

87



**UNIVERSIDAD NACIONAL AUTÓNOMA  
DE MÉXICO**

**FACULTAD DE INGENIERIA**

**“ELEMENTOS PARA EL DISEÑO DE  
CONVERTIDORES CD-CA MONOFÁSICOS  
DE BAJO CONTENIDO ARMÓNICO”**

**T E S I S**

QUE PARA OBTENER EL TÍTULO DE:

**INGENIERO MECÁNICO ELECTRICISTA,  
ÁREA ELÉCTRICA Y ELECTRÓNICA**

PRESENTA

**ERIC VILORIA LOPEZ**

DIRECTOR DE TESIS: ING. JOSE RODELMAR OCAMPO SANTILLAN



MEXICO, D. F.

ABRIL DE 2001

290645



Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

## **DEDICATORIAS**

A mis padres:

**Profra. Natividad de Jesús López Narváez**

**Profr. Efrén Viloría Sánchez**

A mis hermanos:

**Ing. Constantino Viloría López**

**Ing. Cristian Juvenal Viloría López**

A mis tíos:

**Profra. Marcela López Narváez**

**Sr. Heriberto Rodríguez Esquivel**

A todos ellos: gracias por su apoyo y comprensión.

---

# ÍNDICE

---

---

<b>Introducción general</b> .....	xi
Antecedentes .....	xiii
Definición del problema .....	xiv
Objetivos .....	xvi
Contenido .....	xvii
<b>Capítulo 1. Electrónica de Potencia</b> .....	1
Introducción .....	3
1.1. Finalidad de la Electrónica de Potencia .....	3
1.2. Los sistemas de conversión potencia .....	4
1.2.1. Convertidores ca-cd .....	5
1.2.2. Convertidores ca-ca .....	6
1.2.3. Convertidores cd-cd .....	7
1.2.4. Convertidores cd-ca .....	7
1.3. Dispositivos semiconductores de potencia .....	8
1.3.1. El diodo .....	8
1.3.2. El SCR .....	9
1.3.3. El BJT .....	12
1.3.4. El MOSFET .....	14
1.3.5. El IGBT .....	15
<b>Capítulo 2. Análisis de Fourier</b> .....	17
Introducción .....	19
2.1. La serie trigonométrica de Fourier .....	19
2.1.1. La serie de cosenos .....	20
2.1.2. La serie de senos .....	21
2.2. Señales periódicas unipolares y bipolares .....	21
2.3. Serie de Fourier de una señal unipolar con un solo pulso por semiciclo .....	22
2.4. Serie de Fourier de una señal unipolar con varios pulsos por semiciclo .....	23
2.5. Serie de Fourier de una señal bipolar con varios pulsos por semiciclo .....	25
2.6. Parámetros de rendimiento basados en la serie de Fourier .....	26
2.6.1. Factor armónico de la $n$ -ésima componente ( $HF_n$ ) .....	27

---

2.6.2. Distorsión armónica total(THD).....	27
2.6.3. Factor de distorsión de orden m ( $DF_m$ ).....	27
2.6.4. Armónica de menor orden(LOH).....	28
<b>Capítulo 3. Configuraciones de convertidores cd-ca.....</b>	<b>29</b>
Introducción.....	31
3.1. Convertidores cd-ca.....	31
3.2. Configuración push-pull.....	32
3.3. Configuración de medio puente.....	34
3.4. Configuración de puente completo.....	36
3.5. Inversores alimentados con fuente de tensión.....	38
3.6. Inversores con enlace en cd variable.....	39
3.7. Inversores alimentados con fuente de corriente.....	40
3.8. Inversores trifásicos.....	42
<b>Capítulo 4. Técnicas de modulación en convertidores cd-ca.....</b>	<b>45</b>
Introducción.....	47
4.1. Consideraciones relacionadas con la modulación de ancho de pulso.....	47
4.2. Modulación de un solo ancho de pulso.....	48
4.3. Modulación uniforme de ancho de pulso (UPWM) unipolar.....	51
4.4. Modulación senoidal de ancho de pulso (SPWM) unipolar.....	53
4.5. Modulación senoidal modificada de ancho de pulso (MSPWM) unipolar.....	56
4.6. Modulación unipolar por inyección del tercer armónico (THIPWM).....	58
4.7. Modulación unipolar inyección de armónicos (HIPWM).....	60
4.8. Modulación trapezoidal de ancho de pulso (TPWM) unipolar.....	62
4.9. Modulación uniforme de ancho de pulso (UPWM) bipolar.....	64
4.10. Modulación senoidal de ancho de pulso (SPWM) bipolar.....	66
4.11. Modulación senoidal modificada de ancho de pulso (MSPWM) bipolar.....	68
4.12. Modulación bipolar por inyección del tercer armónico (THIPWM).....	70
4.13. Modulación bipolar por inyección de armónicos (HIPWM).....	72
4.14. Modulación trapezoidal de ancho de pulso (TPWM) bipolar.....	74
4.15. Comparación de las técnicas de modulación.....	76

---

<b>Capítulo 5. Técnicas para la eliminación de armónicos</b> .....	79
Introducción.....	81
5.1. El método de Newton.....	81
5.1.1. El método de Newton en sistemas de ecuaciones simultáneas.....	83
5.2. Eliminación de armónicos en señales unipolares.....	84
5.3. Eliminación de armónicos en señales bipolares.....	86
5.4. Comparación de las señales unipolares y bipolares.....	89
<b>Capítulo 6. Filtros</b> .....	91
Introducción.....	93
6.1. Tipos de filtros utilizados en inversores de potencia.....	93
6.1.1. El filtro C.....	94
6.1.2. El filtro L.....	95
6.1.3. El filtro LC serie.....	96
6.1.4. El filtro LC paralelo.....	99
6.1.5. El filtro LC serie-paralelo.....	101
6.2. Comparación de los filtros.....	103
<b>Capítulo 7. Elementos para la evaluación</b> .....	105
Introducción.....	107
7.1. El programa de simulación.....	107
7.1.1. La ventana de entrada.....	108
7.1.2. La ventana de modulación de ancho de pulso.....	109
7.1.3. La ventana de eliminación de armónicos.....	111
7.1.4. La ventana de filtros.....	112
7.2. El módulo de evaluación.....	114
7.2.1. Selección de la configuración inversora.....	114
7.2.2. Generación de las señales de excitación.....	115
7.2.3. Implementación del módulo de evaluación.....	116
7.3. Selección de los elementos del filtro de salida.....	119
7.4. El grabado del patrón de conmutación en la memoria.....	120

---

<b>Capítulo 8. Resultados</b> .....	123
Introducción .....	125
8.1. Simulación de las técnicas de modulación y eliminación.....	125
8.2. Oscilogramas de modulación de ancho de pulso .....	138
8.2.1. Modulación de un solo ancho de pulso.....	138
8.2.2. Modulación uniforme de ancho de pulso (UPWM) unipolar .....	140
8.2.3. Modulación senoidal de ancho de pulso (SPWM) unipolar .....	142
8.2.4. Modulación senoidal modificada (MSPWM) unipolar .....	144
8.2.5. Modulación unipolar por inyección del tercer armónico (THIPWM) .....	146
8.2.6. Modulación unipolar por inyección de armónicos (HIPWM) .....	148
8.2.7 Modulación trapezoidal (TPWM) unipolar .....	150
8.3. Oscilogramas de la técnica de eliminación de armónicos.....	152
8.3.1. Técnica unipolar de eliminación de armónicos .....	152
8.4. Comparación de las técnicas de modulación y eliminación .....	154
<b>Capítulo 9. Conclusiones y propuestas</b> .....	157
9.1. Semiconductores .....	159
9.2. Estructuras de conversión cd-ca.....	160
9.3. Filtros .....	160
9.4. Técnicas de modulación y eliminación.....	160
9.5 Propuestas para un convertidor cd-ca con bajo contenido armónico.....	161
9.5.1. Convertidor cd-ca con banco de baterías de 48 voltios .....	162
9.5.2. Convertidor cd-ca con banco de baterías de 192 voltios .....	163
<b>Apéndices</b> .....	165
A.1. Siglas.....	167
A.2. Código fuente del programa Marcela.....	168
A.3. Código fuente del programa Memoria .....	216
<b>Referencias</b> .....	225



---

# **INTRODUCCIÓN GENERAL**

---

## Antecedentes

La inversión de potencia es la conversión de potencia de cd a potencia de ca a un deseado valor de tensión, o corriente, y frecuencia [12]. El encargado de realizar tal función de inversión es el convertidor cd-ca, también conocido como inversor.

La configuración del circuito inversor de potencia consiste de dispositivos semiconductores de potencia que funcionan como interruptores. El inversor también tiene un circuito para el control de conmutación que proporciona los pulsos necesarios para encender y apagar cada interruptor en el momento y la secuencia adecuados. Los interruptores son repetidamente operados en tal forma que la fuente de cd en las terminales de entrada del inversor aparece como ca en sus terminales de salida.

Cuando un inversor es requerido para una conversión de potencia práctica o para una aplicación de control de potencia, las siguientes características de la salida de ca necesitan ser consideradas [11].

1. La frecuencia de ca, que es ajustada por medio de la frecuencia de conmutación de los interruptores.
2. Magnitud de la tensión de ca, controlada ya sea por la modificación de la entrada de cd o por medio de la modulación de ancho de pulso.
3. Forma de onda de la tensión de ca, siendo usualmente la forma senoidal la más deseada para muchas aplicaciones.
4. El número de fases, que dependiendo del tipo de carga que se maneje pueden ser tres o una, siendo el inversor trifásico o monofásico.

Al ser un inversor una eficiente y conveniente fuente de ca, ya sea que se requiera una frecuencia fija o variable, encuentra un amplio uso en equipo industrial. Para vislumbrar el papel de los inversores en la tecnología de la electrónica de potencia, se revisará brevemente algunas aplicaciones importantes.

Gran parte de la aplicación de inversores es el control de velocidad de motores de ca. La velocidad de un motor de ca, del tipo de inducción o síncrono, está determinado por la frecuencia de la ca. Por tanto, la velocidad puede ser convenientemente ajustada alimentando al motor con un inversor de frecuencia variable [11].

Una fuente de poder ininterrumpible está generalmente formada por un rectificador, un banco de baterías, un inversor y un filtro. Para abastecer cargas muy críticas tales como computadoras usadas para controlar procesos importantes y algunos equipos médicos, puede ser necesario utilizar fuentes de poder ininterrumpibles. Ellas ofrecen protección durante fallas en la línea, así como también regulación de tensión durante condiciones de sobretensión. También son excelentes en términos de supresión de transitorios en la línea y de disturbios armónicos [6].

En el rango de alta potencia, la corriente alterna es rectificadas para proporcionar transmisión a grandes distancias como corriente directa. Al final de la línea de transmisión, un convertidor cd-ca opera a 60 Hz para generar la corriente alterna del sistema otra vez. Tal arreglo brinda ventajas económicas y operacionales a una compañía eléctrica. Sistemas electrónicos de potencia de este tipo se encuentran en un amplio rango de equipo industrial y de consumo [2].

### **Definición del problema**

Reducir el contenido armónico en una señal de tensión, o de corriente, es uno de los retos más difíciles del diseño de convertidores cd-ca. Los armónicos no sólo reducen el factor de potencia de la señal de ca, sino que también interfieren con la apropiada operación del convertidor, u otros equipos, apareciendo como ruido en circuitos de control.

Los filtros de paso bajo consistentes de un capacitor en paralelo y un inductor en serie son utilizados para eliminar las componentes de ca, causadas por la conmutación, de la salida de convertidores cd-cd. Sin embargo, un filtro de paso bajo no es tan efectivo cuando se aplica a convertidores cd-ca. La causa es que la razón de la frecuencia de conmutación de entrada a la frecuencia de salida es infinita en convertidores cd-cd, pero finita para convertidores cd-ca. Para un convertidor cd-cd la cantidad de la atenuación del

rizo está limitada sólo por el tamaño del inductor y el capacitor, o quizás por el control deseado del ancho de banda. Sin embargo, para el convertidor cd-ca el tamaño y la efectividad de los elementos del filtro están determinados por factores tales como la atenuación y el defasamiento que pueden ser tolerados por la componente fundamental. Como resultado de esto, a menudo se controlan los interruptores de un convertidor cd-ca para lograr una reducción activa de los armónicos. Existen varias maneras de lograr esto. Una de ellas es la eliminación de armónicos, en la cual se controlan los interruptores para eliminar ciertos armónicos. Otra forma es la cancelación de armónicos, en donde se suman las salidas de dos o más convertidores para así cancelar armónicos predeterminados. Una alternativa a la reducción de armónicos es moverlos a frecuencias lo suficientemente altas para hacer posible el filtrado con componentes pequeños. Esto es llamado modulación por ancho de pulso [5].

Entre los motivos principales para reducir el contenido armónico de la salida de un inversor, remitiéndonos nuevamente al control de motores de ca y al diseño de fuentes de potencia ininterrumpibles, están las siguientes.

Los inversores usados en el control de frecuencia variable en motores de ca producen tensiones que contienen componentes de alta frecuencia que son armónicos de la frecuencia fundamental. Las pérdidas adicionales debido a estos armónicos están en el rango del 10% al 20% de las pérdidas totales. Además, la presencia de armónicos en la excitación del estator resulta en una componente de torque pulsante. Si los torques pulsantes están a bajas frecuencias pueden causar problemas como variaciones de velocidad y fatiga del eje [6].

En una fuente de potencia ininterrumpible la salida filtrada del inversor es normalmente especificada para contener una muy pequeña distorsión armónica, incluso aunque la mayoría de las cargas son altamente no lineales e inyectan grandes corrientes armónicas dentro del inversor. Por tanto, el inversor debe de permitir casi un control instantáneo sobre su forma de onda de salida de ca. El contenido armónico se especifica por medio de la llamada distorsión armónica total. Típicamente la distorsión armónica total máxima debe estar entre un 3% y un 5%, con cada armónico expresado como una razón de la frecuencia fundamental para ser menor al 3%. Es importante minimizar el contenido

armónico de la salida del inversor. Esto disminuye el tamaño del filtro, lo cual no sólo resulta en un ahorro de costo sino también en una mejora de la respuesta dinámica de la fuente de potencia ininterrumpible ante los cambios en la carga [6].

## Objetivos

El objetivo mas importante de este trabajo es analizar los elementos que constituyen un convertidor cd-ca monofásico, a fin de seleccionar los componentes y las técnicas más adecuadas para generar una señal de salida de bajo contenido armónico.

El método a seguir consiste en una revisión de los elementos que conforman un inversor de potencia, así como de la evaluación de algunas de las técnicas de modulación y eliminación de armónicos existentes. Para evaluar las técnicas de modulación se planea realizar un programa de simulación basado en el análisis de Fourier y construir un módulo de evaluación en el que se puedan observar resultados más tangibles.

Entre los resultados esperados, son cuatro los que se destacan y se pretende obtener:

1. Se espera que con la revisión de los componentes que pueden formar parte de un inversor monofásico, al final del trabajo se tenga la capacidad para decidir cuales son los más adecuados.
2. Se planea realizar un análisis de algunas de las técnicas de modulación y eliminación de armónicos existentes para decidir cuales tienen un desempeño más satisfactorio en un inversor monofásico de potencia.
3. Comprobar que un programa de simulación por computadora proporciona gran ayuda en el diseño de un inversor de potencia monofásico.
4. Comprobar, con la construcción de un módulo de evaluación, la información teórica reunida durante el desarrollo de la tesis.

## Contenido

El trabajo consta de 9 capítulos en los cuales se tratan los diferentes componentes de un inversor de potencia.

En el capítulo 1 se describe la estructura general de un convertidor de potencia y se muestran los diferentes tipos de convertidores que existen, también se comentan algunos tipos de interruptores de estado sólido actuales.

El capítulo 2 trata del análisis de Fourier, se definen los dos tipos de señales que genera un convertidor cd-ca y se muestra la forma en que dichas señales pueden ser sintetizadas con la serie trigonométrica.

Las topologías de los convertidores cd-ca y la forma en la que son alimentadas se muestran en el capítulo 3. Se describen los inversores monofásicos tipo push-pull, de medio puente y de puente completo y la forma en que a partir de un inversor monofásico se puede construir un inversor trifásico.

El capítulo 4 presenta las diferentes técnicas de modulación de ancho de pulso. Se muestra la forma en que las señales de excitación son generadas, las señales de salida que se producen y las características armónicas para cada técnica.

Las técnicas utilizadas para la eliminación de armónicos en inversores de potencia se detallan en el capítulo 5. Estas técnicas requieren del uso de ecuaciones no lineales, por esta razón en este capítulo se incluye la forma en que el método de Newton se aplica a la resolución de sistemas de ecuaciones simultáneas.

En el capítulo 6 se comentan los diferentes tipos de filtros que podrían ser utilizados para mejorar las señales de salida de un inversor de potencia. Se habla específicamente de filtros pasivos de primer, segundo y tercer orden. Se estudia las ventajas o desventajas de cada uno definiendo en donde tienen una mejor aplicación.

Como parte de este trabajo se elaboró un sistema de simulación de inversores y un módulo de evaluación. El sistema de simulación se desarrolló en Delphi 3 para su ejecución en un ambiente visual, mientras que el módulo de evaluación tiene como principal

característica el grabado de los patrones de conmutación en una memoria EPROM. El capítulo 7 describe el funcionamiento de ambos.

Los resultados obtenidos con los sistemas de simulación y el módulo de evaluación, principalmente gráficos, están incluidos en el capítulo 8. Los resultados son comentados de tal manera que se determina cuales son las técnicas más adecuadas para su uso en convertidores cd-ca monofásicos.

El capítulo 9 contiene las conclusiones finales. Se comentan los resultados a los que se llegaron en cada uno de los capítulos anteriores y se proponen dos estructuras para la construcción de un convertidor cd-ca monofásico de bajo contenido armónico.

Para finalizar, en los apéndices se incluyen los significados de las siglas utilizadas en el escrito y los códigos fuente tanto del sistema de simulación como del programa para generar el contenido de la memoria del módulo de evaluación.

---

# CAPÍTULO 1

---

ELECTRÓNICA DE POTENCIA



## **Introducción**

La electrónica de potencia tiene un papel importante en todo lugar donde se utilice energía eléctrica y se requiera realizar un cambio en la tensión, corriente o frecuencia a partir de los valores nominales disponibles. Es una tecnología que brinda soluciones a la industria en busca de controles cada vez más rápidos y eficaces para la energía eléctrica.

La electrónica de potencia tuvo su origen en la década de 1950 con el desarrollo del rectificador controlado de silicio (SCR) que era capaz de manejar potencias del orden de los kilowatts con velocidades mayores a las de cualquier interruptor electromecánico. Hoy, la electrónica de potencia tiene la capacidad de controlar cientos de megawatts y otros dispositivos han sido desarrollados para formar parte, junto con el SCR, de la familia de los interruptores de potencia de estado sólido

### **1.1. Finalidad de la electrónica de potencia**

En esencia lo que la electrónica de potencia hace es acondicionar la energía de una fuente para cubrir las necesidades de una carga.

La fuente de energía es generalmente de corriente alterna con tensión y frecuencia fijas y en algunos casos es una fuente de tensión constante de corriente directa. Las cargas pueden ser muchas y variadas, desde lámparas incandescentes hasta motores de gran potencia. Debido a que no todas las fuentes son adecuadas para alimentar a todas las cargas, el acondicionamiento de potencia es realizado muy frecuentemente.

Como ejemplo de la necesidad de conversión de potencia se tiene a los motores síncronos cuyo funcionamiento requiere de alimentaciones con frecuencia variable necesaria para incrementar o reducir su velocidad. El suministro público de energía eléctrica tiene una frecuencia fija, así que es necesario un sistema de conversión de potencia que sea capaz de proporcionar la tensión modificada con la frecuencia requerida.

## 1.2. Los sistemas de conversión de potencia

Un sistema de conversión de potencia convierte un tipo o nivel de tensión o de corriente en otro. Sin importar que tipo de conversión realice, la configuración del sistema es usualmente la misma. La figura 1.1 ilustra la configuración general de un sistema de conversión en forma de un diagrama de bloques. En el sistema de conversión el elemento principal es el interruptor semiconductor que se encuentra en el bloque de acondicionamiento de potencia, y dado que estos interruptores deben ser encendidos y apagados en una secuencia especial, es necesario un bloque de control. Existe también un bloque impulsor que se encarga de proporcionar la corriente y la tensión necesarios para encender y apagar los interruptores.

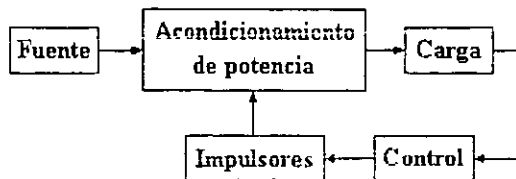


Figura 1.1. Sistema de conversión de potencia.

Los sistemas de conversión de potencia son clasificados por la relación que existe entre su entrada y su salida [4]. De acuerdo con lo anterior, existen cuatro tipos de sistemas de conversión de potencia:

1. Convertidores ca-cd.
2. Convertidores ca-ca.
3. Convertidores cd-cd.
4. Convertidores cd-ca.

### 1.2.1. Convertidores ca-cd

Los convertidores ca-cd se conocen también como circuitos rectificadores y pueden ser del tipo controlados o no controlados.

Un circuito rectificador no controlado convierte la tensión de ca en una tensión fija de cd como se muestra en la figura 1.2. En este circuito los interruptores 1 y 4 conducen durante los semiciclos rotulados con un 1 y los interruptores 2 y 3 durante los semiciclos rotulados con un 2. Los dispositivos que se utilizan para este sistema son los diodos.

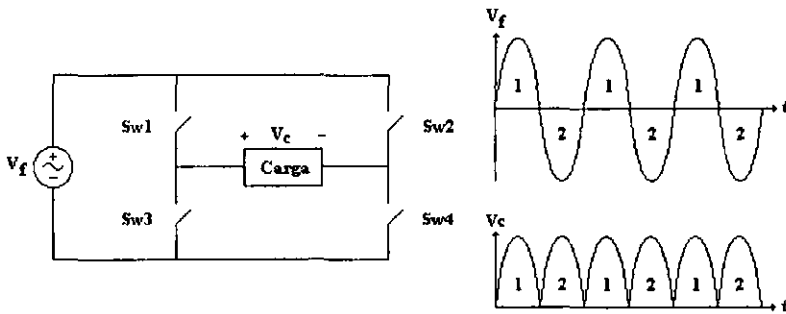


Figura 1.2. Circuito rectificador no controlado.

Un circuito rectificador controlado de ca-cd convierte la tensión de ca en una tensión de cd, pero el valor promedio de la tensión de salida se puede controlar variando el tiempo de conducción de sus interruptores, como se puede ver en la figura 1.3. Los interruptores sólo conducen en la parte del semiciclo que está rayada. En este circuito los interruptores 1 y 4 conducen durante los semiciclos rotulados con un 1 y los interruptores 2 y 3 durante los semiciclos rotulados con un 2. El semiconductor utilizado en estos sistemas es el SCR

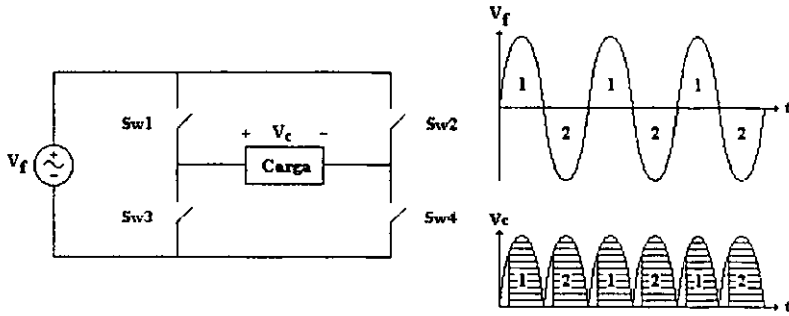


Figura 1.3. Circuito rectificador controlado.

### 1.2.2. Convertidores ca-ca

Los convertidores ca-ca se utilizan para obtener una tensión de corriente alterna variable a partir de una fuente de corriente alterna fija. La figura 1.4 muestra un convertidor ca-ca donde la tensión de salida se controla mediante la variación del tiempo de conducción del interruptor, el cual sólo conduce en la región sombreada. Este tipo de convertidores se conocen también como controladores de tensión de corriente alterna. El semiconductor utilizado en este convertidor es el triac.

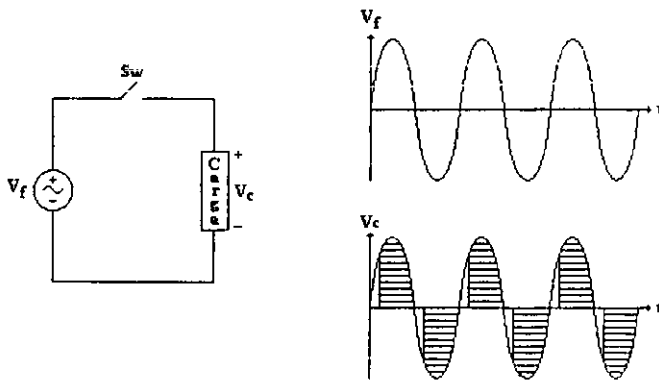


Figura 1.4. Convertidor ca-ca.

### 1.2.3. Convertidores cd-cd

Los convertidores cd-cd también se conocen como pulsadores o reguladores de conmutación, en la figura 1.5 aparece un convertidor de este tipo. La tensión promedio de salida  $V_m$  se controla con el tiempo de conducción del interruptor  $S_w$ , que es generalmente un transistor.

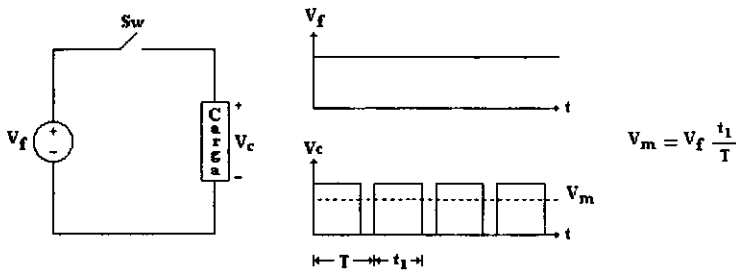


Figura 1.5. Convertidor cd-cd.

### 1.2.4. Convertidores cd-ca

Los convertidores cd-ca también se conocen como inversores. En la figura 1.6 se muestra un inversor con cuatro interruptores que pueden ser transistores. La entrada es una tensión de cd y su salida es una tensión de ca. Los interruptores 1 y 4 conducen durante los semiciclos de salida rotulados con un 1 y los interruptores 2 y 3 durante los semiciclos de salida rotulados con un 2. La salida se controla al variar el tiempo de conducción de los interruptores.

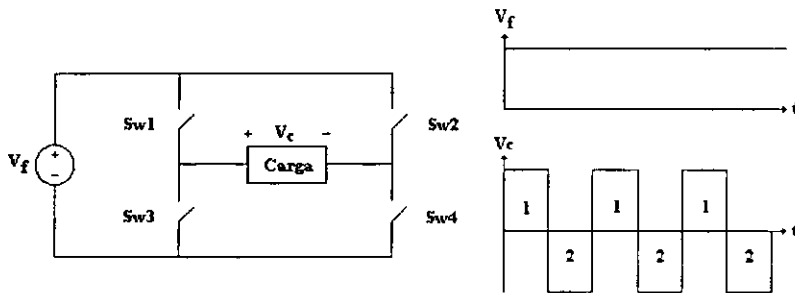


Figura 1.6. Convertidor cd-ca.

### 1.3. Dispositivos semiconductores de potencia

Como se mencionó, los dispositivos semiconductores son los elementos principales de un sistema de conversión de potencia electrónico. Entre los dispositivos más comunes se encuentran el diodo rectificador; el rectificador controlado de silicio, SCR; el transistor bipolar de juntura, BJT; el transistor de efecto de campo de metal, óxido y semiconductor, MOSFET; y recientemente el transistor bipolar de compuerta aislada, IGBT.

#### 1.3.1. El diodo

Los interruptores no controlados son aquellos en los que las condiciones de encendido y apagado son determinadas por las corrientes y las tensiones del circuito en los que están colocados [4], esto significa que no existe manera de encender a estos elementos en forma externa. Esta clasificación de interruptores abarca todos los diodos de potencia existentes.

Las figuras 1.7(a) y 1.7(b) muestran el símbolo del diodo y su curva característica  $i-v$  en estado estable. Cuando el diodo está polarizado directamente comienza a conducir con sólo una pequeña caída en él, la cual es del orden de un voltio. Cuando el diodo es inversamente polarizado, sólo una pequeña corriente de fuga fluye a través del dispositivo hasta que la tensión de ruptura es alcanzada. En operación normal la tensión de polarización inversa no debe alcanzar el valor de ruptura.

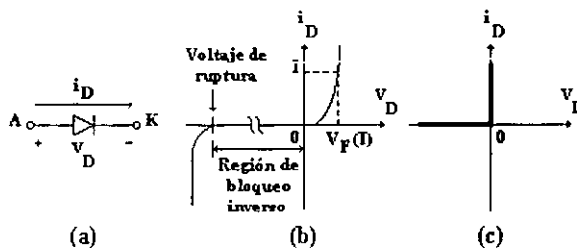


Figura 1.7. El diodo semiconductor de potencia. a) Símbolo. b) Curva característica real. c) Curva característica idealizada.

En vista de que la pequeña corriente de fuga en estado de bloqueo (polarización inversa) y la pequeña caída de tensión en el estado de conducción (polarización directa) son despreciables

en comparación con las corrientes y las tensiones del circuito en el cual el diodo es utilizado, la característica  $i-v$  puede ser idealizada como se muestra en la figura 1.7(c). Dicha característica puede ser usada para analizar la topología de los convertidores, pero no debe ser usada para el diseño real cuando por ejemplo los requerimientos de disipación de calor son calculados.

En la transición de encendido, el diodo puede ser considerado como un interruptor ideal porque enciende rápidamente comparado con los transitorios en el circuito de potencia. Sin embargo, en la transición de apagado existe una corriente inversa durante un tiempo de recuperación inversa  $t_{rr}$ , como es indicado en la figura 1.8, antes de caer a cero. Esto puede llevar a sobretensiones en circuitos inductivos. En muchos circuitos, esta corriente inversa no afecta las características del convertidor, así que el diodo puede ser también considerado ideal durante el transitorio de apagado [6].

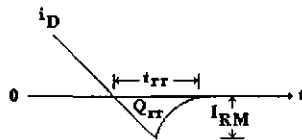


Figura 1.8. Tiempo de recuperación inversa en el diodo.

### 1.3.2. El SCR

Los dispositivos semicontrolados son interruptores electrónicos usados en circuitos electrónicos de potencia donde es necesario un control sobre el encendido del interruptor pero no del apagado [4]. En estos dispositivos es necesaria una señal externa de disparo para hacerlos conducir y su apagado depende totalmente de las condiciones de tensión y corriente del circuito del que forman parte. Estos elementos son una gran parte de la familia de los tiristores, como es el caso del rectificador controlado de silicio.

El símbolo para el SCR y su característica  $i-v$  son mostrados en la figura 1.9(a) y 1.9(b). La corriente principal fluye del ánodo ( $A$ ) al cátodo ( $K$ ). En su estado de apagado, el SCR puede bloquear una tensión directa y no conducir, como se muestra en la figura 1.9(b), para la porción del estado de apagado de la característica  $i-v$ .

El SCR puede ser llevado al estado de encendido por la aplicación de un pulso positivo de corriente de compuerta de corta duración. La relación  $i-v$  resultante es mostrada en la porción del estado de encendido de la característica de la figura 1.9(b). La caída de tensión en polarización directa es de sólo unos cuantos voltios, típicamente de 1 a 3 V dependiendo de la capacidad del dispositivo para bloquear tensión.

Una vez que el dispositivo ha empezado a conducir, permanece encendido y la corriente de compuerta puede ser retirada. El SCR no puede ser apagado por medio de la compuerta, y una vez encendido se comporta como un diodo. Sólo se apaga cuando la corriente de ánodo intenta hacerse negativa bajo la influencia del circuito en el cual el dispositivo está conectado y entonces la corriente se anula. Esto permite a la compuerta recuperar el control a fin de encender el dispositivo en algún momento después de que el SCR ha entrado otra vez a la región de bloqueo directo.

En polarización inversa, con tensiones inferiores al valor inverso de ruptura, sólo una pequeña corriente de fuga fluye en el SCR como se muestra en la figura 1.9(b). Generalmente las cantidades de tensión de bloqueo directo e inverso son las mismas. Usando los mismos argumentos que para los diodos, el SCR puede ser representado por las curvas características idealizadas mostradas en la figura 1.9(c) a fin de realizar el análisis en las topologías de convertidores [6].

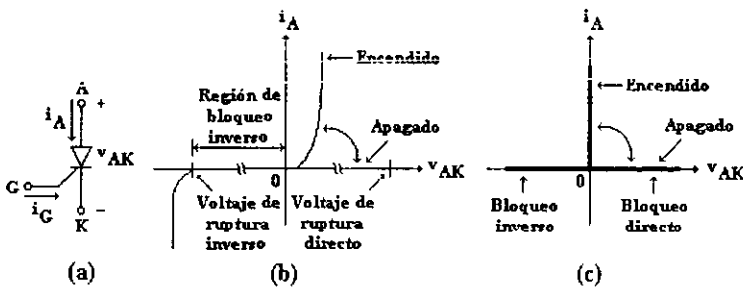


Figura 1.9. El rectificador controlado de silicio. a) Símbolo. b) Curva característica real. c) Curva característica idealizada.



En aplicaciones como en el circuito mostrado en la figura 1.10, el control puede ser ejercido en cualquier instante durante el semiciclo positivo de la señal de la fuente. Cuando la corriente del SCR intenta hacerse negativa debido a que la tensión de la fuente también lo hace, en el SCR ideal la corriente de ánodo se anula de inmediato. Sin embargo, como se precisa en las hojas de especificaciones y se muestra en la figura 1.10, la corriente del SCR se invierte a sí misma antes de anularse. El parámetro importante no es el tiempo que le toma a la corriente hacerse cero a partir de su valor negativo, en cambio lo es el intervalo de tiempo de apagado  $t_q$  definido en la figura 1.10 del cruce por cero de la corriente al cruce por cero de la tensión en el SCR. Durante  $t_q$  una tensión inversa debe ser mantenida a través del SCR y sólo después de este tiempo el SCR es capaz de bloquear una tensión directa sin caer dentro del estado de encendido. Si una tensión directa es aplicado al SCR antes de que este periodo de tiempo haya pasado, el dispositivo puede ser prematuramente encendido y dañado, así como también el circuito en el que está conectado. Las hojas de especificaciones proporcionan un tiempo  $t_q$  con una tensión inversa aplicada durante este intervalo, al igual que una razón de crecimiento de tensión más allá de este intervalo. Este intervalo  $t_q$  es algunas veces llamado el tiempo de recuperación por conmutación del SCR. El SCR está disponible con especificaciones de más de 1200 V y 2000 A y funcionan con frecuencias de hasta 2 kHz.

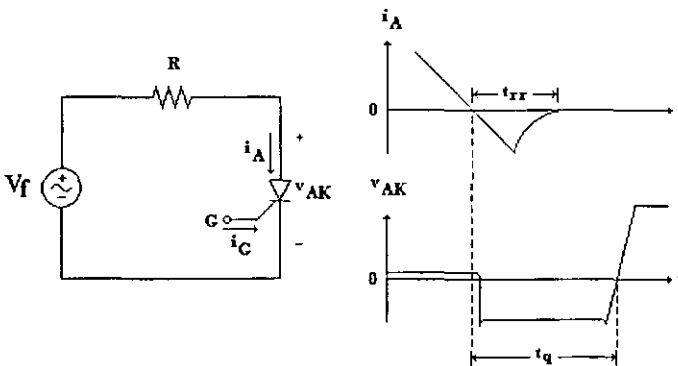


Figura 1.10. Tiempo de recuperación inversa y de recuperación por conmutación en un rectificador controlado de silicio.

1.3.3. El BJT

Existen diferentes tipos de dispositivos semiconductores de potencia que pueden ser encendidos o apagados por señales de tensión o corriente aplicadas a la terminal de control del dispositivo. Estos dispositivos denominados interruptores totalmente controlados son representados de manera general por el símbolo mostrado en la figura 1.11. No fluye corriente cuando el interruptor está apagado y cuando está encendido la corriente puede fluir únicamente en la dirección de la flecha. El BJT, el MOSFET y el IGBT son elementos que pertenecen a este grupo de dispositivos.

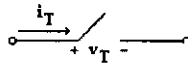


Figura 1.11. Interruptor totalmente controlado.

El símbolo para un transistor bipolar de unión tipo NPN es mostrado en la figura 1.12(a) y su características  $i-v$  en estado estable son mostradas en la figura 1.12(b).

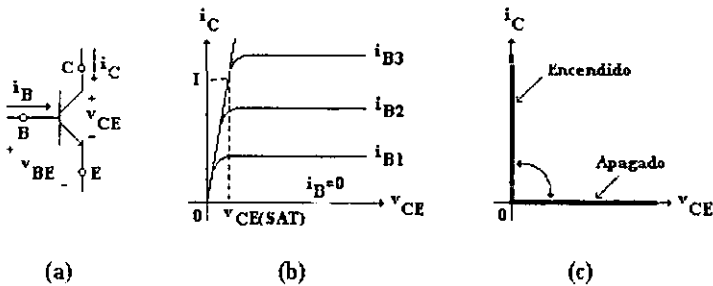


Figura 1.12. El BJT. a) Símbolo. b) Característica real. c) Característica ideal.

Como se muestra en las características  $i-v$ , una suficientemente grande corriente de base da como resultado el que el dispositivo esté totalmente encendido. Esto requiere que el circuito de control proporcione una corriente adecuada de base para que

$$I_B = \frac{I_C}{h_{FE}} \quad (1.1)$$

donde  $h_{FE}$  es la ganancia cd del dispositivo.

La tensión de encendido  $V_{CE(SAT)}$  de los transistores de potencia está usualmente en el rango de entre 1 y 2 voltios, así que la pérdida de potencia por conducción en los BJT es pequeña. Las características  $i-v$  idealizadas de la operación del BJT son mostradas en la figura 1.12(c).

Los BJT son dispositivos controlados por corriente y la corriente de base debe ser aplicada continuamente para mantenerlos encendidos [6]. La ganancia de corriente  $h_{FE}$  es usualmente de 5 a 10 en transistores de potencia así que estos dispositivos son conectados en configuración Darlington o triple Darlington, como se muestra en la figura 1.13, para lograr una ganancia de corriente más grande. Algunas desventajas de esta configuración son un valor más grande de  $V_{CE(SAT)}$  y más bajas velocidades de conmutación.

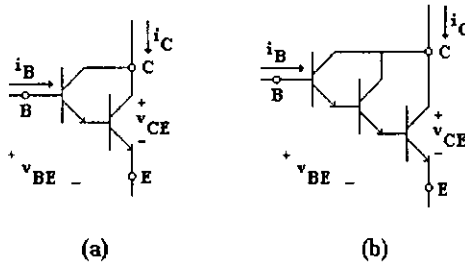


Figura 1.13. a) Configuración Darlington. b) Configuración triple Darlington.

Los BJT tanto en unidades simples como en configuración Darlington en un solo circuito integrado tienen significativos tiempos de almacenamiento durante la transición de apagado. Las frecuencias típicas de conmutación están en valores alrededor de los 10 kHz, por lo tanto los BJT son dispositivos más lentos que los MOSFET.

Los BJT disponibles pueden soportar arriba de 1200 V y corrientes de algunos cientos de amperios. A pesar de un coeficiente negativo de su resistencia de encendido con la temperatura, los modernos BJT fabricados con buen control de calidad pueden ser conectados en paralelo teniendo ciertos cuidados en el diseño del circuito como dar un cierto margen de corriente, esto es, donde teóricamente cuatro transistores conectados en paralelo serían suficientes

debido a una distribución equitativa de la corriente, cinco deben ser usados para tolerar una leve corriente de desbalance.

### 1.3.4. El MOSFET

El símbolo para un MOSFET de canal N es mostrado en la figura 1.14(a). Es un dispositivo controlado por tensión [6], como es indicado por las características  $i-v$  mostradas en la figura 1.14(b). El dispositivo está totalmente encendido y es aproximadamente un interruptor cerrado cuando la tensión de compuerta a fuente es suficientemente grande. El MOSFET está apagado cuando la tensión de compuerta a fuente está abajo de un valor de umbral  $V_{GS(TH)}$ . Las curvas características idealizadas del dispositivo operando como interruptor son mostradas en la figura 1.14(c).

