripción del PLL.

3.4.1.1. Generación de tablas.

De acuerdo a la teoría descrita anteriormente, es necesario generar señales de referencia que actuará como v_o , con esto, cada ΔT de la SVM será equivalente a un grado en un periodo de 360° de una señal seno y otra coseno, que sirven para poder realizar la transformación de Park necesaria para ser utilizada como un Detector de Fase. Para obtener estas señales se generan un par de tablas en donde el argumento de las funciones seno y coseno es θ , es decir, las tablas van desde $\text{sen}(1)$ y $\text{cos}(1)$ hasta $\text{sen}(360)$ y $\text{cos}(360)$, con esto se tienen las señales de referencia descritas anteriormente.

3.4.1.2. Lectura de Voltajes.

Para poder realizar las transformaciones de Clarke y Park, se realiza una lectura de las tres señales de voltaje, v_R, v_S, v_T .

3.4.1.3. Transformaciones.

La transformación de Clarke genera una señal $V_i = [V_\alpha, V_\beta]$ la cual es la referencia de la señal a la cual se necesita sincronizar. Posteriormente se toma de las tablas generadas la señal $V_o = [\text{SENO}(\theta), \text{COSE}(\theta)]$ y se realiza la transformación de Park que genera el la señal v_q que es procesada por el compensador.

3.4.1.4. Cálculo de compensador.

Este bloque calcula el valor del compensador, V_{vco} luego de ser tratada la señal v_q . De acuerdo a la Figura 2.5 se puede realizar el cálculo de las transformaciones y la Ecuación Diferencia para el filtro en los espacios libres de los Tiempos Activos, esto puede ser al principio de ΔT y en $\frac{\Delta T}{2}$, que es donde AS determina la parte descendente para así evitar el traslape de operaciones de la SVM, por ello el cálculo se divide en dos partes obteniéndose la señal V_{vco} que modifica el VCO. La señal V_{vco} modifica el valor de ΔT , con lo cual la frecuencia fundamental de la SVM cambia y de esta manera se sincroniza con la frecuencia de la señal trifásica de referencia.

3.5. Resumen.

En este capítulo se describió el principio de operación de un PLL y se puntualizó su principio de operación. A su vez, se expuso la metodología necesaria para hacer uso de las

transformaciones de Clarke y de Park empleadas dentro de un PLL digital. De esta forma se describió el principio del PLL desarrollado y se explicó cómo se implementa en un algoritmo de control para un microcontrolador.

3.6.Referencias.

- [3-1] Pindado R., "Phase Locked-Loop (PLL): Fundamento y aplicaciones", Universidad Politécnica de Cataluña.
- [3-2] Kroupa, Venceslav F., "Principles of phase locked loops" Institute of Radio Engineering and Electronics, Academy of Sciences of the Czech Republic.
- [3-3] Wolaver, Dan H. "Phase-Locked Loop Circuit design". Prentice Hall.
- [3-4] Data Sheet dsPIC33FJXXMCX10A
- [3-5] Application note, Freescale Semiconductor Document number AN535
- [3-6] Perrot, H. Michael. "Tutorial on Digital Phase-Locked Loops" CICC 2009.
- [3-7] Fischette, Denis. "Practical tips for Phase Locked Loop Design", First time, Every time.

CAPÍTULO 4

The logo for ESIME (Escuela Superior de Ingeniería y Arquitectura) is a large, light green watermark in the background. It features the word "ESIME" in a bold, blocky font, with a gear-like border around the text. In the center of the gear, there is a stylized green graphic consisting of overlapping loops and a lightning bolt shape.

4. Implementación digital.

4.1.Introducción

En este capítulo se hace una descripción general de la implementación de los algoritmos de la modulación de vector de espacio y el PLL digital implementados en un microcontrolador del 16 bits dsPIC16FJ256MC710A

4.2. Breve descripción del microcontrolador.

La familia de dispositivos dsPIC33F emplea una arquitectura de 16 bits con algunas características de un DSP. El resultado es un dispositivo capaz de trabajar a altas velocidades, hacer cálculos repetitivamente y control.

Algunos de los módulos con los que cuentan los microcontroladores de la familia dsPIC33F son los siguientes

- **ADC.-** Es un módulo que provee 32 entradas analógicas en dos diferentes resoluciones, 10 y 12 bits. Además, contiene voltajes de referencia seleccionables.
- **Timer de Propósito General.-** Contiene los elementos base para ser usados con el módulo PWM. Contiene 9 timers de 16 bits (Timer1 hasta Timer9). 8 de los timers pueden ser configurados para generar timers de 32 bits. Tienen diferentes formatos de operación.
- **Output Compare PWM.-** Es un módulo que compara la base de tiempo con uno o dos registros comparadores para generar pulsos o pulsos PWM.
- **SPI™ Module.-** Es un módulo interfaz útil para comunicar dispositivos codificadores/decodificadores de audio, Convertidores AD y DA, etc.
- **DMA.-** Memoria de acceso directo. Permite copiar de manera eficiente datos leídos desde los periféricos, buffers o variables de la RAM con un mínimo uso del CPU.
- **DSP Engine.-** Consiste en dos multiplicadores de 17X17 bits, un rotador y un restador/sumados de 40 bits para dos acumuladores del mismo tamaño.
- **I²C™ Module.-** Es un módulo interfaz útil para comunicarse con otros periféricos o microcontroladores.
- **SPI™ Module.-** Es un módulo interfaz útil para comunicarse con otros periféricos o microcontroladores. Asimismo es útil para comunicación con EEPROMs, controladores de displays, Convertidores ADC, etc.
- La tarjeta de desarrollo *Explorer 16a* ofrece una manera de comprobar el dsPIC33F y la familia de controladores de motores. Esta tarjeta incluye:
 - *100 pines dsPIC33F.*

- Comunicación USB.
- Comunicación RS-232
- Tabla de LEDs para indicaciones generales.
- LCD alfanumérico de 16X2
- Sensor de temperatura
- Tarjeta de expansión.

4.3. Implementación de algoritmo de SVPWM.

El algoritmo descrito en el Capítulo 2 fue implementado en un DSC de 16 bits, modelo dsPIC33FJ256MC710A de Microchip y el código del programa puede ser consultado en el Apéndice A. El desarrollo del algoritmo de control está basado en el Diagrama de Flujo de la Figura 4.1 y su operación puede ser comprendida apoyándose en la Figura 2.5.

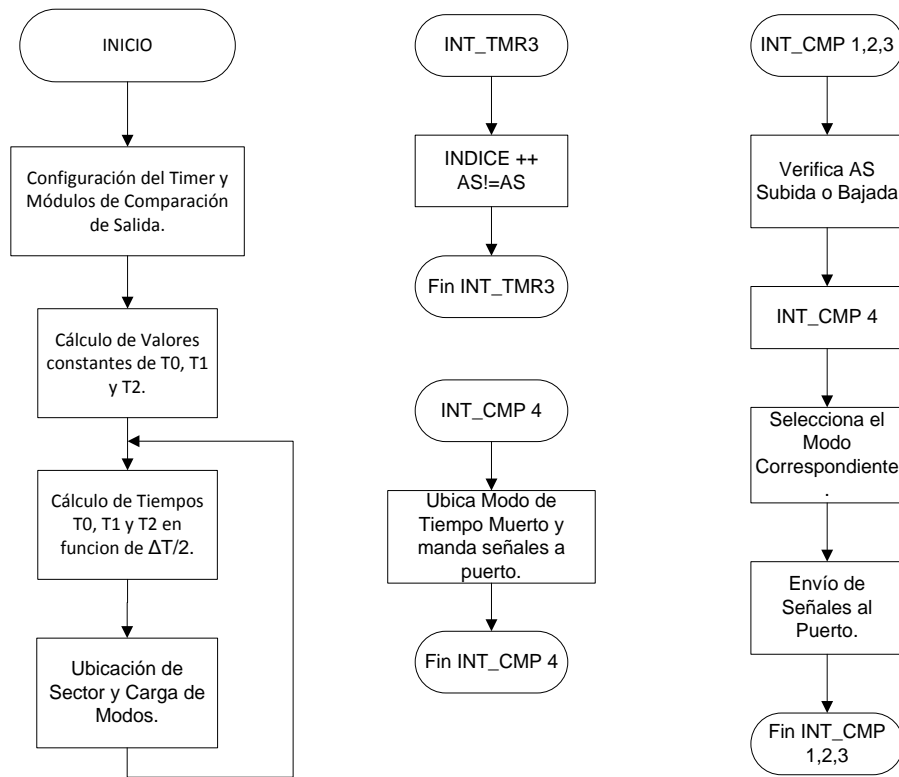


Figura 4.1 Diagrama de Flujo que describe la forma de programación del algoritmo para un Inversor Trifásico Tradicional.

El primer paso del programa es la configuración del *Timer* y los Módulos de Comparación de Salida que son utilizados a lo largo del programa. El siguiente paso es calcular el valor de la parte constante de T0, T1 y T2, ya que como se observa en la Tabla 2.1 , estos valores son constantes para cada sector hasta antes de ser multiplicados por $\frac{\Delta T}{2}$ que es un valor que estará en constante cambio. A continuación se multiplica la parte constante por $\frac{\Delta T}{2}$ y se calculan los valores T0, T1 y T2, para el pulso actual. Posteriormente se cargan los modos M_{n+1} , M_n y M_{DT} que son utilizados por los Módulos de Comparación para ser enviados a los puertos de salida.

La subrutina de Interrupción INT_ TMR3 se genera cada vez que el contador del *Timer* llega hasta su valor $\frac{\Delta T}{2}$. Durante esta interrupción la bandera AS indica si el contador del Timer se ubica en la parte ascendente o descendente de la señal portadora Triangular. Si AS indica ascendencia, el contador INDICE aumenta y los valores T0, T1 y T2 son asignados a los Registros de Comparación. Por lo contrario, si AS determina la parte descendente, los valores T0, T1 y T2 son calculados para ser reasignados a los Registros de los Módulos de Comparación.

Las interrupciones de los Módulos de Comparación de Salida INT_CMP 1,2,3 son generadas cada vez que el contador del Timer alcanza los valores de los Registros de estos módulos. Estas interrupciones, a su vez, generan la interrupción INT_CMP 4 la cual genera un tiempo muerto. Cuando la INT_CMP 4 llega a su fin, se cargan los modos correspondientes y se envían al puerto de salida.

La interrupción del Módulo de Comparación de Salida INT_CMP 4 es generada cuando en INT_CMP 1,2,3 hay un cambio de modo en los transistores. La función principal de esta interrupción es generar un tiempo muerto y una combinación propia del cambio de modo M_{DT} .

4.3.1. Configuración del Timer.

El DSC dsPIC33FJ256MC710A tiene nueve Timers de 16 bits los cuales están divididos en 3 tipos, tipo A, B y C. Para la implementación del algoritmo de control se utilizó el Timer 3 del Tipo C, ya que este Timer puede sincronizar su contador con los Módulos de Comparación de Salida. La Tabla 4.1 muestra la configuración del Timer 3 para este algoritmo.

Tabla 4.1 Configuración del Timer 3 para el algoritmo de SVM para el inversor Trifásico.

Nombre del Registro	Descripción	Configuración
TCS	Selecciona el Reloj interno del DCS.	0
TGATE	Modo de Disparo Deshabilitado.	0
TCKPS	Preescala 1:1	0b00
TMR3	Limpia el Registro del contador del Timer 3.	0x0000
PR3	Carga el valor del periodo.	1194
T1IP	Indica el Nivel de Prioridad del Timer 3.	0x01

4.3.2. Configuración de los Módulos de Comparación de Salida.

Este Módulo compara el valor del Timer con el valor de uno o dos de los registros de comparación, dependiendo del modo de operación seleccionado. El estado de la terminal de salida cambia cuando el valor del registro del Timer cruza con el valor de registro de comparación. El Módulo de Comparación genera un pulso o una secuencia de pulsos cambiando el estado del pin de salida en cada cruce con el registro del Timer. Cuando existe un cambio en el pin de salida se genera una interrupción en la cual se pueden manipular el estado de las tareas actuales. La Tabla 4.2 muestra la configuración de los Módulos de Comparación.