Los MOSFET requieren de una aplicación continua de tensión entre compuerta y fuente de una amplitud adecuada a fin de estar en el estado de encendido. No fluye corriente de compuerta a fuente excepto durante la transición de encendido a apagado y viceversa cuando las capacitancias de compuerta a fuente y de compuerta a drenaje están siendo cargadas o descargadas, en cuyo caso las corrientes llegan a ser significativas sobre todo debido al efecto Miller de la capacitancia de compuerta a drenaje. Los tiempos de conmutación son muy cortos, estando en el rango de unas cuantas decenas de nanosegundos dependiendo del tipo de dispositivo y a costa de un incremento en el transitorio de corriente de compuerta.

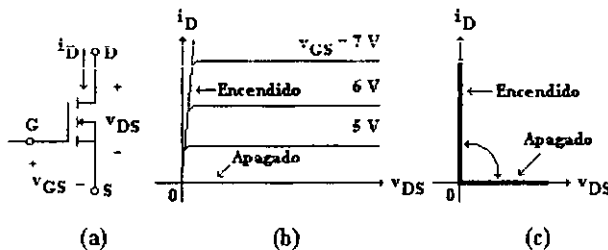


Figura 1.14. El MOSFET. a) Símbolo. b) Curva característica real. c) Curva característica ideal.

La resistencia de encendido  $r_{DS(ON)}$  del MOSFET entre el drenaje y la fuente se incrementa rápidamente con el rango de tensión de bloqueo del dispositivo. La resistencia de encendido como función de la tensión de bloqueo  $BV_{DSS}$  puede ser expresada como

$$r_{DS(OV)} = kB(V_{DSS})^{2.5a2.7} \quad (1.2)$$

donde  $k$  es una constante que depende de la geometría del dispositivo. A causa de esto, sólo dispositivos con pequeñas razones de tensión están disponibles cuando tienen baja resistencia de encendido y así pequeñas pérdidas por conducción. Desde el punto de vista de pérdidas totales, los MOSFET compiten con los BJT en el rango de 300 a 400 V sólo si la frecuencia de conmutación se encuentra entre 30 y 100 kHz o más. Sin embargo, nada definitivo se puede decir acerca del rango de frecuencia porque depende de las tensiones de operación, con bajas tensiones favoreciendo a los MOSFET.

Los MOSFET están disponibles en razones de tensión de más de 1000 V pero con corrientes pequeñas y con corrientes arriba de los 100 A pero con tensiones bajas. La máxima tensión de compuerta a fuente es de  $\pm 20$  V, aunque dispositivos que pueden ser controlados con señales de 5 V están siendo diseñados.

Los MOSFET son fácilmente conectados en paralelo porque su resistencia de encendido tiene un coeficiente positivo con la temperatura. Esto causa que la conducción de corriente a una alta temperatura sea equitativa con los otros MOSFET conectados en paralelo.

### 1.3.5. El IGBT

El símbolo para un IGBT es mostrado en la figura 1.15(a) y sus características  $v-i$  son mostradas en la figura 1.15(b). Los IGBT tienen algunas de las ventajas de los MOSFET y los BJT combinadas [6]. Similar al MOSFET, el IGBT tiene una gran impedancia de compuerta, con lo cual sólo requiere una pequeña cantidad de energía para conmutar al dispositivo, aunque debido a las capacitancias de entrada también tiene transitorios de corriente que no deben ser despreciados, sobre todo si se requieren tiempos de conmutación pequeños. Como el BJT, el IGBT tiene una pequeña caída de tensión de encendido incluso en dispositivos con grandes razones de bloqueo de tensión (por ejemplo  $V_{OV} = 3$  V en un dispositivo que soporta 1000 V).

Teóricamente, los IGBT pueden bloquear tensiones negativas como se muestra en la figura 1.15(c), aunque prácticamente tal capacidad de bloqueo se limita a unas cuantas decenas de voltios.

Los IGBT tienen frecuencias de funcionamiento del orden de los 20 kHz y están disponibles con valores tan grandes como 1200 V y 400 A. Dispositivos con valores de tensión arriba de 2000 V y corrientes de algunos cientos de amperios están siendo diseñados.

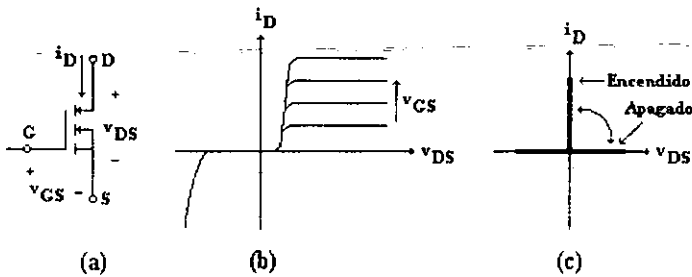


Figura 1.15. El IGBT. a) Símbolo. b) Curva característica real. c) Curva característica ideal.

---

# **CAPÍTULO 2**

---

**ANÁLISIS DE FOURIER**

## Introducción

Las funciones periódicas aparecen en una gran variedad de problemas físicos: las vibraciones de un resorte, el movimiento de los planetas alrededor del sol, las mareas y el movimiento de las olas.

En los circuitos eléctricos se presentan muchas señales periódicas; por ejemplo, la corriente alterna o la señal generada por un oscilador. Para este tipo de señales, que a menudo no son senoidales, el análisis con la serie trigonométrica de Fourier proporciona un método aproximado que facilita su estudio y permite determinar ciertas características y propiedades útiles para su uso. Una serie trigonométrica es una serie de la forma

$$\frac{1}{2} a_0 + a_1 \cos x + b_1 \operatorname{sen} x + \dots + a_n \cos nx + b_n \operatorname{sen} nx + \dots \quad (2.1)$$

donde los coeficientes  $a_n$  y  $b_n$  son constantes. Si estas constantes satisfacen ciertas condiciones que se especifican más adelante, entonces la serie se llama serie de Fourier.

### 2.1. La serie trigonométrica de Fourier

La serie de Fourier de una función  $f(t)$  definida en el intervalo  $-p < t < p$  está dada por

$$f(t) \approx \frac{a_0}{2} + \sum_{n=1}^{\infty} \left( a_n \cos \frac{n\pi}{p} t + b_n \operatorname{sen} \frac{n\pi}{p} t \right) \quad (2.2)$$

donde los coeficientes  $a_n$  y  $b_n$  están dados por

$$a_0 = \frac{1}{p} \int_{-p}^p f(t) dt \quad (2.3)$$

$$a_n = \frac{1}{p} \int_{-p}^p f(t) \cos \frac{n\pi}{p} t dt \quad (2.4)$$

$$b_n = \frac{1}{p} \int_{-p}^p f(t) \operatorname{sen} \frac{n\pi}{p} t dt \quad (2.5)$$



Existen dos simplificaciones que permiten una obtención más rápida de la representación de una función mediante la serie de Fourier. Para aplicar estas simplificaciones es necesario saber si la función que se quiere representar es par o impar.

Para que una función sea par debe cumplir con la condición impuesta por la ecuación 2.6. Por otro lado, para que la función sea impar debe cumplir con la condición impuesta por la ecuación 2.7.

$$f(-x) = f(x) \quad (2.6)$$

$$f(-x) = -f(x) \quad (2.7)$$

Cuando una función es par, esta se puede desarrollar como una serie de cosenos y cuando es impar se puede desarrollar como una serie de senos [13]. Cuando una función se desarrolla como serie de senos o de cosenos el resultado es una menor dificultad en los cálculos.

### 2.1.1. La serie de cosenos

La serie de Fourier de una función par en el intervalo  $-p < t < p$  es la serie de cosenos

$$f(t) \approx \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos \frac{n\pi}{p} t \quad (2.8)$$

donde el coeficiente  $a_n$  está dado por

$$a_0 = \frac{2}{p} \int_0^p f(t) dt \quad (2.9)$$

$$a_n = \frac{2}{p} \int_0^p f(t) \cos \frac{n\pi}{p} t dt \quad (2.10)$$

### 2.1.2. La serie de senos

La serie de Fourier de una función impar en el intervalo  $-p < t < p$  es la serie de senos

$$f(t) \approx \sum_{n=1}^{\infty} b_n \operatorname{sen} \frac{n\pi}{p} t \quad (2.11)$$

donde el coeficiente  $b_n$  está dado por

$$b_n = \frac{2}{p} \int_0^p f(t) \operatorname{sen} \frac{n\pi}{p} t dt \quad (2.12)$$

### 2.2. Señales periódicas unipolares y bipolares

El comportamiento de algunas señales periódicas permite que sean clasificadas en señales unipolares y señales bipolares de acuerdo a la forma en que tales señales oscilan entre sus valores extremos en cada semiciclo.

Una señal de tipo unipolar tiene oscilaciones que varían desde cero hasta un valor máximo positivo en un semiciclo y desde cero hasta un valor máximo negativo para el semiciclo restante. Una señal típica que ejemplifica este comportamiento se muestra en la figura 2.1, donde la señal cambia tres veces desde cero hasta  $-A$  para un semiciclo mientras que para el otro semiciclo varía tres veces desde cero hasta  $+A$ .

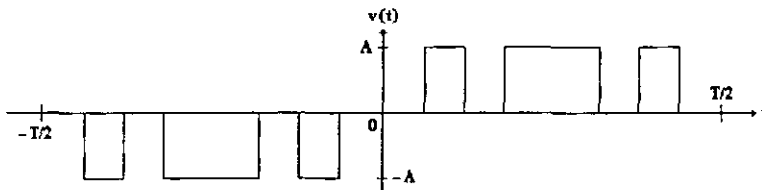


Figura 2.1. Señal unipolar.

Una señal bipolar, a diferencia de una señal unipolar, varía sus oscilaciones en cada semiciclo desde un valor máximo negativo hasta un valor máximo positivo, o bien, desde un valor máximo positivo hasta un valor máximo negativo. La figura 2.2 muestra una señal bipolar con tres pulsos por semiciclo variando desde  $-A$  hasta  $+A$  para el semiciclo a la izquierda y desde  $+A$  hasta  $-A$  para el semiciclo a la derecha.

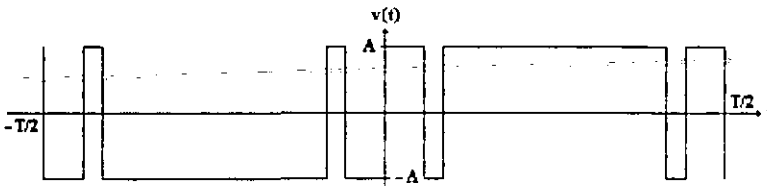


Figura 2.2. Señal bipolar.

### 2.3. Serie de Fourier de una señal unipolar con un solo pulso por semiciclo

La señal unipolar periódica más simple se presenta en la figura 2.3. Se trata de una señal que contiene un pulso por cada semiciclo. Esta función no es lineal, así que su descripción matemática se realiza por intervalos de la siguiente manera

$$f(t) = \begin{cases} A & \text{para } -\frac{T}{2} + \tau < t < -\tau \\ -A & \text{para } \tau < t < \frac{T}{2} - \tau \end{cases} \quad (2.13)$$

Esta señal es importante porque a partir de ella y por superposición es posible representar señales unipolares periódicas con pulsos múltiples por semiciclo.

Si se observa la figura 2.3 se puede determinar que la función es impar y que su representación corresponde a una serie de senos. De la ecuación 2.11, la ecuación 2.12, la función definida por intervalos en 2.13 y la figura 2.3 se obtienen las siguientes expresiones

$$f(t) \approx \sum_{n=1}^{\infty} b_n \operatorname{sen} \frac{2n\pi}{T} t$$

$$b_n = \frac{2}{p} \int_{\tau}^{\tau+T-\tau} A \operatorname{sen} \frac{2n\pi}{T} t dt$$

Al resolver la integral y evaluar los límites se llega a la serie de Fourier buscada.

$$f(t) \approx \sum_{n=1}^{\infty} b_n \operatorname{sen} \frac{2n\pi}{T} t \quad \text{para } n = 1, 3, 5, \dots \quad (2.14)$$

$$b_n = \frac{4A}{n\pi} \cos \frac{2n\pi}{T} \tau \quad \text{para } n = 1, 3, 5, \dots \quad (2.15)$$

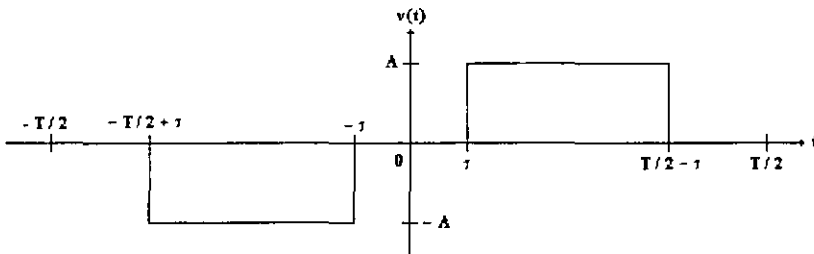


Figura 2.3. Señal unipolar periódica de un solo pulso por semiciclo.

#### 2.4. Serie de Fourier de una señal unipolar con varios pulsos por semiciclo

La figura 2.4(d) muestra una señal unipolar periódica con pulsos múltiples por semiciclo. Dicha figura puede ser representada como la suma de las series de Fourier de las figuras 2.4(a), 2.4(b) y 2.4(c). Con la figura 2.4 se hace evidente que el número de series de Fourier que han de ser sumadas corresponde al número de pulsos por semiciclo de la señal que se

quiere sintetizar. Para obtener la figura 2.4(d) se resta a la figura 2.4(a) la figura 2.4(b) para finalmente sumar la figura 2.4(c).

Las tres series de Fourier que han de sumarse para obtener la señal presentada en la figura 2.4(d) pueden ser expresadas en una sola serie de Fourier de la siguiente manera

$$f_d(t) \approx \sum_{n=1}^{\infty} B_n \operatorname{sen} \frac{2n\pi}{T} t \quad \text{para } n = 1, 3, 5, \dots$$

$$B_n = \sum_{m=1}^3 \frac{4A}{n\pi} (-1)^{m+1} \cos \frac{2n\pi}{T} \tau_m \quad \text{para } n = 1, 3, 5, \dots$$

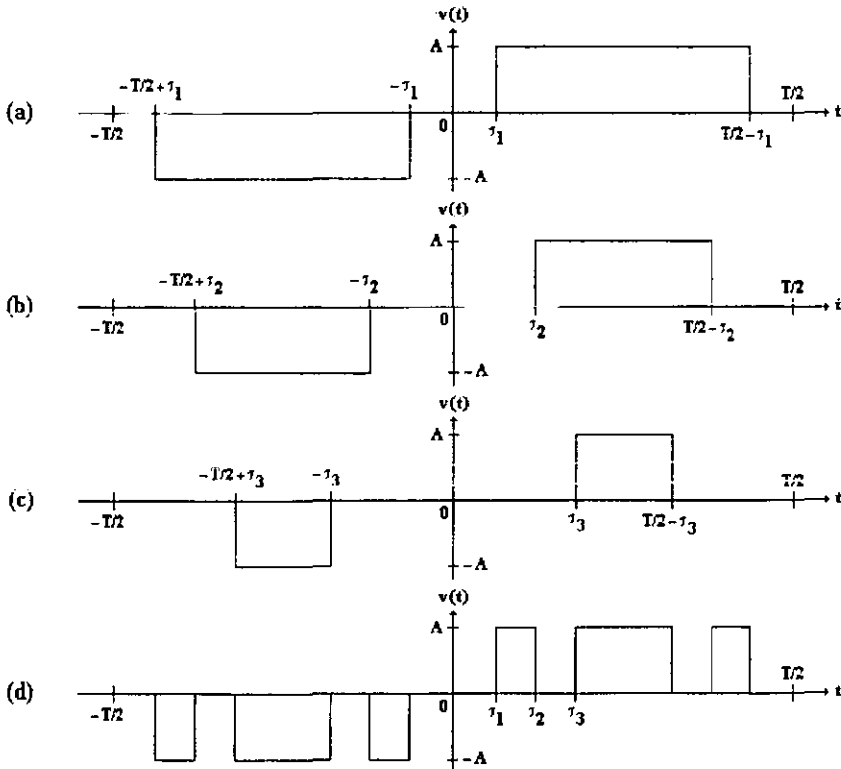


Figura 2.4. Señal unipolar periódica con pulsos múltiples por semiciclo. La señal final en d) se obtiene con la suma de a) con c) y la resta de b).

En forma general, cuando se desea expresar una señal  $f(t)$  del tipo presentado en la figura 2.4(d) con  $k$  pulsos por semiciclo se tienen las dos siguientes expresiones

$$f(t) \approx \sum_{n=1}^{\infty} B_n \operatorname{sen} \frac{2n\pi}{T} t \quad \text{para } n = 1, 3, 5, \dots \quad (2.16)$$

$$B_n = \sum_{m=1}^k \frac{4A}{n\pi} (-1)^{m+1} \cos \frac{2n\pi}{T} \tau_m \quad \text{para } n = 1, 3, 5, \dots \quad (2.17)$$

### 2.5. Serie de Fourier de una señal bipolar con varios pulsos por semiciclo

La figura 2.5(d) muestra una señal bipolar periódica con pulsos múltiples por semiciclo. Esta señal, al igual que la unipolar, puede ser representada como la suma de las series de Fourier de las figuras 2.5(a), 2.5(b) y 2.5(c). Para este tipo de señal el número de series de Fourier que han de ser sumadas corresponde también al número de pulsos por semiciclo de la señal que se quiere sintetizar. Para obtener la figura 2.5(d) se resta a la figura 2.5(a) la figura 2.5(b) para finalmente sumar la figura 2.5(c). Es importante remarcar que en esta ocasión la amplitud de las señales no es uniforme, ya que a fin de obtener la señal deseada es necesario que las señales de las figuras 2.5(b) y 2.5(c) tengan una amplitud dos veces mayor que la amplitud de la figura 2.5(a).

Agrupando las tres series de Fourier que describen a las figuras 2.5(a), 2.5(b) y 2.5(c) en una sola ecuación, la figura 2.5(d) pueden ser expresadas en una sola serie de Fourier con las siguientes ecuaciones

$$f_d(t) \approx \sum_{n=1}^{\infty} B_n \operatorname{sen} \frac{2n\pi}{T} t \quad \text{para } n = 1, 3, 5, \dots$$

$$B_n = \frac{4A}{n\pi} \left[ 1 + 2 \sum_{m=1}^k (-1)^m \cos \frac{2n\pi}{T} \tau_m \right] \quad \text{para } n = 1, 3, 5, \dots$$

En forma general, cuando se desea expresar una señal  $f(t)$  del tipo bipolar como la presentada en la figura 2.5(d) con  $k$  pulsos por semiciclo se tienen las dos siguientes expresiones:

$$f(t) \approx \sum_{n=1}^{\infty} B_n \operatorname{sen} \frac{2n\pi}{T} t \quad \text{para } n = 1, 3, 5, \dots \quad (2.18)$$

$$B_n = \frac{4A}{n\pi} \left[ 1 + 2 \sum_{m=1}^{k-1} (-1)^m \cos \frac{2n\pi}{T} \tau_m \right] \quad \text{para } n = 1, 3, 5, \dots \quad (2.19)$$

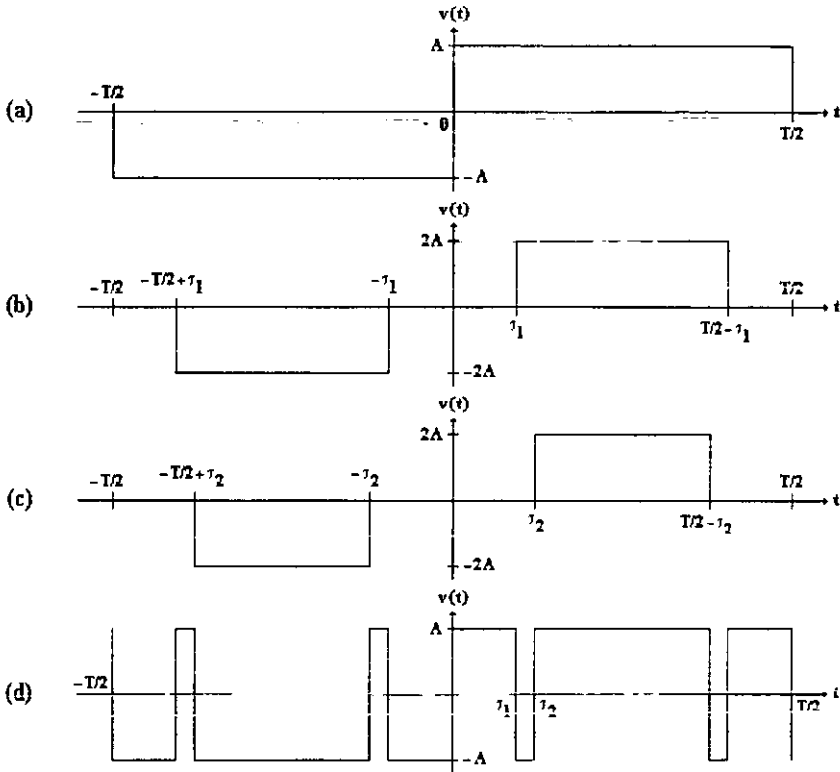


Figura 2.5. Señal bipolar periódica con pulsos múltiples por semiciclo. La señal final en d) se obtiene con la suma de a) con c) y la resta de b).

### 2.6. Parámetros de rendimiento basados en la serie de Fourier

La serie de Fourier expresa una señal periódica como una sumatoria de señales senoidales cuyas frecuencias son múltiplos de la denominada armónica fundamental. Tomando

en cuenta a dicha armónica y al resto de las señales, se tienen los siguientes parámetros de rendimiento [9].

1. Factor armónico de la  $n$ -ésima componente ( $HF_n$ ).
2. Distorsión armónica total ( $THD$ ).
3. Factor de distorsión de orden  $m$  ( $DF_m$ ).
4. Armónica de menor orden ( $LOH$ ).

### 2.6.1. Factor armónico de la $n$ -ésima componente ( $HF_n$ )

El factor armónico es una medida de la contribución armónica individual y se define como

$$HF_n = \frac{V_n}{V_1} \quad (2.20)$$

donde  $V_1$  es el valor rms de la componente fundamental y  $V_n$  es el valor rms de la  $n$ -ésima componente armónica.

### 2.6.2. Distorsión armónica total ( $THD$ )

La distorsión armónica total es una medida de la similitud entre la forma de onda y su componente fundamental, se define como

$$THD = \frac{1}{V_1} \left( \sum_{n=2}^{\infty} V_n^2 \right)^{\frac{1}{2}} \quad (2.21)$$

### 2.6.3. Factor de distorsión de orden $m$ ( $DF_m$ )

Este parámetro deriva directamente del parámetro  $THD$ . Si la señal que se desea evaluar se pasa por un filtro, las armónicas de orden más alto se atenuarán con mayor eficacia. El factor de distorsión es un indicador de la cantidad de distorsión armónica total que queda en una forma de onda particular después de que las armónicas de esa forma de onda hayan sido sujetas a una



atenuación de orden  $m$ , es decir, divididas por  $n^m$ . Por lo tanto, el valor  $DF_m$  es una medida de la eficacia en la reducción de las componentes armónicas no deseadas, sin necesidad de especificar los valores del filtro utilizado, y se define como

$$DF_m = \frac{1}{V_1} \left[ \sum_{n=2}^{\infty} \left( \frac{V_n}{n^m} \right)^2 \right]^{\frac{1}{2}} \quad (2.22)$$

El factor de distorsión de orden  $m$  de una componente armónica individual se obtiene a partir de la expresión anterior como

$$DF_{nm} = \frac{V_n}{V_1 n^m} \quad (2.23)$$

Es importante aclarar que los factores de distorsión como indicadores de contenido armónico después del filtrado sólo proporcionan datos numéricos proporcionales a los valores reales. Por ejemplo, si se tienen señales unipolares diferentes pero con un número de pulsos por semiciclo igual en todas ellas, y se obtiene los factores de distorsión de primer orden, los resultados numéricos no serán iguales a los que se obtendrían si se sometieran realmente dichas señales a un filtro de primer orden. Sin embargo, se podría encontrar una constante  $K$  (válida para todas las señales) que al ser multiplicada por los factores de distorsión calculados diera en forma aproximada los valores reales de distorsión armónica total. De esta manera, los factores de distorsión son útiles para comparar características armónicas entre señales similares.

#### 2.6.4. Armónica de menor orden (LOH)

La armónica de menor orden es aquella componente cuya frecuencia es la más cercana a la fundamental, y cuya amplitud es mayor que o igual al 3% de la componente fundamental.

---

# **CAPÍTULO 3**

---

**CONFIGURACIONES DE CONVERTIDORES CD-CA**

## Introducción

En los convertidores cd-ca existen tres configuraciones que son clásicamente las más utilizadas, estas son la push-pull, de medio puente y de puente completo. También existen tres formas de alimentar al inversor, se puede alimentar con corriente, con una tensión fija o con una tensión variable, pero independientemente del tipo de topología y de alimentación la finalidad es la misma, el arreglo debe permitir el flujo de corriente y la inversión de su sentido a través de una carga en forma periódica y estable.

### 3.1. Convertidores cd-ca

Como ya se mencionó, la función de un inversor es cambiar una tensión de entrada de cd a una tensión de salida de ca. Esta tensión de salida debe cumplir con el requisito de ser simétrica con una magnitud y frecuencia predeterminadas. Tanto la tensión de salida como la frecuencia pueden ser fijos o variables. Si se modifica la tensión de entrada de cd y la ganancia del inversor se mantiene constante, es posible obtener una tensión de salida variable. Por otra parte, se puede obtener una tensión de salida variable si se varía la ganancia del inversor y se mantiene constante la tensión de cd de entrada. La ganancia del inversor se puede definir como la relación de la tensión de salida de ca a la tensión de entrada de cd.

En los inversores ideales la forma de onda de salida debería ser senoidal, sin embargo, en los inversores reales muchas veces no es senoidal y para aplicaciones de baja y mediana potencia se pueden aceptar las señales con forma de onda cuadrada. Para aplicaciones de alta potencia es necesaria la forma de onda senoidal de baja distorsión [9]. Dada la disponibilidad de los dispositivos semiconductores de alta velocidad, es posible reducir la deformación de la tensión de salida mediante algunas técnicas de conmutación.

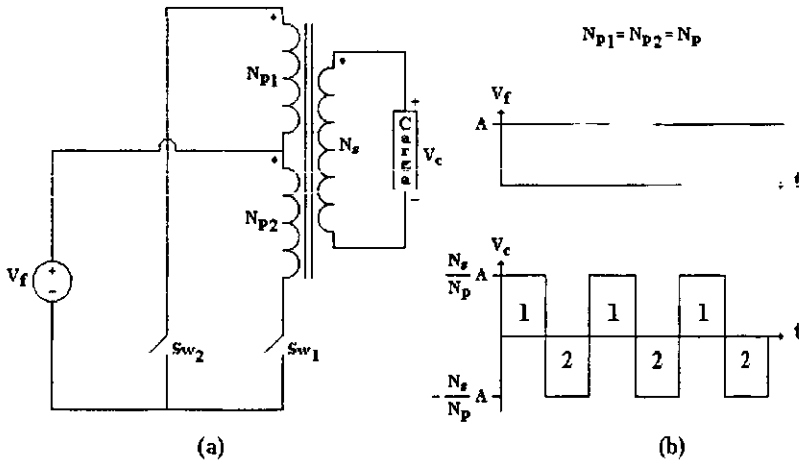
Los inversores se clasifican básicamente en monofásicos y trifásicos. Un inversor es monofásico cuando su salida está constituida por una única señal de ca. Un inversor es trifásico cuando su salida consta de tres señales de ca de la misma magnitud y frecuencia pero desplazadas en el tiempo, de manera equitativa, una con respecto a otra. En general, un inversor trifásico se construye a partir de tres inversores monofásicos.

Se debe tener también en consideración la forma en que el inversor es alimentado. El inversor se llama *inversor alimentado por tensión* si la tensión de entrada se conserva constante; *inversor alimentado por corriente* si es la corriente de entrada quien se conserva constante; e *inversor enlazado en cd variable* si el nivel de entrada es controlable [9].

El uso de inversores es muy común en aplicaciones industriales tales como la propulsión de motores de ca a velocidad variable, las fuentes de energía eléctrica de respaldo y de poder, así como en las alimentaciones ininterrumpibles de potencia. La entrada puede ser una batería, una celda de combustible, una celda solar u otra fuente de cd.

### 3.2. Configuración Push-Pull

La topología push-pull precisa de un transformador para su funcionamiento y tiene la ventaja de proporcionar aislamiento entre la carga y la fuente de potencia. Esta configuración utiliza dos interruptores de potencia y una sola fuente de poder. La figura 3.1 muestra la estructura de este inversor.



Para esta topología la ganancia depende principalmente del transformador. Considerando un modelo ideal para el transformador y una relación de transformación  $N_s/N_p$  se realiza el análisis mostrado a continuación.

Cerrando el interruptor  $S_{w1}$  durante un tiempo  $T/2$ , que corresponde a los semiciclos marcados con un 1, se establece una tensión en el primario  $P_1$  que es igual a la tensión  $A$  de la fuente  $V_f$

$$v_{p1} = A \quad (3.1)$$

La tensión a través de  $P_1$  se refleja en los otros dos embobinados, lo que da como resultado que

$$v_s = v_c = A \frac{N_s}{N_p} \quad (3.2)$$

$$v_{p2} = A \quad (3.3)$$

$$v_{S_{w2}} = 2A \quad (3.4)$$

Al cerrar el interruptor  $S_{w2}$  un tiempo  $T/2$ , correspondiente a los semiciclos marcados con un 2, se establece una tensión en el primario  $P_2$  que es igual a la tensión  $A$  de la fuente  $V_f$

$$v_{p2} = -A \quad (3.5)$$

La tensión a través de  $P_2$  se refleja en los otros dos embobinados, lo que da como resultado que

$$v_s = v_c = -A \frac{N_s}{N_p} \quad (3.6)$$

$$v_{p1} = -A \quad (3.7)$$

$$v_{S_{w1}} = 2A \quad (3.8)$$

Cuando los dos interruptores están abiertos la corriente en ambos primarios es cero, al igual que en el secundario. Los dos interruptores nunca están cerrados al mismo tiempo.

El valor medio cuadrático, o rms, de la tensión de salida se puede determinar a partir de

$$V_c = \left( \frac{2}{T} \int_0^T \frac{N_s}{N_p} A^2 dt \right)^{1/2} = \frac{N_s}{N_p} A \quad (3.9)$$

La tensión instantánea de salida se puede expresar por medio de una serie de Fourier como

$$v_c = \sum_{n=1,2,3,\dots}^{\infty} \frac{N_s}{N_p} \frac{4A}{n\pi} \text{sen } n\omega t \quad (3.10)$$

Para  $n = 1$ , la ecuación anterior proporciona el valor rms de la componente fundamental

$$V_1 = \frac{N_s}{N_p} \frac{4A}{2\pi} = 0.90 \frac{N_s}{N_p} A \quad (3.11)$$

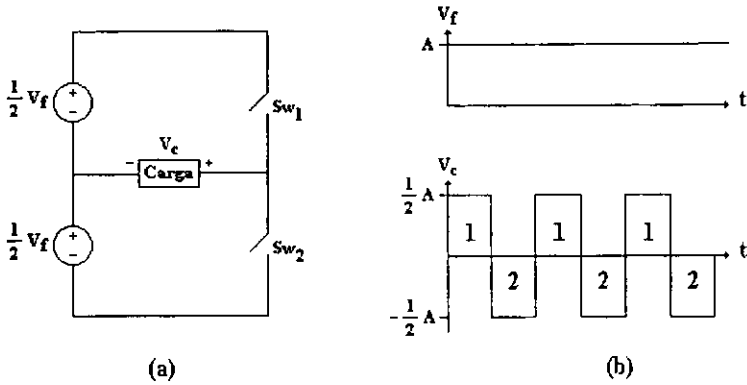
### 3.3. Configuración de medio puente

Mediante la figura 3.2 se puede explicar el principio de funcionamiento del inversor monofásico de medio puente. El circuito inversor está formado por dos interruptores y una fuente de cd de tres terminales y, aunque no es necesario, puede utilizarse con transformador.

El funcionamiento del circuito es el siguiente [9]. Sólo cuando el interruptor  $Sw_1$  está activo durante el tiempo  $T/2$ , correspondiente a los semiciclos marcados con un 1, las tensiones instantáneas a través de la carga y en el interruptor  $Sw_2$  son

$$v_i = \frac{1}{2} A \quad (3.12)$$

$$v_{Sw_2} = A \quad (3.13)$$



**Figura 3.2. Inversor de medio puente. a) Topología de medio puente. b) Señales de entrada y salida.**

Si sólo el interruptor  $Sw_2$  está activo durante un tiempo  $T/2$ , que corresponde a los semiciclos marcados con un 2, las tensiones instantáneas que aparecen en la carga y en el interruptor son

$$v_c = -\frac{1}{2}A \quad (3.14)$$

$$v_{Sw_2} = A \quad (3.15)$$

El circuito de control debe diseñarse de tal forma que  $Sw_1$  y  $Sw_2$  no estén activos simultáneamente. Como se vio, este inversor requiere de una fuente de cd de tres terminales y cuando un interruptor está inactivo, su tensión inversa es de  $A$ , en vez de  $A/2$ .

El valor medio cuadrático de la tensión en la salida del inversor se puede encontrar a partir de

$$V_c = \left( \frac{2}{T} \int_0^T \frac{A^2}{4} dt \right)^{1/2} = \frac{A}{2} \quad (3.16)$$

La tensión instantánea se puede expresar con el desarrollo de una serie de Fourier como

$$v_c = \sum_{n=1,3,5,\dots}^{\infty} \frac{2A}{n\pi} \text{sen } n\omega t \quad (3.17)$$

Para  $n = 1$  la ecuación anterior proporciona el valor rms de la componente fundamental como

$$V_1 = \frac{2A}{2\pi} = 0.45A \quad (3.18)$$

### 3.4. Configuración de puente completo

Un inversor de puente completo aparece en la figura 3.3. Está formado por cuatro interruptores y una fuente de cd. Con este tipo de sistema, como con la configuración de medio puente, se puede o no utilizar un transformador.

El funcionamiento del circuito es el siguiente [9]. Cuando los interruptores  $Sw_1$  y  $Sw_2$  se activan simultáneamente un tiempo  $T/2$ , semiciclos marcados con un 1, la tensión  $A$  de la fuente  $V_f$  aparece a través de la carga al igual que en los interruptores que no están activados. Es decir

$$v_c = A \quad (3.19)$$

$$v_{Sw_1} = A \quad (3.20)$$

$$v_{Sw_4} = A \quad (3.21)$$

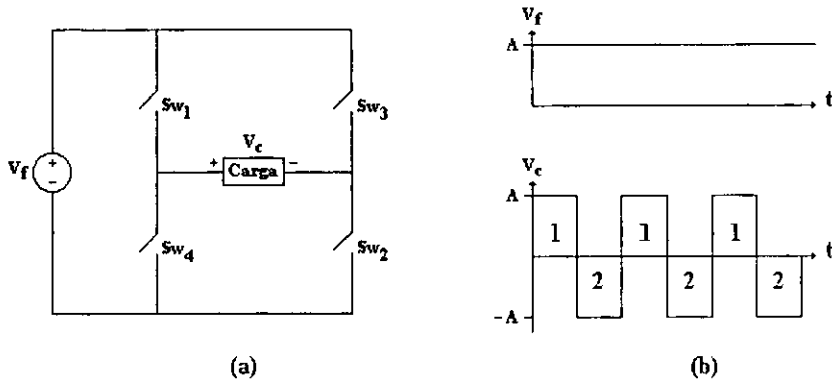
Si los interruptores  $Sw_3$  y  $Sw_4$  se activan al mismo tiempo, la tensión a través de la carga se invierte, y adquiere el valor  $-A$ , por otro lado, los interruptores no activados tienen que soportar la tensión de la fuente, esto es

$$v_c = -A \quad (3.22)$$

$$v_{Sw_1} = A \quad (3.23)$$

$$v_{Sw_2} = A \quad (3.24)$$





**Figura 3.3. Inversor de puente completo. a) Topología de puente completo. b) Señales de entrada y de salida.**

La tensión rms de salida se puede determinar a partir de

$$V_c = \left( \frac{2}{T} \int_0^T A^2 dt \right)^{1/2} = A \quad (3.25)$$

La tensión instantánea de salida se puede expresar por medio de una serie de Fourier como

$$v_c = \sum_{n=1,2,3,\dots}^{\infty} \frac{4A}{n\pi} \sin n\omega t \quad (3.26)$$

Para  $n = 1$ , la ecuación anterior proporciona el valor rms de la componente fundamental

$$V_1 = \frac{4A}{2\pi} = 0.90A \quad (3.27)$$

El hecho de que se remarque la importancia de la componente fundamental se debe a que en la mayor parte de las aplicaciones (por ejemplo, los propulsores de motores eléctricos) la potencia de salida debida a tal componente es la potencia útil y la potencia debida a las corrientes armónicas es disipada en forma de calor aumentando la temperatura de la carga.

### 3.5. Inversores alimentados con fuente de tensión

Las tres configuraciones mostradas hasta el momento corresponden a circuitos inversores alimentados con fuente de tensión. Los análisis fueron hechos con base en elementos ideales donde los interruptores conmutan de manera inmediata y soportan tanto tensiones como corrientes de cualquier magnitud y se supone que la corriente y la tensión están en fase. Cuando se utilizan interruptores reales, como los transistores, con cargas inductivas es necesario modificar las señales de conmutación y adicionar elementos semiconductores para permitir el flujo de corriente cuando tensión y corriente están fuera de fase [9].

La figura 3.4 muestra un inversor de potencia de puente completo con cuatro transistores y cuatro diodos. Cuando los transistores  $Q_1$  y  $Q_2$  se activan al mismo tiempo, la tensión de alimentación  $A$  aparece en la carga. Si los transistores  $Q_3$  y  $Q_4$  se activan simultáneamente, la tensión en la carga se invierte, adquiriendo el valor  $-A$ . El funcionamiento del sistema es casi igual al del circuito presentado en la figura 3.3, excepto que en este último inversor se ha añadido un tiempo de retardo  $\tau$  que sirve tanto para controlar la tensión de salida como para permitir que los transistores puedan conmutar de forma segura. También se han añadido cuatro diodos pues para una carga inductiva la corriente no puede cambiar inmediatamente con la tensión de salida. Para este tipo de carga, cuando  $Q_1$  y  $Q_2$  son desactivados en  $t = T/2$  la corriente de carga seguirá fluyendo a través de  $D_3$  y  $D_4$ . Lo mismo sucede cuando los transistores  $Q_3$  y  $Q_4$  dejan de conducir en  $t = T$ , la corriente de carga continúa fluyendo a través de los diodos  $D_1$  y  $D_2$ . Cuando los diodos conducen, la energía es retroalimentada a la fuente de cd por lo que estos elementos se conocen como diodos de retroalimentación.

Como se mencionó anteriormente, se ha añadido un tiempo de retardo en las señales de conmutación. Al añadir este retardo se modifica la señal de salida lo que a su vez modifica la amplitud de la componente fundamental, como se puede demostrar al observar el desarrollo en serie de Fourier de la tensión de salida en las ecuaciones 3.28 y 3.29.

$$f(t) \approx \sum_{n=1}^{\infty} b_n \sin \frac{2n\pi}{T} t \quad \text{para } n = 1, 3, 5, \dots \quad (3.28)$$

$$b_n = \frac{4A}{n\pi} \cos \frac{2n\pi}{T} \tau \quad \text{para } n = 1, 3, 5, \dots \quad (3.29)$$

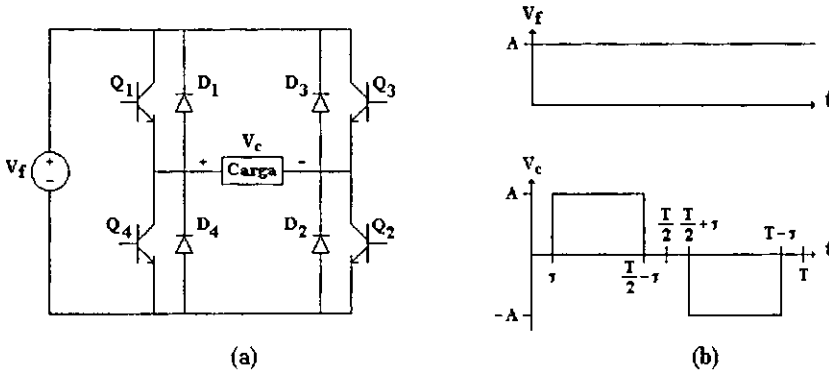


Figura 3.4. Inversor transistorizado con fuente de voltaje. a) Configuración de puente completo con fuente de voltaje. b) Voltajes de entrada y de salida.

Sustituyendo  $n = 1$  en la ecuación 3.29 se obtiene el valor de la componente fundamental de la señal de salida, el cual depende en forma contraria al tiempo de retardo, como se ve en la ecuación 3.30.

$$b_n = \frac{4A}{\pi} \cos \frac{2\pi}{T} \tau \quad (3.30)$$

Se ha mencionado que el tiempo de retardo permite que los transistores conmuten de forma segura, esto se debe a que los interruptores reales necesitan de algún tiempo para pasar del estado de encendido al de apagado y viceversa. A menudo, cuando el tiempo de retardo es utilizado para permitir la conmutación de los interruptores recibe el nombre de tiempo muerto.

### 3.6. Inversores con enlace en cd variable

La salida de un inversor se puede controlar variando los tiempos de retardo, o lo que es lo mismo, el ancho de los pulsos, pero para este tipo de inversor estarían presentes algunos armónicos no deseados. El ancho de los pulsos se puede mantener fijo para eliminar o reducir algunos armónicos y la tensión de salida se puede controlar variando el nivel de tensión de

entrada en cd [9]. Una disposición como esta aparece en la figura 3.5 y se conoce como inversor de enlace en cd variable. Esta configuración requiere de un paso adicional del convertidor y se trata de un convertidor de cd-cd, además la potencia no puede ser retroalimentada a la fuente de cd.

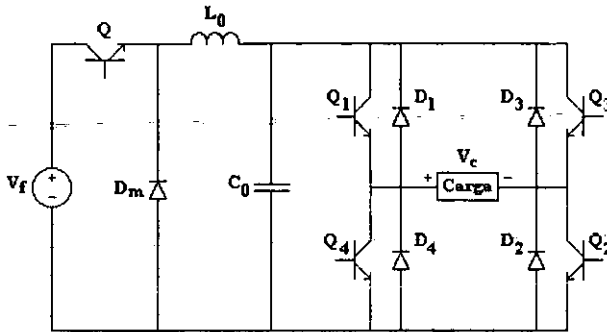


Figura 3.5. Inversor con enlace en cd variable.

A fin de obtener la calidad y las armónicas de salida deseadas, puede determinarse matemáticamente la forma de la tensión de salida. La alimentación de cd es variable con el objeto de proporcionar una salida controlable en ca.

### 3.7. Inversores alimentados con fuente de corriente

En los inversores alimentados con tensión se obliga a la corriente a fluctuar de positiva a negativa y posteriormente a la inversa. A fin de hacer frente a las cargas inductivas se necesitan interruptores de potencia con diodos en antiparalelo, en tanto que en un inversor de corriente la entrada se comporta como su nombre lo indica. La corriente de salida se mantiene constante independientemente de la carga del inversor, y la tensión de salida es obligada a cambiar. El diagrama del circuito inversor monofásico transistorizado aparece en la figura 3.6. Dado que debe haber un flujo continuo de corriente desde la alimentación, siempre deberán conducir dos interruptores, uno de la parte superior y otro de la parte inferior. La secuencia de conducción es 12, 23, 34 y 41, como se muestra en la figura 3.7. Se requieren diodos en serie con los transistores para bloquear tensiones inversas en ellos.

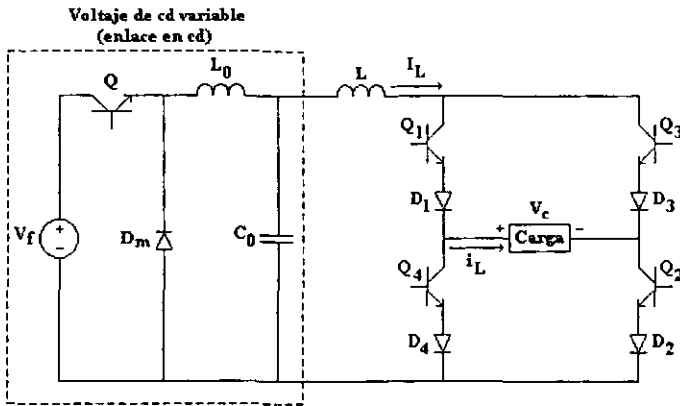


Figura 3.6. Inversor alimentado con fuente de corriente.

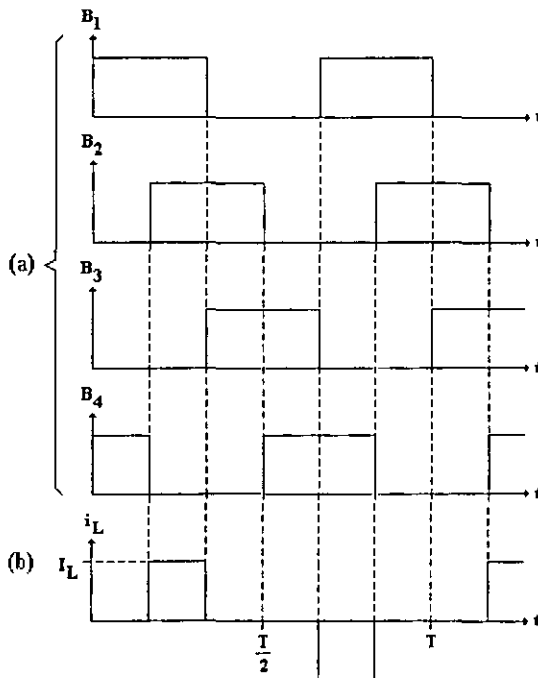


Figura 3.7. Señales en un inversor alimentado con corriente.  
a) Señales de base. b) Corriente en la carga.

Cuando conducen dos dispositivos en brazos diferentes, la corriente de alimentación fluye a través de la carga. Cuando conducen dos dispositivos en el mismo brazo, la corriente de la fuente se desvía de la carga.

Un inversor de fuente de corriente [9] es la contraparte de un inversor de fuente de tensión. En un inversor de fuente de tensión la forma de onda de tensión es similar a la corriente de línea de un inversor de fuente de corriente. El inversor de fuente de corriente tiene la ventaja de que un corto circuito o un disparo equivocado en los interruptores no provocan problemas serios pues la corriente está controlada.

Esta clase de inversor requiere de un inductor relativamente grande para exhibir las características de una fuente de corriente. Debido a la transferencia de corriente de un par de interruptores a otro, se requiere de un filtro de entrada para suprimir las crestas en la salida.

### 3.8. Inversores trifásicos

Los inversores trifásicos se utilizan normalmente en aplicaciones de alta potencia [9]. Tres inversores monofásicos (push-pull, de medio puente o de puente completo) pueden utilizarse para formar una configuración trifásica.

Los inversores de tipo push-pull y de puente completo pueden conectarse en paralelo con transformadores, tal como se muestra en la figura 3.8. Las señales de disparo de cada inversor monofásico deben adelantarse o retrasarse  $180^\circ$  una con respecto a las otras, a fin de obtener tensiones trifásicas balanceadas. Los enbobinados primarios del transformador deben aislarse unos con respecto a otros, en tanto que los secundarios pueden quedar conectados en estrella o en delta. Por lo general, los secundarios se conectan en estrella a fin de eliminar armónicos múltiplos de 3 que aparecen en las tensiones de salida.

El diagrama de la conexión en estrella que representa a los secundarios de la figura 3.8 conjuntamente con una carga, también en estrella, se muestra en la figura 3.9.

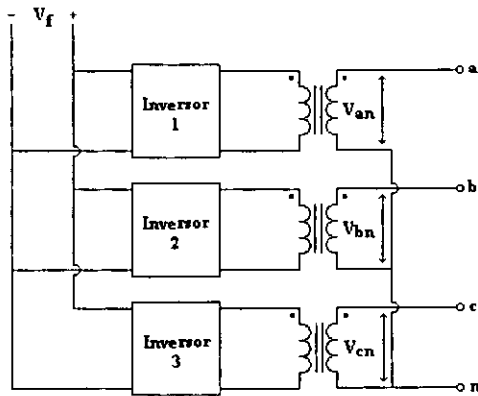


Figura 3.8. Inversor de potencia trifásico formado por tres inversores monofásicos push-pull o de puente completo.

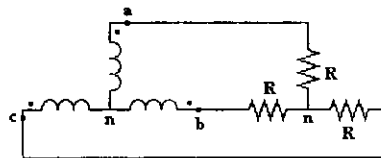


Figura 3.9. Diagrama de una conexión en estrella en un inversor trifásico con transformadores.

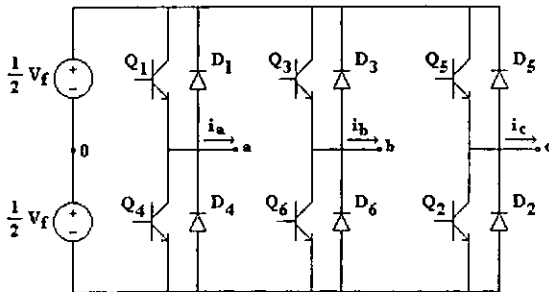


Figura 3.10. Inversor de potencia trifásico formado por tres inversores monofásicos de medio puente.

Tres inversores monofásicos de medio puente se pueden conectar como se muestra en la figura 3.10 para formar un inversor trifásico. La configuración consta de seis transistores y seis diodos. La configuración de medio puente es la que más a menudo se utiliza para implementar inversores trifásicos y su tipo de carga generalmente está conectada en estrella, lo que mantiene sus señales de salida libres del tercer armónico y sus múltiplos.



---

# **CAPÍTULO 4**

---

**TECNICAS DE MODULACIÓN EN CONVERTIDORES CD-CA**

## Introducción

La modulación por ancho de pulso (PWM) tiene como finalidad controlar la amplitud de la salida así como el contenido armónico en los inversores de potencia. Los primeros inversores tuvieron patrones de conmutación simples, pero a medida que los interruptores de estado sólido fueron mejorando, las técnicas de modulación utilizadas se hicieron más complejas ya que se favoreció la implementación de la teoría desarrollada. Actualmente son más de diez las técnicas de modulación que existen, algunas de las cuales serán descritas en este capítulo.

### 4.1. Consideraciones relacionadas con la modulación de ancho de pulso

En este punto, es conveniente hacer algunas consideraciones y definir algunos términos que son comunes en la modulación por ancho de pulso [4] antes de revisar las distintas técnicas de modulación.

La modulación por ancho de pulso proporciona una forma de reducir la distorsión armónica total en una carga. Un inversor con un esquema PWM, con algún tipo de filtrado, puede alcanzar más fácilmente los requisitos de THD que un inversor de onda cuadrada. La señal PWM sin filtrar comúnmente tendrá un contenido armónico más grande que la señal de un inversor de onda cuadrada, pero los armónicos estarán a frecuencias mayores, haciendo más fácil su filtrado.

Para generar las señales de conmutación son necesarias dos formas de onda. La primera es una señal de referencia, algunas veces llamada señal moduladora o señal de control, la segunda es la señal portadora que es una señal triangular que controla la frecuencia de conmutación de los interruptores.

La serie de Fourier de una salida PWM tiene una frecuencia fundamental la cual es la misma que la de la señal de referencia. Existen frecuencias armónicas en y alrededor de los múltiplos de la frecuencia de conmutación. Las magnitudes de algunos armónicos son bastante grandes, algunas veces más grandes que el fundamental. Sin embargo, debido a que estos armónicos están localizados en altas frecuencias, un simple filtro de paso bajo puede ser efectivo para eliminarlos. Incrementando la frecuencia de la portadora se incrementa el valor de las frecuencias a las cuales los armónicos ocurren, lamentablemente la desventaja de usar una

frecuencia de conmutación muy alta es que se incrementan las pérdidas en los interruptores utilizados para implementar el inversor.

Las técnicas de modulación se evalúan por su contenido armónico, específicamente por los factores de distorsión de primer y segundo orden. La evaluación se realiza de esta manera porque las cargas a menudo o son motores, que por su impedancia inductiva se comportan como filtros de primer orden, o se conectan a los inversores mediante filtros de segundo orden para atenuar más el contenido armónico.

El índice de modulación en frecuencia  $m_f$  es definido en la ecuación 4.1 como la razón de la frecuencia de la portadora a la frecuencia de la señal de referencia

$$m_f = \frac{f_{portadora}}{f_{referencia}} = \frac{f_c}{f_r} \quad (4.1)$$

El índice de modulación en amplitud  $m_a$  está definido por la ecuación 4.2 como la razón de la amplitud de la señal de referencia a la amplitud de la señal portadora. La amplitud del armónico fundamental  $V_1$  es proporcional a  $m_a$ . Así, la amplitud de la componente fundamental de la salida PWM es controlada por  $m_a$ . Esto es muy importante en el caso de una fuente de tensión de cd no regulada alimentando al inversor, debido a que el valor de  $m_a$  puede ser ajustado para compensar las variaciones de la fuente de alimentación, produciendo una salida de amplitud constante. Alternativamente  $m_a$  puede ser usada para variar la amplitud de la salida.

$$m_a = \frac{A_{referencia}}{A_{portadora}} = \frac{A_r}{A_c} \quad (4.2)$$

## 4.2. Modulación de un solo ancho de pulso

En el control por modulación de un solo ancho de pulso [9], existe un solo pulso por cada medio ciclo, el ancho del pulso se hace variar a fin de controlar la tensión de salida del inversor. La figura 4.1(a) muestra la generación de las señales de excitación y la figura 4.1(b) la tensión de salida para un inversor monofásico.

Las señales de excitación se generan al comparar una señal rectangular de referencia de amplitud  $A_r$  con una onda portadora de amplitud  $A_c$ . La frecuencia de la señal de referencia determina la frecuencia fundamental de salida. Si se varía  $A_r$  desde 0 hasta  $A_c$ , el ancho de pulso  $\delta$  puede ser modificado desde 0 hasta  $T/2$ .

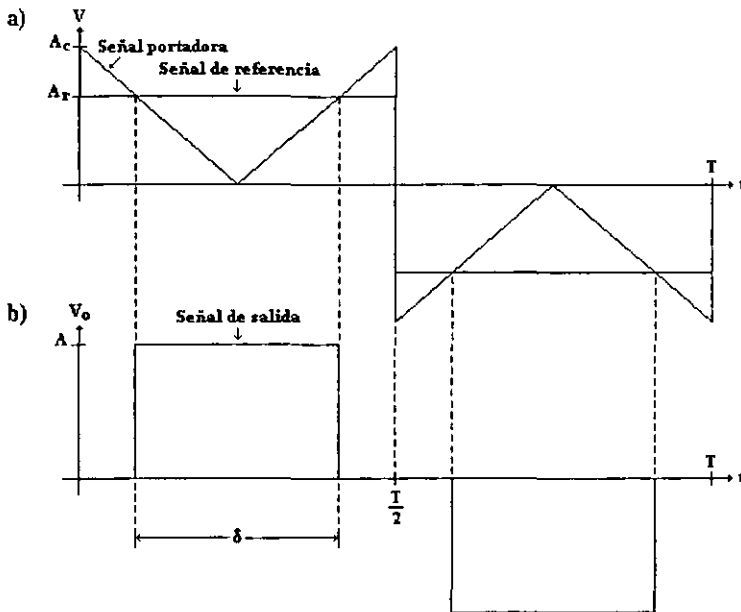


Figura 4.1. Modulación de un solo ancho de pulso. a) Señal de referencia y señal portadora. b) Señal de salida.

La figura 4.2 muestra el perfil de armónicas con la variación del índice de modulación  $m_a$ . La armónica dominante es la tercera, y el factor de distorsión aumenta en forma significativa a un bajo nivel de salida. La figura 4.3 representa la distorsión armónica total y la figura 4.4 los factores de distorsión de primer y segundo orden.

Este tipo de técnica es la más simple que existe y entre las ventajas con las que cuenta están la sencillez y pocas pérdidas por conmutación en los elementos interruptores. Entre sus desventajas se encuentran un gran contenido armónico y la no linealidad de la componente fundamental con respecto al índice de modulación. Los sistemas inversores baratos generalmente hacen uso de esta técnica.

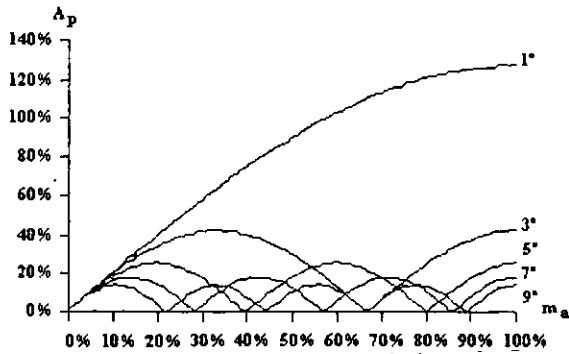


Figura 4.2. Modulación de un ancho de pulso. Perfil armónico.

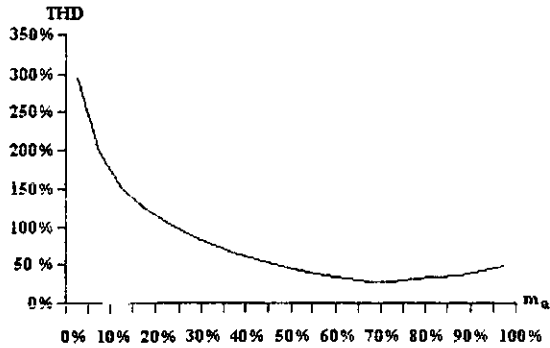


Figura 4.3. Modulación de un ancho de pulso. THD.

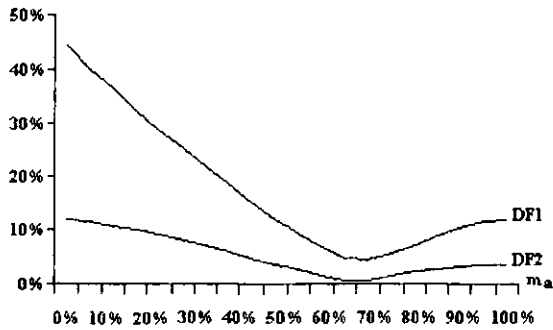


Figura 4.4. Modulación de un ancho de pulso. DF1 y DF2.

### 4.3. Modulación uniforme de ancho de pulso (UPWM) unipolar

Utilizando varios pulsos en cada medio ciclo de la señal de salida puede reducirse el contenido armónico [9]. La generación de señales de excitación para activar y desactivar los transistores aparece en la figura 4.5(a), mediante la comparación de una señal de referencia con una onda portadora triangular. La frecuencia de la señal de referencia establece la frecuencia de la señal de salida  $f_o$ , y la frecuencia de la portadora  $f_c$  determina el número de pulsos  $p$  por cada ciclo. El índice de modulación en amplitud controla la tensión de salida.

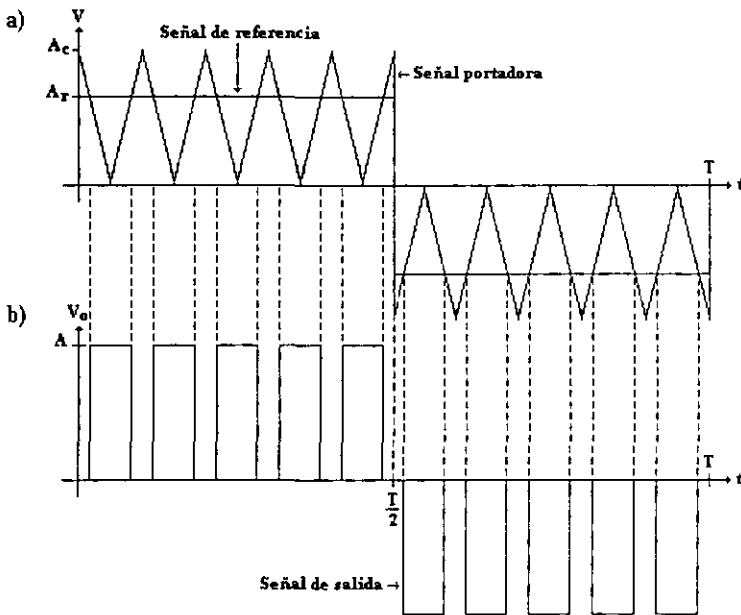


Figura 4.5. Modulación uniforme de ancho de pulso (UPWM) unipolar.  
 a) Señal de referencia y señal portadora. b) Señal de salida.

La variación del índice de modulación  $m_a$  desde 0 hasta 1 varía el ancho de pulso desde 0 hasta  $T/2p$  y la tensión pico de salida del armónico fundamental desde 0 hasta  $A$ . La tensión de salida para los inversores monofásicos en puente aparece en la figura 4.5(b).

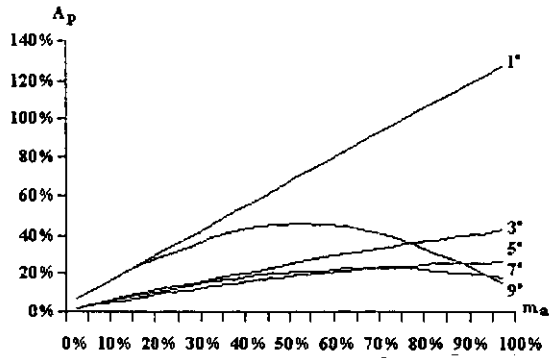


Figura 4.6. UPWM unipolar. Perfil armónico.

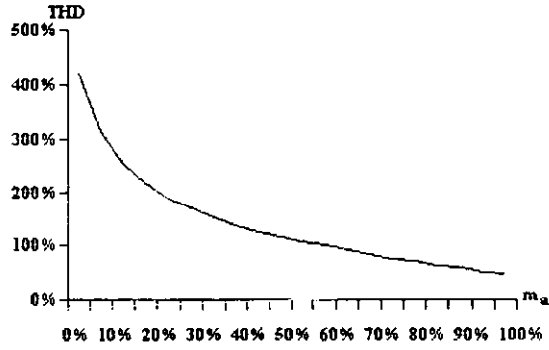


Figura 4.7. UPWM unipolar. THD.

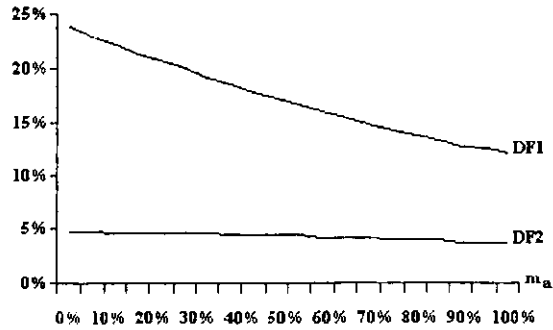


Figura 4.8. UPWM unipolar. DF1 y DF2.

La figura 4.6 muestra el perfil armónico contra la variación del índice de modulación para cinco pulsos por cada medio ciclo. El orden de las armónicas es el mismo que para la modulación de un solo pulso. El contenido armónico total aumenta, figura 4.7, y los factores de distorsión de primer y segundo orden se hacen más lineales en comparación con la modulación de un solo pulso, figura 4.8. Sin embargo, debido al gran número de conmutaciones de los interruptores de potencia las pérdidas por ese concepto aumentan. Con valores de  $p$  mayores, las amplitudes de las armónicas de menor orden serán menores, pero se incrementará la amplitud de algunas armónicas de orden más alto. En general, esta técnica de modulación proporciona resultados similares a la técnica de un solo ancho de pulso.

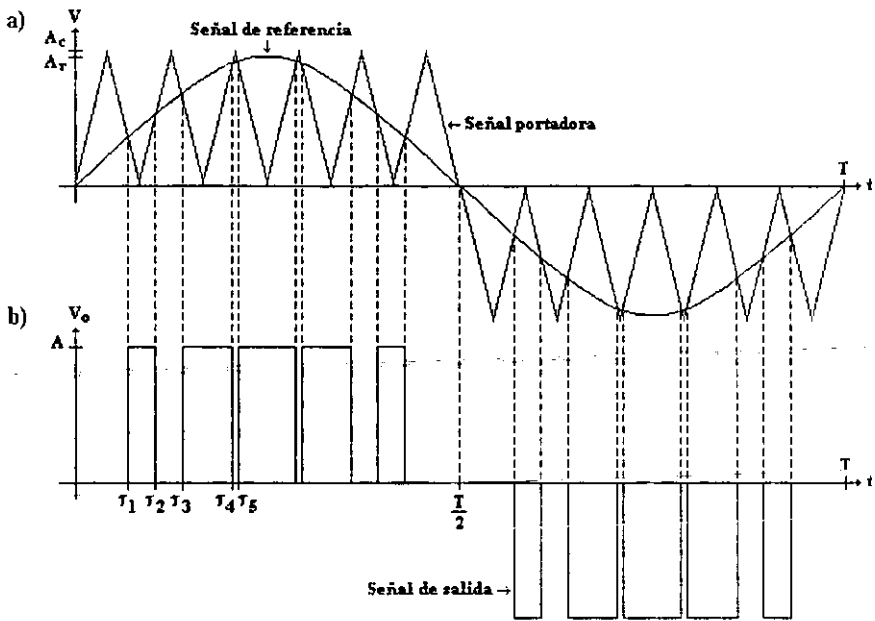
#### 4.4. Modulación senoidal de ancho de pulso (SPWM) unipolar

En vez de mantener igual el ancho de todos los pulsos, como en el caso de la modulación uniforme de ancho de pulso, el ancho de cada pulso varía en proporción con la amplitud de una onda senoidal evaluada en el centro del mismo pulso [1], [9]. El factor de distorsión y las armónicas de menor orden se reducen en forma significativa. Las señales de excitación, según se muestran en la figura 4.9(a), se generan al comparar una señal senoidal de referencia con una onda portadora triangular de frecuencia  $f_c$ . Este tipo de modulación se utiliza por lo regular en aplicaciones industriales; se abrevia SPWM. La frecuencia de la señal de referencia  $f_r$ , determina la frecuencia de salida  $f_o$  del inversor y su amplitud pico  $A_r$ , controla el índice de modulación  $m_a$ , y en consecuencia, la tensión rms de salida  $V_o$ . El número de pulsos por medio ciclo depende de la frecuencia portadora. La tensión instantáneo de salida se muestra en la figura 4.9(b).

La tensión rms de salida puede controlarse si se varía el índice de modulación  $m_a$ . Es fácil observar que el área de cada pulso corresponde aproximadamente al área bajo la onda senoidal entre los puntos medios adyacentes de los periodos inactivos de las señales de excitación

El perfil armónico se muestra en la figura 4.10 para cinco pulsos por cada medio ciclo. El contenido armónico total aumenta, figura 4.11, pero los factores de distorsión se reducen significativamente en comparación con la modulación uniforme de un solo ancho de pulso, véase la figura 4.12. Este tipo de modulación elimina todas las armónicas menores a  $2p-1$ . Para  $p=5$ , la armónica de menor orden es la novena.





**Figura 4.9. Modulación senoidal de ancho de pulso (SPWM) unipolar.**  
 a) Señal de referencia y señal portadora. b) Señal de salida.

La señal de salida de un inversor contiene armónicas. El PWM empuja a las armónicas al rango de las altas frecuencias alrededor del valor de conmutación  $f_c$  y sus múltiplos, es decir, alrededor de las armónicas  $m_f, 2m_f, 3m_f$  y así sucesivamente.

Se puede determinar la tensión pico aproximada de la fundamental de salida para el control SPWM, a partir de la ecuación 4.3, siempre y cuando  $0 \leq m_a \leq 1$ .

$$V_{m1} = m_a A \quad (4.3)$$

A fin de aumentar el valor de la señal fundamental de salida,  $m_a$  deberá incrementarse más allá de 1 llegando a la sobremodulación. La sobremodulación básicamente lleva a una operación de onda cuadrada y añade más armónicos en comparación con el rango lineal (con  $m_a \leq 1$ ). En aplicaciones que requieren baja distorsión la sobremodulación suele evitarse.

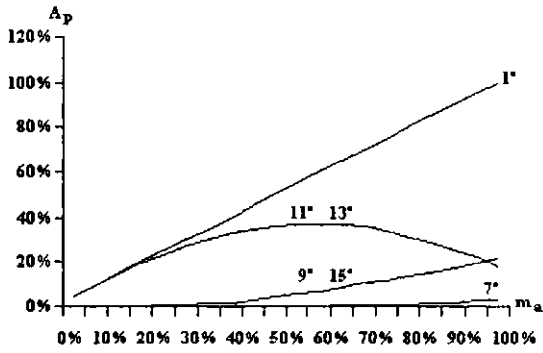


Figura 4.10. SPWM unipolar. Perfil armónico.

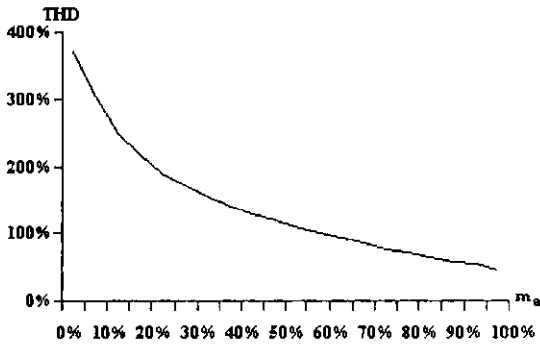


Figura 4.11. SPWM unipolar. THD

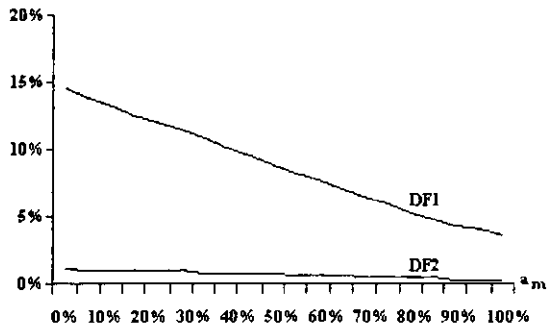


Figura 4.12. SPWM unipolar. DF1 y DF2.

### 4.5. Modulación senoidal modificada de ancho de pulso (MSPWM) unipolar

Una observación cuidadosa de la modulación senoidal del ancho de pulso permite ver que el ancho de los pulsos más cercanos al pico de la onda senoidal no cambian en forma significativa con la variación del índice de modulación. Esto se debe a las características de una onda senoidal, la técnica SPWM se puede modificar de tal manera que la onda portadora se aplique durante el primer y el último intervalo de  $60^\circ$  de cada medio ciclo. Este tipo de modulación se conoce como modulación senoidal modificada del ancho de pulso [1], [9] y se muestra en la figura 4.13(a), la salida se muestra en la figura 4.13(b). La componente fundamental se incrementa y las características armónicas mejoran. Esto reduce el número de conmutaciones de los dispositivos de potencia y las pérdidas por conmutación. El perfil armónico se muestra en la figura 4.14 para cinco pulsos por cada medio ciclo. La información del contenido armónico total y de los factores de distorsión de primer y segundo orden se presentan en las figuras 4.15 y 4.16 respectivamente.

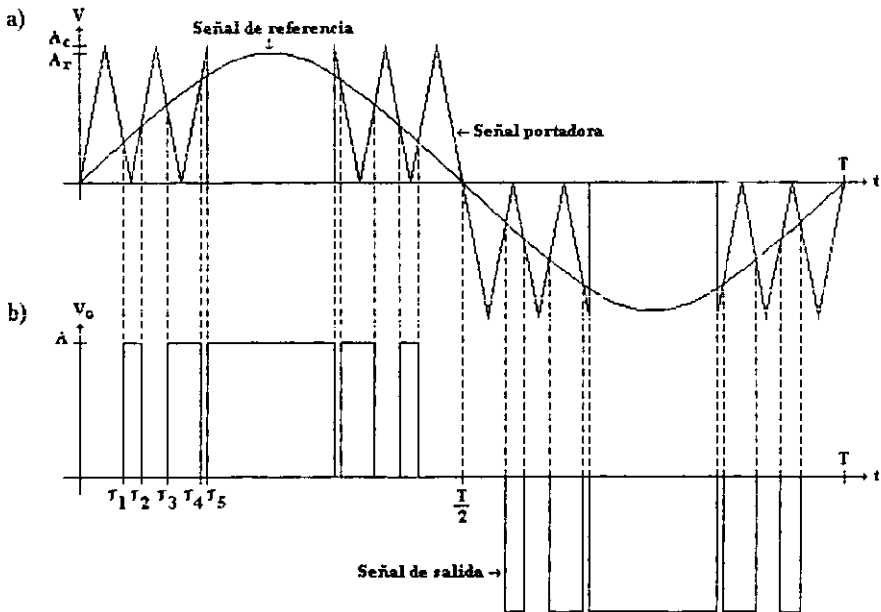


Figura 4.13. Modulación senoidal modificada de ancho de pulso (MSPWM) unipolar. a) Señal de referencia y señal portadora. b) Señal de salida.

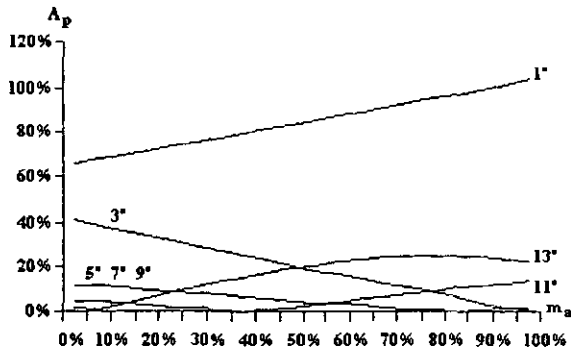


Figura 4.14. MSPWM unipolar. Perfil armónico.

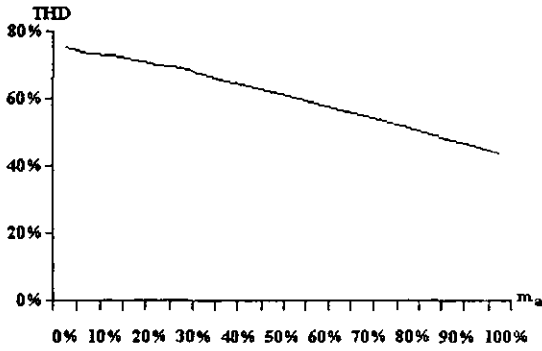


Figura 4.15. MSPWM unipolar. THD.

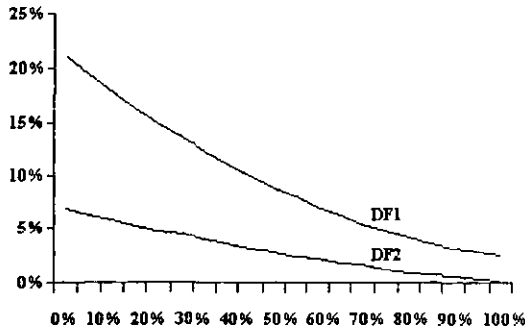


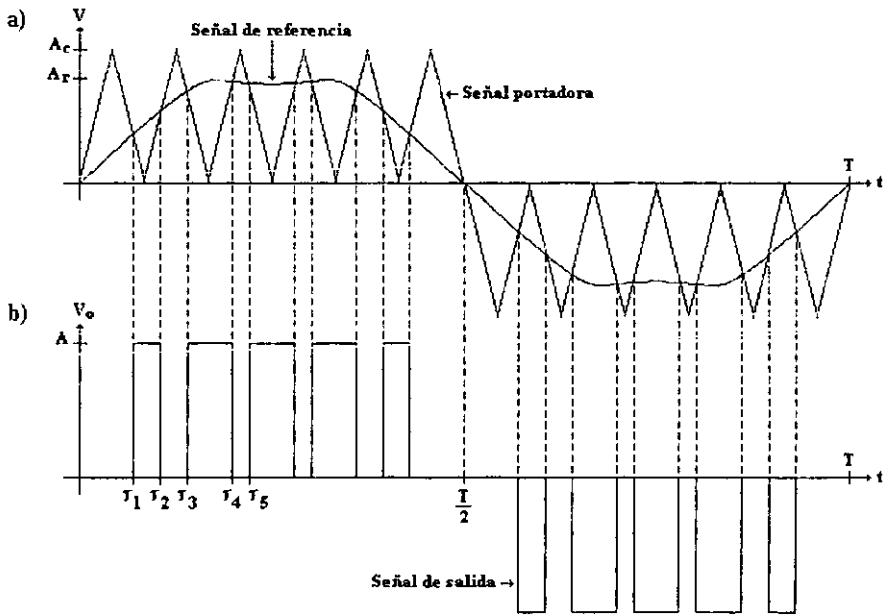
Figura 4.16. MSPWM unipolar. DF1 y DF2.

**4.6. Modulación unipolar por inyección del tercer armónico (THIPWM)**

Esta técnica de modulación mejorada ha sido derivada directamente de la técnica SPWM a través de la adición del tercer armónico con una amplitud del 17% de la amplitud de la onda senoidal pura de referencia [1], [9]. El resultado es una forma de onda aplanada, figura 4.17(a), que permite la sobremodulación con respecto a la técnica SPWM original y el incremento de la componente fundamental de salida. La expresión analítica para la forma de onda de referencia está dada por la ecuación 4.4. La forma de onda de la tensión de salida se presenta en la figura 4.17(b) y el perfil armónico para cinco pulsos por semiciclo en la figura 4.18

$$y = 1.15 \text{sen } \omega t + 0.19 \text{sen } 3\omega t \quad (4.4)$$

La figura 4.19 representa al contenido armónico total de la señal de salida y la figura 4.20 a los factores de distorsión de primer y segundo orden.



**Figura 4.17. Modulación de ancho de pulso por inyección del tercer armónico (THIPWM) unipolar. a) Señal de referencia y señal portadora. b) Señal de salida.**

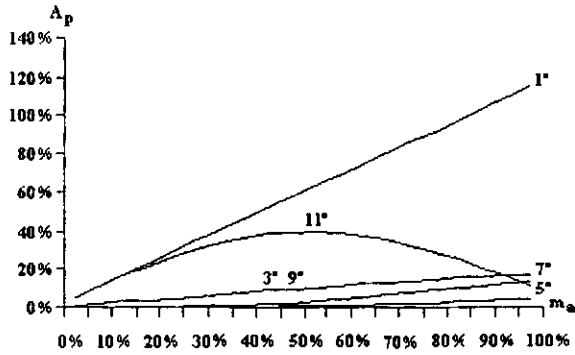
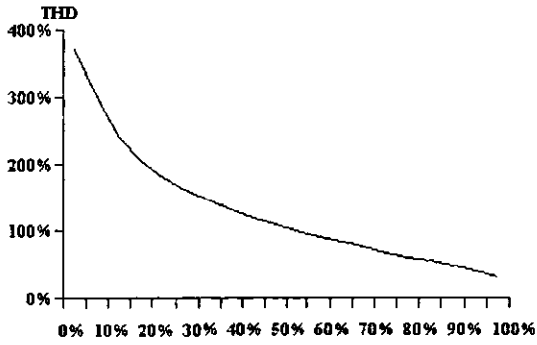


Figura 4.18. THIPWM unipolar. Perfil armónico.