Tabla 4.2 Configuración del Modulo de comparación

Nombre del Registro	Descripción	Configuración
OCTSEL	Selecciona el Timer 3 como base	1
OCM	Modo de operación "Toggle".	0b011
OCnIE*	Habilita Interrupción en el módulo.	1

4.4.Verificación numérica del PLL.

Para comprobar el funcionamiento del PLL se realizó una simulación en SIMULINK. De acuerdo a la Figura 4.2 se definen tres señales que actúan como el voltaje de suministro, se realizan las transformaciones del Clarke y Park usando las señales de referencia seno y coseno que son generadas por el VCO Digital. Al obtenerse la señal V_d , se procesa por la función de transferencia de la ecuación 3.10 para generar $V_{control}$, la cual se resta de un valor constante de

2388 que es el registro del *Timer* que genera la frecuencia central del PLL de 60 Hz, y de esta manera se modifica la frecuencia fundamental del PLL.

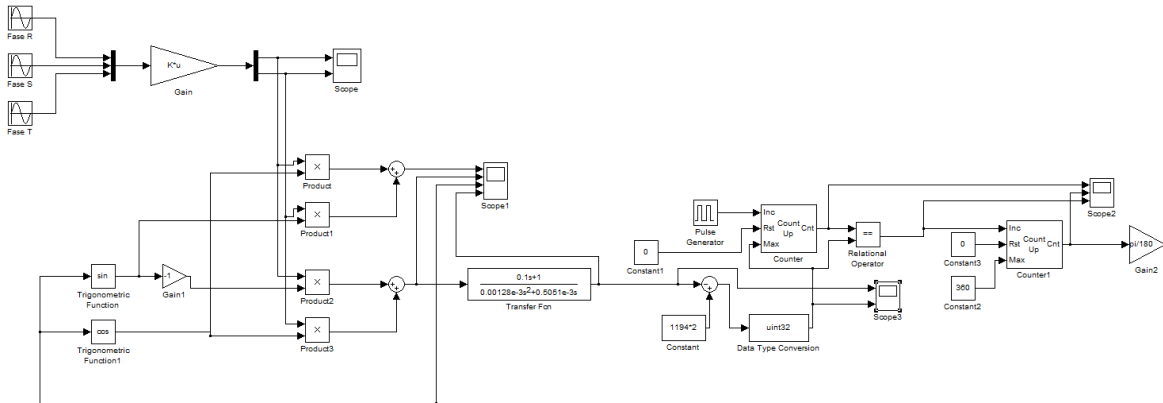


Figura 4.2 Diagrama a bloques en SIMULINK del PLL.

La relación para obtener el valor del registro al que debe estar operando el “*Timer*” para poder modificar la frecuencia del PLL es la siguiente.

TMR3= frecuencia del microcontrolador es de 51.59MHz

$$TMR3 = \frac{\text{Frecuencia del dsPIC}}{\text{Frecuencia de conmutación del PLL}}$$

Donde: Frecuencia del dsPIC = 51.59 MHz

Frecuencia de conmutación del PLL=12*30*Frecuencia fundamental del PLL

La Tabla 4.3 muestra la relación de la Frecuencia fundamental del PLL y el valor *Timer* para generar está frecuencia en el dsPIC.

Tabla 4.3 Frecuencia fundamental del PLL en relación al “*Timer*” del dsPIC.

Frecuencia fundamental del PLL	Valor del Registro del “ <i>Timer</i> ”
56 Hz	2559
58 Hz	2470
60 Hz	2388
62 Hz	2311
64 Hz	2239

La verificación en SIMULINK arroja los resultados mostrados en las siguientes gráficas. Por ejemplo, la Figura 4.3 muestra el valor del Registro del *Timer* para una frecuencia fundamental del PLL de 62 Hz.

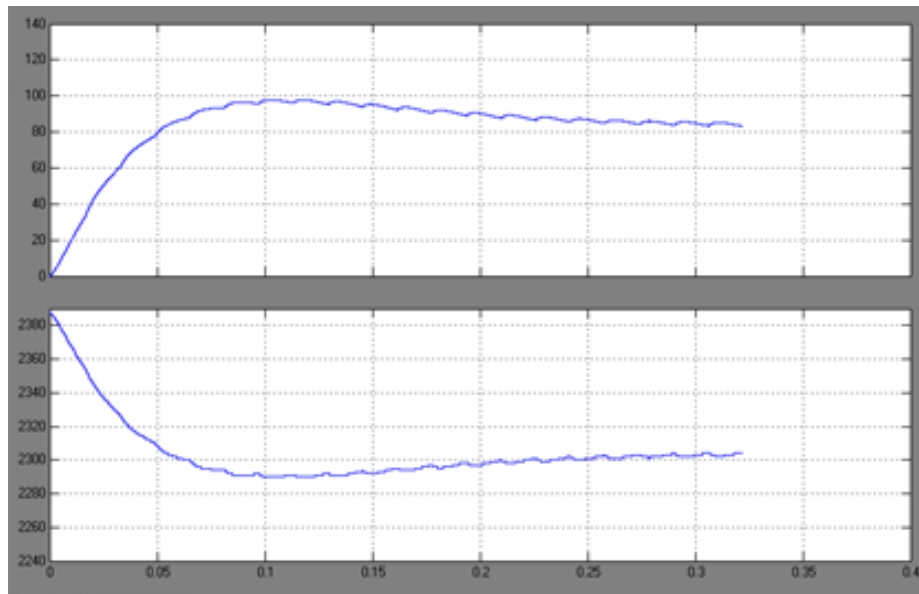


Figura 4.3 Valor del Registro del *Timer* para una frecuencia fundamental del PLL de 62 Hz.

La Figura 4.4 muestra el valor del Registro del *Timer* para una frecuencia fundamental del PLL de 58 Hz.

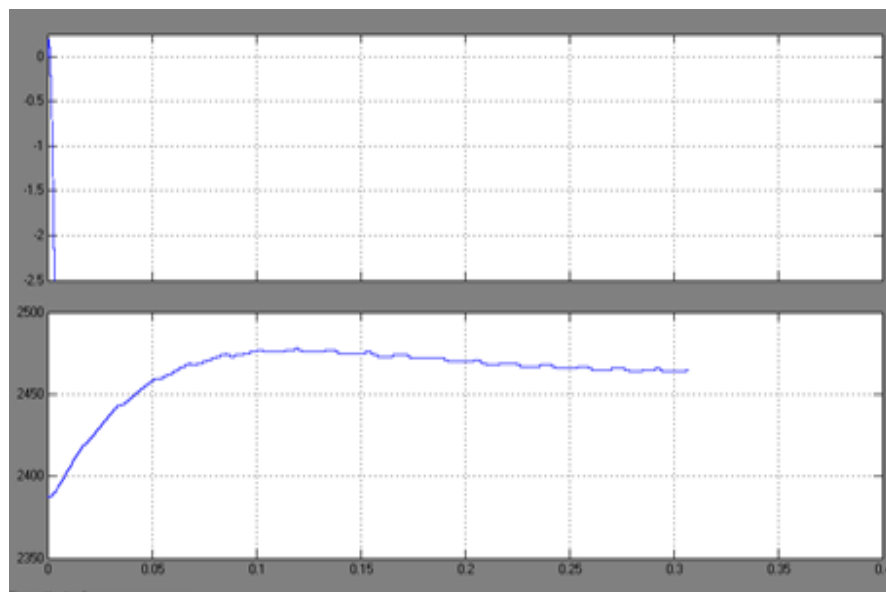


Figura 4.4 Valor del Registro del *Timer* para una frecuencia fundamental del PLL de 58 Hz.

La Figura 4.5 muestra el valor del Registro del *Timer* para una frecuencia fundamental del PLL de 60 Hz.

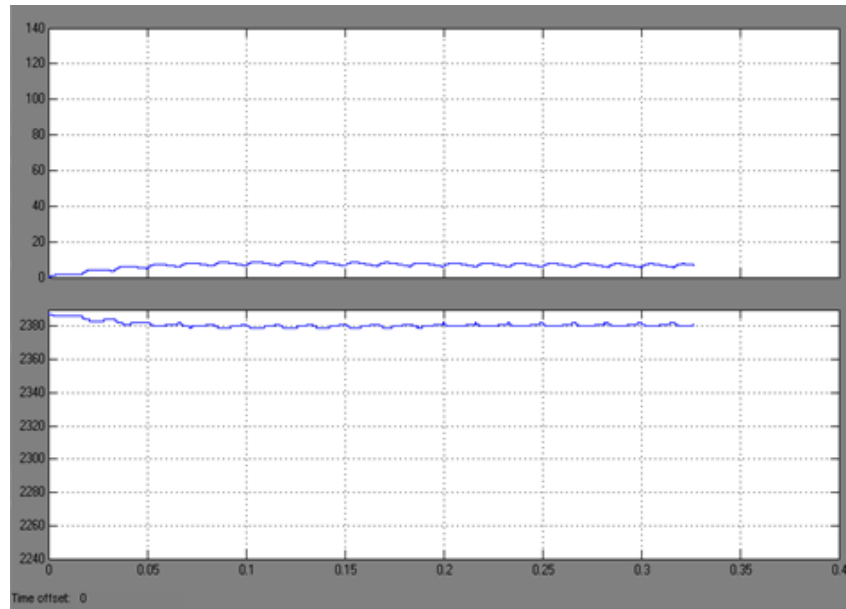


Figura 4.5. Valor del Registro del *Timer* para una frecuencia fundamental del PLL de 60 Hz.

En las figuras anteriores se pudo observar que el valor del registro del *Timer* es muy cercano al estimado en los cálculos.

4.5. Implementación Digital del PLL.

El algoritmo descrito anteriormente fue implementado en un DSC de 16 bits, modelo dsPIC33FJ256MC710A de Microchip y el código del programa puede ser consultado en el Apéndice A. El desarrollo del algoritmo de control está basado en el Diagrama de Flujo de la Figura 4.6 y su operación puede ser comprendida apoyándose en la Figura 3.13.

La lógica del programa inicia con la configuración del *Timer*, ADC y DMA que son utilizados a lo largo del programa. El siguiente paso es calcular los valores constantes de T0, T1 y T2, pues como se observa en la Tabla 2.1, estos valores son fijos para cada sector hasta antes de ser multiplicados por $\Delta T/2$ que es el valor que se modifica a partir de la determinación del valor de la señal V_{vco} . A continuación se toma en cuenta $\Delta T/2$ y se calculan los valores T0, T1 y T2, para el pulso actual.

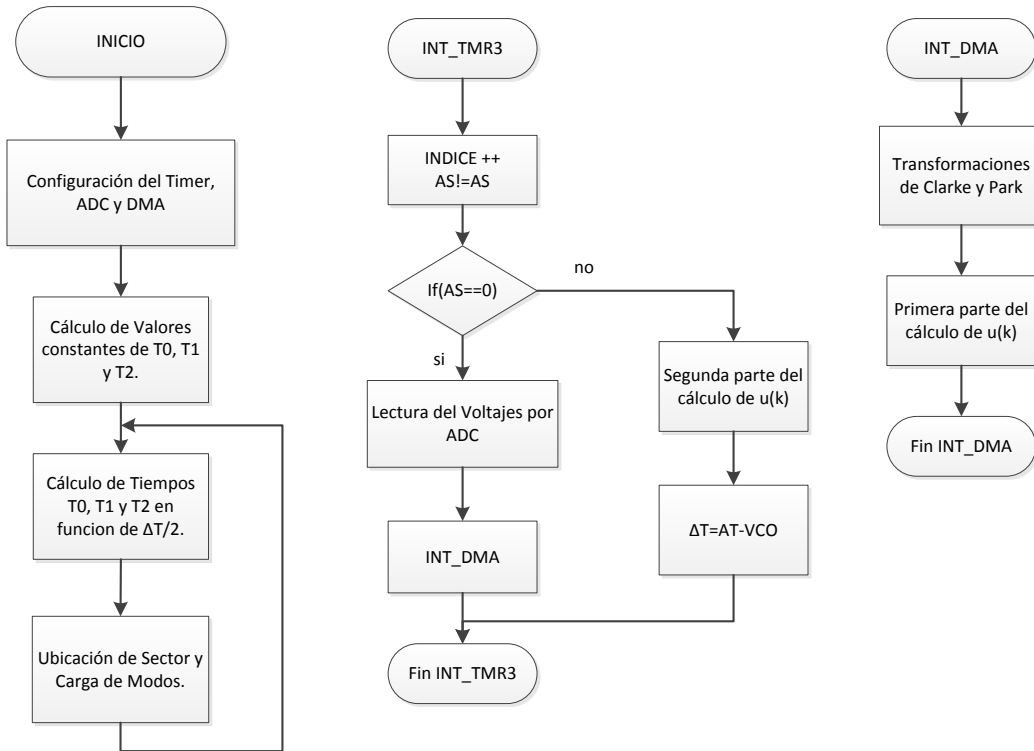


Figura 4.6 Diagrama a Bloques del programa.

La subrutina de Interrupción INT_ TMR3 se genera cada vez que el contador del Timer llega hasta su valor $\frac{\Delta T}{2}$. Durante esta interrupción la bandera AS indica si el contador del Timer se ubica en la parte ascendente o descendente de la señal portadora Triangular. Si AS indica ascendencia, el contador INDICE aumenta y se realiza la primera lectura de 3 canales simultáneos por el ADC correspondientes a v_R , v_S y v_T y se genera la subrutina de interrupción INT_DMA. Por lo contrario, si AS determina la parte descendente, se realiza la segunda parte del cálculo de $u(k)$ que determina el valor de la señal V_{vco} .

La subrutina de Interrupción INT_ DMA registra las lecturas de los voltajes y realiza las operaciones de las transformaciones de Clarke y Park. Posteriormente se realiza el cálculo de la primera parte de $u(z)$ que continua cuando la bandera AS indica descendencia.

4.5.1. Configuración del ADC.

Se configura el ADC para realizar una lectura de 4 canales simultáneos aunque solo se utilizan 3 y sus correspondientes lecturas. El ADC es activado cada vez que el bit SAMP es encendido, esto sucede cuando la bandera AS es igual a 1. Las lecturas son transferidas al CPU a través del DMA.

Tabla 4.4 Configuración del ADC para el algoritmo del PLL.

Nombre del Registro	Descripción	Configuración
FORM	Formato de Lectura de datos Q15.	3
ASAM	Inicia el muestreo cuando el bit SAMP es puesto a 1.	0
AD12B	Lectura de ADC de 10 bits.	0
SIMSAM	Muestreo simultaneo.	1
VCFG	Voltajes de referencia AVdd y AVss.	4
AD1CSSL	Configura a AIN0, AIN1, AIN2, AIN3 como entradas.	0x000F

4.5.2. Configuración del DMA.

EL DMA (por sus siglas en inglés, Direct Memory Access), realiza una transferencia de datos entre los periféricos y el CPU. Este proceso no afecta el desempeño del DSC y por tanto es de gran utilidad al realizar las lecturas simultáneas de los canales del ADC.

Tabla 4.5 Configuración del DMA para el algoritmo del PLL.

Nombre del Registro	Descripción	Configuración
AMODE	Configura el DMA para Modo Indirecto	3
MODE	Configura el DMA en Modo Continuo	0
DMAOCNT	Lectura de 3 datos	4
DMAOREQ	Selecciona el ADC como fuente de datos.	13

4.6. Resumen.

En este capítulo se describió la implementación digital de los algoritmos para SVPWM y para el PLL Digital desarrollados en los Capítulos 2 y 3 respectivamente. Además se hace una verificación numérica del PLL usando un software de simulación. Se desarrollaron tablas donde se muestra como quedan configurados los principales registros del microcontrolador usado.

4.7. Referencias.

- [4-1] dsPIC30F Family Reference Manual - Microchip
- [4-2] MATLAB Documentation - MathWorks
- [4-3] Simulink Documentation - MathWorks

CAPÍTULO 5

ESIME

5. Pruebas y resultados.

5.1.Introducción.

En este capítulo se describen los resultados obtenidos al hacer práctica la implementación del PLL de manera digital. Se muestran las señales de control implementadas en el microcontrolador mostrando las combinaciones para un sector y los tiempos muertos para evitar cortos circuitos en el inversor. También se hace hincapié en el desarrollo de la tarjeta de conexiones del dsPIC y la tarjeta acondicionadora de señales para el ADC. Se describen las posibles formas de verificación de ciertas operaciones aritméticas que se realizan dentro del

microcontrolador. Por última instancia, se muestra la sincronización de las señales v_R y v_{RN} del convertidor.

5.2.Verificación de las señales de control para el inversor tradicional.

Para verificar las señales de control que se obtuvieron del microcontrolador se hizo uso de un osciloscopio digital Agilent Technologies modelo MSO6012A con un ancho de banda de 100MHz y 16 canales digitales. Se programó el Puerto G del dsPIC para controlar los transistores del inversor.

Las señales de control se obtuvieron del algoritmo mostrado en el Apéndice A tienen una frecuencia fundamental de 60 Hz, como se muestra en la Figura 5.1, dando como resultado una frecuencia de 21.6 kHz por cada pulso para la SVPWM como se observa en la Figura 5.2. Esta misma figura muestra estas señales de control únicamente para el primer sextante de la modulación.



Figura 5.5.1 a) Señales de control mostrando su frecuencia fundamental de 60 Hz b) Ampliación en tiempo de las señales de control.

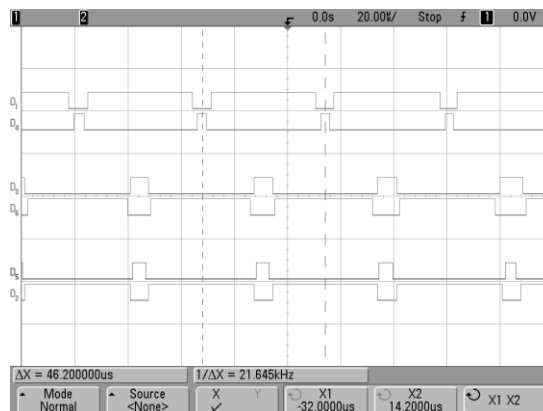


Figura 5.2 Señales de control para el primer sextante de la SVPWM.

Para garantizar que no existan cortos circuitos en el inversor es necesario introducir tiempos muertos. La Figura 5.3 muestra la introducción de tiempos muertos para cada una de las conmutaciones en las señales de control. Este tiempo muerto apaga primeramente el transistor que está encendido, deja pasar un tiempo inactivo de $2 \mu\text{s}$ y enciende el otro transistor.

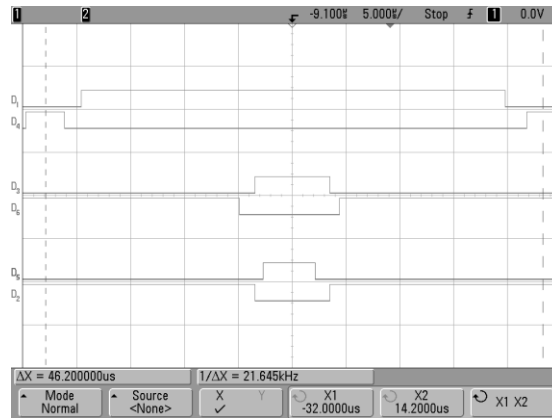


Figura 5.3 Tiempos muertos en las señales de control.

El tiempo muerto en cada una de las transiciones genera otras combinaciones de transistores, estas nuevas combinaciones se generan a partir de las combinaciones previa y posterior y aplicarles una operación lógica AND para enviar esta nueva combinación al puerto G. La operación lógica AND cumple con las necesidades de generar la nueva combinación necesaria. La Figura 5.4 muestra en un diagrama la forma de operación.

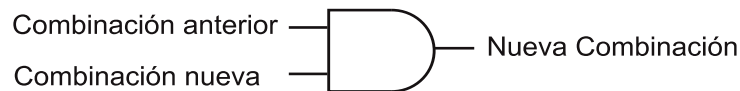


Figura 5.4 Compuerta AND como generador de combinación de tiempo muerto.

5.3.Verificación experimental del inversor Trifásico y la modulación SVPWM.

Para comprobar el funcionamiento del inversor se utilizó uno ya armado en el Laboratorio de Calidad y Conversión de la Energía que funciona como filtro activo de potencia. Este filtro activo y el su acercamiento hacia el inversor se muestran en las Figuras 5.5 y 5.6 respectivamente.

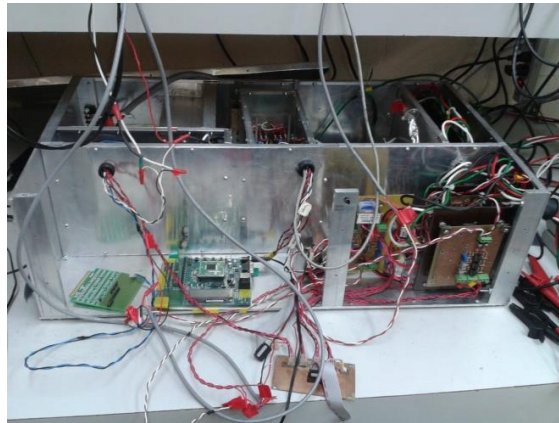


Figura 5.5 Filtro activo de potencia.

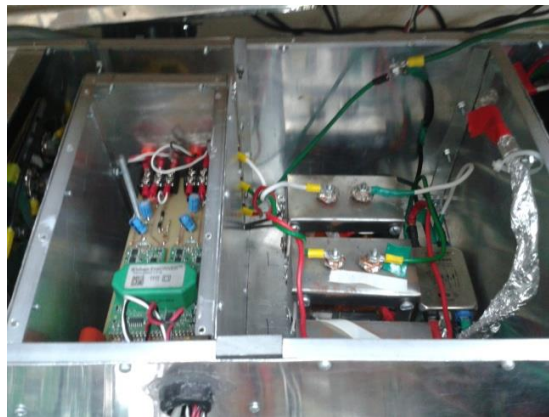


Figura 5.6 Inversor trifásico del filtro activo.

El inversor mostrado está construido con un módulo de IGBT's BSM25GD120DN2 de la marca INFINEON y tres drivers 2SC0108T de la marca CONCEPT que se utilizan para controlar la conmutación de los transistores. La Figura 5.7 muestra en un diagrama a bloques las etapas que conforman este inversor

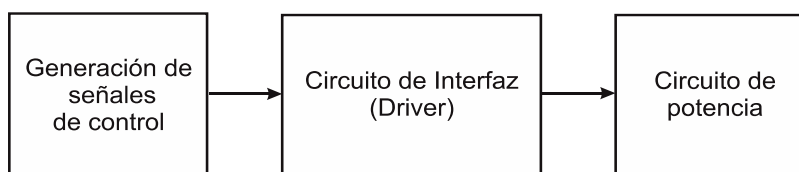


Figura 5.7 Diagrama a bloques del inversor.

La Figura 5.8 muestra los voltajes V_{RN} , V_{SN} y V_{TN} en ese mismo orden, observándose el claro desfase de 120° correspondiente a una señal trifásica. Utilizando la base de tiempo que se ve

en la misma figura, se puede constatar que el periodo de las señales tiene una duración de 16.6 ms lo que arroja una frecuencia de 60 Hz.