4.19. THIPWM unipolar. THD.

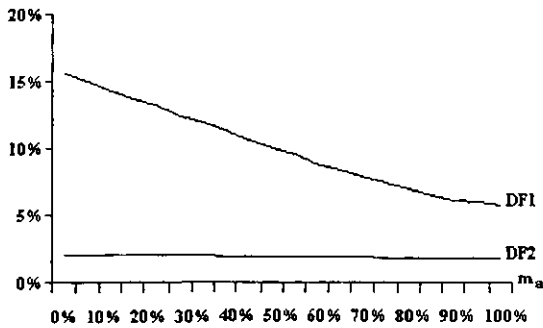


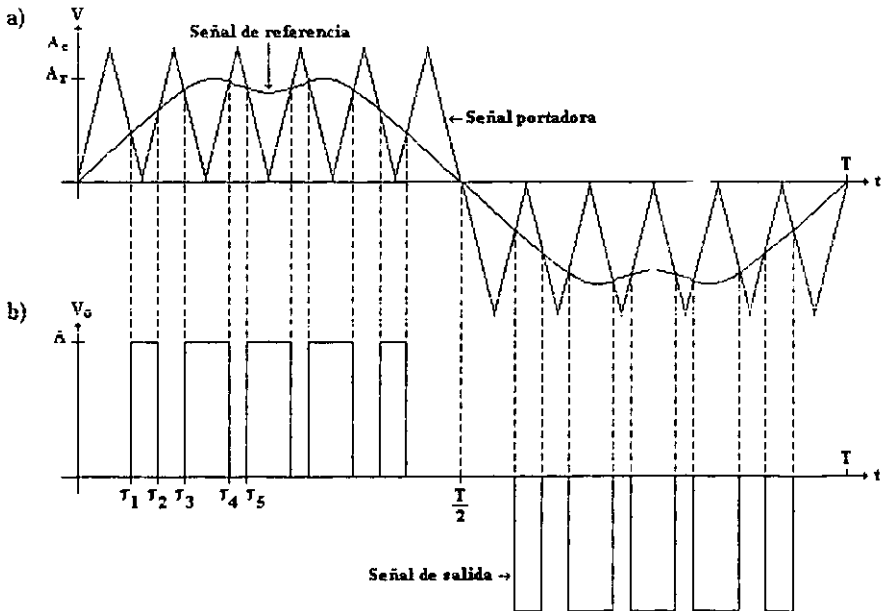
Figura 4.20. THIPWM unipolar. DF1 y DF2.

**4.7. Modulación unipolar por inyección de armónicos (HIPWM)**

Esta técnica es una variación de la modulación por inyección del tercer armónico. La variación se obtiene por la inyección de armónicos adicionales en la respectiva señal de referencia [1], [9]. La señal aplanada que se obtiene, figura 4.21(a), permite la sobremodulación con respecto a la técnica SPWM y mejorando los resultados de la técnica THIPWM. La expresión analítica para la señal de referencia está dada por la ecuación 4.5. La señal de salida para esta técnica de modulación se muestra en la figura 4.21(b) y su perfil armónico, para 5 pulsos por semiciclo, en la figura 4.22.

$$y = 1.15 \text{sen } \omega t + 0.27 \text{sen } 3\omega t - 0.029 \text{sen } 9\omega t \quad (4.5)$$

La figura 4.23 muestra el contenido armónico total de la señal de salida y la figura 4.25 el comportamiento de los factores de distorsión de primer y segundo orden con respecto al índice de modulación.



**Figura 4.21. Modulación de ancho de pulso por inyección de armónicos (HIPWM) unipolar. a) Señal de referencia y señal portadora. b) Señal de salida.**

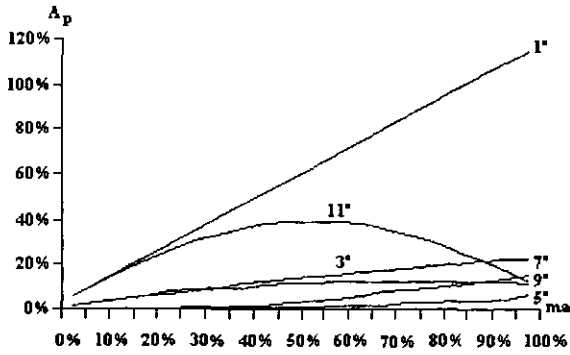


Figura 4.22. HIPWM unipolar. Perfil armónico.

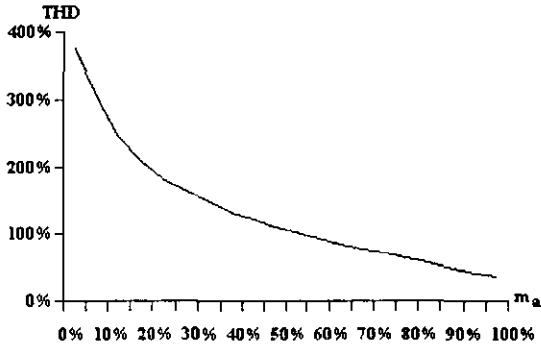


Figura 4.23. HIPWM unipolar. THD.

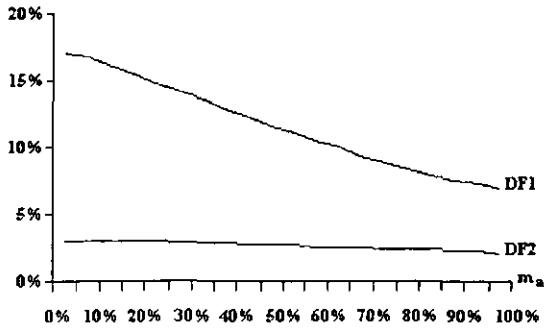


Figura 4.24. HIPWM unipolar. DF1 y DF2.



### 4.8. Modulación trapezoidal de ancho de pulso (TPWM) unipolar

En la modulación trapezoidal [9], [10] las señales de excitación se generan al comparar una onda portadora triangular con una onda moduladora trapezoidal, como se muestra en la figura 4.25(a). La onda trapezoidal puede ser obtenida a partir de una onda triangular si se limita su amplitud a  $\pm A_r$ . La señal modulada de salida se muestra en la figura 4.25(b).

Este tipo de modulación aumenta hasta 1.05A la tensión pico de salida de la componente fundamental, pero la salida contiene armónicas de menor orden. El perfil armónico para la modulación trapezoidal se muestra en la figura 4.26 para cinco pulsos por semiciclo, en donde puede observarse que la amplitud de la componente fundamental no tiene un comportamiento lineal con respecto a la modulación en amplitud. La gráfica 4.27 muestra la distorsión armónica total mientras que la figura 4.28 presenta el comportamiento de los factores de distorsión de primer y segundo orden.

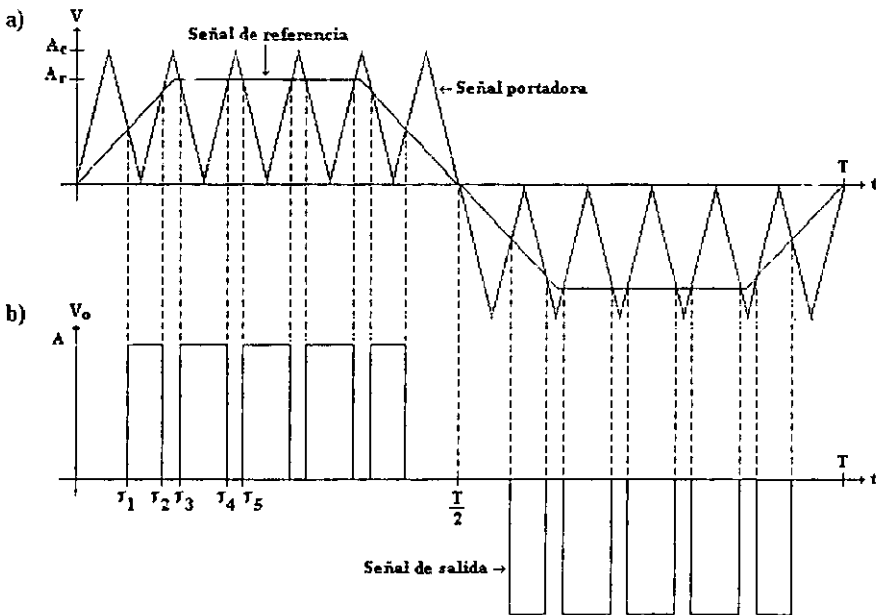


Figura 4.25. Modulación trapezoidal de ancho de pulso (TPWM) unipolar.

a) Señal de referencia y señal portadora. b) Señal de salida.

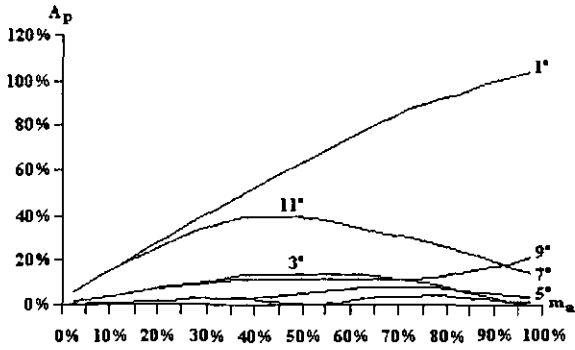


Figura 4.26. TPWM unipolar. Perfil armónico.

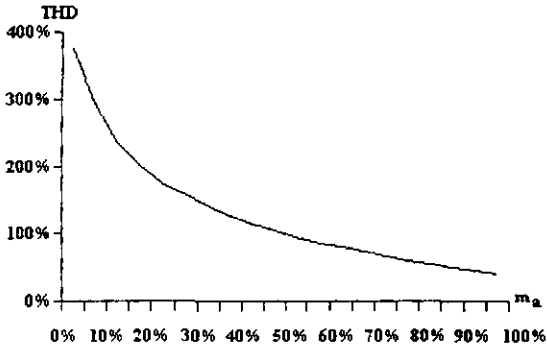


Figura 4.27. TPWM unipolar. THD.

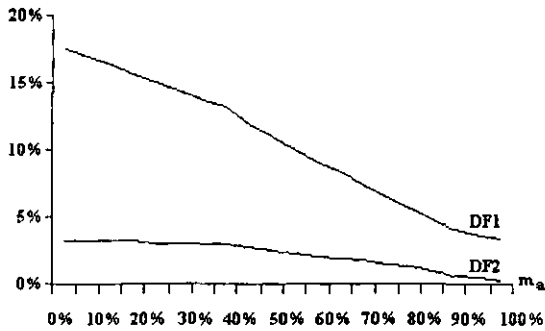


Figura 4.28. TPWM unipolar. Perfil armónico.

### 4.9. Modulación uniforme de ancho de pulso (UPWM) bipolar

En la modulación uniforme de ancho de pulso bipolar la señal de excitación se genera al comparar una señal triangular con una señal rectangular de amplitud variable, como se muestra en la figura 4.29(a). La señal de salida se muestra en la figura 4.29(b).

El perfil armónico de esta modulación se presenta en la figura 4.30, en donde se puede constatar el comportamiento lineal de la componente fundamental. El contenido armónico total, mostrado en la figura 4.31 es mayor que en su contraparte unipolar, al igual que los factores de distorsión mostrados en la figura 4.32.

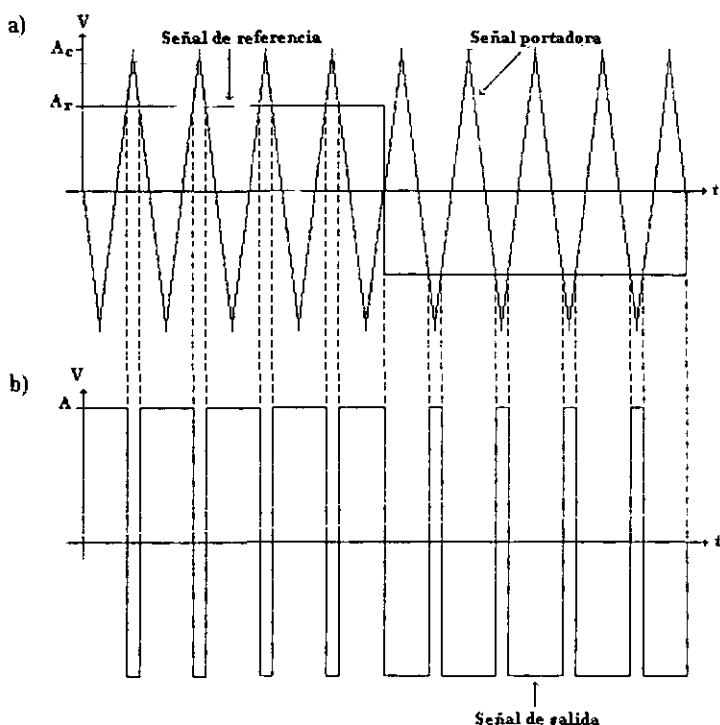


Figura 4.29. Modulación uniforme de ancho de pulso (UPWM) bipolar  
 a) Señal de referencia y señal portadora. b) Señal de salida.

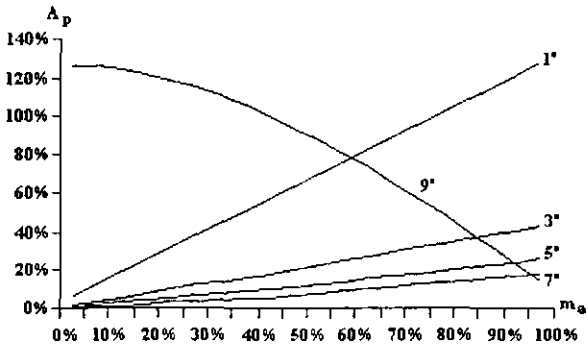


Figura 4.30. UPWM bipolar. Perfil armónico.

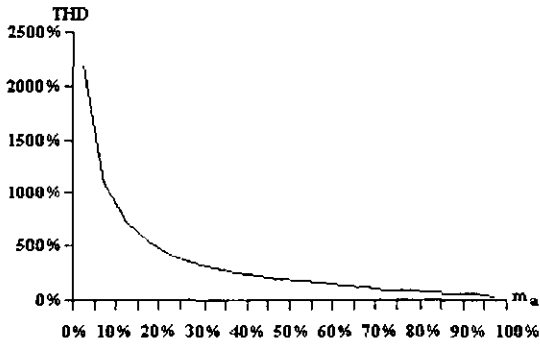


Figura 4.31. UPWM bipolar. THD.

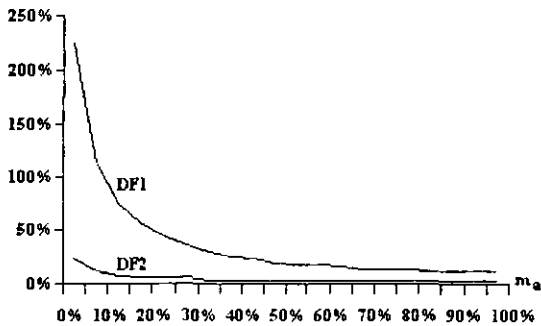


Figura 4.32. UPWM bipolar. DF1 y DF2.

### 4.10. Modulación senoidal de ancho de pulso (SPWM) bipolar

La figura 4.33(a) muestra la forma en que la señal de referencia senoidal es comparada con la señal portadora para generar las señales de excitación, en tanto que la figura 4.33(b) muestra la señal de salida que debe ser generada por el inversor. El perfil armónico de la figura 4.34 muestra que esta modulación atenúa en menor proporción los armónicos de orden menor en comparación con la modulación senoidal unipolar, y da como resultado que el contenido armónico total, mostrado en la figura 4.35, y los factores de distorsión de primer y segundo orden de la figura 4.36 sean también mayores.

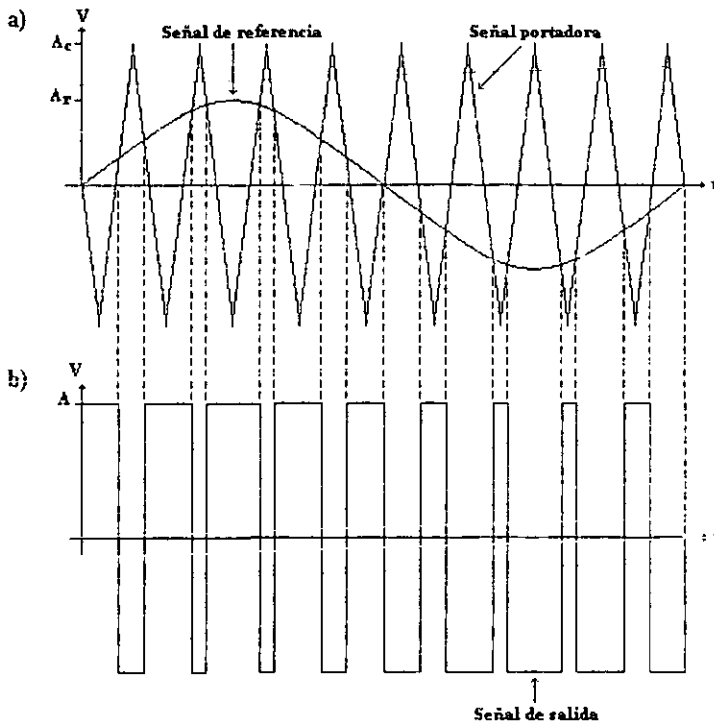


Figura 4.33. Modulación senoidal de ancho de pulso (SPWM) bipolar  
 a) Señal de referencia y señal portadora. b) Señal de salida.

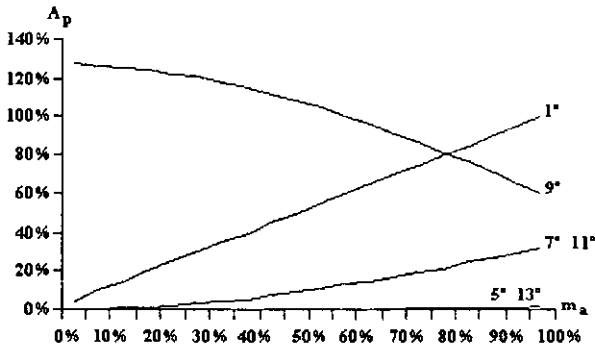


Figura 4.34. SPWM bipolar. Perfil armónico.

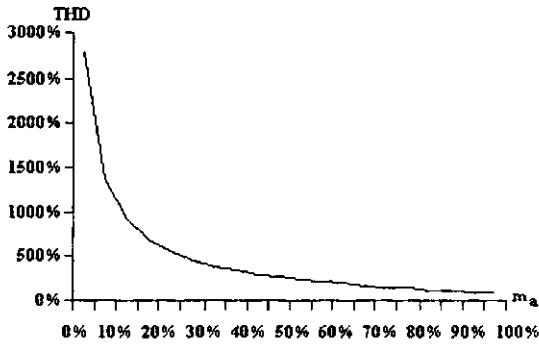


Figura 4.35. SPWM bipolar. THD.

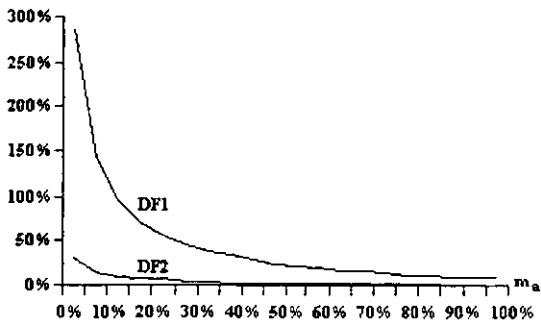
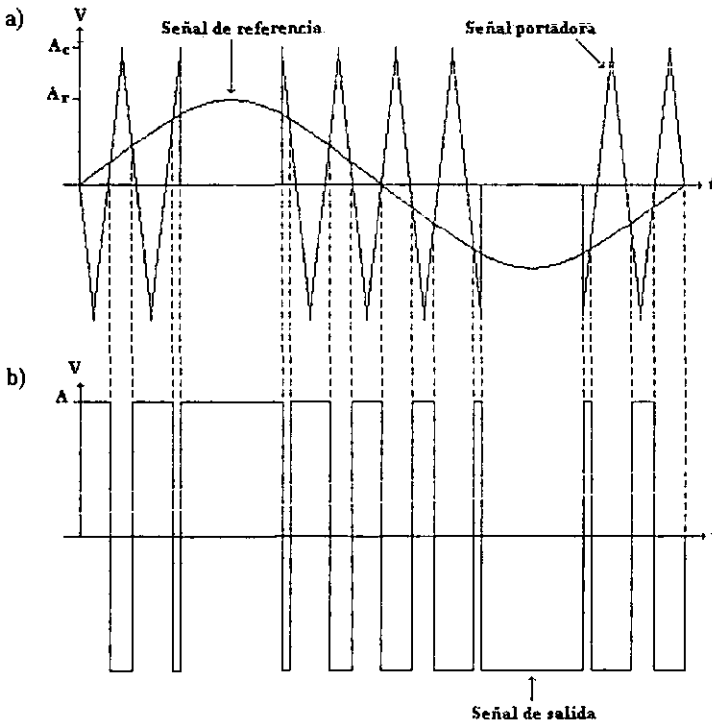


Figura 4.36. SPWM bipolar. DF1 y DF2.

**4.11. Modulación senoidal modificada de ancho de pulso (MSPWM) bipolar**

La característica distintiva de esta técnica es que la modulación sólo se realiza durante el primer y tercer intervalo de  $60^\circ$  de cada semiciclo [1], [9]. Visto de otra forma, la señal portadora sólo existe en estos intervalos, como se puede apreciar en la figura 4.37(a). La señal generada por este tipo de modulación se presenta en la figura 4.37(b). Las señales generadas permiten aumentar la amplitud de la componente fundamental, tal como se muestra en el perfil armónico de la figura 4.38. Sin embargo, el contenido armónico total aumenta, figura 4.39, en comparación con la misma técnica en su versión unipolar, y lo mismo sucede con los factores de distorsión de primer y segundo orden presentados en la figura 4.40.



**Figura 4.37. Modulación senoidal modificada de ancho de pulso (MSPWM) bipolar a) Señal de referencia y señal portadora. b) Señal de salida.**

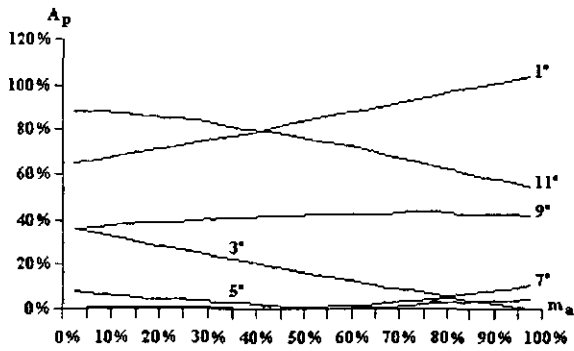


Figura 4.38. MSPWM bipolar. Perfil armónico.

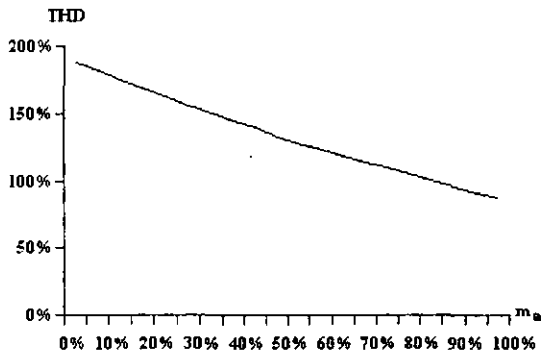


Figura 4.39. MSPWM bipolar. THD.

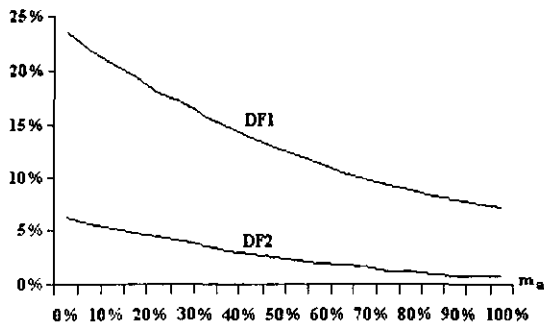


Figura 4.40. MSPWM bipolar. DF1 y DF2.



### 4.12. Modulación bipolar por inyección del tercer armónico (THIPWM)

La adición del tercer armónico a la señal de referencia para esta técnica de modulación tiene la misma finalidad que en su contraparte unipolar; aumentar la amplitud de la componente fundamental [1], [9]. La expresión matemática para la señal de referencia también es la misma. Las formas de onda para las señales portadora y de referencia se presentan en la figura 4.41(a), mientras que la señal de salida se presenta en la figura 4.41(b). El perfil armónico para cinco pulsos por semiciclo se muestra en la figura 4.42. La información referente a la distorsión armónica total y a los factores de distorsión de primer y segundo orden está contenida en las figuras 4.43 y 4.44.

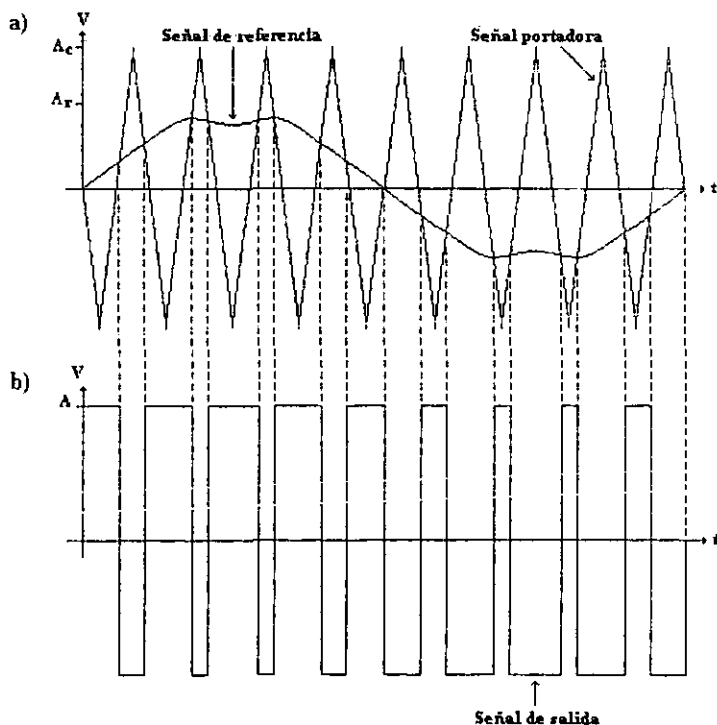


Figura 4.41. Modulación de ancho de pulso por inyección del tercer armónico (THIPWM) bipolar. a) Señal de referencia y señal portadora. b) Señal de salida.

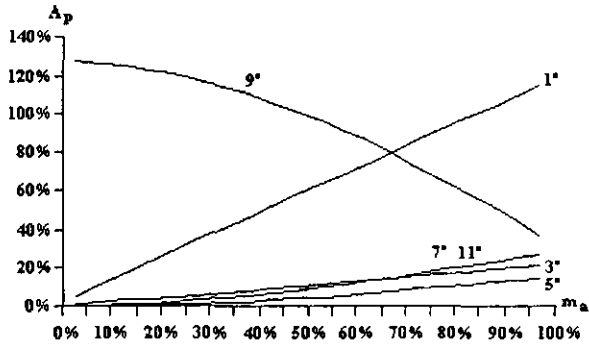


Figura 4.42. THIPWM bipolar. Perfil armónico.

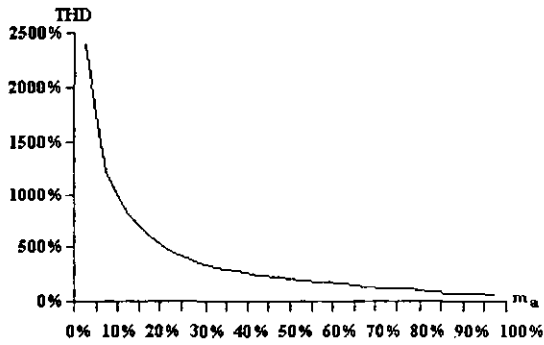


Figura 4.43. THIPWM bipolar. THD.

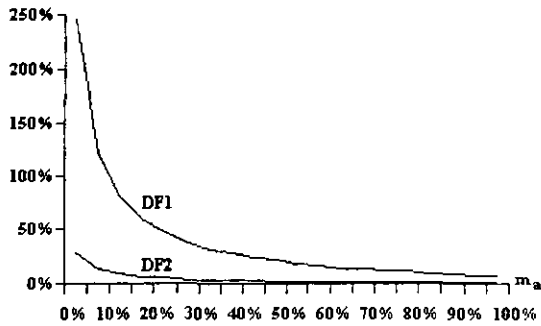


Figura 4.44. THIPWM bipolar. DF1 y DF2.

### 4.13. Modulación bipolar por inyección de armónicos (HIPWM)

En esta técnica a la señal de referencia se le agregan los armónicos tercero y quinto, con la finalidad de aumentar la componente fundamental en la señal de salida [1], [9]. Los resultados obtenidos son similares a los de la técnica anterior, como se ve en las figuras 4.45(a) y 4.45(b) donde se muestra tanto la generación de las señales de excitación como la señal de salida del inversor. El perfil armónico se muestra en la figura 4.46.

El comportamiento del contenido armónico total con respecto al índice de modulación se presenta en la figura 4.47, y al igual que para las técnicas anteriores, se puede constatar un aumento substancial en su valor, al igual que en los factores de distorsión de primer y segundo orden de la figura 4.48.

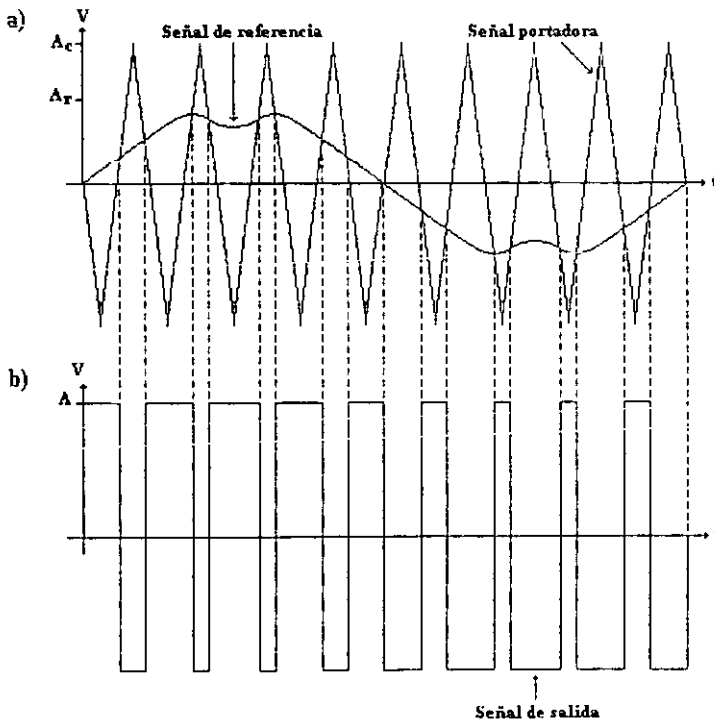


Figura 4.45. Modulación de ancho de pulso por inyección de armónicos (HIPWM) bipolar. a) Señal de referencia y señal portadora. b) Señal de salida.

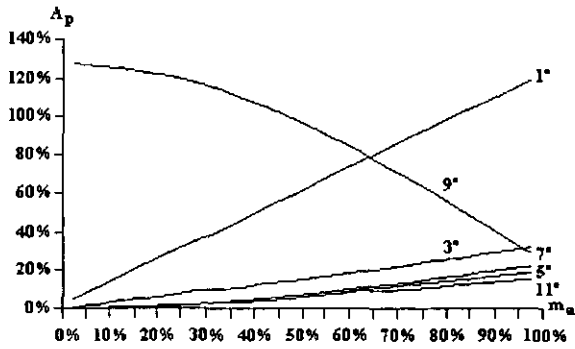


Figura 4.46. HIPWM bipolar. Perfil armónico.

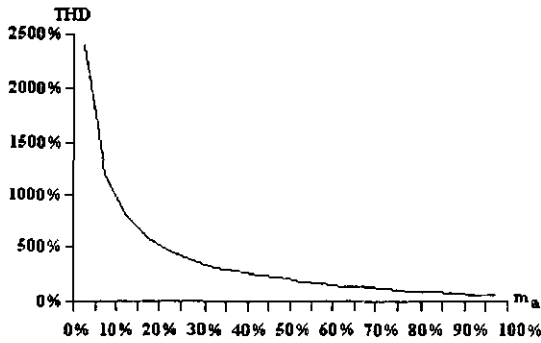
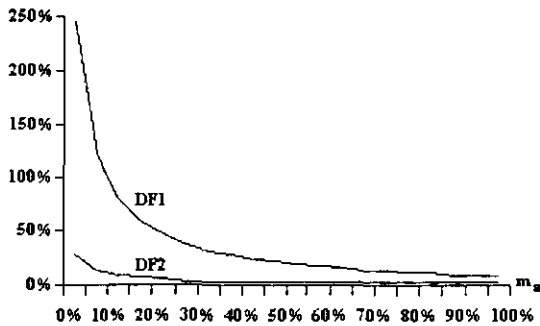


Figura 4.47. HIPWM bipolar. THD.



4.48. HIPWM bipolar. DF1 y DF2.

#### 4.14. Modulación trapezoidal de ancho de pulso (TPWM) bipolar.

La comparación de una señal de referencia trapezoidal con una señal portadora triangular, mostrada en la figura 4.49(a), es el origen de esta técnica de modulación [9], [10]. En la figura 4.49(b) se observa la señal de salida que se obtiene con esta modulación. La figura 4.50 muestra el perfil armónico para cinco pulsos por semiciclo.

El contenido armónico total es grande y su comportamiento se observa en la figura 4.51. Al igual que todas las técnicas bipolares anteriores, se observa un aumento en los factores de distorsión de la figura 4.52 con respecto a las técnicas unipolares, principalmente en el factor de distorsión de primer orden.

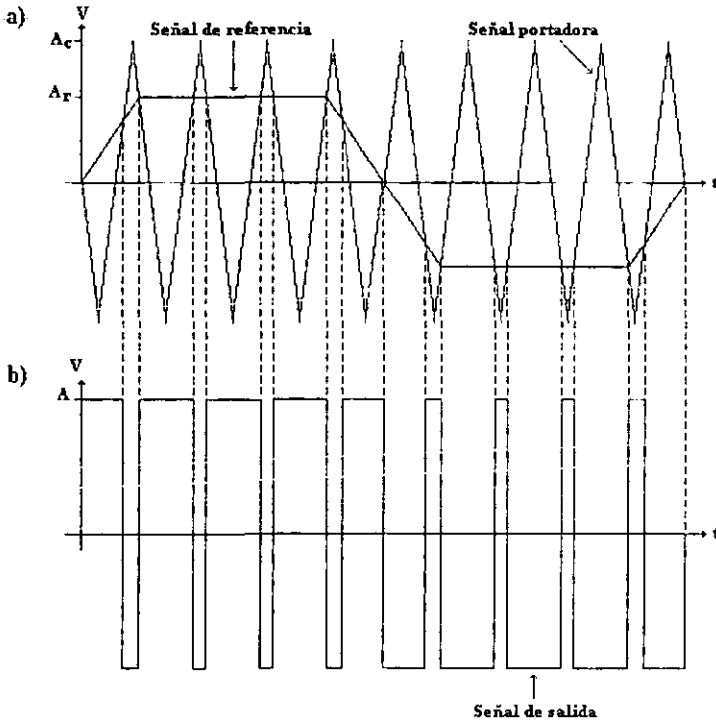


Figura 4.49. Modulación trapezoidal de ancho de pulso (TPWM) bipolar. a) Señal de referencia y señal portadora. b) Señal de salida.

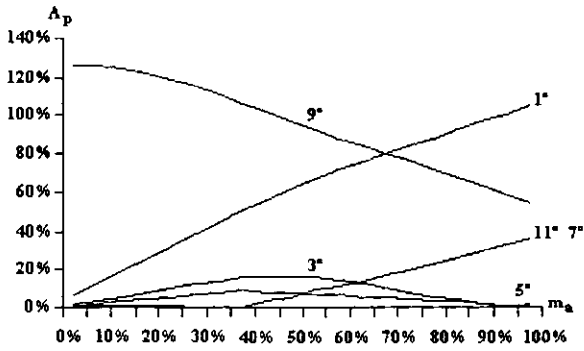


Figura 4.50. TPWM bipolar. Perfil armónico.

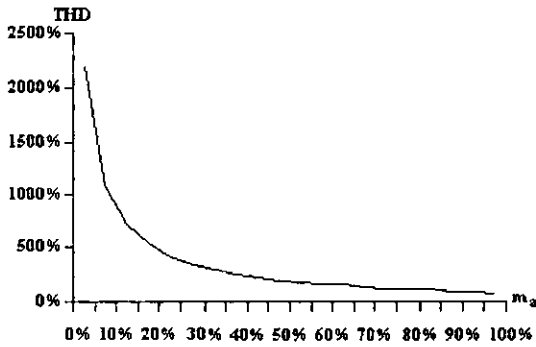


Figura 4.51. TPWM bipolar. THD.

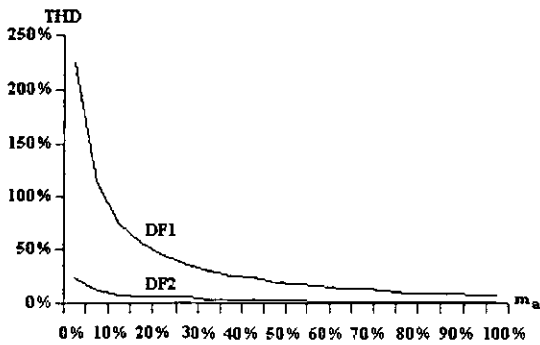


Figura 4.52. TPWM bipolar. DF1 y DF2.

**4.15. Comparación de las técnicas de modulación**

Con la finalidad de valorar las técnicas de modulación vistas se elaboraron tres tablas comparativas. Las tablas 4.1 y 4.2 muestran los primeros nueve armónicos para las distintas técnicas de modulación con 5 pulsos por semiciclo y un índice de modulación del 90%. La tabla 4.1 contiene a la componente fundamental y a los primeros cuatro armónicos mientras que la tabla 4.2 tiene los datos de los siguientes cinco armónico. La distorsión armónica total y los factores de distorsión de primer y segundo orden son el contenido de la tabla 4.3. En las tres tablas la información incluye a señales unipolares y bipolares. En las primeras dos tablas las señales unipolares son marcadas con una letra U y las bipolares con una letra B, en la tabla 4.3 la distinción entre ambos tipos de señales se da desde el encabezado.

Técnica	Armónicos				
	1°	3°	5°	7°	9°
UN PULSO	125.76%	37.82%	18.01%	8.26%	2.21%
UPWM (U)	114.95%	39.35%	25.15%	20.63%	25.73%
UPWM (B)	114.53%	38.00%	22.57%	15.83%	31.85%
SPWM (U)	89.97%	0.01%	0.11%	2.14%	17.69%
SPWM (B)	89.95%	0.01%	1.19%	26.81%	71.28%
MSPWM (U)	99.03%	4.34%	0.91%	2.39%	0.02%
MSPWM (B)	99.59%	3.16%	3.56%	7.53%	42.70%
THIPWM (U)	103.31%	16.07%	3.69%	11.09%	14.62%
THIPWM (B)	103.76%	18.91%	11.92%	22.95%	51.54%
HIPWM (U)	103.32%	21.10%	4.29%	12.31%	12.37%
HIPWM (B)	107.20%	28.18%	16.94%	18.89%	44.44%
TPWM (U)	97.52%	4.69%	3.25%	6.03%	16.55%
TPWM (B)	96.73%	1.75%	2.11%	28.81%	63.55%

**Tabla 4.1. Contenido armónico en técnicas de modulación.**

Técnica	Armónicos				
	11°	13°	15°	17°	19°
UN PULSO	1.81%	4.45%	6.00%	6.67%	6.62%
UPWM (U)	1.18%	6.16%	7.56%	9.22%	17.14%
UPWM (B)	9.22%	6.92%	4.09%	5.12%	19.27%
SPWM (U)	25.51%	25.50%	17.22%	1.03%	10.58%
SPWM (B)	26.70%	0.93%	17.64%	25.52%	25.78%
MSPWM (U)	11.39%	24.20%	1.62%	25.88%	12.78%
MSPWM (B)	59.35%	23.81%	3.07%	11.23%	22.67%
THIPWM (U)	20.42%	16.46%	8.71%	3.31%	1.28%
THIPWM (B)	19.62%	1.22%	11.78%	17.13%	24.04%
HIPWM (U)	22.05%	15.67%	4.07%	5.52%	1.55%
HIPWM (B)	13.91%	3.69%	4.89%	13.39%	23.90%
TPWM (U)	20.91%	21.04%	16.65%	0.17%	10.75%
TPWM (B)	29.34%	0.34%	19.40%	21.73%	21.22%
TE A (U)	0.00%	18.67%	22.33%	8.82%	23.30%
TE A (B)	29.79%	56.33%	36.76%	4.21%	0.24%

Tabla 4.2. Contenido armónico en técnicas de modulación.