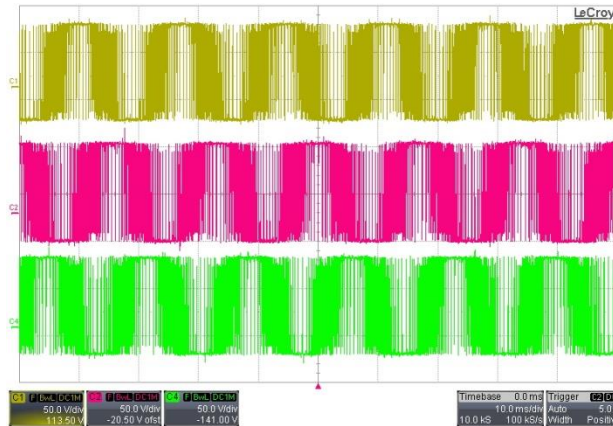


Figura 5.8 Voltajes V_{RN} , V_{SN} y V_{TN} obtenidos con SVPWM.

La Figura 5.9 muestra una ampliación de la Figura 4.9 para un ciclo de la frecuencia fundamental de las señales.

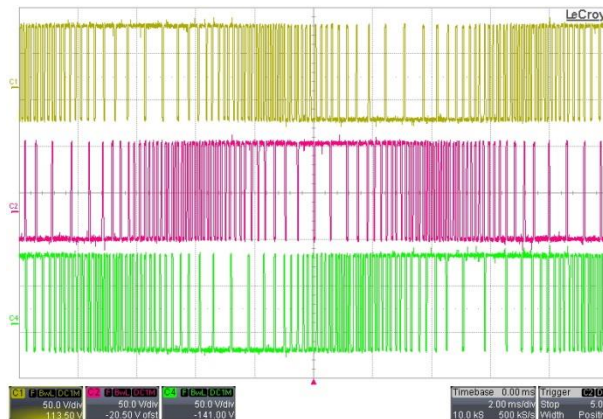


Figura 5.9 Acercamiento de los voltajes V_{RN} , V_{SN} y V_{TN} obtenidos con SVPWM.

5.4.Desarrollo físico del PLL.

Para comprobar el funcionamiento físico del PLL se utilizó una fuente modelo 4500Ls, de la marca California Instruments de 4500 VA. Esta fuente puede variar la amplitud y la frecuencia del voltaje de salida, estas son las principales características por las que se eligió el uso de esta fuente y no la conexión directa al voltaje trifásico del Laboratorio de la calidad y conversión de la energía.

Al tomarse en cuenta que el ADC del dsPIC acepta niveles de voltaje de 0 a 3.3 volts, fue necesario desarrollar una tarjeta acondicionadora de voltaje, mostrada en la Figura 5.10, esta

tarjeta consta de 3 transformadores que reducen el voltaje y un 3 amplificadores operacionales que dan un offset de 1.65 volts para trabajar en el rango indicado por el microcontrolador. Dicha tarjeta fue desarrollada por Onésimo Flores Acoltzi cuyo trabajo se reporta en su tesis de licenciatura.

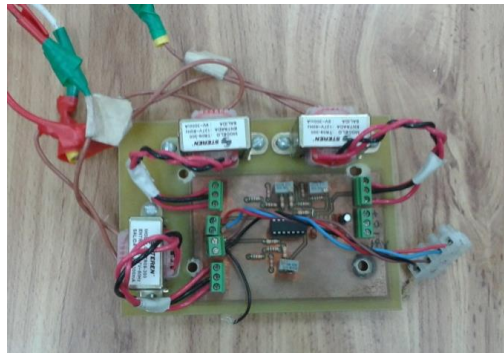


Figura 5.10. Tarjeta acondicionadora de voltaje.

Para poder hacer uso del dsPIC se utilizó la tarjeta de desarrollo Explorer 16 de Microchip. Esta tarjeta cuenta con un conector de expansión que permite poder utilizar y crear diferentes tipos de conexiones que se adecuen al trabajo que se esté realizando. La Figura 5.11 muestra este conector, la tarjeta de desarrollo y el dsPIC.

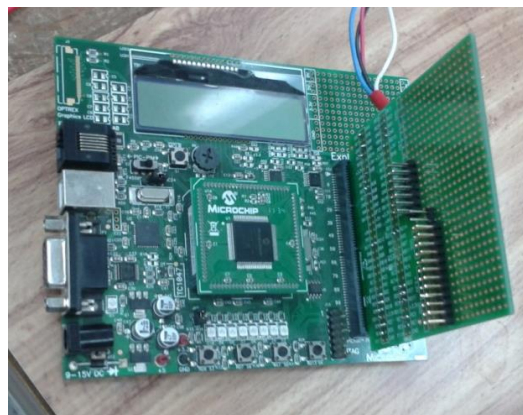


Figura 5.11. dsPIC en la tarjeta de desarrollo Explorer 16 con su conector de expansión.

5.5.Verificación de transformación de Clarke en el dsPIC.

A través de los años, la velocidad de los microcontroladores se ha ido incrementando, sin embargo, algunas aplicaciones requieren de realizar muchas operaciones de precisa exactitud, tal es el caso de este PLL. Los lenguajes de programación juegan un papel muy importante si es que el tiempo se torna vital en dicha aplicación. El lenguaje C, de medio nivel, consume muchos recursos

cuando se realizan operaciones con valores de tipo flotante, es por ello que la programación se lleva a un lenguaje de bajo nivel, disminuyendo el tiempo de ejecución de operaciones.

El dsPIC cuenta con diferentes formatos de recolección de datos por el ADC, estos formatos son utilizados de tal manera que se ajusten a las necesidades del programa. Tomando en cuenta que el dsPIC no cuenta con un convertidor digital-analógico y que los algunos puertos más no son continuos, es decir existen puertos dedicados para tareas específicas, se recurrieron a ideas sencillas para la verificación de operaciones del PLL.

Los registros de los acumuladores del dsPIC son cargados a ciertas variables, pero los formatos en lo que éstos se guardan no son propios para ser entendidos por un usuario ajeno al desarrollo del PLL, de esta manera la transformación de Clarke se realizó de la siguiente manera. Se tomaron registros que ejemplifican las señales α y β y trabajan con una lógica similar al ZCD, es decir, estos registros envían un “1” lógico al puerto cuando las señales sinusoidales están en el ciclo positivo y un “0” lógico cuando las señales están en el ciclo negativo, de esta manera, nuestras señales sinusoidales son ahora pulsos cuadrados.

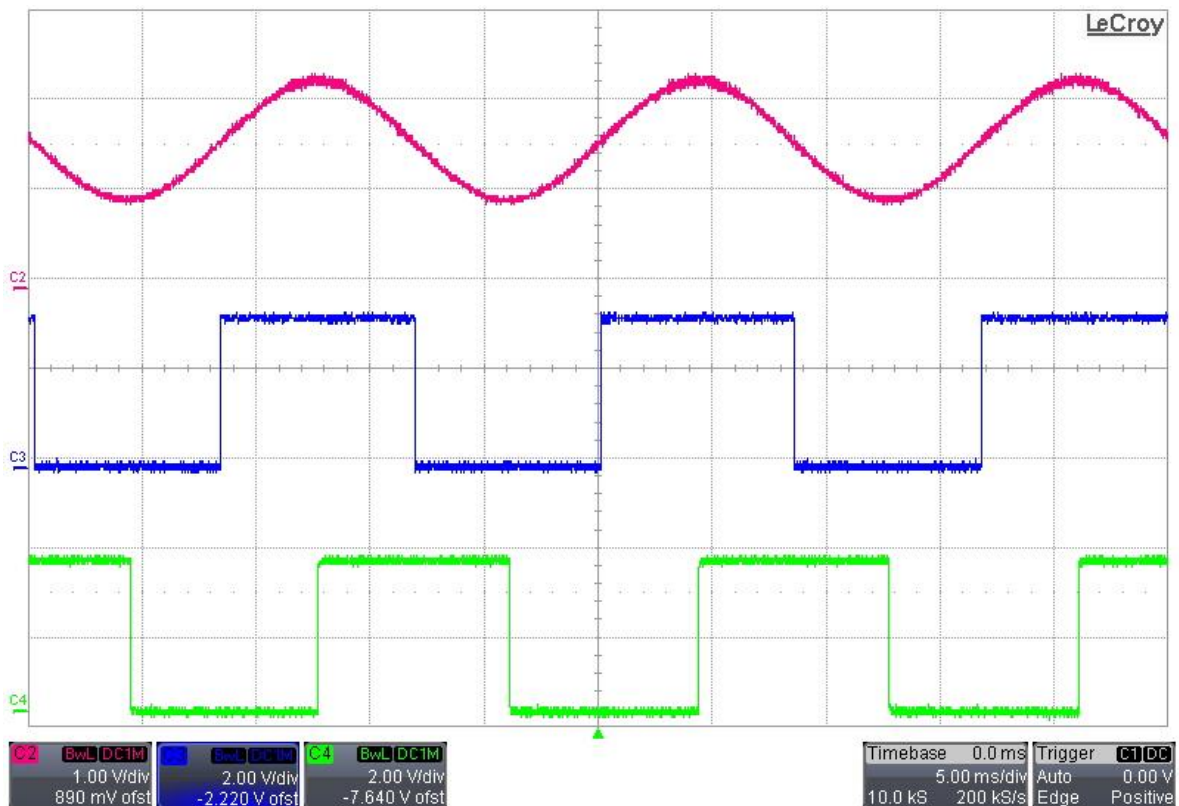


Figura 5.12. Comprobación de las la transformación de Clarke.

La Figura 5.12 muestra en sus señales la transformación de Clarke, la forma de onda sinusoidal es el voltaje V_R , la primera de las señales en orden descendente; la señal cuadrada en medio es la señal α mientras que β es representada por la última señal. Como se observa en la misma figura, las señales α y β están desfasadas 90° una respecto a la otra, mientras que α está en fase con V_R .

5.6.Verificación experimental del PLL.

La verificación experimental se realizó de una manera similar a la comprobación de la transformación de Clarke. Se inició el microcontrolador e inmediatamente se encendió la fuente trifásica, se dejó al trabajar durante aproximadamente 1 seg y se detuvo para comprobar el funcionamiento del PLL digital, tal como se observa en la Figura 5.13.

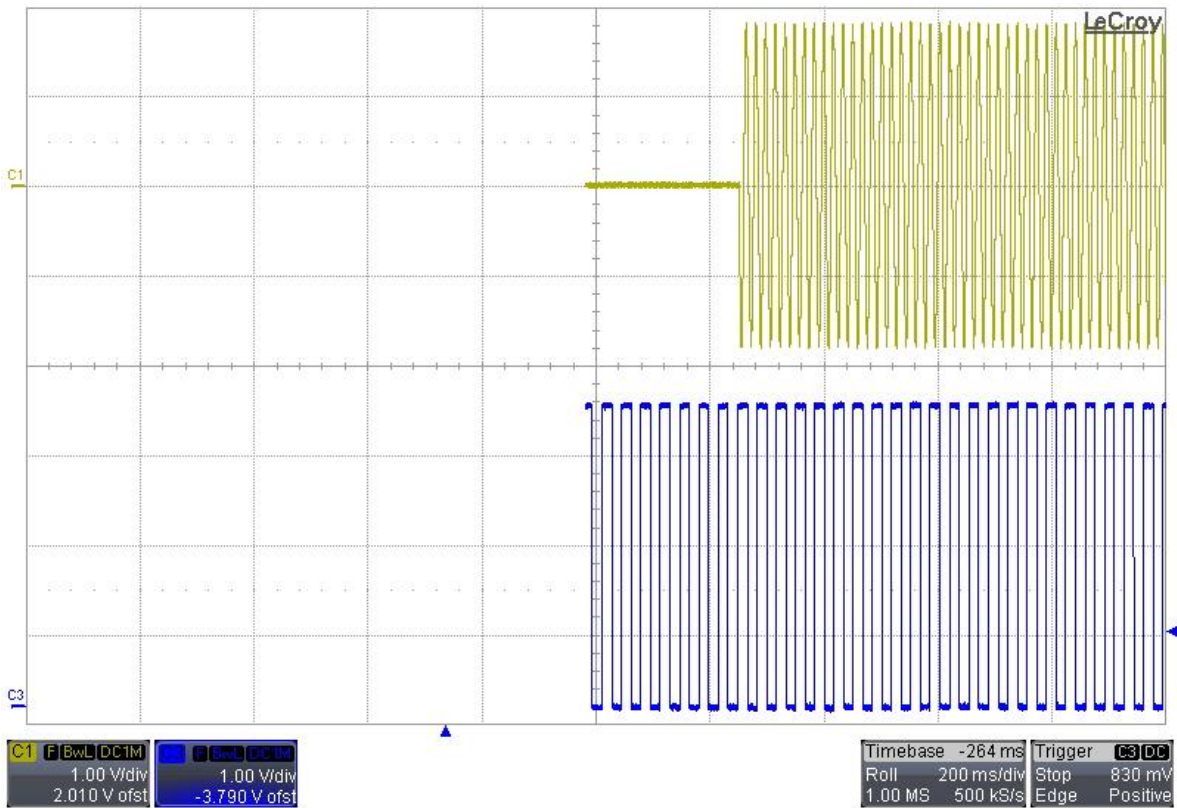


Figura 5.13 Funcionamiento del PLL utilizando una fase y una señal cuadrada

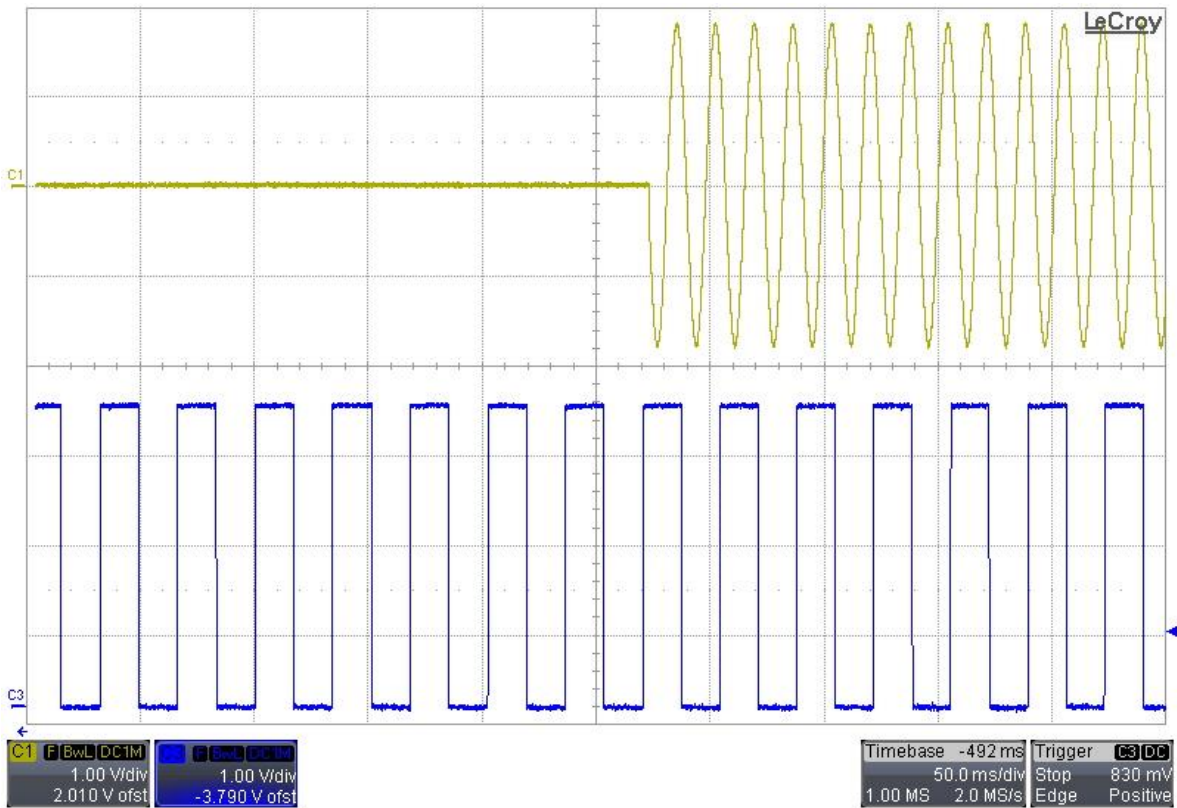


Figura 5.14. Sincronización de la SVPWM con V_R a través del PLL digital.

Como se observa en la Figura 5.14, la señal cuadrada representa un periodo de la frecuencia fundamental de las señales de control, es decir, esta señal tiene una frecuencia inicial de 30 Hz pues cuando la SVPWM genera los 360 pulsos de los 6 sectores la señal cuadrada cambia de “1” a “0” y viceversa. Al encenderse la fuente trifásica se puede ver que el PLL está desfasado 90° con respecto al voltaje V_R , pero también se puede notar que al pasar cierta cantidad de periodos el PLL hace que la señal fundamental de la SVPWM quede en fase con V_R .

Las Figuras 5.15 y 5.16 muestran las respectivas ampliaciones de la Figura 5.13. La distorsión en la onda cuadrada se debe al número de muestras tomadas por el osciloscopio y por consiguiente su poca legibilidad en la forma de onda.

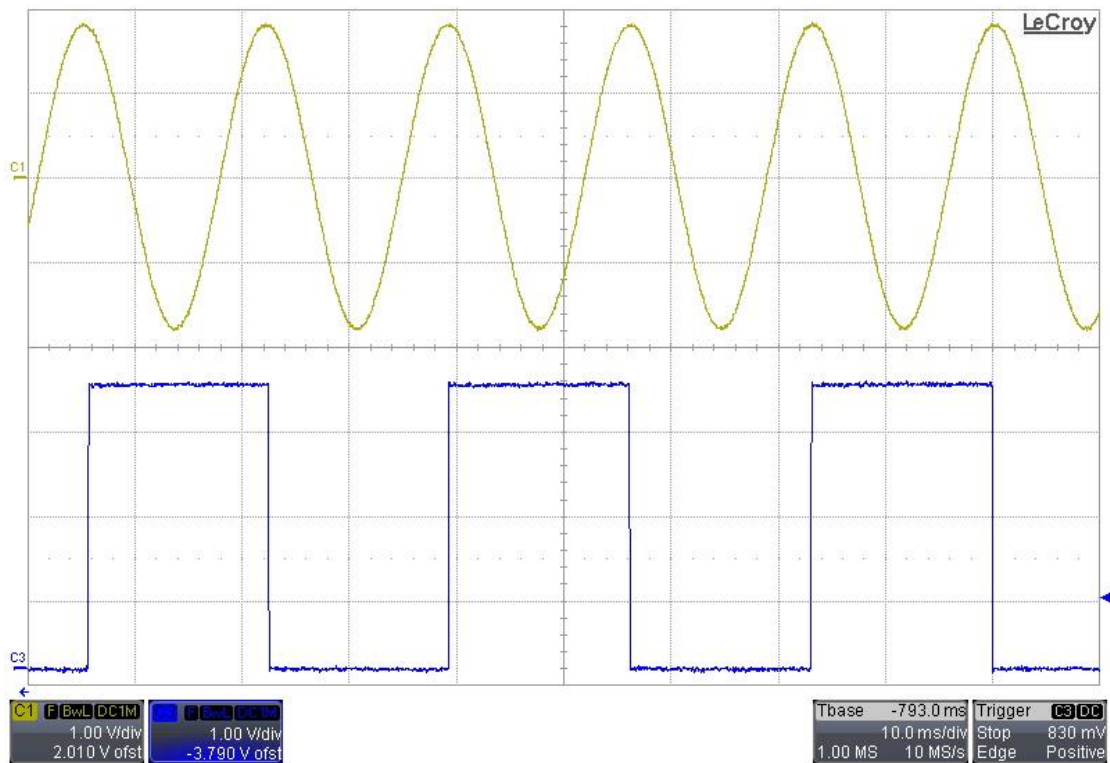


Figura 5.15 Sincronización de la SVPWM.

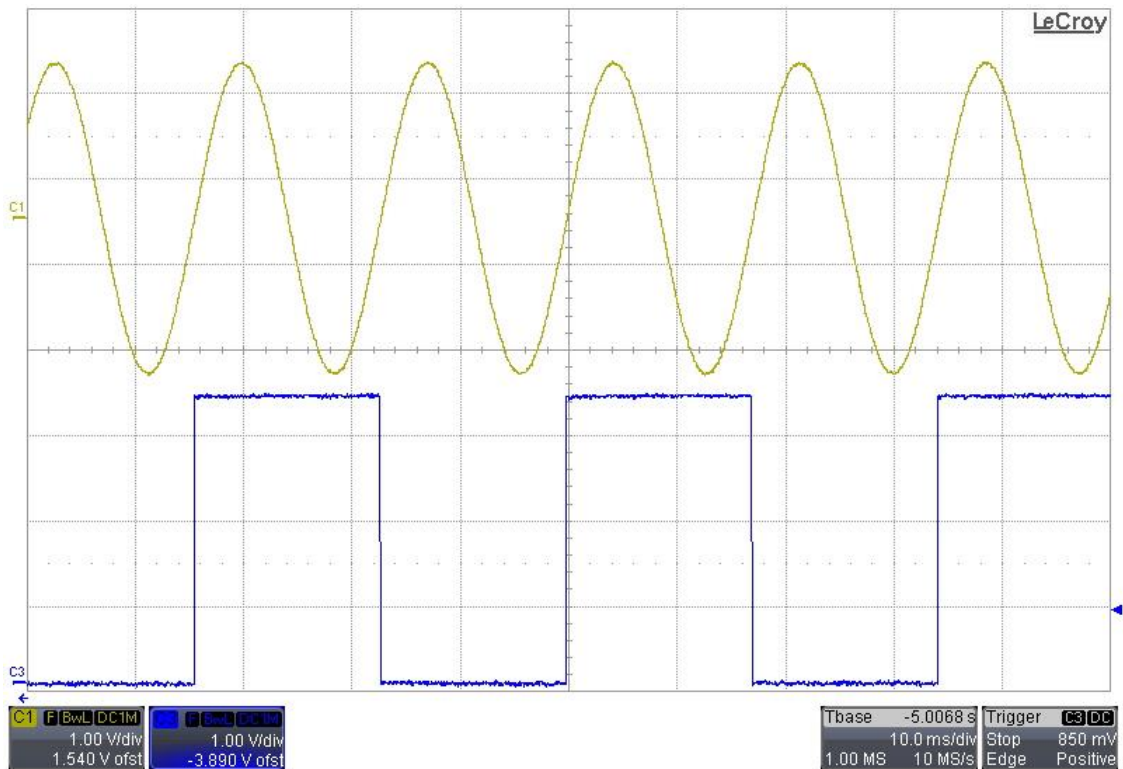


Figura 5.16 Acercamiento a las formas de onda sincronizadas por el PLL digital.

En la Figura 5.17 se hace una superposición de las señales del PLL y la señal de entrada V_R para ver más claramente el enfazamiento.