Técnica	Unipolar			Bipolar		
	THD	DF <sub>1</sub>	DF <sub>2</sub>	THD	DF <sub>1</sub>	DF <sub>2</sub>
UN PULSO	37.17%	10.49%	3.39%			
UPWM	60.16%	12.80%	3.93%	72.37%	12.42%	3.80%
SPWM	61.66%	4.40%	0.39%	121.26%	10.65%	1.19%
MSPWM	49.96%	3.20%	0.53%	100.76%	7.92%	0.85%
THIPWM	49.85%	6.19%	1.77%	92.53%	9.59%	2.22%
HIPWM	51.46%	7.62%	2.30%	85.96%	10.98%	3.06%
TPWM	53.33%	4.06%	0.65%	106.58%	9.36%	1.08%

Tabla 4.3. Parámetros de evaluación en técnicas de modulación.



De acuerdo con las tablas comparativas de este apartado, las técnicas de modulación que tienen un mejor desempeño son las técnicas de modulación senoidal, senoidal modificada y trapezoidal. Estas técnicas son las que tienen armónicos de menor amplitud cercanos a la fundamental y también son las que poseen factores de distorsión más pequeños. De las técnicas restantes, la de un solo pulso y la de pulsos uniformes son las que peores factores de distorsión tienen. Las técnicas de inyección del tercer armónico y de armónicos seleccionados parecen tampoco tener muy buenas características, pero de acuerdo a la forma en la que la amplitud de sus armónicos se distribuye su uso parece ser exclusivo de inversores trifásicos con carga en estrella, en donde el tercer armónico y sus múltiplos se cancelan.

---

# **CAPÍTULO 5**

---

**TÉCNICAS PARA LA ELIMINACIÓN DE ARMÓNICOS**

**ESTA TESIS NO SALE  
DE LA BIBLIOTECA**

## Introducción

Con el uso de técnicas de modulación de ancho de pulso se hace evidente que la amplitud de los armónicos varía de acuerdo al patrón de conmutación y en algunos casos incluso llega a desaparecer. Esto sugiere que con un patrón de conmutación adecuado es posible eliminar tantos armónicos como se desee. La metodología seguida para lograr tal fin implica la resolución de ecuaciones simultáneas no lineales, y dada la dificultad de resolver tales ecuaciones, implica también el uso de métodos numéricos, en este caso el método de Newton. La eliminación de armónicos es posible tanto en señales unipolares como bipolares y el procedimiento en ambos casos es similar.

### 5.1. El método de Newton

El método de Newton suministra un procedimiento muy cómodo para calcular la raíz de una ecuación cuando se tiene una aproximación del valor de la raíz [3]. La figura 5.1 muestra un punto  $P(a, f(a))$  de la gráfica  $f(x)$ , sea  $PT$  la tangente en  $P$ . Evidentemente, la abscisa  $a'$  del punto  $T$  de intersección de la tangente con el eje de las  $x$ , es un valor aproximado del punto de intersección del eje de las  $x$  con la gráfica, y por lo tanto, de la raíz correspondiente de  $f(x) = 0$ . El método de Newton determina la abscisa de  $P$ .

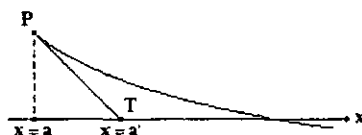


Figura 5.1. El método de Newton.

La forma de hallar esa abscisa es la siguiente:

Las coordenadas del punto  $P$  son

$$x_1 = a_0 \quad (5.1)$$

$$y_1 = f(a_0) \quad (5.2)$$

La pendiente de la tangente  $PT$  es

$$m_1 = f'(a_0) \quad (5.3)$$

De acuerdo con la ecuación de la recta, la ecuación que representa a  $PT$  es

$$y - f(a_0) = f'(a_0)(x - a_0) \quad (5.4)$$

Si  $y = 0$  y se despeja el valor de  $x$ , obtenemos la fórmula de aproximación de Newton en la ecuación 5.5.

$$a_1 = a_0 - \frac{f(a_0)}{f'(a_0)} \quad (5.5)$$

Con  $a_1$  hallado por medio de la ecuación 5.5, podemos reemplazar  $a_0$  por  $a_1$  en el segundo miembro y obtener como segunda aproximación a la ecuación 5.6.

$$a_2 = a_1 - \frac{f(a_1)}{f'(a_1)} \quad (5.6)$$

El procedimiento se puede continuar y obtener una serie de valores  $a_0, a_1, a_2, \dots$ , que se aproximan cada vez más a la raíz exacta.

El método de Newton tiene una estructura aún más útil que le brinda generalización, para obtenerla lo primero que se debe hacer es reordenar la ecuación 5.6. Esta reorganización da origen a la ecuación 5.7.

$$f'(a_0)(a_1 - a_0) + f(a_0) = 0 \quad (5.7)$$

En la ecuación 5.7 a la diferencia  $a_1 - a_0$  se le considera como un incremento  $\Delta a$  el cual habrá de ser determinado para posteriormente ser sumado a  $a_0$  a fin de determinar  $a_1$ . El procedimiento anterior proporciona una forma alternativa a la ecuación 5.6, ahora compuesta por las ecuaciones 5.8 y 5.9.

$$f'(a_0)\Delta a + f(a_0) = 0 \quad (5.8)$$

$$a_1 = a_0 + \Delta a \quad (5.9)$$

**5.1.1. El método de Newton en sistemas de ecuaciones simultáneas**

El método de Newton para ecuaciones con una variable independiente queda completamente descrito con las ecuaciones 5.8 y 5.9, pero este es sólo un caso particular. En el caso general, el método se aplica a sistemas de  $n$  ecuaciones simultáneas con  $n$  variables independientes.

Para el caso generalizado del método de Newton cada uno de los elementos de las ecuaciones 5.8 y 5.9 es reemplazado por una matriz. La derivada de la función se convierte en una matriz de derivadas parciales del sistema de ecuaciones, el incremento cambia a una matriz de incrementos y la función evaluada en una aproximación inicial se transforma en una matriz compuesta por las ecuaciones simultáneas evaluadas en un conjunto de primeras aproximaciones. Así, el conjunto de nuevas aproximaciones se obtiene sumando las aproximaciones iniciales a la matriz de incrementos. Las ecuaciones matriciales 5.10 y 5.11 resumen todo el procedimiento, en ellas  $F$  es el conjunto de ecuaciones simultáneas,  $\partial F / \partial X$  es la matriz de derivadas que en el método es evaluada en  $X_0$ ,  $X_0$  es la matriz de aproximaciones iniciales y  $X_1$  es la matriz de nuevas aproximaciones.

$$\frac{\partial F}{\partial X} \Delta X + F_{X_0} = 0 \quad (5.10)$$

$$X_1 = X_0 + \Delta X \quad (5.11)$$

Los elementos de las expresiones anteriores son definidos en su estructura con las ecuaciones restantes, desde la ecuación 5.12 hasta la 5.18.

$$F = \begin{cases} f_1(x_1, x_2, \dots, x_n) = 0 \\ f_2(x_1, x_2, \dots, x_n) = 0 \\ \vdots \\ f_n(x_1, x_2, \dots, x_n) = 0 \end{cases} \quad (5.12) \quad X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad (5.13)$$

$$X_0 = \begin{bmatrix} x_{1,0} \\ x_{2,0} \\ \vdots \\ x_{n,0} \end{bmatrix} \quad (5.14)$$

$$\Delta X = \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \vdots \\ \Delta x_n \end{bmatrix} \quad (5.15)$$

$$X_1 = \begin{bmatrix} x_{1,1} \\ x_{2,1} \\ \vdots \\ x_{n,1} \end{bmatrix} \quad (5.16)$$

$$\frac{\partial F}{\partial X_{X_0}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1 X_0} & \frac{\partial f_1}{\partial x_2 X_0} & \dots & \frac{\partial f_1}{\partial x_n X_0} \\ \frac{\partial f_2}{\partial x_1 X_0} & \frac{\partial f_2}{\partial x_2 X_0} & \dots & \frac{\partial f_2}{\partial x_n X_0} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1 X_0} & \frac{\partial f_n}{\partial x_2 X_0} & \dots & \frac{\partial f_n}{\partial x_n X_0} \end{bmatrix} \quad (5.17) \quad F_{X_0} = \begin{bmatrix} f_1(X_0) \\ f_2(X_0) \\ \vdots \\ f_n(X_0) \end{bmatrix} \quad (5.18)$$

Al igual que en el caso de las ecuaciones con una sola variable, el procedimiento se puede continuar obteniendo conjuntos de valores  $X_0, X_1, X_2, \dots$ , cada vez más cercanos a las raíces del sistema.

### 5.2. Eliminación de armónicos en señales unipolares

El desarrollo en serie de Fourier de una señal unipolar se obtiene con las expresiones 5.19 y 5.20. A fin de eliminar los armónicos deseados es necesario plantear un grupo de ecuaciones con tantos elementos como armónicos vayan a ser eliminados. Este sistema se genera a partir de la ecuación 5.20 al ser igualada a cero.

$$f(t) \approx \sum_{n=1}^{\infty} B_n \operatorname{sen} \frac{2n\pi}{T} t \quad \text{para } n=1,3,5,\dots \quad (5.19)$$

$$B_n = \sum_{m=1}^k \frac{4A}{n\pi} (-1)^{m+1} \cos \frac{2n\pi}{T} \tau_m \quad \text{para } n=1,3,5,\dots \quad (5.20)$$

Suponiendo que  $n_1, n_2, \dots, n_k$  son los  $k$  armónicos que serán eliminados, entonces, de la expresión 5.20 se obtienen las siguientes ecuaciones simultáneas

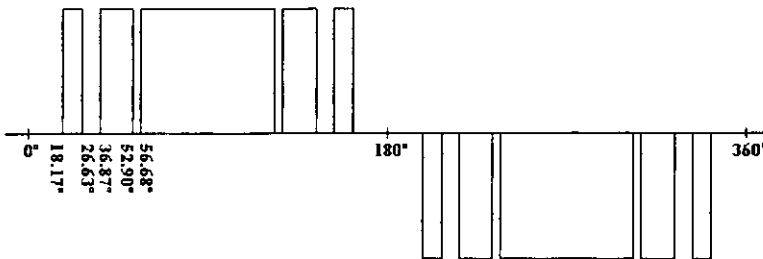
$$F = \begin{cases} f_1(T) = \sum_{m=1}^k (-1)^{m+1} \cos \frac{2n_1\pi}{T} \tau_m = 0 \\ f_2(T) = \sum_{m=1}^k (-1)^{m+1} \cos \frac{2n_2\pi}{T} \tau_m = 0 \\ \vdots \\ f_k(T) = \sum_{m=1}^k (-1)^{m+1} \cos \frac{2n_k\pi}{T} \tau_m = 0 \end{cases} \quad (5.21)$$

A partir de este sistema de ecuaciones se procede a obtener la matriz de derivadas  $\frac{\partial F}{\partial T}$  para aplicar el método de Newton, pues el sistema de ecuaciones no es lineal.

$$\frac{\partial F}{\partial T} = \begin{bmatrix} -\frac{2n_1\pi}{T} \operatorname{sen} \frac{2n_1\pi}{T} \tau_1 & \frac{2n_1\pi}{T} \operatorname{sen} \frac{2n_1\pi}{T} \tau_2 & \dots & \pm \frac{2n_1\pi}{T} \operatorname{sen} \frac{2n_1\pi}{T} \tau_k \\ -\frac{2n_2\pi}{T} \operatorname{sen} \frac{2n_2\pi}{T} \tau_1 & \frac{2n_2\pi}{T} \operatorname{sen} \frac{2n_2\pi}{T} \tau_2 & \dots & \pm \frac{2n_2\pi}{T} \operatorname{sen} \frac{2n_2\pi}{T} \tau_k \\ \vdots & \vdots & \dots & \vdots \\ -\frac{2n_k\pi}{T} \operatorname{sen} \frac{2n_k\pi}{T} \tau_1 & \frac{2n_k\pi}{T} \operatorname{sen} \frac{2n_k\pi}{T} \tau_2 & \dots & \pm \frac{2n_k\pi}{T} \operatorname{sen} \frac{2n_k\pi}{T} \tau_k \end{bmatrix} \quad (5.22)$$

Con esta técnica de eliminación [8] en señales unipolares es posible eliminar tantos armónicos como número de pulsos por semiciclo existan. Como particularidad, se debe mencionar que el conjunto de valores que sirven como primeras aproximaciones deben ser diferentes, mayores a  $0^\circ$  y menores que  $90^\circ$ , y ser proporcionados en un orden ascendente. El método funciona apropiadamente siempre y cuando los armónicos a eliminar no sean consecuentes en cantidad par, para ese caso, el método numérico nunca converge. Para un número impar de armónicos a eliminar, el método siempre obtiene una solución.

La figura 5.2 muestra una señal con cinco pulsos por semiciclo en la cual se han eliminado los armónicos  $3^\circ$ ,  $5^\circ$ ,  $7^\circ$ ,  $9^\circ$  y  $11^\circ$ . La resolución del sistema de ecuaciones da como solución cinco ángulos de disparo por semiciclo, cuyos valores se encuentran entre  $0^\circ$  y  $90^\circ$ . Para poder determinar el resto de los ángulos de disparo se recurre a la simetría de la señal.



5.2. Eliminación de armónicos. Señal unipolar.

Para el tipo de señal determinada la amplitud pico de la componente fundamental es de 102.15%, tomando en cuenta que la tensión de alimentación de cd es del 100%, mientras que la distorsión armónica total es del 48.03%. El factor de distorsión de primer orden es del 2.53% y el de segundo orden de 0.16%. El comportamiento de los primeros 24 armónicos se puede observar en la tabla 5.1, en donde la amplitud pico de cada armónico tiene como referencia la tensión de alimentación y se presenta como un porcentaje.

Bn(03): 000.00%	Bn(05): 000.00%	Bn(07): 000.00%	Bn(09): 000.00%
Bn(11): 000.00%	Bn(13): 018.67%	Bn(15): 022.33%	Bn(17): 008.82%
Bn(19): 023.30%	Bn(21): 007.59%	Bn(23): 001.22%	Bn(25): 001.06%
Bn(27): 011.99%	Bn(29): 006.33%	Bn(31): 006.08%	Bn(33): 002.14%
Bn(35): 005.57%	Bn(37): 008.19%	Bn(39): 006.46%	Bn(41): 005.30%
Bn(43): 000.62%	Bn(45): 003.36%	Bn(47): 002.60%	Bn(49): 000.50%

Tabla 5.1. Amplitud de los armónicos en una señal unipolar.

### 5.3. Eliminación de armónicos en señales bipolares

Para una señal bipolar las ecuaciones de la serie de Fourier son la 5.23 y la 5.24. Al igual que en el caso de una señal unipolar, es necesario utilizar el método de Newton para determinar los ángulos de disparo a fin de eliminar los armónicos deseados.

$$f(t) \approx \sum_{n=1}^{\infty} B_n \operatorname{sen} \frac{2n\pi}{T} t \quad \text{para } n = 1, 3, 5, \dots \quad (5.23)$$

$$B_n = \frac{4A}{n\pi} \left[ 1 + 2 \sum_{m=1}^{k-1} (-1)^m \cos \frac{2n\pi}{T} \tau \right] \quad \text{para } n = 1, 3, 5, \dots \quad (5.24)$$

Al igualar la ecuación 5.24 a cero se obtiene el sistema de ecuaciones  $F$  con  $k-1$  ecuaciones simultáneas correspondientes a los  $k-1$  armónicos que se desean eliminar.



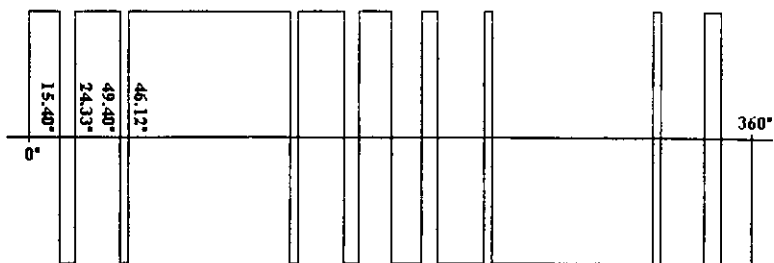
$$F = \begin{cases} f_1(T) = 1 + 2 \sum_{m=1}^{k-1} (-1)^m \cos \frac{2n_1 \pi}{T} \tau_m = 0 \\ f_2(T) = 1 + 2 \sum_{m=1}^{k-1} (-1)^m \cos \frac{2n_2 \pi}{T} \tau_m = 0 \\ \vdots \\ f_k(T) = 1 + 2 \sum_{m=1}^{k-1} (-1)^m \cos \frac{2n_{k-1} \pi}{T} \tau_m = 0 \end{cases} \quad (5.25)$$

Posteriormente, a partir del sistema de ecuaciones anterior se obtiene la matriz de derivadas  $\partial F / \partial T$ .

$$\frac{\partial F}{\partial T} = \begin{bmatrix} \frac{4n_1 \pi}{T} \operatorname{sen} \frac{2n_1 \pi}{T} \tau_1 & -\frac{4n_1 \pi}{T} \operatorname{sen} \frac{2n_1 \pi}{T} \tau_2 & \cdots & \pm \frac{4n_1 \pi}{T} \operatorname{sen} \frac{2n_1 \pi}{T} \tau_k \\ \frac{4n_2 \pi}{T} \operatorname{sen} \frac{2n_2 \pi}{T} \tau_1 & -\frac{4n_2 \pi}{T} \operatorname{sen} \frac{2n_2 \pi}{T} \tau_2 & \cdots & \pm \frac{4n_2 \pi}{T} \operatorname{sen} \frac{2n_2 \pi}{T} \tau_k \\ \vdots & \vdots & \cdots & \vdots \\ \frac{4n_k \pi}{T} \operatorname{sen} \frac{2n_k \pi}{T} \tau_1 & -\frac{4n_k \pi}{T} \operatorname{sen} \frac{2n_k \pi}{T} \tau_2 & \cdots & \pm \frac{4n_k \pi}{T} \operatorname{sen} \frac{2n_k \pi}{T} \tau_k \end{bmatrix} \quad (5.26)$$

A diferencia de la eliminación de armónicos en señales unipolares, en las señales bipolares el número de armónicos que se eliminan es equivalente al número de pulsos por semiciclo menos uno al igual que el número de ángulos de disparo que se obtiene como solución [8], aunque en realidad el número de ángulos de disparo es el mismo que el número de pulsos ya que para este tipo de señales el primer ángulo es siempre cero. Para este tipo de señal el conjunto de primeras aproximaciones también debe ser mayor a  $0^\circ$  y menor a  $90^\circ$ , se debe proporcionar en orden ascendente y sus valores no deben repetirse. En este caso, el método de Newton siempre llega a converger, sin importar si el número de armónicos para eliminar es par o impar, consecuente o no.

Una señal bipolar con cinco pulsos por semiciclo se presenta en la figura 5.3. En esta señal se han calculado los ángulos de disparo para la eliminación de los armónicos 3º, 5º, 7º y 9º. Como sucedió con la señal unipolar, los ángulos de disparo corresponden al primer cuarto de periodo y el resto de los ángulos de disparo se deben determinar tomando en cuenta la simetría de la señal.



### 5.3. Eliminación de armónicos. Señal bipolar.

Para la señal bipolar determinada la amplitud pico de la componente fundamental es de 103.11% con respecto a la tensión de alimentación de cd, mientras que la distorsión armónica total es del 93.79%. El factor de distorsión de primer orden es del 5.75% y el de segundo orden del 0.44%. La amplitud de los primeros 24 armónicos se puede observar en la tabla 5.2, en donde la amplitud pico de cada armónico tiene como referencia la tensión de alimentación y se presenta como un porcentaje de dicha tensión.

Bn(03): 000.00%	Bn(05): 000.00%	Bn(07): 000.00%	Bn(09): 000.00%
Bn(11): 029.79%	Bn(13): 056.33%	Bn(15): 036.76%	Bn(17): 004.21%
Bn(19): 000.24%	Bn(21): 001.05%	Bn(23): 020.23%	Bn(25): 020.02%
Bn(27): 010.41%	Bn(29): 023.28%	Bn(31): 006.63%	Bn(33): 002.83%
Bn(35): 015.70%	Bn(37): 008.91%	Bn(39): 008.50%	Bn(41): 000.78%
Bn(43): 015.22%	Bn(45): 005.18%	Bn(47): 010.78%	Bn(49): 003.44%

Tabla 5.2. Amplitud de los armónicos en una señal bipolar.

#### 5.4. Comparación de las señales unipolares y bipolares

Al hacer una comparación de los dos tipos de señales utilizadas para eliminar armónicos se hace notar que la señal unipolar es más eficiente, pues requiere de un pulso menos por semiciclo que la señal bipolar para eliminar el mismo número de armónicos. Por otro lado, la ventaja que tiene la señal bipolar sobre la unipolar es que siempre es posible la eliminación de armónicos, sin importar que su número sea par y los armónicos consecuentes, caso para el cual la señal unipolar falla. Sin embargo, esta última deficiencia de la señal unipolar no es algo que le reste importancia con respecto a la señal bipolar, pues la primera tiene mejores características tanto en distorsión armónica total como en factores de distorsión, a esto hay que agregar que los tiempos muertos son más fáciles de implementar sin necesidad de modificar la señal teórica calculada.

---

# **CAPÍTULO 6**

---

**FILTROS**

## Introducción

El uso de técnicas de modulación de ancho de pulso y de métodos de eliminación de armónicos minimizan o anulan la amplitud de los armónicos de menor orden, pero aumentan la amplitud de los armónicos de orden superior, y con ello a menudo se incrementa la distorsión armónica total. Sin embargo, el aumento de la amplitud de los armónicos mencionados no representa un problema muy significativo pues pueden ser fácilmente eliminados con el uso de filtros pasivos en la salida del inversor de potencia. Los filtros usados son generalmente de segundo orden aunque su estructura depende del tipo de señal que se maneje.

### 6.1. Tipos de filtros utilizados en inversores de potencia

Los filtros que se utilizan en los sistemas inversores de potencia y la forma en la que se elige a sus elementos depende en gran medida de la señal que se desea filtrar. Cuando la señal tiene un contenido de armónicos de orden menor muy alto, como en el caso de un inversor de señal cuadrada, los filtros se diseñan para resonar en la frecuencia fundamental de la señal de salida, tales filtros tienen el inconveniente de tener elementos de valor muy elevado para frecuencias bajas, por lo mismo su uso es práctico sólo cuando la frecuencia fundamental es mayor a 200 Hz [9]. Con el uso de técnicas de modulación y eliminación los filtros ya no se usan para resonar a la frecuencia fundamental y por lo mismo el valor de sus elementos disminuye haciendo su construcción más fácil.

El orden de los filtros utilizados es normalmente menor al tercero. Se busca en todo momento la simplicidad en su estructura. Los cinco tipos de filtros analizados se listan a continuación:

1. Filtro C (primer orden).
2. Filtro L (primer orden).
3. Filtro LC serie (segundo orden).
4. Filtro LC paralelo (segundo orden).
5. Filtro LC serie-paralelo (tercer orden).

### 6.1.1. Filtro C

La figura 6.1 muestra la estructura del filtro capacitivo. Este filtro está compuesto sólo por un capacitor, aunque la impedancia de salida del convertidor al que está conectado, representada por  $r$ , también forma parte de él ya que sin ella el filtro sería inútil. No es un filtro eficiente y generalmente requiere de valores de capacitancia grandes.

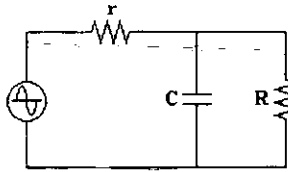


Figura 6.1. Filtro capacitivo.

El análisis en frecuencia de este filtro es simple y está centrado en su función de transferencia mostrada en la ecuación 6.1. A partir de ésta se obtiene la expresión para la respuesta en frecuencia al sustituir  $s$  por  $j\omega$  llegando a la ecuación 6.2. El módulo y el ángulo de defasamiento de las señales que atraviesan el filtro quedan determinadas por las ecuaciones 6.3 y 6.4 respectivamente.

$$H(s) = \frac{R}{rRCs + R + r} \quad (6.1)$$

$$H(j\omega) = \frac{R}{jrRC\omega + R + r} \quad (6.2)$$

$$H(j\omega) = \frac{R}{\left[ (rRC\omega)^2 + (r + R)^2 \right]^{1/2}} \quad (6.3)$$

$$\phi(\omega) = -\arctan \frac{rRC\omega}{r + R} \quad (6.4)$$

La frecuencia de transición o corte está dada por la ecuación 6.5, misma que proporciona el valor de la frecuencia para la cual la ganancia del filtro a disminuido en 3 dB a partir de su ganancia máxima.

$$\omega_T = \frac{\left[ 2 - \left( 1 + \frac{r}{R} \right)^2 \right]^{1/2}}{rC} \quad (6.5)$$

### 6.1.2. Filtro L

La figura 6.2 muestra la estructura del filtro inductivo que está compuesto por un inductor y la carga. El valor del inductor tiende a ser alto por lo que su uso no es del todo recomendable.

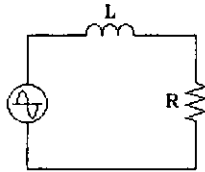


Figura 6.2. Filtro inductivo.

Al igual que para el caso del filtro capacitivo su análisis es sencillo. La función de transferencia del filtro es la ecuación 6.6 y a partir de ella se obtiene la respuesta en frecuencia al sustituir  $s$  por  $j\omega$  llegando a la ecuación 6.7. El módulo y el ángulo de defasamiento para las señales que atraviesan el filtro quedan definidas por las ecuaciones 6.8 y 6.9 respectivamente.

$$H(s) = \frac{R}{Ls + R} \quad (6.6)$$

$$H(j\omega) = \frac{R}{jL\omega + R} \quad (6.7)$$

$$H(j\omega) = \frac{R}{\left[ (L\omega)^2 + R^2 \right]^{1/2}} \quad (6.8)$$

$$\phi(\omega) = -\arctan \frac{L\omega}{R} \quad (6.9)$$

La frecuencia de transición o corte se obtiene mediante la ecuación 6.10, misma que proporciona el valor de la frecuencia para la cual la ganancia del filtro a disminuido en 3 dB a partir de su ganancia máxima.

$$\omega_r = \frac{R}{L} \quad (6.10)$$

### 6.1.3. Filtro LC serie

El filtro LC serie de la figura 6.3 tiene su principal aplicación en inversores resonantes [4], [9]. En un inversor resonante en serie, un inductor y un capacitor son colocados en serie con la carga. Los interruptores del inversor producen una señal de tensión periódica cuadrada, y la combinación de inductancia y capacitancia es seleccionada de tal forma que la frecuencia de resonancia es la misma que la frecuencia de conmutación de los interruptores.

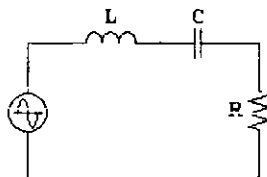


Figura 6.3. Filtro LC serie.

El análisis comienza considerando la respuesta en frecuencia del circuito RLC de la figura 6.3. La ecuación 6.11 es la función de transferencia del sistema, su respuesta en frecuencia está dada por la ecuación 6.12 y las ecuaciones 6.13 y 6.14 proporcionan, respectivamente, el módulo y el ángulo de defasamiento de la respuesta en frecuencia.



$$H(s) = \frac{R}{Ls + \frac{1}{Cs} + R} \quad (6.11)$$

$$H(s) = \frac{R}{jL\omega - j\frac{1}{C\omega} + R} \quad (6.12)$$

$$H(j\omega) = \frac{R}{\left[ \left( L\omega - \frac{1}{C\omega} \right)^2 + R^2 \right]^{\frac{1}{2}}} \quad (6.13)$$

$$\phi(\omega) = -\arctan \frac{LC\omega^2 - 1}{RC\omega} \quad (6.14)$$

Existe una característica muy importante en este circuito llamada frecuencia de resonancia. Cuando el sistema está en resonancia, las impedancias del capacitor y del inductor se cancelan y la carga aparece sólo como un resistor. Matemáticamente, la frecuencia de resonancia está dada por la ecuación 6.15.

$$\omega_r = \frac{1}{LC} \quad (6.15)$$

Además de la frecuencia de resonancia, una descripción más completa del sistema incluye al ancho de banda  $B$  y al factor de calidad  $Q$  del filtro. Debido a la respuesta típica de un filtro en resonancia, mostrada en la figura 6.4, existen dos frecuencias de corte, siendo  $\omega_l$  la frecuencia de corte inferior y  $\omega_u$  la frecuencia de corte superior. A la diferencia entre estas dos frecuencias, ecuación 6.16, se le denomina ancho de banda. La ecuación 6.17 define también a  $B$ , pero en función de los elementos del filtro LC serie.

$$B = \omega_u - \omega_l \quad (6.16)$$

$$B = \frac{R}{L} \quad (6.17)$$

Por otro lado, el factor de calidad es una medida de la selectividad del filtro y se define con la ecuación 6.18 como la razón de la frecuencia de resonancia al ancho de

banda. Mientras mayor sea el factor de calidad  $Q$ , el circuito será más selectivo en torno a la frecuencia de resonancia, lo cual se muestra en forma gráfica en la figura 6.5.

$$Q = \frac{1}{R} \cdot \frac{L}{C} \quad (6.18)$$

Diseñar circuitos con altos factores de calidad requiere de valores altos de inductancia y valores bajos de resistencia y, puesto que los inductores prácticos incluyen significativas resistencias, es difícil y costoso diseñarlos.

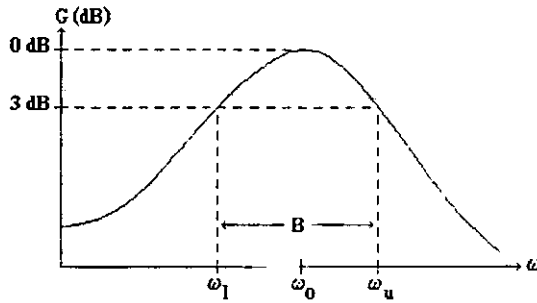


Figura 6.4. Respuesta típica de un filtro en resonancia.

Si el tercer armónico y algunos otros de orden superior son removidos, la tensión a través de la carga mejora, y permite también el diseño de un filtro con un factor de calidad menor con un inductor más pequeño.

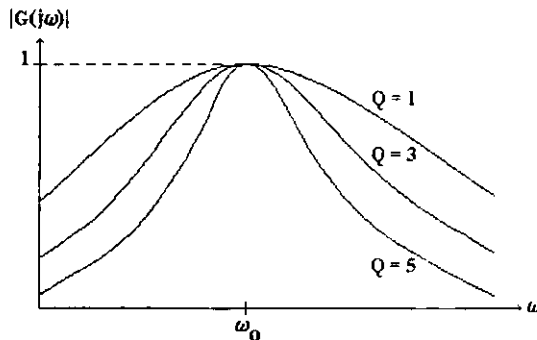


Figura 6.5. Respuesta en frecuencia del filtro LC serie.

### 6.1.4. Filtro LC paralelo

La figura 6.6 es un filtro LC paralelo [4], [9], llamado así por el capacitor colocado en paralelo con el resistor de carga. Este filtro es más flexible que el LC serie, ya que puede utilizarse para resonar a la frecuencia fundamental del inversor si la señal de salida es cuadrada, o bien, sin resonar a la frecuencia fundamental para eliminar armónicos de orden superior cuando la señal de salida del inversor ha sido modulada para eliminar armónicos de orden inferior.

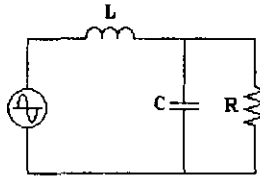


Figura 6.6. Filtro LC paralelo.

Los parámetros de descripción de este filtro son similares a los del filtro LC serie. La función de transferencia del filtro está dada por la ecuación 6.19. La respuesta en frecuencia está expresada por la ecuación 6.20 de donde se derivan las ecuaciones 6.21 y 6.22 para el módulo y el ángulo de defasamiento respectivamente.

$$H(s) = \frac{R}{RLCs^2 + Ls + R} \quad (6.19)$$

$$H(j\omega) = \frac{R}{jL\omega - RLC\omega^2 + R} \quad (6.20)$$

$$|H(j\omega)| = \frac{R}{\left[ (R - RLC\omega)^2 + (L\omega)^2 \right]^{1/2}} \quad (6.21)$$

$$\phi(\omega) = -\arctan \frac{L\omega}{R - RLC\omega} \quad (6.22)$$

Para este filtro también es posible hablar de frecuencia de resonancia  $\omega_r$  y del factor de calidad  $Q$ , aunque debido al comportamiento del filtro mostrado en la figura 6.6, para  $Q < 1$  conviene más hablar de una frecuencia de corte y no de una de resonancia. La frecuencia de resonancia del filtro está dada por la ecuación 6.23 y la del ancho de banda, siempre que sea pertinente hablar de él, por la ecuación 6.24. De las ecuaciones anteriores se puede obtener la ecuación 6.25 para el factor de calidad.

$$\omega_r = \frac{1}{LC} \quad (6.23)$$

$$B = \frac{1}{RC} \quad (6.24)$$

$$Q = R \frac{C}{L} \quad (6.25)$$

En condiciones sin carga  $R = \infty$  y  $Q = \infty$ , por lo que la tensión de salida es una función de la carga y puede resultar muy alta si no se eleva la frecuencia de operación. Si debido a una falla en la carga el capacitor  $C$  queda en corto circuito, la corriente quedará limitada por el inductor  $L$  y se puede decir que este es un inversor a prueba de cortos circuitos en forma natural, y es deseable para aplicaciones con requisitos severos en condiciones de falla.

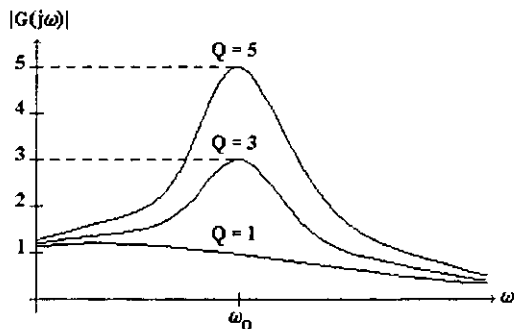


Figura 6.7. Respuesta en frecuencia del filtro LC paralelo.

### 6.1.5. Filtro LC serie-paralelo

El filtro LC serie-paralelo [4], [9] tiene un capacitor en serie y un capacitor en paralelo con la carga. A pesar de ser un filtro de tercer orden, su análisis es similar a los filtros LC discutidos anteriormente. El filtro se presenta en la figura 6.8.

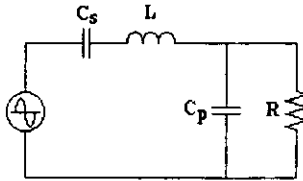


Figura 6.8. Filtro LC serie-paralelo.

La función de transferencia del sistema está dada por la ecuación 6.26, en donde se ha considerado que ambos capacitores son iguales. La respuesta en frecuencia se encuentra a través de la ecuación 6.27, de donde provienen las ecuaciones que definen al módulo y al ángulo de defasamiento que corresponden a las ecuaciones 6.28 y 6.29 respectivamente.

$$H(s) = \frac{RCs}{RC^2Ls^3 + LCs^2 + 2RCs + 1} \quad (6.26)$$

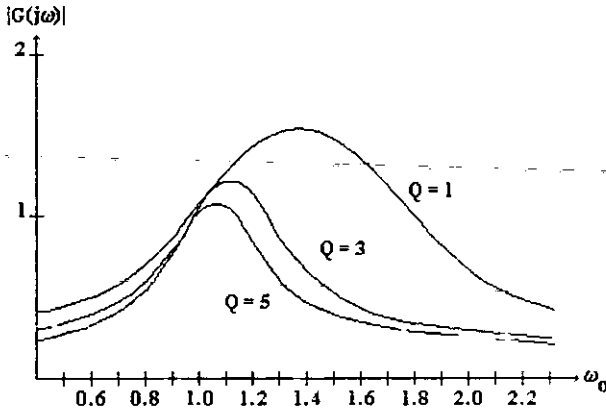
$$H(j\omega) = \frac{jRC\omega}{-jRC^2L\omega^3 - LC\omega^2 + 2jRC\omega + 1} \quad (6.27)$$

$$H(j\omega) = \frac{RC\omega}{\left[ (1 - LC\omega^2)^2 + (2RC\omega - RC^2L\omega^3) \right]^{1/2}} \quad (6.28)$$

$$\phi(\omega) = \frac{\pi}{2} - \arctan \frac{2RC\omega - RC^2L\omega^3}{1 - LC\omega^2} \quad (6.29)$$

Al ser similar el comportamiento de este sistema al de los sistemas de segundo orden se pueden definir en él características tales como frecuencia de resonancia, ancho de banda y factor de calidad, aunque se debe tener cuidado pues el comportamiento es

semejante mas no idéntico, como se muestra en la figura 6.9, donde la respuesta en frecuencia es función de la frecuencia de resonancia determinada matemáticamente.



**Figura 6.9.** Respuesta en frecuencia del filtro LC serie-paralelo.

Las ecuaciones 6.30, 6.31 y 6.32 corresponden a las expresiones matemáticas para determinar la frecuencia de resonancia, ancho de banda y factor de calidad respectivamente, aunque no son siempre exactas, pues el filtro no es de segundo orden.

$$\omega_r = \frac{1}{LC} \quad (6.30)$$

$$B = \frac{R}{L} \quad (6.31)$$

$$Q = \frac{1}{R} \frac{L}{C} \quad (6.32)$$

El nombre de filtro serie-paralelo sugiere la combinación de las características de los filtros serie y paralelo de segundo orden, y es cierto, este filtro reúne tales características, pero tiene la desventaja de utilizar un mayor número de componentes.

## 6.2. Comparación de los filtros

La eficacia de un filtro capacitivo depende principalmente del elemento en serie que tenga, ya que mientras mayor sea el elemento en serie menor será el valor del capacitor utilizado. En el caso de un convertido cd-ca el elemento en serie es la impedancia de salida del convertidor, la cual es generalmente menor a  $1 \Omega$ . Con este valor de impedancia para que el capacitor tenga un efecto notable sobre la señal de salida es necesario un valor elevado de capacitancia que, al estar en paralelo con la carga, incrementa considerablemente el consumo de corriente y disminuye la eficiencia del inversor.

Un filtro L de primer orden tiene un mejor funcionamiento que un filtro C pues, al ser un elemento que se conecta en serie, no incrementa el consumo de corriente. Sin embargo, su uso tiene la desventaja de requerir un valor grande de inductancia que disminuye la ganancia del inversor.

En el caso de los filtros LC serie y LC serie-paralelo el principal inconveniente es que su respuesta en frecuencia es más adecuada para inversores resonantes y no para aquellos que utilizan técnicas de modulación de ancho de pulso y eliminación de armónicos en donde se prefiere el uso de un filtro de paso bajo. Otro obstáculo en su utilización es que debido a los grandes valores requeridos para resonar a frecuencias bajas, su uso sólo se recomienda para frecuencias superiores a 200 Hz.

Para finalizar, sólo resta comentar las características del filtro LC paralelo. En este filtro la atenuación de los armónicos es compartida por el inductor y el capacitor, lo que da como resultado una disminución del valor de los elementos en comparación con los filtros de primer orden L y C. Por otro lado, al utilizar la red LC paralela como un filtro de paso bajo con técnicas de modulación y eliminación, la frecuencia de corte puede situarse lo suficientemente lejos de la frecuencia fundamental como para permitir el uso de componentes más pequeños que aquellos empleados en los filtros resonantes. El uso de componentes más pequeños significa, entre otras cosas, una mejor respuesta ante los cambios de carga y una mejor eficiencia del filtro, pues se limita la corriente que consume el capacitor a la frecuencia fundamental.

---

# **CAPÍTULO 7**

---

**ELEMENTOS PARA LA EVALUACIÓN**



## Introducción

La simulación tiene como finalidad la comprobación de la teoría al permitir observar resultados sin la necesidad de construir modelos reales. Entre las ventajas que ofrece, la principal es la facilidad para intercambiar los parámetros del modelo teórico.

Por otro lado, la construcción de pequeños módulos de evaluación permite también la comprobación de la teoría, pero sin las complicaciones que siempre genera trabajar con un proyecto que puede ser costoso y delicado.

### 7.1. El programa de simulación

La computadora como elemento de diseño ofrece grandes beneficios. Actualmente, los entornos de programación han evolucionado, de tal forma, que realizar programas de simulación no es difícil. Para diseñar un programa de simulación que permita analizar técnicas de modulación en inversores monofásicos el elemento esencial es el análisis de Fourier, y una plataforma de programación que, para ir con la tendencia actual, tenga componentes visuales.