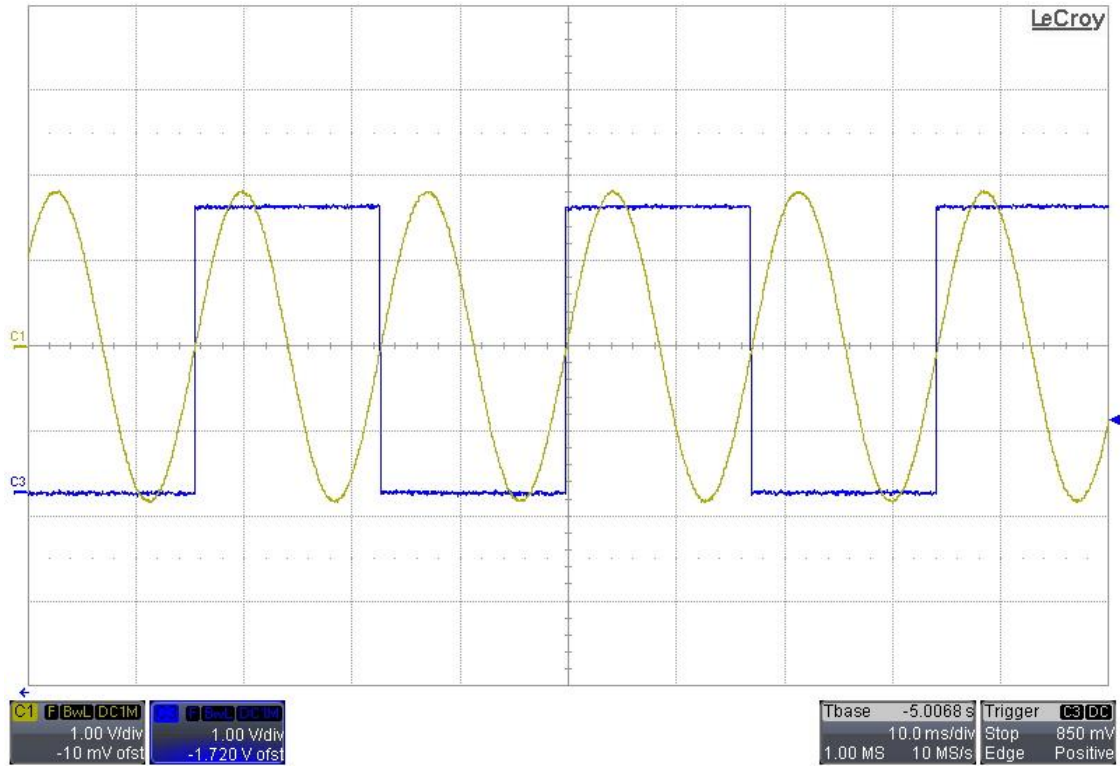


Figura 5.17 Superposición de la señal del PLL y V_R .

Para comprobar el funcionamiento del PLL digital en conjunto con el inversor trifásico se puede observar la Figura 5.18 la cual sincroniza el voltaje V_{RN} a la fase R de la fuente trifásica. Hay que señalar que el PLL solo se sincroniza en el rango de frecuencias que va de 57.5 Hz a 58.89 Hz. Esto se debe a la pérdida de información en la recopilación de datos del ADC y en la transformación que se requiere en la ecuación diferencia, pues para tener la exactitud requerida se realizaron arreglos de registros de 16 bits para obtener registros de 32 bits, así como una compleja operación con los acumuladores del dsPIC, de esta manera el tiempo en el que se realizan todas las operaciones necesarias para controlar un ciclo de la SVPWM y las operaciones del PLL es suficiente.

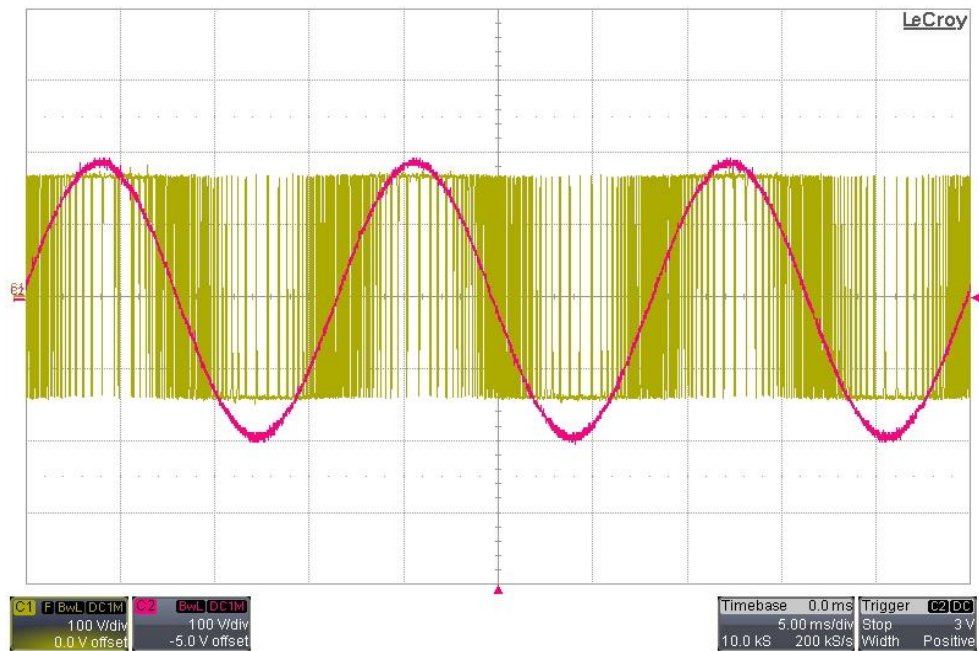


Figura 5.18. Sincronización de V_{RN} con la fase R de la fuente trifásica usando el PLL.

La Figura 5.19 muestra un acercamiento de la Figura 5.18.

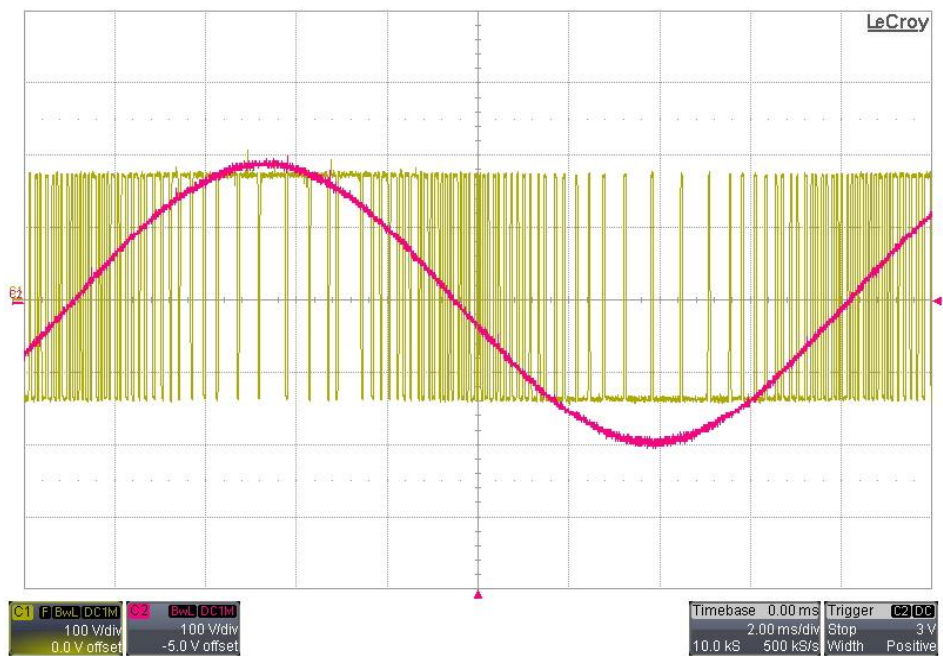


Figura 5.19. Acercamiento de las señales sincronizadas.

5.7. Resumen.

El capítulo muestra a detalle la manera en la que fue implementado el PLL, las principales características del convertidor que fue sincronizado, el equipo utilizado así como los circuitos y conexiones necesarias para el desarrollo. Además, se señalan y se muestran las formas de onda que comprueban el funcionamiento de dicha implementación. Finalmente se describe la comprobación del PLL y se muestran las señales que aseguran su funcionamiento.

CAPÍTULO 6

6. Observaciones y Conclusiones

6.1.Introducción.

En este capítulo se desarrollan las conclusiones y observaciones generales referentes a este trabajo de tesis, así como el trabajo a futuro existente a partir de este trabajo de investigación.

6.2.Trabajo Futuro.

Para trabajos futuros se pretende utilizar un DSP de punto flotante de la Texas Instruments con una frecuencia de 150MHz para ser implementado ahí el algoritmo del PLL. Esto ayudaría a que las operaciones propias del PLL se realizaran en menor tiempo comparado con la velocidad del dsPIC pues este DSP no cambia de formatos y realiza operaciones de este tipo en un ciclo de máquina. Al ser utilizado el DSP como PLL se puede sincronizar con el dsPIC el cual se puede utilizar para realizar diferentes tipos de modulaciones para convertidores de potencia.

6.3.Conclusiones y observaciones.

El capítulo completo muestra un desarrollo de varias etapas de trabajo, desde el desarrollo circuitos acondicionadores de voltaje hasta métodos de programación complejos. El desarrollo del programa en el dsPIC combina dos lenguajes de programación, Ensamblador y C, los cuales también utilizan diferentes formatos de datos, entre ellos los usados comúnmente como *int*, *float*, *unsigned int*, *doublé* y otros menos conocidos como los son el *Q15* y *fractional*. Estos formatos se usan para combinar registros de 16 bits y así lograr una precisión en las operaciones propias del PLL disminuyendo el tiempo de las mismas. Es por esta misma razón que el dsPIC pierde mucha información al cambiar de formatos en cada rutina de trabajo y el PLL no se sincroniza utilizando todo el rango por el cual fue diseñado.

ANEXO A



7. Código del programa del SVPMW y PLL digital.

A continuación se presenta el código del programa desarrollado de manera híbrida usando los lenguajes de programación C y Ensamblador, compilado en C30 de Microchip.

```
#include "p33FJ256MC710A.h"
#define __EXPLORER16_BRD__ 1 // Explorer 16 Development Board
#include <libq.h>
#include <math.h>
#include "common.h"
#include "float.h"
#include "stdio.h"
#include "ctype.h"
#define PI 3.141592654
#define __EXPLORER16_BRD__ 1 // Explorer 16 Development Board

_FOSCSEL(FNOSC_FRC); // FRC Oscillator
_FOSC(FCKSM_CSECMD & OSCIOFNC_OFF & POSCMD_NONE);
_FWDT(FWDTEN_OFF); // Watchdog Timer Enabled/disabled by user
software
_Q15 _Q15cos(_Q15 ANGULO);
_Q15 _Q15sin(_Q15 GRADO);
_Q15 _Q15sinPI(_Q15 GRADO);
_Q15 _Q15cos(_Q15 GRADO);
_Q15 _Q15cosPI(_Q15 GRADO);
_Q15 _Q15asin(_Q15 R);

void initTM3(void);
void initTM2(void);
void initCMP(void);
void initADC(void);
void initDMAC0(void);
unsigned int SEN0[363], COSENO[363], alfa, beta, q, aux;
unsigned int R, S, T;
int y, z=90, i=0, k, i, SPUL[14], PERIODO2;
int AD, GRADO, PERIODO, INDICE, PULSO, AS, JJ, PUL, MAX, gra1, gra2, gra3;
int VT0, VT1, VT2, VT3, VT13, VT01, VT02, VT12, VT23, SECTOR, PLL;
unsigned int T0, T1, T2, TV0, h, a, U, b, VCO=0, d, x;
unsigned int UK1P1, UK1P2, UKP1, UKP2, UK2P1, UK2P2, EK1P1, EK1P2, EKP1, EKP2,
EK2P1, EK2P2;
unsigned int A11, A12, A21, A22, A31, A32, A41, A42, A51, A52, RES2, RES1, AUX;
int SV0=0x008A, SV1=0x0089, SV2=0x0085, SV3=0x0086, SV4=0x0046, SV5=0x004A,
SV6=0x0049, SV7=0x0045;
int S01=0x0088, S02=0x0080, S03=0x0082, S04=0x0002, S05=0x000A, S06=0x0008;
int S17=0x0001, S27=0x0005, S37=0x0004, S47=0x0044, S57=0x0040, S67=0x0041;
int S12=0x0081, S23=0x0084, S34=0x0006, S45=0x0042, S56=0x0048, S61=0x0009;
int S1=0, S2=0, S3=0, S4=0, S5=0, S6=0;
signed long RESULTADO, FIN;
float ANGULO, M=0.84831, p, r, s, Q;
float TV1[363], TV2[363], seno, coseno;
int PERIODO2, EXP;

int main ( void )
{
    // Configure Oscillator to operate the device at 40Mhz
    // Fosc= Fin*M/(N1*N2), Fcy=Fosc/2
    // Fosc= 8M*40/(2*2)=80Mhz for 8M input clock
    PLLFBD=54; // M=40
    CLKDIVbits.PLLPOST=0; // N1=2
    CLKDIVbits.PLLPRE=0; // N2=2
    OSCTUN=0; // Tune FRC oscillator, if FRC is used
    // Disable Watch Dog Timer
```

```

    RCONbits.SWDTEN=0;
    // Clock Switch to incorporate PLL
    __builtin_write_OSCCONH(0x01); // Initiate Clock Switch to
                                     // FRC with PLL
(NOSC=0b001)
    __builtin_write_OSCCONL(0x01); // Start clock switching
    while (OSCCONbits.COSC != 0b001); // Wait for Clock switch to occur
    // Wait for PLL to lock
    while(OSCCONbits.LOCK!=1) {};

    TRISG = 0;
    TRISE = 0;
    PORTE = 0;
    PORTG = 0;
    TRISC=0x000E;
    TRISA = 0;
    PORTA = 0;
    TRISD = 0;
    PORTD = 0;
    AS = 1;
    GRADO = 1;
    PUL=60;
    MAX=(PUL*6)+1;
    i=1;
    PLL=1;
//-----INICIALIZACION - DE - VARIABLES - PARA - ECUACION - DIFERENCIA-----
    A11=0x01FB;
        A12=0x6EDF;
        A21=0x00FB;
        A22=0x6EDF;
        A31=0x0187;
        A32=0xA0E6;
        A41=0x0000;
        A42=0x2E63;
        A51=0x0187;
        A52=0x727F;

        UK1P1=0;
        UK1P2=0;
        UKP1=0;
        UKP2=0;
        UK2P1=0;
        UK2P2=0;
        EK1P1=0;
        EK1P2=0;
        EKP1=0;
        EKP2=0;
        EK2P1=0;
        EK2P2=0;

//----- C A L C U L O - D E - T I E M P O S - P A R A - S V P W M-----
    while(i<7)
    {
        SPUL[i]=PUL*i;
        i++;
    }

    while (GRADO<MAX)
    {
        ANGULO = GRADO*PI/((MAX-1)/2);
        coseno=cos (GRADO*PI/180);
        seno=sin (GRADO*PI/180);
        COSENO[GRADO]= _Q15ftoi (coseno);
    }

```

```

        SENO[GRADO]= _Q15ftoi(seno);

        if (GRADO<=SPUL[1]) //SECTOR 1
        {
            TV1[GRADO]=cos (ANGULO + (PI/6));
            TV2[GRADO]=cos (ANGULO - (PI/2));
            GRADO++;
        }

        else if (GRADO>SPUL[1] && GRADO <=SPUL[2]) //SECTOR 2
        {
            TV1[GRADO]=cos (ANGULO - (PI/6));
            TV2[GRADO]=cos (ANGULO - (5*PI/6));
            GRADO++;
        }

        else if (GRADO>SPUL[2] && GRADO <=SPUL[3]) //SECTOR 3
        {
            TV1[GRADO]=cos (ANGULO - (PI/2));
            TV2[GRADO]=cos (ANGULO - (7*PI/6));
            GRADO++;
        }

        else if (GRADO>SPUL[3] && GRADO <=SPUL[4]) //SECTOR 4
        {
            TV1[GRADO]=cos (ANGULO - (5*PI/6));
            TV2[GRADO]=cos (ANGULO - (3*PI/2));
            GRADO++;
        }

        else if (GRADO>SPUL[4] && GRADO <=SPUL[5]) //SECTOR 5
        {
            TV1[GRADO]=cos (ANGULO - (7*PI/6));
            TV2[GRADO]=cos (ANGULO - (11*PI/6));
            GRADO++;
        }

        else if (GRADO>SPUL[5] && GRADO <=SPUL[6]) //SECTOR 6
        {
            TV1[GRADO]=cos (ANGULO - (3*PI/2));
            TV2[GRADO]=cos (ANGULO - (PI/6));
            GRADO++;
        }
    }

    PORTG = SV0;
    h=1;
    JJ=1;
    d=1;
    INDICE = 1;
    PULSO = 1;
    initTM3();
    initCMP();
    initADC();
    initDMAC0();
    T3CONbits.TON = 1; // Start Timer
    PR3 = 1194;
    PERIODO2=2389;

    while(1)
    {
        if (AS==1) {

```

```

        //PERIODO2=PR3*2;
        a=TV1[INDICE]*PR3*M;
        b=TV2[INDICE]*PR3*M;
        TV0 = (TV1[INDICE]*PR3*M) + (TV2[INDICE]*PR3*M);
        T0 = (PR3-TV0)/2;
        T1 = a + T0;
        T2 = b + T1;
    }
    if (PULSO<=SPUL[1]){
        SECTOR=1; VT0=SV0;
        VT02=S02; VT01=S01; VT1=SV1; VT13=S17; VT12=S12; VT2=SV2; VT23=S27; VT3=SV7;}
    else if (PULSO>SPUL[1] && PULSO<=SPUL[2]){SECTOR=2; VT0=SV0; VT02=S02;
        VT01=S03; VT1=SV3; VT13=S37; VT12=S23; VT2=SV2; VT23=S27; VT3=SV7;}
    else if (PULSO>SPUL[2] && PULSO<=SPUL[3]){SECTOR=3; VT0=SV0; VT02=S04;
        VT01=S03; VT1=SV3; VT13=S37; VT12=S34; VT2=SV4; VT23=S47; VT3=SV7;}
    else if (PULSO>SPUL[3] && PULSO<=SPUL[4]){SECTOR=4; VT0=SV0; VT02=S04;
        VT01=S05; VT1=SV5; VT13=S57; VT12=S45; VT2=SV4; VT23=S47; VT3=SV7;}
    else if (PULSO>SPUL[4] && PULSO<=SPUL[5]){SECTOR=5; VT0=SV0; VT02=S06;
        VT01=S05; VT1=SV5; VT13=S57; VT12=S56; VT2=SV6; VT23=S67; VT3=SV7;}
    else if (PULSO>SPUL[5] && PULSO<=SPUL[6]){SECTOR=6; VT0=SV0; VT02=S06;
        VT01=S01; VT1=SV1; VT13=S17; VT12=S61; VT2=SV6; VT23=S67; VT3=SV7;}
    }
}

//---- I N T E R R U P C I O N - D E L - T I M E R -3- -----(GENERACION
DIENTE DE SIERRA)
void __attribute__((__interrupt__, no_auto_psv)) _T3Interrupt(void)
{
    IFS0bits.T3IF = 0;
    AS=!AS;
    if (AS==1)
    {
        PORTEbits.RE0=1;
        AD1CON1bits.SAMP = 1;
        OC1R = T0;
        OC4R= OC1R + z;
        OC2R = T1;
        OC3R = T2;
        PORTG = VT0;
        INDICE++;
        PORTEbits.RE3=PLL;
        if (INDICE==MAX)
        {
            INDICE = 1;
            PLL=!PLL;
        }
    }
    else if (AS==0)
    {
        PORTEbits.RE0=0;
        AD1CON1bits.SAMP = 0;
        OC3R=PR3-OC3R;
        OC4R= OC3R + z;
        OC2R=PR3-OC2R;
        OC1R=PR3-OC1R;

        //-----MULTIPLICACION DE E[K]-----
        if(EKP1>=0x8000)
        {
            S3=1;
            EKP1=EKP1-0x8000;
        }
    }
}

```

```

asm(" clr A");
asm(" mov  _EKP2,w7      ");
asm(" mov  _A32,w5      ");
asm(" mac  w5*w7, A      ");
asm(" sftac A, #16      ");
asm(" mov  _EKP1,w7      ");
asm(" mov  _A32,w5      ");
asm(" mac  w5*w7, A      ");
asm(" mov  _EKP2,w7      ");
asm(" mov  _A31,w5      ");
asm(" mac  w5*w7, A      ");
asm(" sftac A, #-8      ");
asm(" sac A, #0, w2     ");
asm(" mov  w2,_RES2     ");
RES2=RES2*256;
RES2=RES2/256;
asm(" sftac A, #8       ");
asm(" sftac A, #16      ");
asm(" mov  _A31,w7      ");
asm(" mov  _EKP1,w5      ");
asm(" mac  w5*w7, A      ");
asm(" sftac A, #-8      ");
asm(" mov  #0x0001,w7    ");
asm(" mov  _RES2,w5      ");
asm(" mac  w5*w7, A      ");
asm(" sac A, #0, w2     ");
asm(" mov  w2,_RES1     ");
asm(" sftac A, #-16     ");
asm(" sftac A, #-8      ");
asm(" sftac A, #8       ");
asm(" sac A, #0, w2     ");
asm(" mov  w2,_RES2     ");
//-----
if(S3==1)
{
RESULTADO=RESULTADO-((RES1*0x10000)+RES2);
S3=0;
EKP1=EKP1+0x8000;
}
else
{
RESULTADO=RESULTADO+((RES1*0x10000)+RES2);
}
//-----MULTIPLICACION DE E[K-1]-----
if(EK1P1>=0x8000)
{
S4=1;
EK1P1=EK1P1-0x8000;
}
asm(" clr A");
asm(" mov  _EK1P2,w7      ");
asm(" mov  _A42,w5      ");
asm(" mac  w5*w7, A      ");
asm(" sftac A, #16      ");
asm(" mov  _EK1P1,w7      ");
asm(" mov  _A42,w5      ");
asm(" mac  w5*w7, A      ");
asm(" mov  _EK1P2,w7      ");
asm(" mov  _A31,w5      ");
asm(" mac  w5*w7, A      ");
asm(" sftac A, #-8      ");
asm(" sac A, #0, w2     ");
asm(" mov  w2,_RES2     ");

```

```

RES2=RES2*256;
RES2=RES2/256;
asm(" sftac A, #8 ");
asm(" sftac A, #16 ");
asm(" mov _A41,w7 ");
asm(" mov _EK1P1,w5 ");
asm(" mac w5*w7, A ");
asm(" sftac A, #-8 ");
asm(" mov #0x0001,w7 ");
asm(" mov _RES2,w5 ");
asm(" mac w5*w7, A ");
asm(" sac A, #0, w2 ");
asm(" mov w2,_RES1 ");
asm(" sftac A, #-16 ");
asm(" sftac A, #-8 ");
asm(" sftac A, #8 ");
asm(" sac A, #0, w2 ");
asm(" mov w2,_RES2 ");

//-----
if(S4==1)
{
RESULTADO=RESULTADO-((RES1*0x10000)+RES2);
S4=0;
EK1P1=EK1P1+0x8000;
}
else
{
RESULTADO=RESULTADO+((RES1*0x10000)+RES2);
}
//-----MULTIPLICACION DE E[K-2]-----
if(EK2P1>=0x8000)
{
S5=1;
EK2P1=EK2P1-0x8000;
}
asm(" clr A");
asm(" mov _EK2P2,w7 ");
asm(" mov _A52,w5 ");
asm(" mac w5*w7, A ");
asm(" sftac A, #16 ");
asm(" mov _EK2P1,w7 ");
asm(" mov _A52,w5 ");
asm(" mac w5*w7, A ");
asm(" mov _EK2P2,w7 ");
asm(" mov _A51,w5 ");
asm(" mac w5*w7, A ");
asm(" sftac A, #-8 ");
asm(" sac A, #0, w2 ");
asm(" mov w2,_RES2 ");
RES2=RES2*256;
RES2=RES2/256;
asm(" sftac A, #8 ");
asm(" sftac A, #16 ");
asm(" mov _A51,w7 ");
asm(" mov _EK2P1,w5 ");
asm(" mac w5*w7, A ");
asm(" sftac A, #-8 ");
asm(" mov #0x0001,w7 ");
asm(" mov _RES2,w5 ");
asm(" mac w5*w7, A ");
asm(" sac A, #0, w2 ");
asm(" mov w2,_RES1 ");