Para la creación del programa se eligió el desarrollador de aplicaciones Delphi 3, que es un sistema que hace uso del lenguaje de programación Object Pascal y que permite desarrollar fácilmente una interfaz visual. La versión final del programa se encuentra en la forma de un archivo ejecutable denominado *Marcela.exe*.

El programa de simulación está estructurado en ventanas, en cada una de las cuales existe un menú que permite acceder a las ventanas restantes. Adicionalmente, cada ventana está diseñada para realizar una acción cuyas características se pueden especificar por medio de datos numéricos y selección de opciones múltiples. En total son cuatro las ventanas que componen al programa y que son enumeradas a continuación.

1. La ventana de entrada.
2. La ventana de modulación de ancho de pulso.

3. La ventana de eliminación de armónicos.
4. La ventana de filtros.

### 7.1.1. La ventana de entrada

Al ejecutar el programa se abre la ventana de entrada. Esta ventana, mostrada en la figura 7.1, funge también como área de graficación, en ella se pueden observar las formas de onda de las señales moduladas y de las señales generadas para la eliminación de armónicos, así como las señales obtenidas después del filtrado.

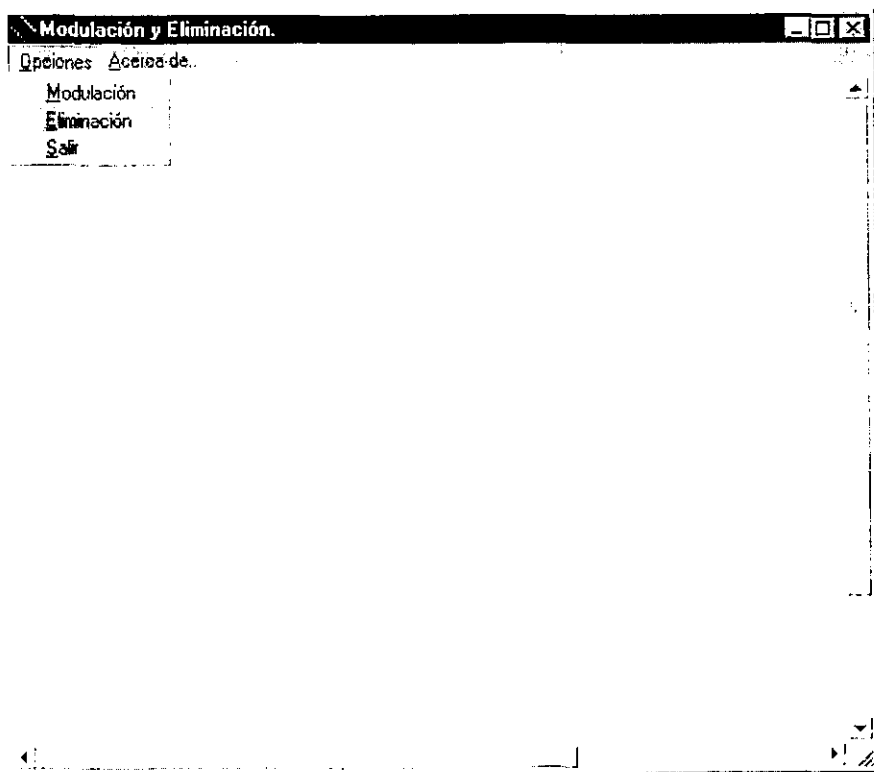


Figura 7.1. Ventana de entrada.

La ventana de entrada cuenta con el menú *Opciones* gracias al cual se puede acceder a las ventanas de modulación de ancho de pulso y de eliminación de armónicos, así como salir del sistema.

### 7.1.2. La ventana de modulación de ancho de pulso

La figura 7.2 muestra la ventana de modulación de ancho de pulso. En ella existe también el menú *Opciones* con el que se pueden abrir las ventanas de filtros, de eliminación de armónicos, ocultar la ventana de modulación y salir del sistema.

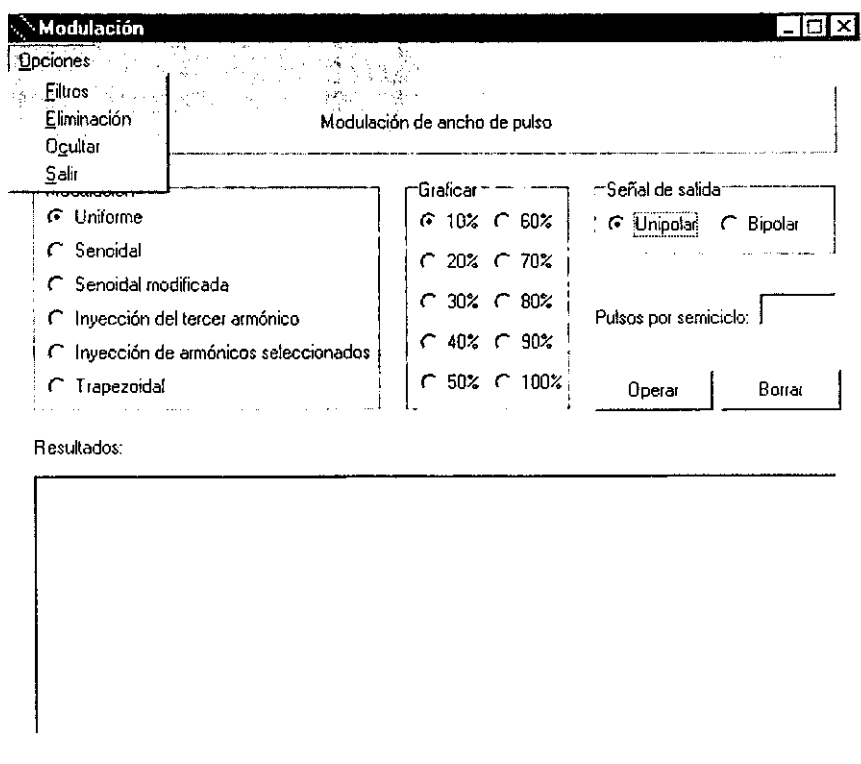


Figura 7.2. Ventana de modulación de ancho de pulso.

En la ventana de modulación de ancho de pulso se pueden seleccionar seis tipos de modulación. Los seis tipos de modulación son:

1. Modulación de ancho de pulso uniforme.
2. Modulación senoidal.
3. Modulación senoidal modificada.
4. Modulación por inyección del tercer armónico.
5. Modulación por inyección de armónicos seleccionados.
6. Modulación trapezoidal.

En todos los tipos de modulación se puede seleccionar uno de dos tipos de señales de salida:

1. Unipolar.
2. Bipolar.

También se puede seleccionar el índice de modulación que se quiere graficar desde el 5% hasta el 100% con incrementos del 5%. El número de pulsos por semiciclo se introduce por medio de un pequeño recuadro de texto que está antecedido por la leyenda *Pulsos por semiciclo*.

Existen dos botones en la ventana. Con el botón *Operar* se inicia la ejecución del programa una vez que se ha seleccionado la técnica de modulación, el tipo de señal de salida y el número de pulsos por semiciclo. Con el botón *Borrar* se limpian todos los recuadros de texto en la ventana.

En el recuadro de texto más grande, precedido por la palabra *Resultados*, se pueden leer todos los datos generados por el sistema, tales como índice de modulación, factores de distorsión de primer y segundo orden, distorsión armónica total, ángulos de disparo y amplitud de la componente fundamental y de sus armónicos.

Para observar resultados gráficos es necesario trasladarse a la ventana de entrada. Lo que gráficamente se observará será la señal antes y después del filtro del inversor. Adicionalmente, y en forma numérica, en la misma pantalla se puede consultar la distorsión armónica total de las señales de entrada y salida así como sus respectivas amplitudes máximas.

### 7.1.3. La ventana de eliminación de armónicos

Al igual que en las ventanas anteriores, en la ventana de eliminación de armónicos se cuenta con el menú *Opciones* que permite abrir otras ventanas, ocultar la ventana actual y salir del sistema.

En lo concerniente a la eliminación de armónicos, se puede seleccionar entre los dos tipos de señales en los que se quieren eliminar:

1. Unipolares.
2. Bipolares.

El número de armónicos que se quiere eliminar se introduce en un recuadro de texto, mientras que los armónicos específicos a eliminar se introducen en un arreglo con tantas localidades como armónicos se desea anular. Ambos, el recuadro de texto y el arreglo, se encuentran etiquetados de tal manera que es fácil identificarlos.

Por otra parte, el programa detrás de la ventana de eliminación de armónicos es un conjunto de métodos numéricos para resolver ecuaciones simultáneas no lineales y está basado en los métodos de Newton, Kramer y en un procedimiento recursivo para invertir matrices. Como conjunto de métodos numéricos, es necesario proporcionarle siempre primeras aproximaciones a las soluciones del sistema, tales aproximaciones se proporcionan a través de un arreglo de texto que tiene el mismo número de localidades que armónicos a eliminar y que está señalado como *Aproximaciones a los ángulos de disparo*. Estas primeras aproximaciones deben de cumplir con los requisitos de encontrarse entre  $0^\circ$  y  $90^\circ$ , no estar repetidas y ser proporcionadas en forma ascendente en cuanto a su valor.

Existen tres botones en la ventana de eliminación. Los botones *Operar* y *Borrar* tienen la misma función que en la ventana de modulación, en tanto que el botón *Gráfica* tiene como finalidad mostrar la ventana de entrada con las señales generadas.

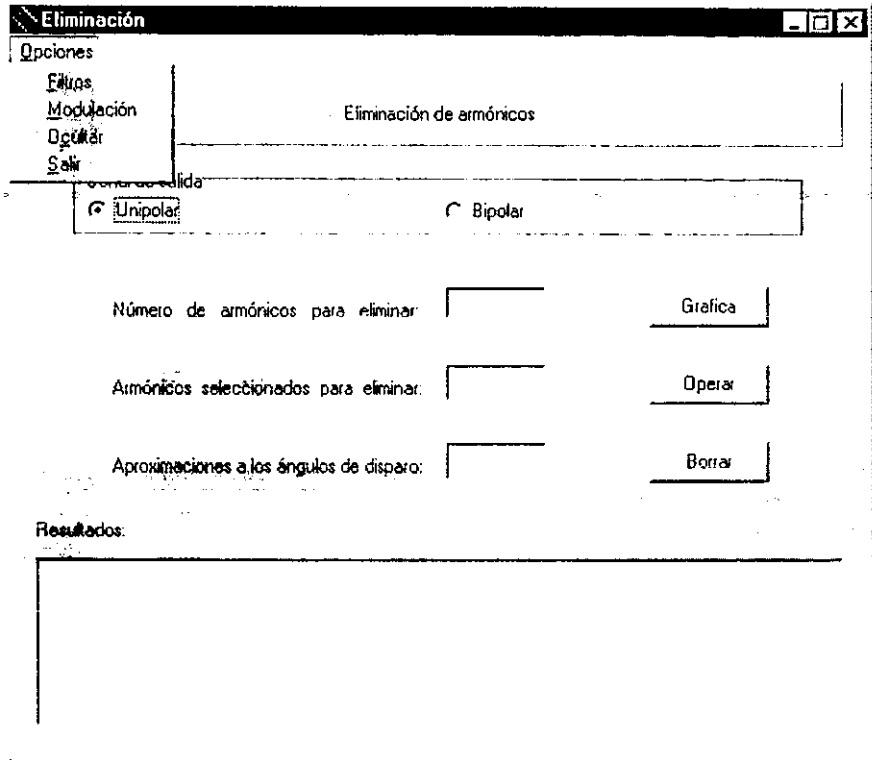


Figura 7.3. Ventana de eliminación de armónicos.

Finalmente, el recuadro de texto más grande muestra resultados numéricos similares a los mostrados en la ventana de modulación de ancho de pulso.

#### 7.1.4. La ventana de filtros

En la ventana de filtros se encuentran disponibles seis tipos de filtros pasivos, en cada uno de los cuales es posible cambiar el valor de sus componentes por medio de cuadros de texto. Los seis tipos de filtros disponibles son:

1. Filtro RC.
2. Filtro LR (R es la salida).
3. Filtro RL (L es la salida).
4. Filtro LC serie.
5. Filtro LC paralelo.
6. Filtro LC serie-paralelo.

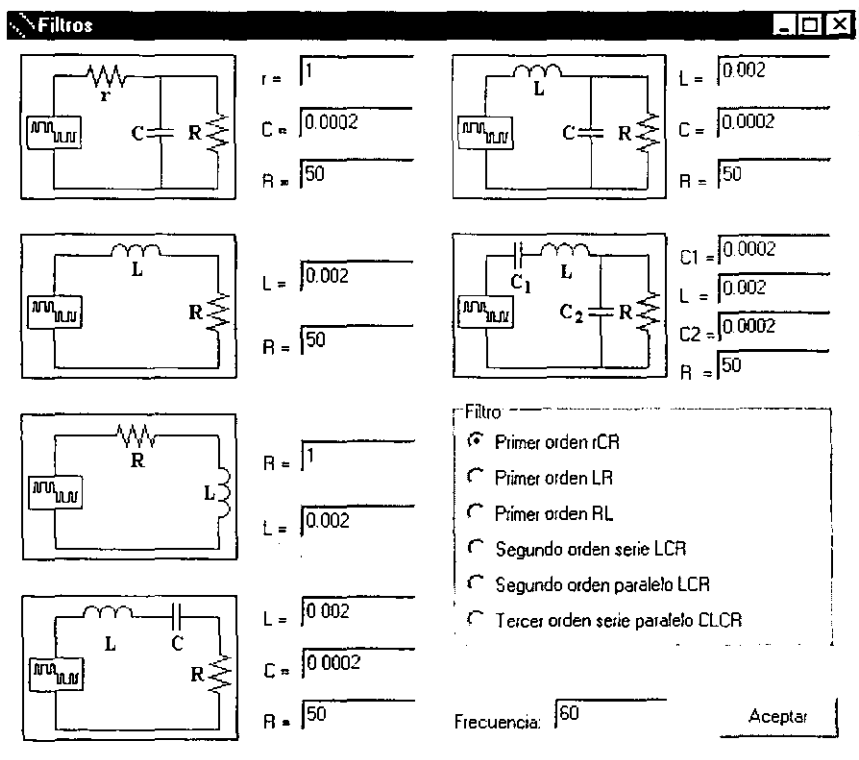


Figura 7.4. Ventana de filtros.

El filtro seleccionado es al que se someterá la señal de salida tanto de la ventana de modulación como de la ventana de eliminación. Los filtros son nombrados en forma descendente y de izquierda a derecha. Los valores de los elementos de cada filtro pueden ser cambiados utilizando los espacios a la derecha de los mismos. Un cuadro de texto es utilizado en la esquina inferior derecha a fin de variar la frecuencia fundamental de las señales utilizadas. El botón *Aceptar* cierra la ventana de filtros conservando los valores introducidos y el tipo de filtro seleccionado.

## **7.2. El módulo de evaluación**

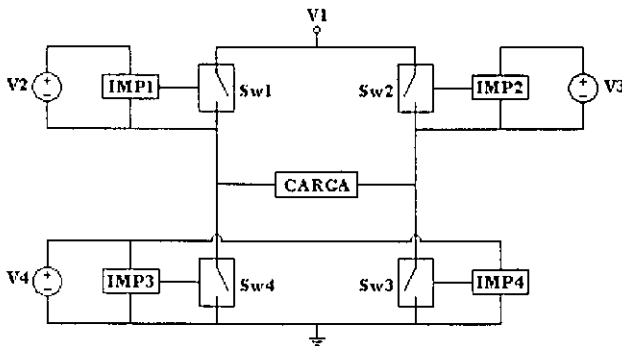
El módulo de evaluación tiene como finalidad proporcionar los elementos necesarios para observar el desempeño de las técnicas de modulación y eliminación en forma directa, así como permitir fácilmente la modificación de tales técnicas.

El módulo de evaluación se encuentra constituido por dos partes. La primera es la topología inversora, estrictamente el convertidor cd-ca, y la segunda es el circuito de control o generador de señales de excitación. De alguna manera el filtro también forma parte del módulo de evaluación, pero su discusión se deja para un apartado posterior.

### **7.2.1. Selección de la topología inversora**

De las tres topologías inversoras existentes se eligió la de puente completo, por ser posible implementar en ella tanto señales unipolares como bipolares. Este tipo de topología hace uso de cuatro interruptores de estado sólido y cuatro fuentes de tensión de corriente directa. Una fuente es usada para alimentar al inversor y las tres restantes se requieren para excitar a los interruptores. Para excitar a cada uno de los interruptores es necesaria una etapa impulsora, la cual suministra la corriente necesaria para encender los dispositivos de potencia además de proporcionar aislamiento con el origen de las señales de excitación. La figura 7.5 muestra, en forma simplificada, todos los elementos de los que se ha hablado hasta ahora.





**Figura 7.5. Configuración inversora seleccionada para el módulo de evaluación.**

### 7.2.2. Generación de las señales de excitación

Para la generación de las señales de excitación se optó por el grabado del patrón de conmutación en una memoria reutilizable de sólo lectura con borrado por luz ultravioleta. El uso de una memoria proporciona una ventaja, para cambiar el tipo de señales de excitación sólo es necesario cambiar la memoria o regrabarla, sin importar si se hace uso de modulación senoidal, trapezoidal o cualquier otra. De otra manera, generar las señales analógicamente requiere de circuitos más complejos y diferentes para cada tipo de modulación.

El generador de señales de excitación está compuesto esencialmente por un contador binario ascendente de 12 bits y la memoria de sólo lectura. Para aumentar el manejo de corriente, las salidas de la memoria se conectan a transistores que operan optoacopladores para proporcionar aislamiento con la configuración inversora. La figura 7.6 es un diagrama de bloques del generador de señales de excitación en donde se muestran sus elementos principales.

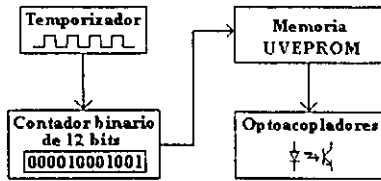


Figura 7.6. Generador de señales de excitación.

### 7.2.3. Implementación del módulo de evaluación

Para la implementación del inversor de puente completo se utilizaron cuatro transistores tipo MOSFET de matricula IRF530, no se utilizaron diodos externos porque los transistores contienen diodos parásitos en antiparalelo que funcionan como diodos de recuperación. La implementación del convertidor cd-ca del módulo de evaluación se muestra en la figura 7.7.

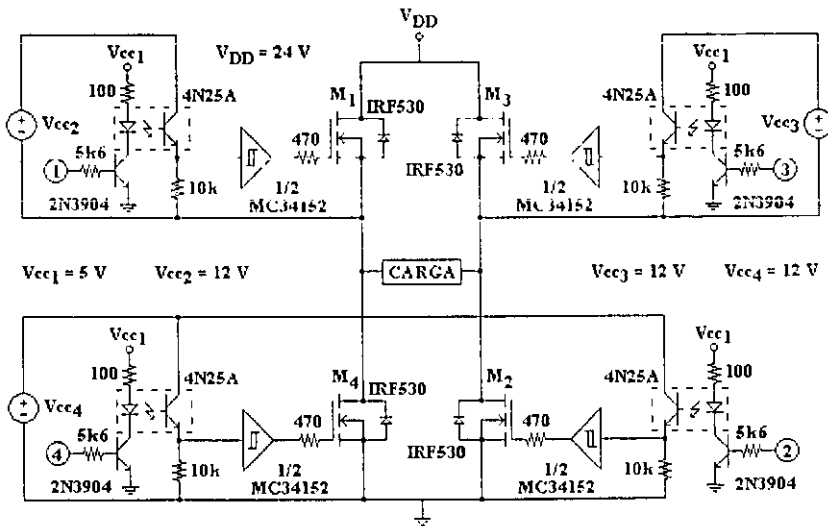


Figura 7.7. Implementación del inversor del modulo de evaluación.

El sistema de control, mostrado en la figura 7.8, está compuesto de 6 circuitos integrados, un LM555, tres 74LS191, una memoria NMC27C16B y un 74LS08. El circuito LM555 está conectado en modo astable para producir una señal de 245 kHz que se aplica a un contador de 12 bits, mismo que está constituido por los tres circuitos 74LS191 que son contadores binarios de cuatro bits conectados en configuración ascendente. Los primeros once bits del contador son utilizados para direccionar la memoria y el bit restante para determinar en donde se aplicará la salida.

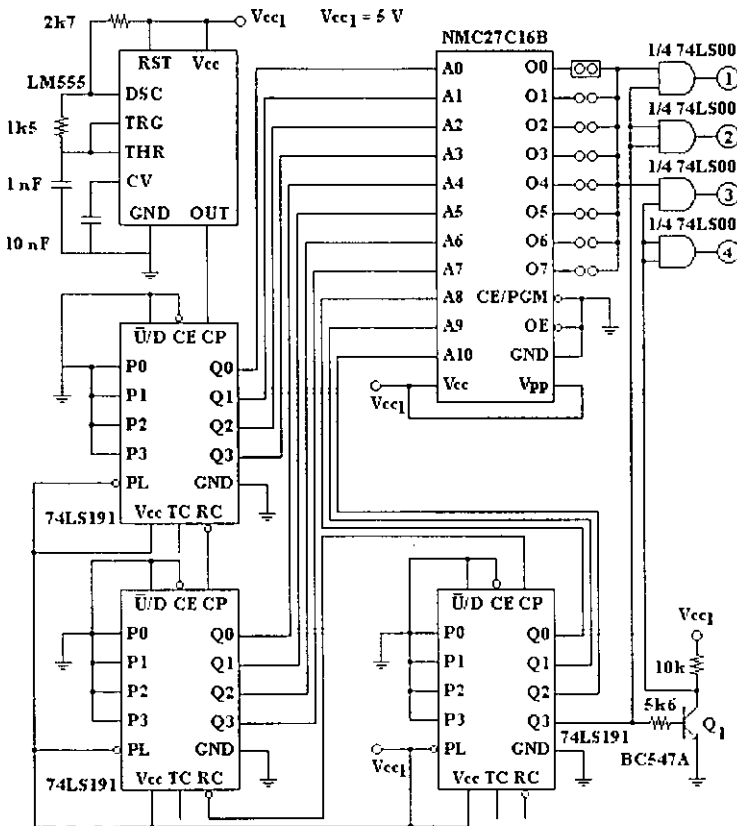


Figura 7.8. Generador de señales del módulo de evaluación.

El circuito NMC27C16B es una memoria UVEPROM con 2048 palabras de 8 bits. En cada bit se puede grabar un patrón de conmutación, por lo que en cada memoria es posible grabar 8 técnicas de modulación con índice modulador fijo. Cada patrón de conmutación contiene la señal de excitación correspondiente a un semiciclo y dado que existen 2048 localidades de memoria, la resolución de las señales es de hasta  $0.0879^\circ$ . Como se busca que la señal sea inyectada alternadamente a los pares de transistores M1-M2 y M3-M4, para tal propósito se utilizan cuatro compuertas AND que funcionan como habilitadores de señales. Dos de estas compuertas, las que van a los transistores M1 y M3, son conectadas a la memoria con una de sus entradas, las otras entradas son conectadas una al bit 12 del contador y la otra a su complemento a través del transistor Q1. De esta manera, la señal de excitación llega durante medio periodo al transistor M1 y durante el semiperiodo restante al transistor M3. A los transistores M2 y M4 se les aplican las señales del bit 12 y su complemento con las dos compuertas AND restantes, de tal manera que únicamente en el semiciclo en el que existe excitación para M1 el transistor M2 permanece encendido. Lo mismo sucede para los transistores M3 y M4.

Para lograr que las señales pasen del circuito de control a los transistores de potencia se utilizaron 4 optoacopladores 4N25A, mostrados en la figura 7.7, ya que las fuentes de las compuertas de los transistores y la del circuito de control son diferentes. Como el circuito 74LS08 proporciona poca corriente de salida para alimentar al optoacoplador, la conexión se realiza con transistores 2N3904 que sirven como amplificadores de corriente. Así, los diodos de los optoacopladores en serie con resistores de  $100 \Omega$  se conectan a los colectores de los transistores que son activados por las compuertas AND.

La salida en los optoacopladores se toma en el emisor del fototransistor en donde se conecta un resistor de  $10 \text{ k}\Omega$  y una de las entradas del circuito MC34152. Los circuitos MC34152 contienen dos dispositivos con entradas Schmitt Trigger que les confieren buena inmunidad al ruido y salidas tipo Poste Tótem que les permite suministrar o absorber una corriente de hasta un amperio. Para terminar, la salida de los MC34152 se conecta por medio de un resistor de  $470 \Omega$  a las compuertas de los transistores del puente completo del inversor.

### 7.3. Selección de los elementos del filtro de salida

Por sus características de atenuación y simplicidad se eligió al filtro LC paralelo como el más adecuado para mejorar la señal de salida del inversor. La carga utilizada es de  $33 \Omega$  totalmente resistiva.

La selección de los elementos del filtro se hace con base a la técnica de eliminación de armónicos con 5 pulsos por semiciclo, la cual contiene una distorsión armónica total de un poco más del 48%. Con esta técnica se eliminan los armónicos  $3^\circ$ ,  $5^\circ$ ,  $7^\circ$ ,  $9^\circ$ , y  $11^\circ$ , siendo el primer armónico presente el  $13^\circ$ . El método de selección de elementos consiste en considerar toda la distorsión armónica reunida en el primer armónico que existe; determinar la atenuación requerida en la frecuencia del armónico; calcular la frecuencia de corte y por último el valor de los elementos del filtro.

Como se desea una distorsión menor al 3% después del filtro, la atenuación requerida es de  $1/16$  o de  $-24.08$  dB en la frecuencia de 780 Hz. Tomando en cuenta que la atenuación de un filtro de segundo orden de paso bajo es de  $-40$  dB/década, se tiene que

$$f_B = \frac{780}{10^{-\frac{-24.08}{-40}}} = 195.06 \text{ Hz.} \quad (7.1)$$

$$\omega_B = 2\pi f_B = 1225.39 \text{ rad/s} \quad (7.2)$$

Para la implementación del filtro es conveniente que el valor del capacitor no sea muy grande ya que significa un consumo mayor de corriente. Un valor recomendado es que su reactancia a la frecuencia fundamental sea al menos 5 veces mayor que la carga con lo cual se logra que la corriente en la carga sea igual que en el inductor [7]. Sin embargo, para este caso se eligió la reactancia del capacitor como 2.5 veces la impedancia de carga a fin de minimizar el valor del inductor. Con este valor de reactancia en el capacitor la corriente en la carga es sólo un 10% menor que en el inductor, así que

$$X_C = 33(2.5) = 82.5 \Omega \quad (7.3)$$

$$C = \frac{1}{X_c \cdot \omega_0} = \frac{1}{82.5(376.99)} = 32.15 \mu F \quad (7.4)$$

Una vez conocido el valor del capacitor, el valor del inductor puede ser fácilmente determinado con la ecuación 7.5.

$$L = \frac{1}{C\omega_H^2} = \frac{1}{32.15 \times 10^{-6} (1225.39)^2} = 20.71 mH \quad (7.5)$$

Con la metodología utilizada, al considerar toda la distorsión armónica concentrada en una frecuencia, se asegura que la distorsión armónica después del filtro sea menor que la requerida. De esta manera, los valores de los elementos pueden ser redondeados en forma segura a valores inferiores. El valor final elegido para el capacitor es de 30  $\mu F$  mientras que para el inductor es de 20 mH.

#### 7.4. El grabado del patrón de conmutación en la memoria

Como consecuencia de las 2048 localidades que tiene la memoria, escribir el archivo que contiene toda la información de los 8 patrones de conmutación que se grabarán es un trabajo lento y no exento de errores. Por este motivo se hizo un programa que proporcione tal archivo para el caso de señales unipolares. El nombre del programa en su versión ejecutable es *Memoria.exe*.

La figura 7.9 muestra la pantalla del programa que, al igual que el programa de simulación, fue realizada con Delphi 3. Los datos necesarios que deben incluirse para ejecutar el programa son:

1. El número de pulsos por semiciclo de la señal para cada bit.
2. Los ángulos de disparo del primer cuarto de periodo para cada bit.
3. Un nombre para el archivo con extensión ASM.

La pantalla cuenta con dos botones. El botón *Borrar* tiene como finalidad limpiar todos los recuadros de texto y el botón *Aceptar* activa el programa una vez que se han proporcionado todos los datos.

Después de ejecutar el programa, y una vez obtenido el archivo con extensión ASM, se procede a su ensamblado con el programa IASMI1 para su posterior grabado en la memoria. Para el grabado de la memoria se utilizó un programador universal marca *E. E. Tools* modelo *Topmax*.

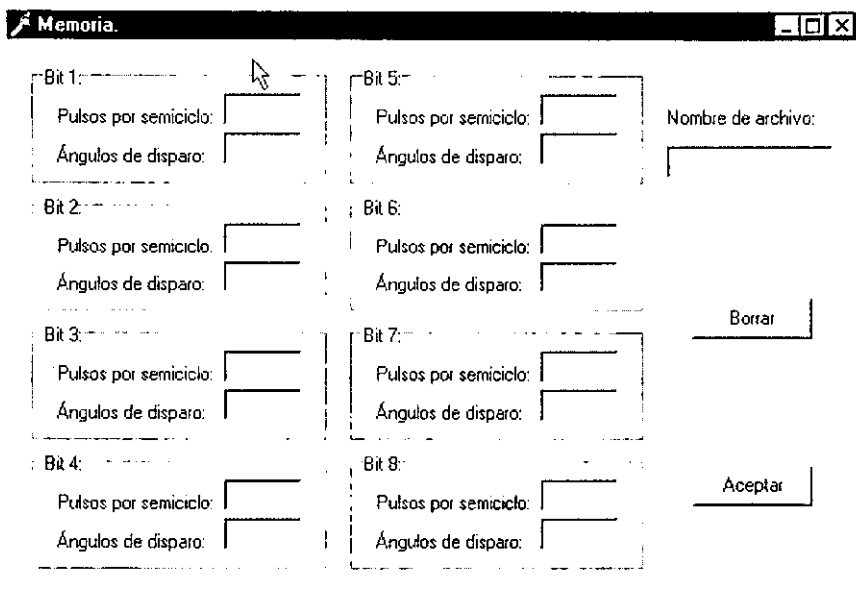


Figura 7.9. Programa para generación de archivo con patrones de conmutación.

---

# **CAPÍTULO 8**

---

**RESULTADOS**



## Introducción

Los resultados de las simulaciones y oscilogramas obtenidos con el módulo de evaluación con las distintas técnicas de modulación y eliminación son presentados en esta parte del trabajo. Las pruebas con el módulo de evaluación incluyen todas las técnicas mencionadas en el modo unipolar por considerarse más adecuadas para un inversor monofásico. La carga utilizada es puramente resistiva y es sometida a un filtro de segundo orden LC del tipo paralelo. Para mostrar como la señal de salida varía con el número de pulsos por semiciclo, las pruebas se realizaron primero con cinco pulsos y después con nueve mientras que el filtro se mantuvo fijo al igual que la fuente de alimentación.

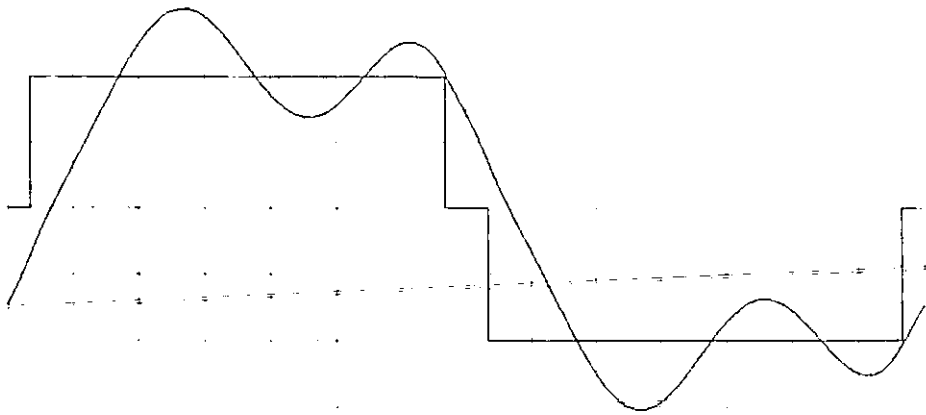
### 8.1. Simulaciones de las técnicas de modulación y eliminación

En la simulación de las técnicas de modulación y eliminación se utilizó un filtro de segundo orden LC paralelo con una carga resistiva. El valor del inductor simulado es de 20 mH y el del capacitor es de 30  $\mu$ F. La carga utilizada es un resistor de 33  $\Omega$ . Las simulaciones se realizaron para los casos de 5 y 9 pulsos por semiciclo para señales unipolares y bipolares. Se simuló también la modulación de un solo ancho de pulso, mostrada en la figura 8.1.

Los gráficos generados con el programa de simulación se muestran en las primeras 22 figuras del capítulo. Desde la figura 8.2 hasta la 8.15 se incluyen tanto las técnicas de modulación como de eliminación para el caso de 5 pulsos por semiciclo, en sus versiones unipolar y bipolar. Las siguientes 7 figuras muestran las gráficas para el caso de 9 pulsos por semiciclo sólo en la versión unipolar que es el tipo de modulación que se implementará en el módulo de evaluación.

La tabla 8.1 muestra la comparación entre las técnicas unipolares y bipolares tanto de modulación como de eliminación. La comparación se basa en dos valores de distorsión armónica, la distorsión armónica THD1 antes del filtro de salida y la distorsión armónica THD2 después del filtro. Todos los valores presentados fueron calculados con el programa de simulación para los casos de 5 y 9 pulsos por semiciclo con una modulación del 90%.

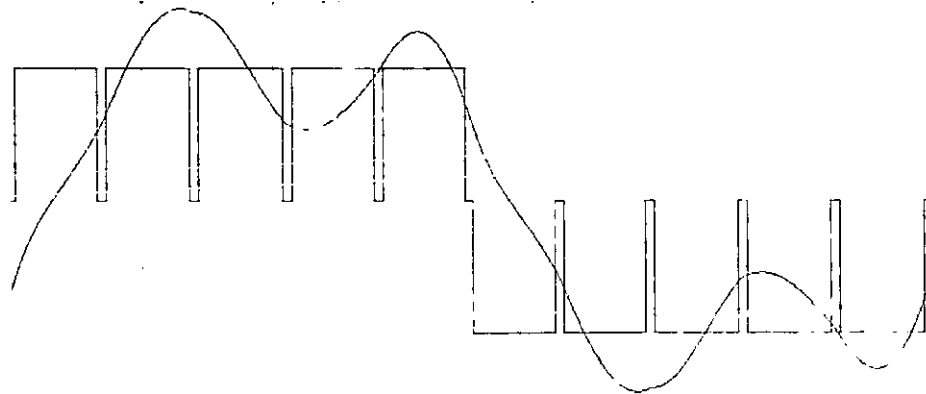
Frecuencia = 60 Hz      THD entrada = 037 17%      THD salida = 040 10 %      Amplitud máxima de salida = 152 %



Señal de entrada ——— Señal de salida ——— Amplitud de la señal de entrada = 100 %

Figura 8.1. Simulación de modulación de un solo pulso por semiciclo. Modulación al 90%.

Frecuencia = 60 Hz      THD entrada = 060 16%      THD salida = 046 77 %      Amplitud máxima de salida = 145 %



Señal de entrada ——— Señal de salida ——— Amplitud de la señal de entrada = 100 %

Figura 8.2. Simulación de UPWM unipolar con 5 pulsos por semiciclo. Modulación al 90%.

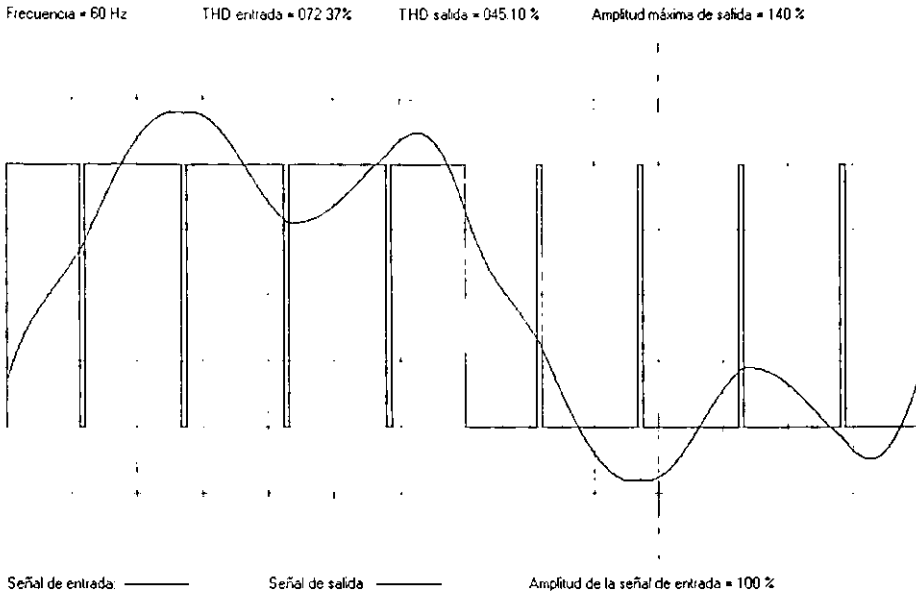


Figura 8.3. Simulación de UPWM bipolar con 5 pulsos por semiciclo. Modulación al 90%.

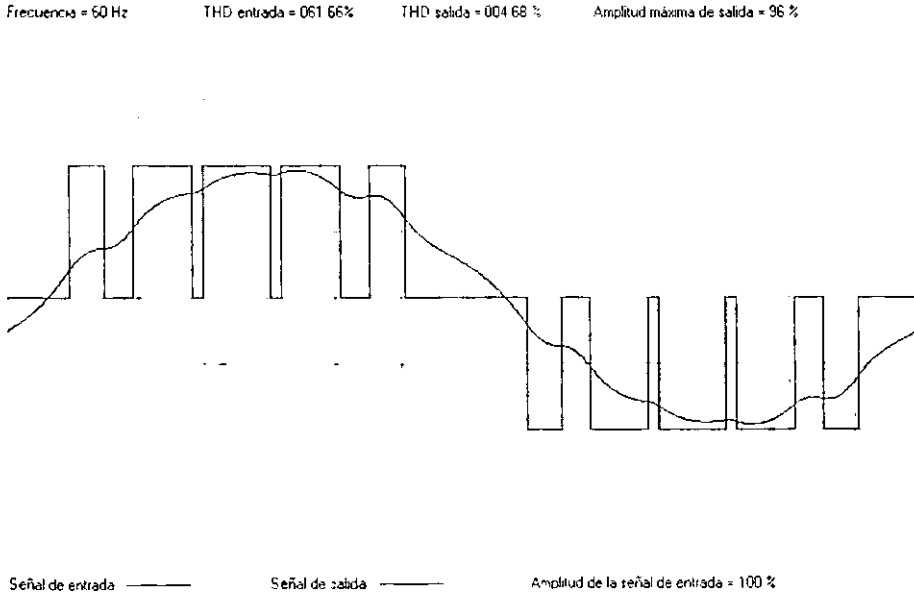
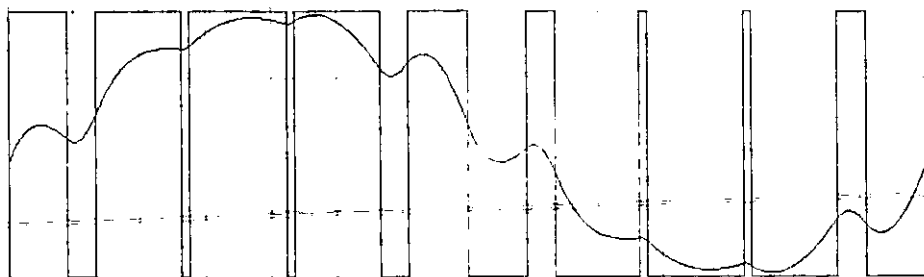


Figura 8.4. Simulación de SPWM unipolar con 5 pulsos por semiciclo. Modulación al 90%.