```

```

asm(" sftac  A, #-16           ");
asm(" sftac  A, #-8           ");
asm(" sftac  A, #8           ");
asm("   sac  A, #0, w2        ");
asm("   mov   w2, _RES2       ");

//-----
if(S5==1)
{
RESULTADO=RESULTADO + ((RES1*0x10000)+RES2);
S5=0;
EK2P1=EK2P1+0x8000;
}
else
{
RESULTADO=RESULTADO- ((RES1*0x10000)+RES2);
}
//---- C O N V E R S I O N   -   D E L   -   R E S U L T A D O
FIN=RESULTADO;
RESULTADO=0;

if(FIN==0x00000000)
{
FIN=0;
}
else if(FIN==0x80000000)
{
FIN=0x81000000;
}
else if(FIN<0x80000000)
{
FIN=FIN*2;
}
else if(FIN>0x80000000)
{
FIN=FIN-0x80000001;
FIN=(0x7FFFFFFF-FIN);
FIN=FIN+0x80000000;
}
UKP2=FIN;
UKP1=(FIN-UKP2)/0x10000;
if(UKP1>=0x8000)
{
AUX=UKP1-0x8000;
VCO=- (AUX/0x0100);
}
else
{
VCO=UKP1/0x0100;
}
//-----R E A S I G N A C I O N   --   D E   --   V A L O R E S-----
UK2P1= UK1P1;
UK2P2= UK1P2;
UK1P1= UKP1;
UK1P2= UKP2;
EK2P1= EK1P1;
EK2P2= EK1P2;
EK1P1= EKP1;
EK1P2= EKP2;
PR3=(PERIODO2-VCO)/2;
}
}

```



```

void __attribute__((__interrupt__,__auto_psv__)) _OC1Interrupt( void )
{
    IFS0bits.OC1IF = 0; // Clear OC1 interrupt flag
    if (AS==1 && OC1R!=OC2R && OC2R!=OC3R){y= 1; PORTG = VT01;
}
    else if (AS==1 && OC3R==OC2R && OC1R!=OC2R ){PORTG = VT01; y= 9;}
    else if (AS==0 && OC1R!=OC2R )
    {
        PORTG = VT01; y=7;
        PULSO++;
        if (PULSO==MAX){PULSO=h;}
    }
}

void __attribute__((__interrupt__,__auto_psv__)) _OC2Interrupt( void )
{
    IFS0bits.OC2IF = 0; // Clear OC2 interrupt flag
    if (AS==1 && OC2R!=OC1R && OC2R!=OC3R){PORTG = VT12;
y=3;}
    else if (AS==0 && OC2R!=OC1R && OC2R!=OC3R){PORTG = VT12; y=6; }
    else if (AS==1 && OC1R==OC2R ){ y=2; PORTG =VT02; }
    else if (AS==0 && OC1R==OC2R && OC2R!=OC3R )
    {
        PORTG = VT02; y=7;
        PULSO++;
        if (PULSO==MAX){PULSO=h;}
    }
}

void __attribute__((__interrupt__,__auto_psv__)) _OC3Interrupt( void )
{
    IFS1bits.OC3IF = 0; // Clear OC3 interrupt flag
    if (AS==1&& OC2R==OC3R){PORTG = VT13; y=4; }
    else if (AS==1&& OC2R!=OC3R){PORTG = VT23; y=4; }
    else if (AS==0&&OC2R!=OC3R ){PORTG = VT23; y=5; }
    else if (AS==0&&OC2R==OC3R ){PORTG = VT13; y=8; }
}

void __attribute__((__interrupt__,__auto_psv__)) _OC4Interrupt( void )
{
    IFS1bits.OC4IF = 0; // Clear OC4 interrupt flag
    if (y==1){PORTG = VT1; OC4R= OC2R+z;}
    else if (y==2){PORTG = VT2; OC4R= OC3R+z; }
    else if (y==3){PORTG = VT2; OC4R= OC3R+z; }
    else if (y==4){PORTG = VT3; }
    else if (y==5){PORTG = VT2; OC4R= OC2R+z; }
    else if (y==6){PORTG = VT1; OC4R= OC1R+z; }
    else if (y==7){PORTG = VT0; }
    else if (y==9){PORTG = VT1; OC4R= OC3R+z;}
    else if (y==8){PORTG = VT1; OC4R= OC1R+z;}
}

//----- I N I C I A L I Z A C I O N - D E L - D M A -----
    unsigned int BufferA[4] __attribute__((space(dma)));
void initDMAC0( void )
{
    DMA0CONbits.AMODE = 0;// Configure DMA for Register indirect mode
    DMA0CONbits.MODE = 0; // Configure DMA for Continuous Ping-Pong mode
    DMA0PAD = (volatile unsigned int)&ADC1BUF0; // Point DMA to ADC1BUF0
    DMA0CNT = 3; // 4 DMA request
    DMA0REQ = 13; // Select ADC1 as DMA Request source
    DMA0STA = __builtin_dmaoffset(BufferA);
    IFS0bits.DMA0IF = 0; //Clear the DMA interrupt flag bit
}

```

```

        IEC0bits.DMA0IE = 1;        //Set the DMA interrupt enable bit
        DMA0CONbits.CHEN=1;        // Enable DMA
    }
//-----I N T E R R U P C I O N   - D E L   -   D M A -----
void __attribute__((interrupt, no_auto_psv)) _DMA0Interrupt(void)
{
    R= BufferA[1];
    S= BufferA[2];
    T= BufferA[3];
//-----T R A N S F O R M A D A   -   D E   -   C L A R K -----
    CORCONbits.US=0;
    CORCONbits.IF=0;
    //Calculation of Alfa
    asm(" clr A");
    asm(" mov    #0x5555,w5    ");
    asm(" mov    _R,w7        ");
    asm(" mac    w7*w5, A     ");

    asm(" mov    #0x2AAA,w5    ");
    asm(" mov    _S,w7        ");
    asm(" msc    w7*w5, A     ");
    asm(" mov    _T,w7        ");
    asm(" mov    #0x2AAA,w5    ");
    asm(" msc    w7*w5, A     ");
    asm(" sac    A, #0, w2    ");
    asm(" mov    w2,_alfa     ");
    // Calculation of Beta
    asm(" clr A");
    asm(" mov    #0x49E6,w5    ");
    asm(" mov    _S,w7        ");
    asm(" mac    w7*w5, A     ");
    asm(" mov    #0x49E6,w5    ");
    asm(" mov    _T,w7        ");
    asm(" msc    w7*w5, A     ");
    asm(" sac    A, #0, w2    ");
    asm(" mov    w2,_beta     ");
    //Calculation of q
    asm(" clr A");
    aux=COSENO[INDICE];
    asm(" mov    _aux,w7      ");
    asm(" mov    _beta,w5    ");
    asm(" mac    w5*w7, A     ");
    aux=SENO[INDICE];
    asm(" mov    _alfa,w5     ");
    asm(" mov    _aux,w7      ");
    asm(" msc    w5*w7, A     ");
    asm(" sac    A, #0, w2    ");
    asm(" mov    w2,_q       ");
//Change from Q15 to K.32
    aux=0;
    if(q==0x0000)
    {
        q=0;
        aux=0;
    }
    else if(q==0x8000)
    {
        q=0x8100;
        aux=0;
    }
    else if(q<0x8000)
    {
        aux=(q*2)*0x100;
    }
}

```

```

q=(q*2)/0x100;
}
else if(q>0x8000)
{
q=q-0x8000;
q=(0x7FFF-q);
q=q*2;
aux=q*0x100;
q=q/0x0100;

q=q+0x8000;
}
EKP1=q;
EKP2=aux;
//-----E C U A C I O N   -   D I F E R E N C I A-----
-----
CORCONbits.US=1;
CORCONbits.IF=1;
CORCONbits.SATA=0;
CORCONbits.ACCSAT=1;
CORCONbits.SATDW=1;
//-----MULTIPLICACION DE U[K-1]-----
if(UK1P1>=0x8000)
{
S1=1;
UK1P1=UK1P1-0x8000;
}
asm(" clr A");
asm(" mov   _UK1P2,w7   ");
asm(" mov   _A12,w5    ");
asm(" mac   w5*w7, A   ");
asm(" sftac A, #16     ");
asm(" mov   _UK1P1,w7  ");
asm(" mov   _A12,w5    ");
asm(" mac   w5*w7, A   ");
asm(" mov   _UK1P2,w7  ");
asm(" mov   _A11,w5    ");
asm(" mac   w5*w7, A   ");
asm(" sftac A, #-8     ");
asm(" sac  A, #0, w2   ");
asm(" mov   w2,_RES2   ");
RES2=RES2*256;
RES2=RES2/256;
asm(" sftac A, #8      ");
asm(" sftac A, #16     ");
asm(" mov   _A11,w7    ");
asm(" mov   _UK1P1,w5  ");
asm(" mac   w5*w7, A   ");
asm(" sftac A, #-8     ");
asm(" mov   #0x0001,w7 ");
asm(" mov   _RES2,w5   ");
asm(" mac   w5*w7, A   ");
asm(" sac  A, #0, w2   ");
asm(" mov   w2,_RES1   ");
asm(" sftac A, #-16    ");
asm(" sftac A, #-8     ");
asm(" sftac A, #8      ");
asm(" sac  A, #0, w2   ");
asm(" mov   w2,_RES2   ");
//-----
if(S1==1)
{
RESULTADO=RESULTADO-((RES1*0x10000)+RES2);
}

```

```

    UK1P1=UK1P1+0x8000;
    S1=0;
}
else
{
    RESULTADO=RESULTADO+((RES1*0x10000)+RES2);
}
//-----MULTIPLICACION DE U[K-2]-----
if(UK2P1>=0x8000)
{
    S2=1;
    UK2P1=UK2P1-0x8000;
}
asm(" clr A");
asm(" mov    _UK2P2,w7    ");
asm(" mov    _A22,w5     ");
asm(" mac    w5*w7, A    ");
asm(" sftac  A, #16     ");
asm(" mov    _UK2P1,w7    ");
asm(" mov    _A22,w5     ");
asm(" mac    w5*w7, A    ");
asm(" mov    _UK2P2,w7    ");
asm(" mov    _A21,w5     ");
asm(" mac    w5*w7, A    ");
asm(" sftac  A, #-8     ");
asm(" sac   A, #0, w2    ");
asm(" mov    w2,_RES2    ");
RES2=RES2*256;
RES2=RES2/256;
asm(" sftac  A, #8      ");
asm(" sftac  A, #16     ");
asm(" mov    _A21,w7    ");
asm(" mov    _UK2P1,w5   ");
asm(" mac    w5*w7, A    ");
asm(" sftac  A, #-8     ");
asm(" mov    #0x0001,w7  ");
asm(" mov    _RES2,w5    ");
asm(" mac    w5*w7, A    ");
asm(" sac   A, #0, w2    ");
asm(" mov    w2,_RES1    ");
asm(" sftac  A, #-16    ");
asm(" sftac  A, #-8     ");
asm(" sftac  A, #8      ");
asm(" sac   A, #0, w2    ");
asm(" mov    w2,_RES2    ");
//-----
if(S2==1)
{
    RESULTADO=RESULTADO+((RES1*0x10000)+RES2);
    S2=0;
    UK2P1=UK2P1+0x8000;
}
else
{
    RESULTADO=RESULTADO-((RES1*0x10000)+RES2);
}
IFS0bits.DMA0IF = 0; //Clear the DMA0 Interrupt Flag
}

```