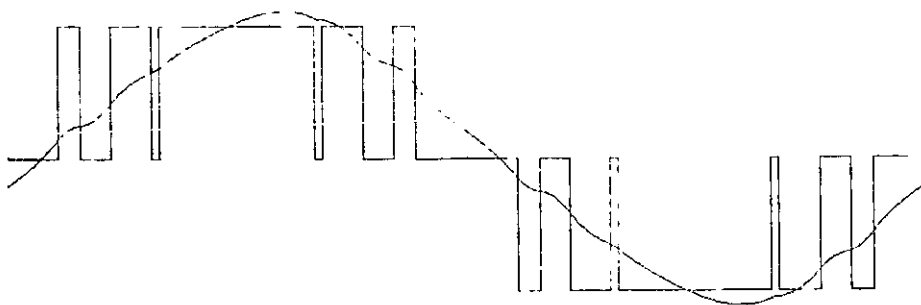
Frecuencia = 60 Hz      THD entrada = 121.26%      THD salida = 014.76%      Amplitud máxima de salida = 97%



Señal de entrada - - -      Señal de salida - - -      Amplitud de la señal de entrada = 100%

Figura 8.5. Simulación de SPWM bipolar con 5 pulsos por semiciclo. Modulación al 90%.

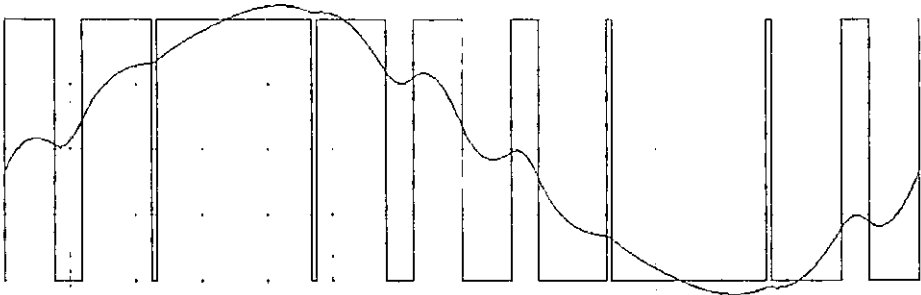
Frecuencia = 60 Hz      THD entrada = 049.96%      THD salida = 014.76%      Amplitud máxima de salida = 111%



Señal de entrada - - -      Señal de salida - - -      Amplitud de la señal de entrada = 100%

Figura 8.6. Simulación de MSPWM unipolar con 5 pulsos por semiciclo. Modulación al 90%.

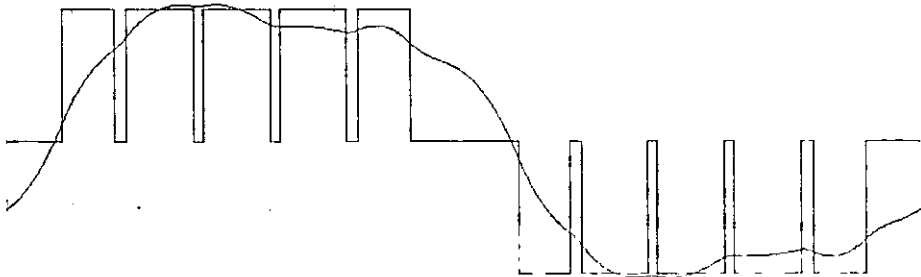
Frecuencia = 60 Hz      THD entrada = 100.76%      THD salida = 010.25 %      Amplitud máxima de salida = 111 %



Señal de entrada ——— Señal de salida: ——— Amplitud de la señal de entrada = 100 %

Figura 8.7. Simulación de MSPWM bipolar con 5 pulsos por semiciclo. Modulación al 90%.

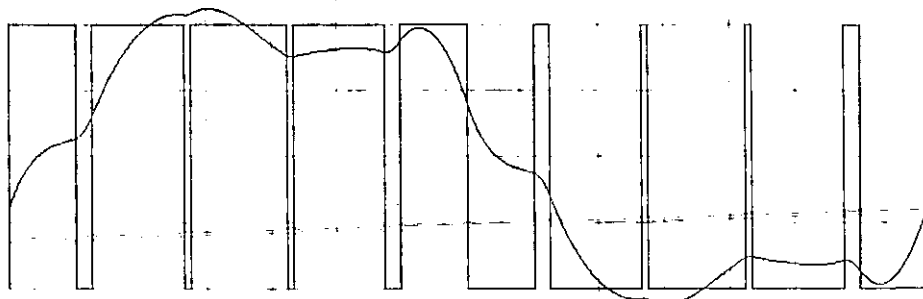
Frecuencia = 60 Hz      THD entrada = 049.85%      THD salida = 020.80 %      Amplitud máxima de salida = 104 %



Señal de entrada ——— Señal de salida: ——— Amplitud de la señal de entrada = 100 %

Figura 8.8. Simulación de THIPWM unipolar con 5 pulsos por semiciclo. Modulación al 90%.

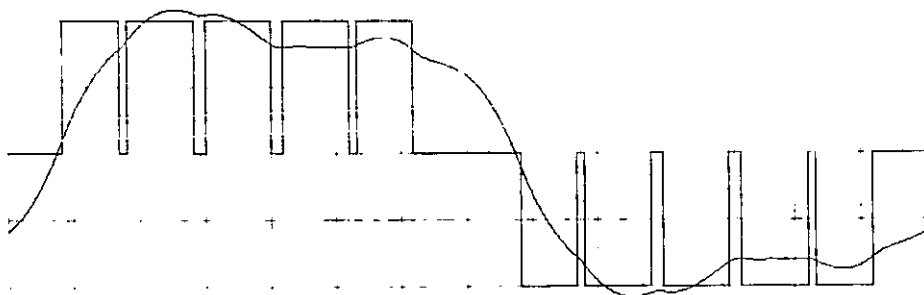
Frecuencia = 60 Hz      THD entrada = 092 53%      THD salida = 026 54 %      Amplitud máxima de salida = 112 %



Señal de entrada — — —      Señal de salida — — —      Amplitud de la señal de entrada = 100 %

Figura 8.9. Simulación de THIPWM bipolar con 5 pulsos por semiciclo. Modulación al 90%.

Frecuencia = 60 Hz      THD entrada = 051 46%      THD salida = 027 06 %      Amplitud máxima de salida = 100 %



Señal de entrada — — —      Señal de salida — — —      Amplitud de la señal de entrada = 100 %

Figura 8.10. Simulación de HIPWM unipolar con 5 pulsos por semiciclo. Modulación al 90%.

Frecuencia = 60 Hz      THD entrada = 085.96%      THD salida = 036.36%      Amplitud máxima de salida = 123%

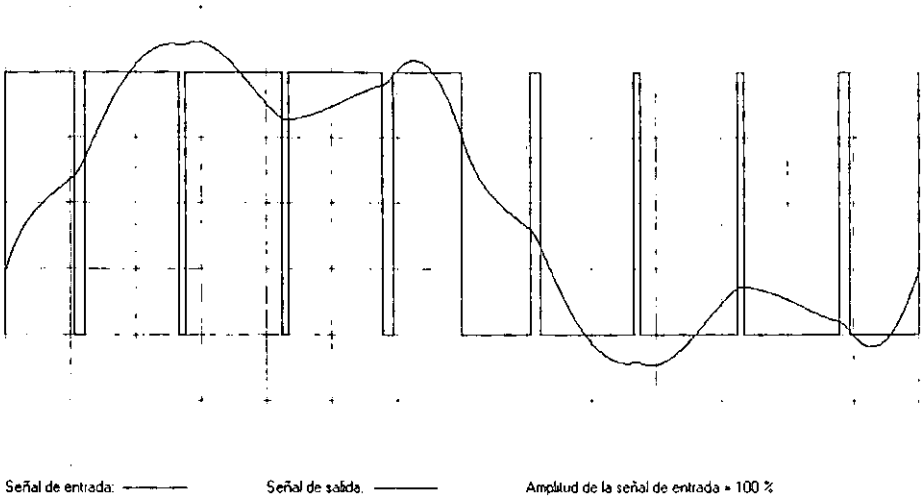


Figura 8.11. Simulación de HIPWM bipolar con 5 pulsos por semiciclo. Modulación al 90%.

Frecuencia = 60 Hz      THD entrada = 053.33%      THD salida = 007.74%      Amplitud máxima de salida = 99%

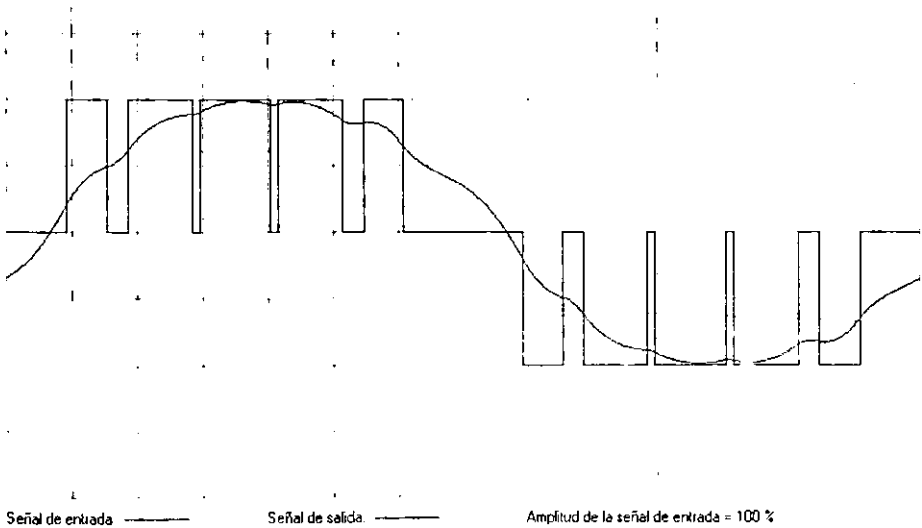
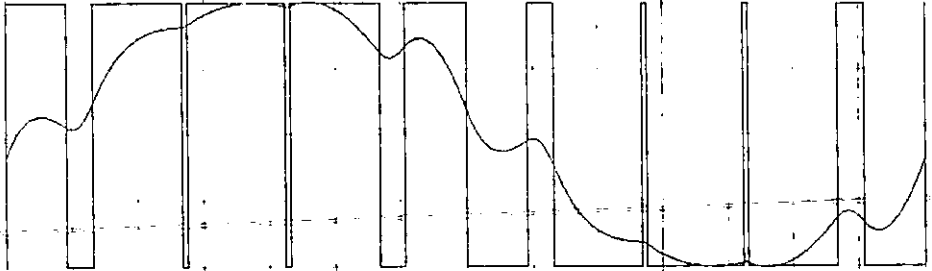


Figura 8.12. Simulación de TPWM unipolar con 5 pulsos por semiciclo. Modulación al 90%.

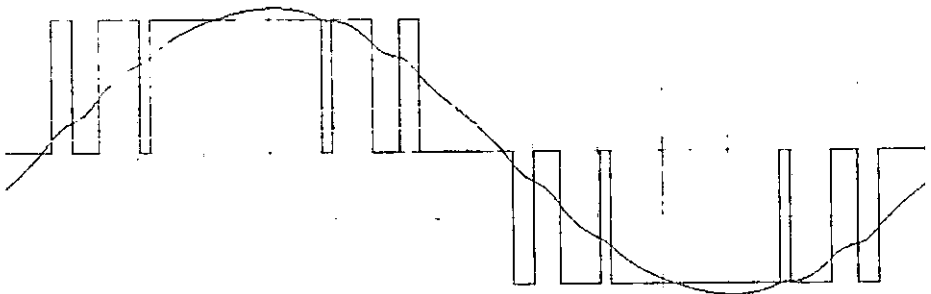
Frecuencia = 60 Hz      THD entrada = 106.58%      THD salida = 013.39%      Amplitud máxima de salida = 101 %



Señal de entrada —      Señal de salida — —      Amplitud de la señal de entrada = 100 %

Figura 8.13. Simulación de TPWM bipolar con 5 pulsos por semiciclo. Modulación al 90%.

Frecuencia = 60 Hz      THD entrada = 046.03%      THD salida = 001.88%      Amplitud máxima de salida = 109 %

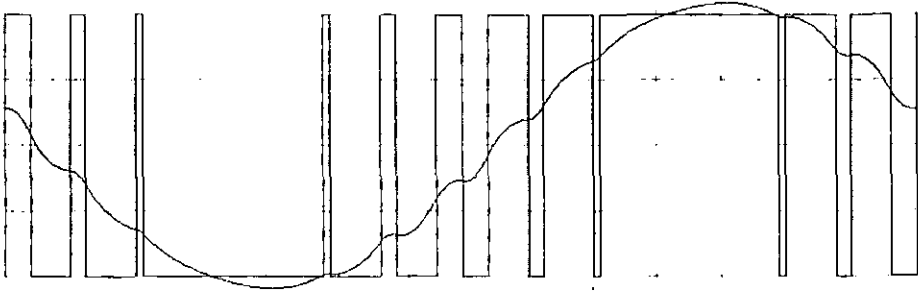


Señal de entrada - - -      Señal de salida —      Amplitud de la señal de entrada = 100 %

Figura 8.14. Técnica unipolar de eliminación de armónicos con 5 pulsos por semiciclo. Simulación.



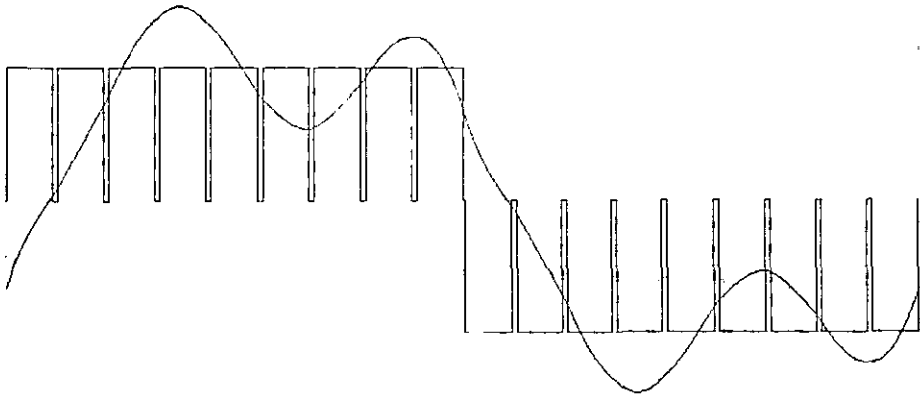
Frecuencia = 60 Hz      THD entrada = 095 34%      THD salida = 003 77 %      Amplitud máxima de salida = 109 %



Señal de entrada      Señal de salida      Amplitud de la señal de entrada = 100 %

Figura 8.15. Técnica bipolar de eliminación de armónicos con 5 pulsos por semiciclo. Simulación.

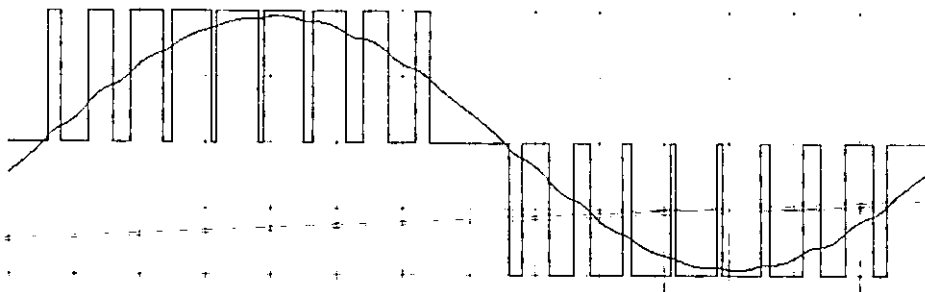
Frecuencia = 60 Hz      THD entrada = 060 63%      THD salida = 045 62 %      Amplitud máxima de salida = 146 %



Señal de entrada      Señal de salida      Amplitud de la señal de entrada = 100 %

Figura 8.16. Simulación de UPWM unipolar con 9 pulsos por semiciclo. Modulación al 90%.

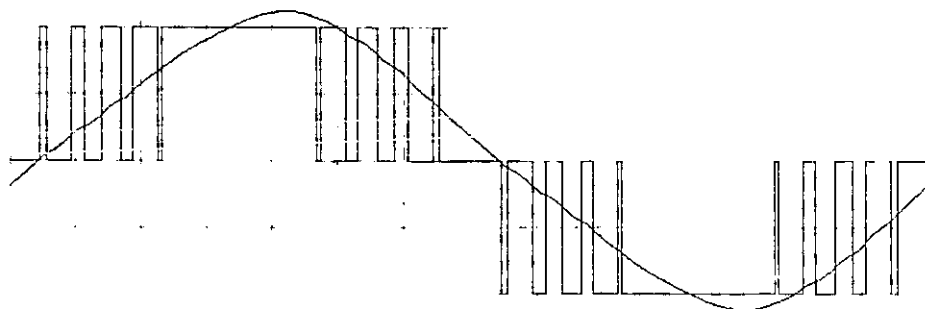
Frecuencia = 60 Hz      THD entrada = 063 40%      THD salida = 001 46 %      Amplitud máxima de salida = 96 %



Señal de entrada ——— Señal de salida: ——— Amplitud de la señal de entrada = 100 %

Figura 8.17. Simulación de SPWM unipolar con 9 pulsos por semiciclo. Modulación al 90%.

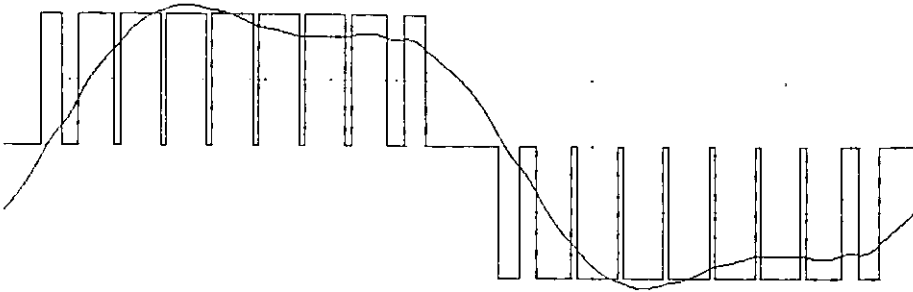
Frecuencia = 60 Hz      THD entrada = 051 72%      THD salida = 006 62 %      Amplitud máxima de salida = 11 7 %



Señal de entrada ——— Señal de salida: ——— Amplitud de la señal de entrada = 100 %

Figura 8.18. Simulación de MSPWM unipolar con 9 pulsos por semiciclo. Modulación al 90%.

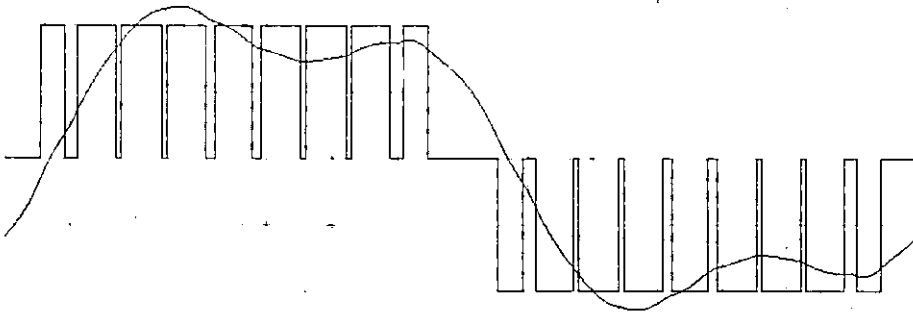
Frecuencia = 60 Hz      THD entrada = 052 99%      THD salida = 021 56 %      Amplitud máxima de salida = 107 %



Señal de entrada ——— Señal de salida ——— Amplitud de la señal de entrada = 100 %

Figura 8.19. Simulación de THIPWM unipolar con 9 pulsos por semiciclo. Modulación al 90%.

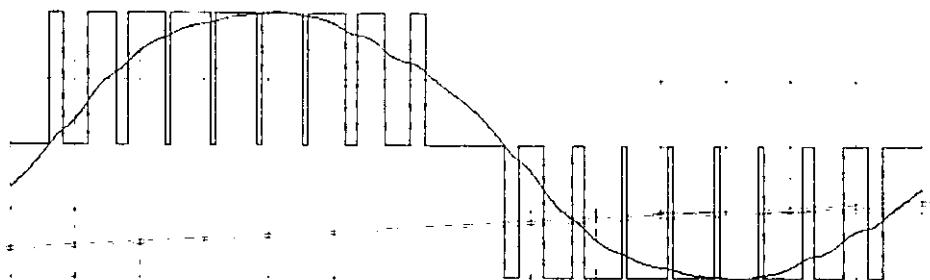
Frecuencia = 60 Hz      THD entrada = 055 42%      THD salida = 030 79 %      Amplitud máxima de salida = 114 %



Señal de entrada - - - - - Señal de salida ——— Amplitud de la señal de entrada = 100 %

Figura 8.20. Simulación de HIPWM unipolar con 9 pulsos por semiciclo. Modulación al 90%.

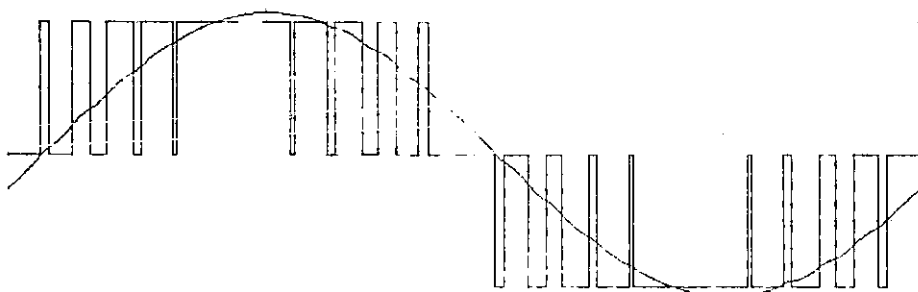
Frecuencia = 60 Hz      THD entrada = 054.43%      THD salida = 005.04%      Amplitud máxima de salida = 101%



Señal de entrada      Señal de salida      Amplitud de la señal de entrada = 100%

Figura 8.21. Simulación de TPWM unipolar con 9 pulsos por semiciclo. Modulación al 90%.

Frecuencia = 60 Hz      THD entrada = 054.43%      THD salida = 005.04%      Amplitud máxima de salida = 107%



Señal de entrada      Señal de salida      Amplitud de la señal de entrada = 100%

Figura 8.22. Técnica unipolar de eliminación de armónicos con 9 pulsos por semiciclo. Simulación.

Técnicas de modulación y eliminación	Unipolar				Bipolar			
	5 pulsos		9 pulsos		5 pulsos		9 pulsos	
	THD1	THD2	THD1	THD2	THD1	THD2	THD1	THD2
UPWM	60.16%	46.77%	60.63%	45.62%	72.37%	45.10%	72.21%	45.12%
SPWM	61.66%	4.68%	63.40%	1.46%	121.2%	14.76%	121.1%	3.62%
MSPWM	49.96%	6.22%	51.22%	6.62%	100.7%	10.25%	101.9%	6.57%
THIPWM	49.85%	20.80%	52.99%	21.56%	92.53%	26.54%	92.99%	21.67%
HIPWM	51.46%	27.06%	55.42%	30.79%	85.96%	36.36%	92.03%	30.55%
TPWM	53.33%	7.74%	54.43%	6.04%	106.5%	13.39%	102.9%	6.73%
Eliminación	48.03%	1.88%	50.40%	0.79%	95.34%	3.77%	97.56%	1.90%

**Tabla 8.1. Comparación de técnicas de modulación y eliminación.**

De la observación de las gráficas del programa de simulación se hace evidente que las técnicas de modulación que mejores resultados tienen son la senoidal, la senoidal modificada y la trapezoidal tanto en la forma unipolar como bipolar. Al revisar la tabla 8.1 se puede comprobar lo anterior, pues las distorsiones armónicas después del filtro son menores con las técnicas mencionadas.

También es evidente que los resultados gráficos de las técnicas de eliminación de armónicos son mejores que los de las técnicas de modulación, tanto con señales unipolares como con señales bipolares. Si se toma en cuenta sólo las técnicas de eliminación se puede apreciar que la técnica de eliminación unipolar posee una distorsión armónica menor que la bipolar para el mismo número de pulsos por semiciclo. En forma numérica los datos se pueden verificar en la tabla 8.1.

Con los datos obtenidos en la simulación, las técnicas de modulación de un pulso, de múltiples pulsos uniformes, por inyección del tercer armónico y por inyección de armónicos seleccionados, bipolares y unipolares, quedan descartadas para su uso en inversores monofásicos.

De esta manera, se eligió el tipo de señales unipolares para su implementación en el módulo de evaluación porque de acuerdo con los resultados obtenidos en el programa de simulación éstas tienen a menudo un contenido armónico menor al de las señales bipolares

para el caso de las técnicas de modulación, además de que en la técnica de eliminación la señal unipolar requiere un pulso menos por semiciclo que la señal bipolar para eliminar el mismo número de armónicos.

## 8.2. Oscilogramas de modulación de ancho de pulso

Las siguientes 26 gráficas muestran la señal de salida y el contenido armónico de las distintas técnicas de modulación unipolar después de pasar por un filtro de segundo orden LC paralelo. El filtro está constituido por un inductor de 20 mH y un capacitor de 30  $\mu\text{F}$ , la carga es puramente resistiva y su valor es de 33  $\Omega$ . La fuente que alimenta al módulo de evaluación es de 24 V. Para fines de comparación, se han incluido todas las técnicas de modulación unipolar, aunque cuatro de ellas ya han sido rechazadas para su uso en inversores monofásicos de acuerdo con los resultados del programa de simulación

Los oscilogramas fueron obtenidos con un osciloscopio digital marca *TEKTRONIX* modelo *Tekscope THS720P* que permite la captura y transferencia de las imágenes generadas y que cuenta además con la posibilidad de calcular la distorsión armónica total de la señal analizada y representarla en forma gráfica.

### 8.2.1. Modulación de un solo ancho de pulso

De entre todas las técnicas utilizadas, ésta es de las que presentan una distorsión armónica mayor, 36.3% para un índice de modulación del 90%. La figura 8.23 muestra la señal de salida después del filtro, la cual continua siendo prácticamente cuadrada, mientras que la gráfica 8.24 muestra el contenido armónico, donde el tercer, quinto y séptimo armónicos aún están presentes y contribuyen casi por completo a la distorsión de la señal porque son difíciles de eliminar. Sin embargo, es posible obtener una señal senoidal con esa técnica de modulación si se recurre a un filtro resonante, aunque con la desventaja de que los valores del inductor y del capacitor son muy grandes.

Esta técnica de modulación es la más simple de todas y por lo tanto es una buena referencia para comparar las técnicas de modulación y eliminación restantes, además, a pesar de no ser una buena técnica de modulación, es la más difundida entre los constructores de inversores en el país debido a la poca complejidad requerida para generar

las señales de control y a que generalmente se fabrican sin filtro de salida, siendo su costo de producción bajo.

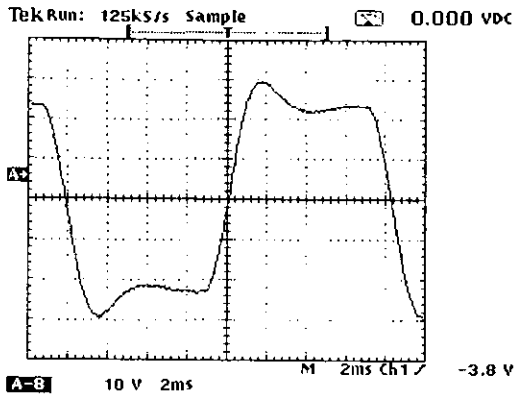


Figura 8.23. Modulación de un solo pulso.  
Señal de salida con modulación al 90%.

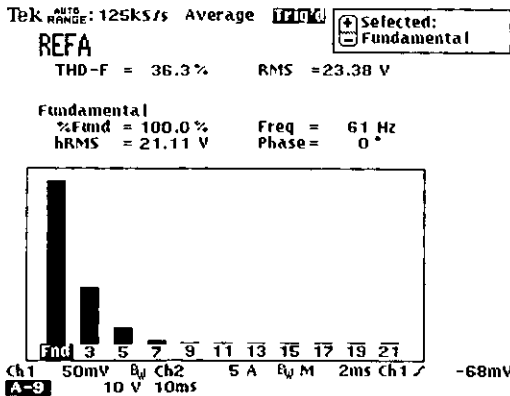


Figura 8.24. Modulación de un solo pulso.  
Contenido armónico de la señal de salida.

### 8.2.2. Modulación uniforme de ancho de pulso (UPWM) unipolar

Esta es la técnica de modulación que presenta un contenido armónico mayor, 40% para el caso de 5 pulsos por semiciclo y 39.8% para el caso de 9 pulsos por semiciclo, en ambos casos con una modulación del 90%. La única ventaja con respecto a la modulación de un solo ancho de pulso es el comportamiento lineal de la componente fundamental. Las gráficas 8.25 y 8.26 pertenecen al caso de 5 pulsos por semiciclo, en tanto que las gráficas 8.27 y 8.28 son de la modulación con 9 pulsos por semiciclo.

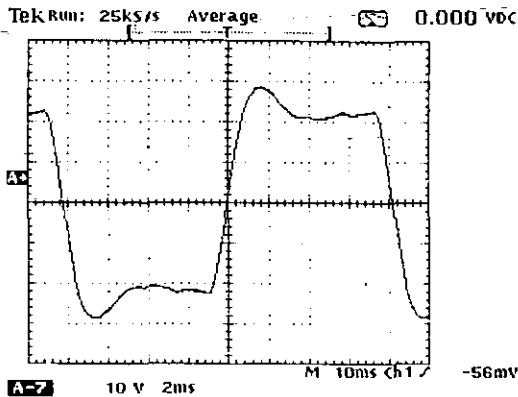


Figura 8.25. UPWM unipolar. Señal de salida con 5 pulsos por semiciclo, modulación al 90%.

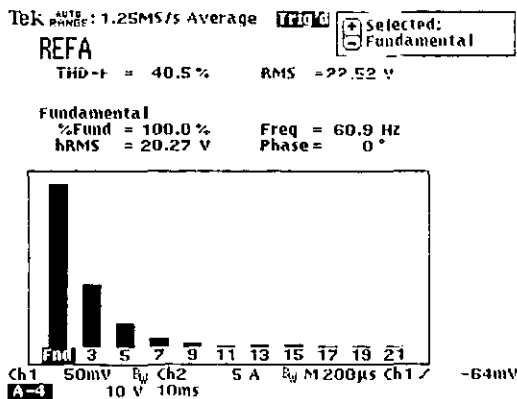


Figura 8.26. UPWM unipolar. THD de salida con 5 pulsos por semiciclo, modulación al 90%.



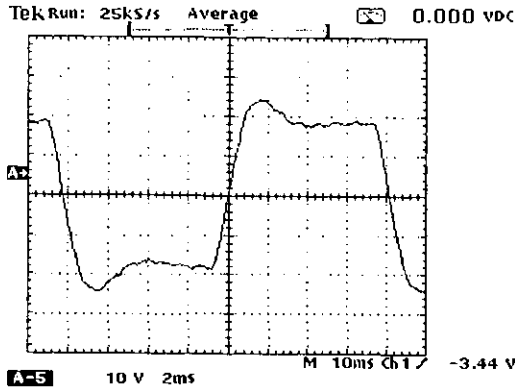


Figura 8.27. UPWM unipolar. Señal de salida con 9 pulsos por semiciclo, modulación al 90%.

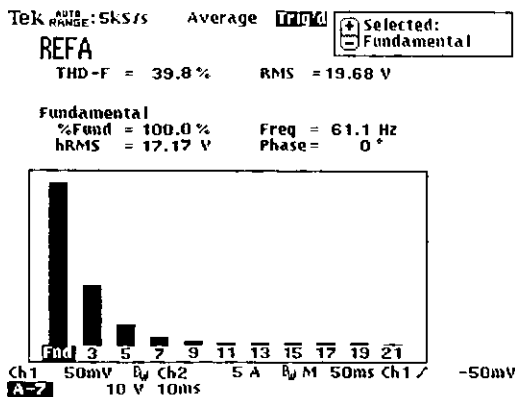


Figura 8.28. UPWM unipolar. THD de salida con 9 pulsos por semiciclo, modulación al 90%.

Bajo las circunstancias mencionadas, este tipo de modulación es menos recomendable que la de un solo pulso para su implementación en inversores monofásicos, ya que los circuitos de control se vuelven más complejos y las señales obtenidas no generan una disminución de los armónicos. En este caso, el factor de distorsión de segundo orden proporciona una información correcta para juzgar el comportamiento de esta técnica de modulación.

### 8.2.3. Modulación senoidal de ancho de pulso (SPWM) unipolar

En esta modulación se observa una mejoría considerable con respecto a las dos anteriores. Las señales de salida para 5 y 9 pulsos por semiciclo se pueden apreciar en las figuras 8.29 y 8.31, y aún en el caso de 5 pulsos por semiciclo, presentado en la figura 8.29, es fácil reconocer una señal senoidal. Para el caso de 5 pulsos por semiciclo la distorsión armónica total es de 5.8% y para 9 pulsos de 2.7%.

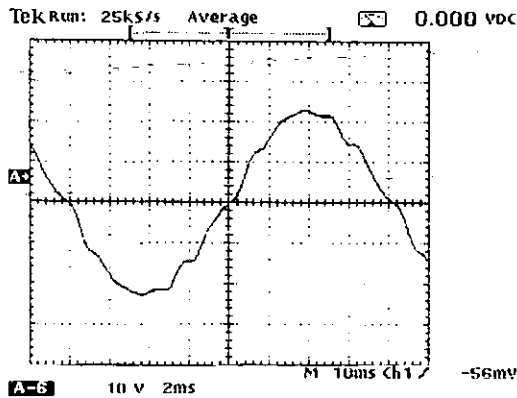


Figura 8.29. SPWM unipolar. Señal de salida con 5 pulsos por semiciclo, modulación al 90%.

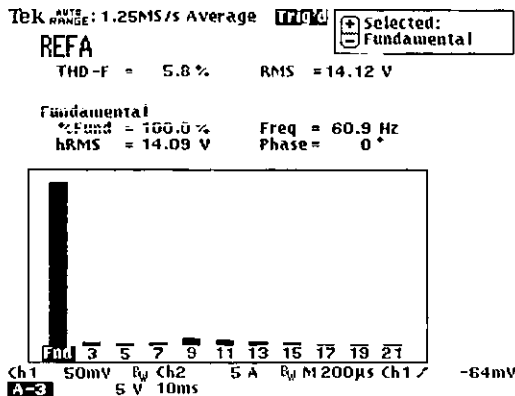


Figura 8.30. SPWM unipolar. THD de salida con 5 pulsos por semiciclo, modulación al 90%.

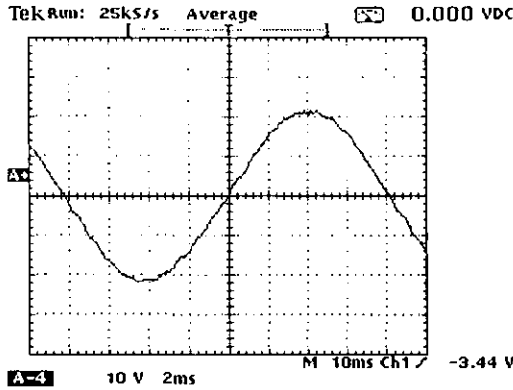


Figura 8.31. SPWM unipolar. Señal de salida con 9 pulsos por semiciclo, modulación al 90%.

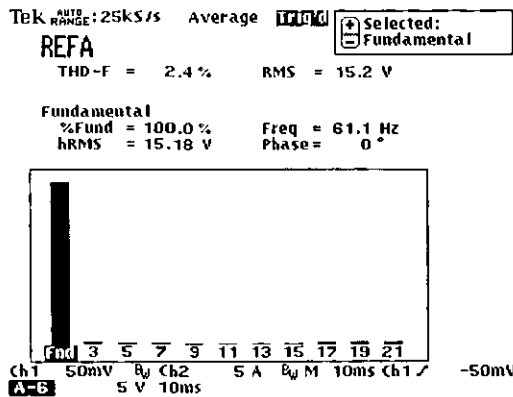


Figura 8.32. SPWM unipolar. THD de salida con 9 pulsos por semiciclo, modulación al 90%.

En esta modulación, la teoría predice que el número de armónicos eliminados es igual al número de pulsos por semiciclo menos 2, lo cual concuerda con los resultados prácticos, pues para una modulación de 5 pulsos se eliminan 3 armónicos y para una modulación de 9 pulsos se eliminan 7, como se puede ver en las figuras 8.30 y 8.32 respectivamente. Se puede concluir que esta técnica es muy recomendable para inversores monofásicos. En cuanto al factor de distorsión de segundo orden, se pueden considerar sus predicciones como válidas.

### 8.2.4. Modulación senoidal modificada (MSPWM) unipolar

Esta técnica muy bien puede competir con la modulación senoidal, pues además de que su contenido armónico no es grande, 4.6% para el caso de 5 pulsos por semiciclo y 5.2% para 9 pulsos, la amplitud de la componente fundamental aumenta, como se puede comprobar en las figuras 8.34 y 8.36. Sin embargo, una revisión de la figura 8.35 revela que la señal tiende a ser más triangular que senoidal, algo que es un poco menos perceptible en la figura 8.33 que muestra la señal de salida para 5 pulsos por semiciclo.

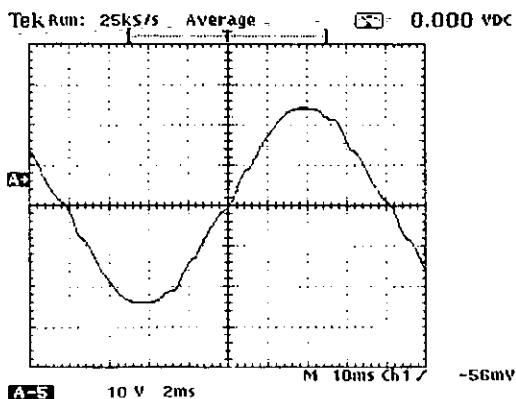


Figura 8.33. MSPWM unipolar. Señal de salida con 5 pulsos por semiciclo, modulación al 90%.

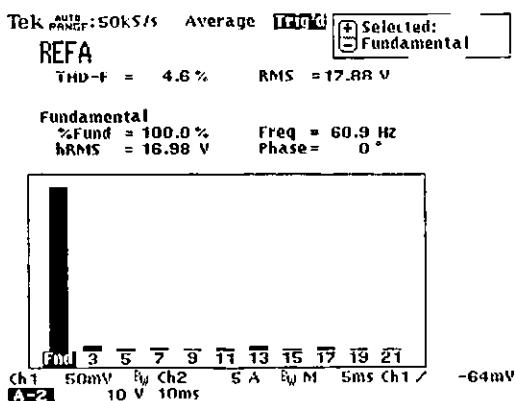


Figura 8.34. MSPWM unipolar. THD de salida con 5 pulsos por semiciclo, modulación al 90%.

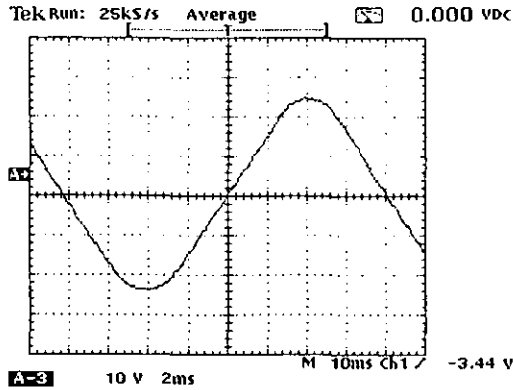


Figura 8.35. MSPWM unipolar. Señal de salida con 9 pulsos por semiciclo, modulación al 90%.

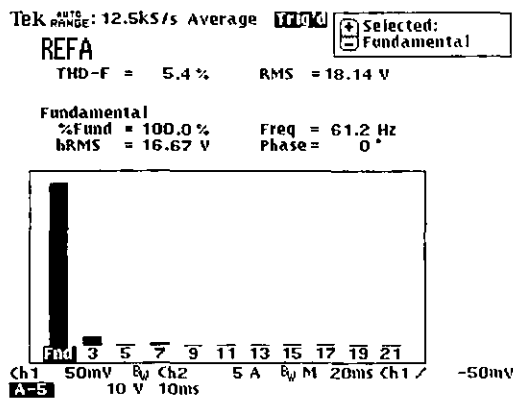


Figura 8.36. MSPWM unipolar. THD de salida con 9 pulsos por semiciclo, modulación al 90%.

Como se dijo, esta técnica puede competir con la modulación senoidal en inversores monofásicos, pero debido a la presencia del tercer armónico, como se aprecia en la figura 8.36, sin duda tendrá un mejor desempeño en inversores trifásicos con carga conectada en estrella, donde el tercer armónico y sus múltiplos serán eliminados. El factor de distorsión de segundo orden proporciona una buena predicción para este caso.

### 8.2.5. Modulación unipolar por inyección del tercer armónico (THIPWM)

Este tipo de modulación no es una buena opción para inversores monofásicos. Las señales de salida se pueden observar en las figuras 8.37 y 8.39 para los casos de 5 y 9 pulsos por semiciclo respectivamente. Es fácil observar que la presencia del tercer armónico distorsiona grandemente la señal y su presencia es notable en las figuras 8.38 y 8.40, en donde se muestra el contenido armónico de la señal después del filtro de salida.

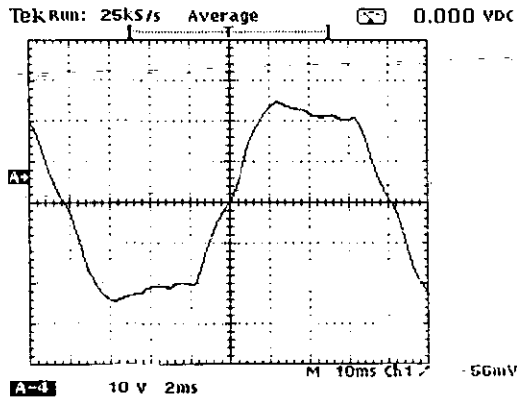


Figura 8.37. THIPWM unipolar. Señal de salida con 5 pulsos por semiciclo, modulación al 90%.

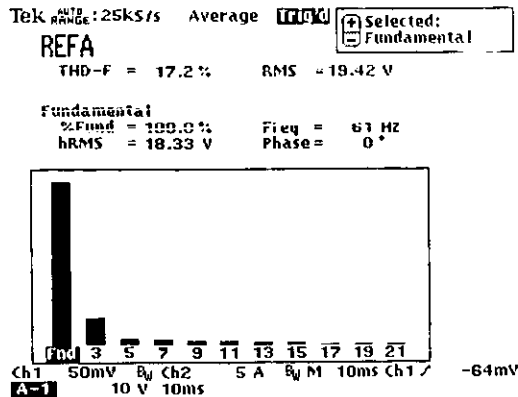


Figura 8.38. THIPWM unipolar. THD de salida con 5 pulsos por semiciclo, modulación al 90%.

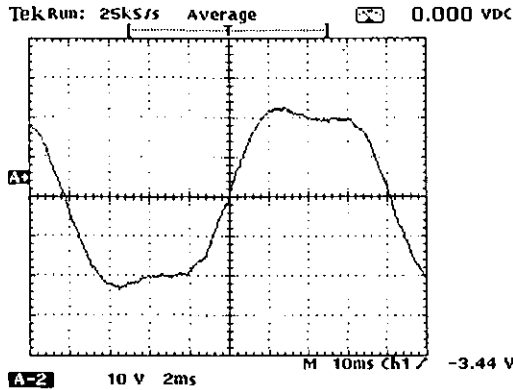


Figura 8.39. THIPWM unipolar. Señal de salida con 9 pulsos por semiciclo, modulación al 90%.

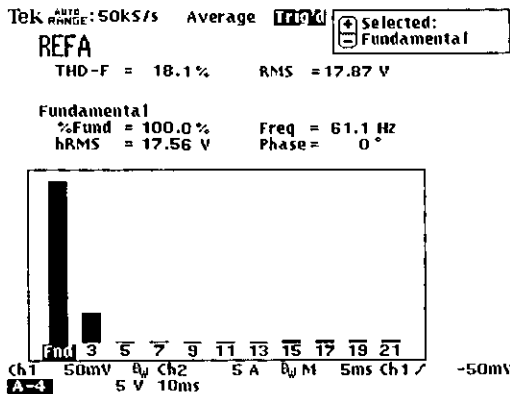


Figura 8.40. THIPWM unipolar. THD de salida con 9 pulsos por semiciclo, modulación al 90%.

La técnica de modulación por inyección del tercer armónico es, sin duda alguna, una técnica de modulación para inversores trifásicos, la presencia del tercer armónico lo confirma. El factor de distorsión de segundo orden tiene aplicación para el caso de un inversor monofásico, siendo una figura de mérito importante para fines de comparación. La principal ventaja de esta técnica para inversores trifásicos es el incremento de la componente fundamental aunada con el bajo contenido armónico presente. Para el caso de inversores monofásicos se puede considerar una técnica nula.

8.2.6. Modulación unipolar por inyección de armónicos (HIPWM)

Las figuras 8.41 y 8.43 muestran las señales de salida para la técnica de modulación por inyección de armónicos para 5 y 9 pulsos por semiciclo. La distorsión de la señal de salida es grande debido principalmente a la presencia del tercer armónico como se aprecia en las figuras 8.42 y 8.44. No es una técnica para inversores monofásicos pues por sus características es similar a la técnica de inyección del tercer armónico y su uso debe restringirse a inversores trifásicos.

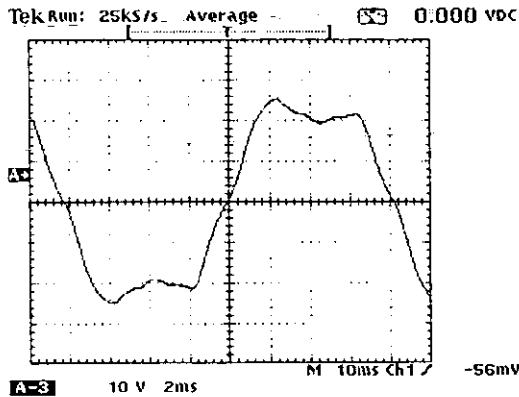


Figura 8.41. HIPWM unipolar. Señal de salida con 5 pulsos por semiciclo, modulación al 90%.

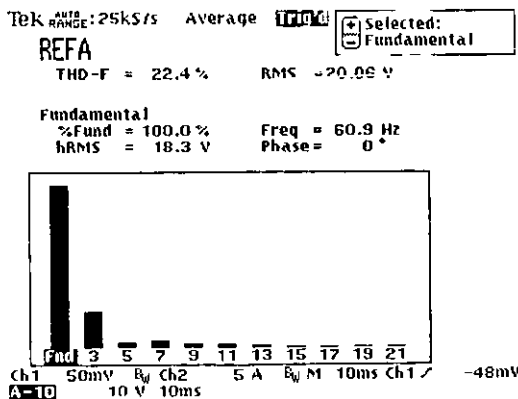


Figura 8.42. HIPWM unipolar. THD de salida con 5 pulsos por semiciclo, modulación al 90%.



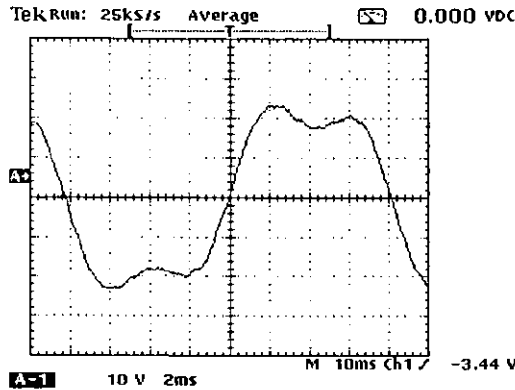


Figura 8.43. HIPWM unipolar. Señal de salida con 9 pulsos por semiciclo, modulación al 90%.

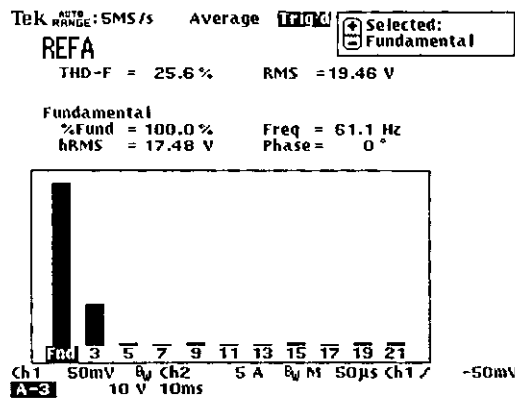


Figura 8.44. HIPWM unipolar. THD de salida con 9 pulsos por semiciclo, modulación al 90%.

Tanto de la técnica de inyección del tercer armónico como de la técnica de inyección de armónicos seleccionados se puede decir que son variaciones de una misma idea, la cual es incrementar la amplitud de la componente fundamental a costa de aumentar también el tercer armónico en la señal de salida, teniendo en cuenta que con su uso en sistemas inversores trifásicos el aumento de dicho armónico no constituye problema alguno pues por la topología del sistema su eliminación está garantizada. Los comentarios para el factor de distorsión de segundo orden son los mismos que para la técnica anterior.

### 8.2.7. Modulación trapezoidal (TPWM) unipolar

Las señales de salida para 5 y 9 pulsos por semiciclo con esta técnica se presentan en las figuras 8.45 y 8.47. Son señales que aunque tienen semejanza con una señal senoidal puede ser que en algunos casos no satisfagan los requisitos de contenido armónico en un inversor monofásico. En las figuras 8.46 y 8.48 se puede observar que la distorsión armónica total es mayor al 5.5% debido a la presencia de armónicos de menor orden, principalmente el tercero y el quinto.

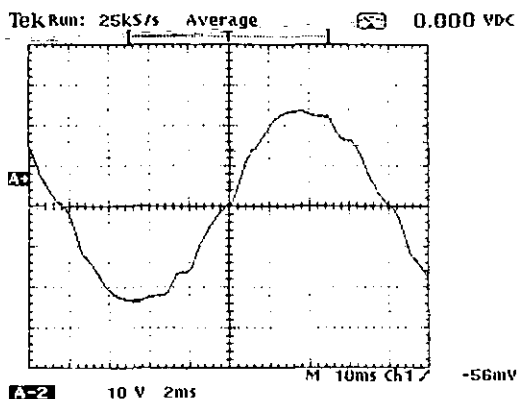


Figura 8.45. TPWM unipolar. Señal de salida con 5 pulsos por semiciclo, modulación al 90%.

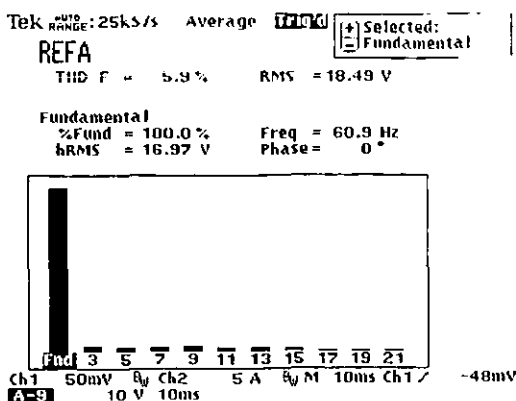


Figura 8.46. TPWM unipolar. THD de salida con 5 pulsos por semiciclo, modulación al 90%.

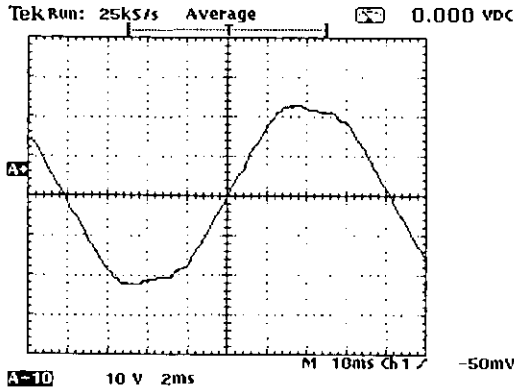


Figura 8.47. TPWM unipolar. Señal de salida con 9 pulsos por semiciclo, modulación al 90%.

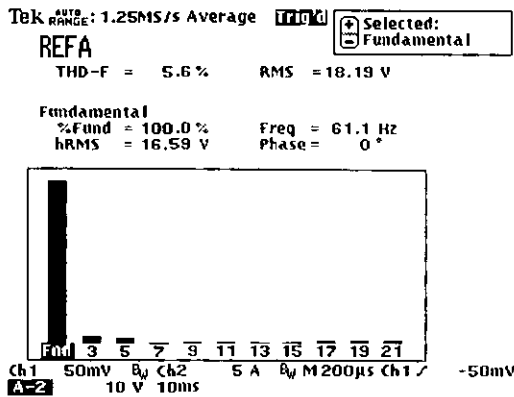


Figura 8.48. TPWM unipolar. THD de salida con 9 pulsos por semiciclo, modulación al 90%.

Con la información proporcionada, se puede concluir que aunque la modulación trapezoidal puede usarse en inversores monofásicos y que sin duda alguna proporcionará mejores resultados que los de la modulación de un solo pulso y de ancho de pulso uniforme, su uso es preferible en inversores trifásicos. Una ventaja adicional de este tipo de modulación, es que es ideal para ser generada y controlada por computadora. El factor de distorsión de segundo orden es válido como figura de evaluación para la modulación discutida.

### 8.3. Oscilogramas de la técnica de eliminación de armónicos

Las 4 gráficas presentadas a continuación pertenecen a la técnica unipolar de eliminación de armónicos para los casos de 5 y 9 pulsos por semiciclo. El filtro y carga utilizados son los mismos que para el caso de las técnicas de modulación.

#### 8.3.1. Técnica unipolar de eliminación de armónicos

La técnica de eliminación de armónicos es la que mejores resultados genera. Las figuras 8.49 y 8.51 presentan las señales de salida filtradas y en ambos casos es evidente una señal senoidal. Teóricamente, esta técnica elimina tantos armónicos como pulsos por semiciclo contiene la señal antes del filtro. Los resultados prácticos son similares a los teóricos. La figura 8.50 muestra el contenido armónico para una señal filtrada con cinco pulsos por semiciclo, en esta figura se puede apreciar como los armónicos desde el 3º hasta el 11º presentan una atenuación significativa, misma que se ve reflejada en un contenido armónico de 2.6%, el más bajo logrado con 5 pulsos por semiciclo. Una situación similar se puede observar en la figura 8.52, donde con 9 pulsos por semiciclo se eliminan 9 armónicos, desde el 3º hasta el 19º, logrando una distorsión armónica total de tan sólo 1%, con un desempeño mejor al de cualquiera de las técnicas de modulación vistas.

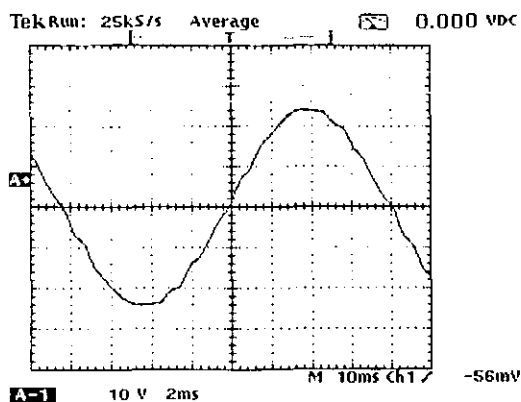


Figura 8.49. Técnica unipolar de eliminación de armónicos. Señal de salida, 5 pulsos por semiciclo.

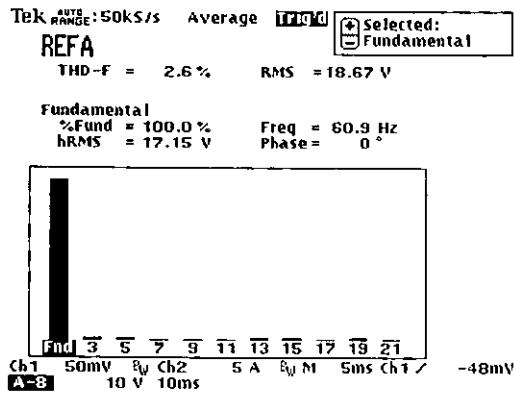


Figura 8.50. Técnica unipolar de eliminación de armónicos. THD de salida, 5 pulsos por semiciclo.

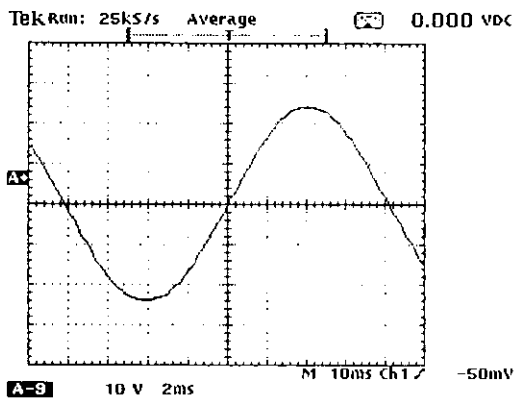


Figura 8.51. Técnica unipolar de eliminación de armónicos. Señal de salida, 9 pulsos por semiciclo.

Además de generar un bajo contenido armónico de salida, otra de las ventajas de la técnica de eliminación es que su aplicación permite eliminar el grupo de armónicos deseado, aun cuando éstos no sean consecutivos. Esta característica es útil en el caso de inversores trifásicos donde no es necesario eliminar el tercer armónico y sus múltiplos, lo anterior permite optimizar el número de pulsos por semiciclo, ya que con una cantidad menor de ellos es posible eliminar una cantidad mayor de armónicos.

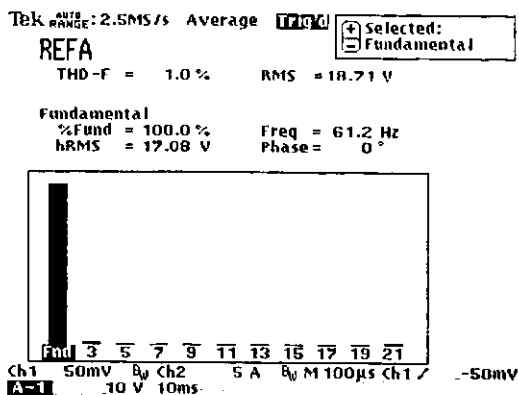


Figura 8.52. Técnica unipolar de eliminación de armónicos. THD de salida, 9 pulsos por semiciclo.

#### 8.4. Comparación de las técnicas de modulación y eliminación

De los resultados obtenidos con el módulo de evaluación, de las distintas técnicas de modulación sólo la modulación senoidal tiene las características adecuadas para su aplicación a un sistema de inversión monofásico con un bajo contenido armónico menor al 3%. El resto de las técnicas de modulación tienen una mejor aplicación en inversores trifásicos y en inversores resonantes. En el caso de la técnica de eliminación de armónicos, y al compararla con la técnica senoidal de modulación, se puede decir que su eficacia es mejor, pues se obtienen resultados similares pero con dos pulsos por semiciclo menos haciendo más eficiente el funcionamiento del inversor.

Para comparar los resultados teóricos con los reales se preparó la tabla 8.2, en ella se presentan los resultados numéricos obtenidos con las simulaciones y con el osciloscopio, únicamente para señales unipolares. Se puede constatar que ambos resultados son muy similares, lo que da validez a las simulaciones.

Por otro lado, se comprobó que el factor de distorsión armónica de segundo orden brinda información muy útil, el comportamiento descrito por este factor para las diferentes técnicas de modulación se acerca mucho a los resultados reales. Para el caso del filtro utilizado, el factor de distorsión de segundo orden debe ser escalado diez veces para apreciar en forma más clara sus predicciones.

Técnica	5 pulsos por semiciclo			9 pulsos por semiciclo		
	THD real	THD2	10 x DF <sub>2</sub>	THD real	THD2	10 x DF <sub>2</sub>
UN PULSO	36.3%	40.10%	33.9%	36.3%	40.10%	33.9%
UPWM	40.5%	46.77%	39.3%	39.8%	45.62%	38.4%
SPWM	5.8%	4.68%	3.9%	2.4%	1.46%	1.3%
MSPWM	4.6%	6.22%	5.3%	5.4%	6.62%	5.6%
THIPWM	17.2%	20.80%	17.7%	18.1%	21.56%	18.4%
HIPWM	22.4%	27.06%	23.0%	25.6%	30.79%	26.3%
TPWM	5.9%	7.74%	6.5%	5.6%	6.04%	5.0%
Eliminación	2.6%	1.88%	1.6%	1.0%	0.79%	0.7%

**Tabla 8.2. Comparación de resultados simulados y reales.**

---

# **CAPÍTULO 9**

---

**CONCLUSIONES Y PROPUESTAS**



## 9.1. Semiconductores

Los semiconductores más comunes en inversores de potencia son el BJT, el MOSFET y el IGBT. Entre las características más sobresalientes de estos dispositivos están las siguientes:

El BJT es un dispositivo controlado por corriente disponible con especificaciones de hasta 1200V, 400 A y frecuencia de funcionamiento de 10 kHz. Tiene una caída de tensión de encendido que llega a ser de 2 V.

El MOSFET es un dispositivo controlado por tensión disponible con especificaciones de hasta 1000 V, 50 A. y frecuencia de funcionamiento de 100 kHz. Tiene una resistencia entre drenaje y fuente que se incrementa con la capacidad de bloqueo de tensión y que como consecuencia incrementa también la caída de tensión de encendido que puede llegar a ser mayor que en los BJT, sobre todo para tensiones superiores a los 400 V.

El IGBT es un dispositivo controlado por tensión disponible con especificaciones de hasta 1200V, 400 A y frecuencia de funcionamiento de 20 kHz. Tiene una caída de tensión de encendido de hasta 3 V.

Con base a los datos anteriores, se eligió al IGBT como el interruptor más adecuado para los convertidores cd-ca. Su principal ventaja es que reúne características del BJT y del MOSFET en un solo dispositivo. Es similar al MOSFET en ser un dispositivo controlado por tensión, lo cual simplifica su manejo, y es semejante al BJT en cuanto a las tensiones y corrientes que soporta así como en la caída de tensión de encendido. Por otro lado, el MOSFET también es una buena elección si su uso se limita a tensiones inferiores a 400 V y se requiere un dispositivo más veloz que el IGBT con un menor manejo de potencia.

Es importante mencionar que los transistores sólo se utilizan en convertidores con potencias menores a 100 kW. Para potencias mayores el SCR y otros tiristores son más adecuados.

## 9.2. Estructuras de conversión cd-ca

De tres topologías inversoras vistas, la de puente completo es la que hace un mejor uso de la fuente de alimentación al utilizarla totalmente, a diferencia de la topología de medio puente, que sólo hace uso de la mitad de la fuente por necesitar una derivación central. La topología push-pull también utiliza por completo la fuente de alimentación, pero tiene la desventaja de que sólo funciona con transformador, en tanto que las topologías restantes pueden funcionar con o sin él. Otra desventaja de la topología push-pull es que debido al uso del transformador somete a los dispositivos de conmutación a una tensión del doble de la fuente de alimentación. Así, al tomar en cuenta el mejor uso de la fuente de alimentación y una menor tensión de ruptura en los transistores se puede concluir que la topología más adecuada para un inversor monofásico es la de puente completo.

## 9.3. Filtros

De los cinco filtros analizados, el más adecuado para las técnicas de modulación y eliminación es el tipo LC paralelo. La red LC paralela es utilizada como un filtro de paso bajo, lo que le da ventajas sobre los filtros LC serie y LC serie-paralelo que, al ser usados para resonar a la frecuencia fundamental, requieren de valores de capacitancia e inductancia que fácilmente llegan a ser 5 veces más grandes. En cuanto a los filtros de primer orden C y L, el filtro LC paralelo muestra también mejores características, entre las que se encuentran una atenuación de 40 dB/década en contra de los 20 dB/década de los filtros de primer orden y una consecuente reducción del valor de sus elementos. En un filtro LC paralelo los valores más pequeños de sus componentes, en comparación con el resto de los filtros, se traducen en una mejor respuesta ante los cambios de la carga y un menor consumo de corriente, siendo este último efecto producido por la disminución de la capacitancia.

## 9.4. Técnicas de modulación y eliminación

Tras examinar el tipo de señales, que pueden obtenerse con las distintas técnicas de modulación, se descartaron las del tipo bipolar por poseer una distorsión armónica total mayor que su contraparte unipolar. Por otro lado, de entre todas las técnicas de modulación examinadas, únicamente la técnica de modulación senoidal puede cubrir los requisitos de

una distorsión armónica total menor al 3%. El resto de las técnicas de modulación contienen armónicos de orden menor, en donde se destaca sobre todo el tercero. La presencia de este armónico hace más adecuadas las técnicas de modulación restantes para sistemas de inversión trifásicos con carga en estrella, en los cuales el armónico tercero y sus múltiplos son eliminados.

Con respecto a la técnica de eliminación utilizada, el modo bipolar fue también descartado por necesitar un pulso por semiciclo más que la técnica unipolar para eliminar el mismo número de armónicos. En el modo unipolar utilizado, los resultados obtenidos mejoraron por completo a los de las técnicas de modulación, incluida la técnica de modulación senoidal. Con esta técnica de eliminación unipolar utilizada, se logró obtener una distorsión armónica total del 1% utilizando 9 pulsos por semiciclo, en tanto que la técnica de modulación senoidal contenía una distorsión de 2.6% para el mismo número de pulsos. Otra característica importante de la técnica de eliminación es que su uso puede ser aplicado a inversores trifásicos, en donde, como se ha mencionado, no es necesario eliminar el tercer armónico y sus múltiplos, característica que hace aún más eficiente a la técnica de eliminación.

En resumen, los datos obtenidos tanto con el programa de simulación como con el módulo de evaluación proporcionan elementos suficientes para decir que la técnica unipolar de eliminación de armónicos es una excelente alternativa para la implementación de un inversor monofásico de bajo contenido armónico.

### **9.5. Propuestas para un convertidor cd-ca de bajo contenido armónico**

Con la información obtenida se propone la construcción de un sistema inversor de potencia monofásico con  $127 V_{rms}$  de salida que haga uso de la técnica unipolar de eliminación de armónicos, un filtro de segundo orden y transistores bipolares de compuerta aislada. La propuesta presenta dos alternativas para la implementación. La primera es utilizar un banco de baterías de 48 V y escalar esta tensión a 180 V con un convertidor cd-cd para alimentar al inversor. La segunda es utilizar directamente un banco de baterías que alimente al convertidor cd-ca. Ambas alternativas presentan ventajas y desventajas en su implementación que serán comentadas a continuación.

### 9.5.1 Convertidor cd-ca con banco de baterías de 48 voltios

Para la construcción de este inversor es necesario escalar la tensión de las baterías a 180 V con un convertidor cd-cd. Para la topología del convertidor cd-cd la opción más adecuada es una estructura de puente completo que alimente a un transformador con una frecuencia de más de 20 kHz. El uso de este tipo de convertidor cd-cd proporciona la ventaja del aislamiento eléctrico entre la carga y el banco de baterías además de que controla la tensión de salida. Por otro lado, tiene la desventaja de disminuir la eficiencia del sistema pues en este tipo de convertidor cd-cd la eficiencia es típicamente del 75%.

La estructura del sistema de inversión propuesta se presenta en forma de diagrama de bloques en la figura 9.1. En el primer bloque se encuentra el banco de baterías formado por cuatro baterías de 12 V cada una, a continuación se encuentra el convertidor cd-cd con transformador y finalmente se tiene al convertidor de cd-ca, en conjunto con el filtro de segundo orden de salida y la carga.

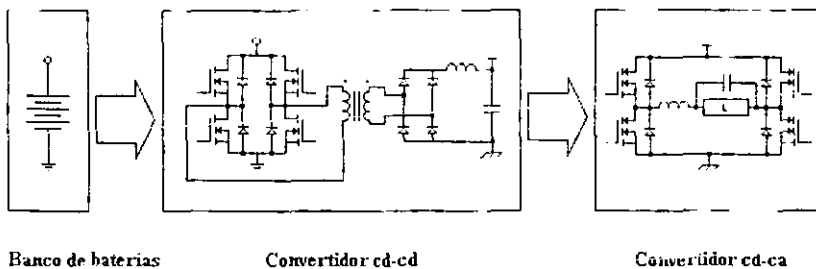


Figura 9.1. Estructura de la primera propuesta de conversión cd-ca.

Al tener en cuenta que la eficiencia del convertidor cd-ca es de alrededor del 90%, la eficiencia para esta estructura de conversión será tan sólo del 68%, aunque con la ventaja del aislamiento entre fuente y carga aunada al uso de un banco de baterías relativamente pequeño, motivos por los cuales a menudo esta estructura es preferida.

### 9.5.2. Convertidor cd-ca con banco de baterías de 192 voltios

Para la construcción de este inversor la tensión de alimentación no tiene la ventaja del aislamiento eléctrico con la carga, pero sí una mejor eficiencia. Para la regulación de la tensión de salida del inversor se utiliza un convertidor cd-cd sin transformador. Este tipo de convertidores tiene una eficiencia de alrededor del 90% que contribuye a un mejor uso del banco de baterías.

El diagrama de bloques para el sistema se presenta en la figura 9.2. El diagrama cuenta con tres bloques, el primer bloque está constituido por 16 baterías de 12 V cada una, enseguida se muestra el convertidor cd-cd utilizado para regular la tensión de salida y finalmente el bloque que contiene al convertidor cd-ca.

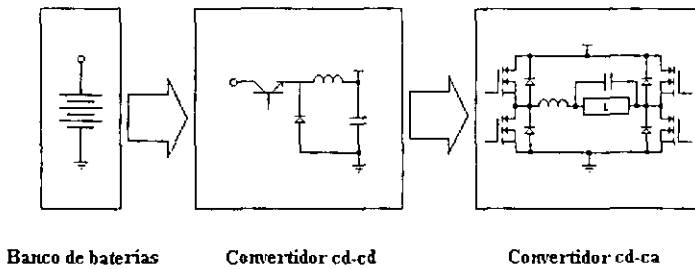


Figura 9.2. Estructura de la segunda propuesta de conversión cd-ca.

Como en el caso de la primera propuesta y considerando las eficiencias del convertidor cd-cd y del convertidor cd-ca la eficiencia total del sistema será de alrededor del 81%, es decir, un 13% mejor que la propuesta anterior.

---

# APÉNDICES

---

### A.1. Siglas

BJT: Bipolar Junction Transistor, transistor bipolar de unión.

DF: Distortion Factor, factor de distorsión.

HF: Harmonic Factor, factor armónico.

HIPWM: Harmonic Injection Pulse Width Modulation, modulación de ancho de pulso por inyección de armónicos.

IGBT: Insulated Gate Bipolar Transistor, transistor bipolar de compuerta aislada.

LOH: Lowest-Order Harmonic, armónica de menor orden.

MOSFET: Metal-Oxide-Semiconductor Field Effect Transistor, transistor de efecto de campo de metal óxido y semiconductor.

MSPWM: Modified Sinusoidal Pulse Width Modulation, modulación senoidal modificada de ancho de pulso.

PWM: Pulse Width Modulation, modulación de ancho de pulso.

SCR: Silicon Controlled Rectifier, rectificador controlado de silicio.

SPWM: Sinusoidal Pulse Width Modulation, modulación senoidal de ancho de pulso.

THD: Total Harmonic Distortion, distorsión armónica total.

THIPWM: Third Harmonic Injection Pulse Width Modulation, modulación de ancho de pulso por inyección del tercer armónico.

TPWM: Trapezoidal Pulse Width Modulation, modulación trapezoidal de ancho de pulso.

UPWM: Uniform Pulse Width Modulation, modulación uniforme de ancho de pulso.

## A.2. Código fuente del programa Marcela

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  Menus, ExtCtrls;

type
  TForm1 = class(TForm)
    MainMenu1: TMainMenu;
    Opciones1: TMenuItem;
    Modulacin1: TMenuItem;
    Eliminacin1: TMenuItem;
    Salir1: TMenuItem;
    Image1: TImage;
    Acercade1: TMenuItem;
    procedure Modulacin1Click(Sender: TObject);
    procedure Eliminacin1Click(Sender: TObject);
    procedure Salir1Click(Sender: TObject);
    procedure Acercade1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation

uses Unit2, Unit3, Unit5;

{$R *.DFM}

procedure TForm1.Modulacin1Click(Sender: TObject);
begin
  Form2.Show;
end;

procedure TForm1.Eliminacin1Click(Sender: TObject);
begin
  Form3.Show;
end;
```



---

```
procedure TForm1.Salir1Click(Sender: TObject);
begin
  Application.Terminate
end;

procedure TForm1.Acercade1Click(Sender: TObject);
begin
  Form5.Show
end;

end.

unit Unit2;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  Menus, ExtCtrls, StdCtrls;

type
  TForm2 = class(TForm)
    Panel1: TPanel;
    MainMenu1: TMainMenu;
    Elimina1: TMenuItem;
    Salir1: TMenuItem;
    Salir2: TMenuItem;
    RadioGroup1: TRadioGroup;
    RadioGroup2: TRadioGroup;
    Label1: TLabel;
    Edit1: TEdit;
    Button1: TButton;
    Button2: TButton;
    Memo1: TMemo;
    Label2: TLabel;
    Salir3: TMenuItem;
    Filtro1: TMenuItem;
    RadioGroup3: TRadioGroup;
    procedure Salir1Click(Sender: TObject);
    procedure Salir2Click(Sender: TObject);
    procedure Salir3Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Button1Click(Sender: TObject);
    procedure Filtro1Click(Sender: TObject);
  private
    { Private declarations }
  end;
end;
```

---

```

public
  { Public declarations }
end;

var
  Form2: TForm2;

implementation

uses Unit3, Unit1, Unit4;

{$R *.DFM}

procedure TForm2.Salir1Click(Sender: TObject);
begin
  Form3.Show
end;

procedure TForm2.Salir2Click(Sender: TObject);
begin
  Form2.Hide
end;

procedure TForm2.Salir3Click(Sender: TObject);
begin
  Application.Terminate
end;

procedure TForm2.Button2Click(Sender: TObject);
begin
  Edit1.Clear;
  Memo1.Clear
end;

procedure TForm2.Button1Click(Sender: TObject);
type
  arreglo = array [1 .. 5000] of real;

var
  IndMod, AncMaxPul, AncPulMod, PriDis, Sumabn, tiempo, DetMatDer: real;
  MitAncMaxPul, y1, y2, y3, y4, y5, Diferencia, tau, AncPen: real;
  THD, SumaTHD, DF1, SumaDF1, DF2, SumaDF2, error1, error2, angulo: real;
  cual, j, i, k, n, NumPulSem, Uno, Seleccion, RenCol, ciclo, g, h, Modula: integer;
  Disparo, bn, SigmaBn, inicio, Fin: arreglo;
  Salir: Char;

const

```

```

T = 1;
Iteracion = 1000;
Error = 0.0006;
Amp = 1;
NumArm = 5000;
Paso = 20;
AmpTra = 1.5;
Iteracion2 = 2000;

```

```

Procedure Desplegamiento2;

```

```

var

```

```

  n, i : integer;

```

```

Begin

```

```

  For n := 1 to NumArm do

```

```

    Begin

```

```

      Sumabn := 0;

```

```

      If n mod 2 = 1 then

```

```

        Begin

```

```

          For i := 1 to NumPulSem do

```

```

            Begin

```

```

              tau := Disparo[i] * T;

```

```

              If i mod 2 = 1 then

```

```

                Begin

```

```

                  If i = 1 then

```

```

                    Begin

```

```

                      bn[n] := 1

```

```

                    End {End If i...}

```

```

                  else

```

```

                    Begin

```

```

                      bn[n] := 2 * cos(2 * n * Pi * tau / T)

```

```

                    End {End Else};

```

```

                  End {End If i...}

```

```

                Else

```

```

                  Begin

```

```

                    bn[n] := -2 * cos(2 * n * Pi * tau / T)

```

```

                  End; {End Else}

```

```

                Sumabn := Sumabn + bn[n];

```

```

            End; {End For i...}

```

```

          End; {End If n...}

```

```

          SigmaBn[n] := Sumabn * 4 * Amp / (n * Pi);

```

```

        End; {End For n...}

```

```

      SumaFHD := 0;

```

```

      SumaDF1 := 0;

```

```

      SumaDF2 := 0;

```

```

      For n := 1 to NumArm do

```

```

        Begin

```

```

          If n > 1 then

```

```

Begin
  SumaTHD := SumaTHD + Sqr(SigmaBn[n]);
  SumaDF1 := SumaDF1 + Sqr(SigmaBn[n] / n);
  SumaDF2 := SumaDF2 + Sqr(SigmaBn[n] / Sqr(n))
End; {End If n...}
End; {End For n...}
THD := Sqr(SumaTHD) / SigmaBn[1];
DF1 := Sqr(SumaDF1) / SigmaBn[1];
DF2 := Sqr(SumaDF2) / SigmaBn[1];
Memo1.Lines.Add('Mod: ' + FormatFloat('000.00', IndMod * 100) + '% '
+ 'Bn(1): ' + FormatFloat('000.00', SigmaBn[1] * 100) + '% '
+ 'DF1: ' + FormatFloat('000.00', DF1 * 100) + '% '
+ 'DF2: ' + FormatFloat('000.00', DF2 * 100) + '% '
+ 'THD: ' + FormatFloat('000.00', THD * 100) + '%');
Memo1.Lines.Add("");
Memo1.Lines.Add('Angulos de disparo.'):
For i := 1 to NumPulSem do
  Begin
    if i mod 3 = 0 then
      begin
        Memo1.Lines.Add('Disparo ' + FormatFloat('00', i - 2) + ' ' +
FormatFloat('000.00', 360 * Disparo[i - 2]) + ' grados '
÷ 'Disparo ' + FormatFloat('00', i - 1) + ' ' + FormatFloat('000.00', 360 *
Disparo[i - 1]) + ' grados '
+ 'Disparo ' + FormatFloat('00', i) + ' ' + FormatFloat('000.00', 360 * Disparo[i])
+ ' grados')
      end;
    if (i = NumPulSem) and (NumPulSem mod 3 = 2) then
      begin
        Memo1.Lines.Add('Disparo ' + FormatFloat('00', i - 1) + ' ' +
FormatFloat('000.00', 360 * Disparo[i - 1]) + ' grados '
+ 'Disparo ' + FormatFloat('00', i) + ' ' + FormatFloat('000.00', 360 * Disparo[i])
+ ' grados')
      end;
    if (i = NumPulSem) and (NumPulSem mod 3 = 1) then
      begin
        Memo1.Lines.Add('Disparo ' + FormatFloat('00', i) + ' ' + FormatFloat('000.00',
360 * Disparo[i]) + ' grados')
      end;
    End; {End For i...}
  Memo1.Lines.Add("");
  Memo1.Lines.Add('Amplitud de los armónicos.'):
  For i := 1 to 24 do
    Begin
      if i mod 4 = 0 then
        begin

```

```

Memo1.Lines.Add('Bn(' + FormatFloat('00'. 2 * (i - 3) + 1) + ')': ' +
FormatFloat('000.00', Abs(SigmaBn[2 * (i - 3) + 1]) * 100) + '%
+ 'Bn(' + FormatFloat('00'. 2 * (i - 2) + 1) + ')': ' + FormatFloat('000.00',
Abs(SigmaBn[2 * (i - 2) + 1]) * 100) + '%
+ 'Bn(' + FormatFloat('00'. 2 * (i - 1) + 1) + ')': ' + FormatFloat('000.00',
Abs(SigmaBn[2 * (i - 1) + 1]) * 100) + '%
+ 'Bn(' + FormatFloat('00'. 2 * i + 1) + ')': ' + FormatFloat('000.00',
Abs(SigmaBn[2 * i + 1]) * 100) + '%
end;

```

```
End; {End For i...}
```

```
Memo1.Lines.Add("")
```

```
End;
```

```
Procedure Limites2;
```

```
var
```

```
  i : integer;
```

```
Begin
```

```
  IndMod := k / Paso;
```

```
  AncPen := T / (4 * NumPulSem - 2);
```

```
  For i := 1 to (NumPulSem - 1) do
```

```
    Begin
```

```
      If i = 1 then
```

```
        Begin
```

```
          Inicio[i] := AncPen / 2;
```

```
          Fin[i] := AncPen + AncPen / 2
```

```
        End {End If i...}
```

```
      Else
```

```
        Begin
```

```
          Inicio[i] := Fin[i - 1];
```

```
          Fin[i] := Fin[i - 1] + AncPen
```

```
        End; {End Else}
```

```
      End; {End For i...}
```

```
End;
```

```
Procedure Limites;
```

```
var
```

```
  i : integer;
```

```
Begin
```

```
  IndMod := k / Paso;
```

```
  MitAncMaxPul := T / (4 * (NumPulSem + 1));
```

```
  For i := 1 to (NumPulSem + 1) do
```

```
    Begin
```

```
      If i = 1 then
```

```
        Begin
```

```
          Inicio[i] := 0;
```

```
          Fin[i] := MitAncMaxPul
```

```
        End {End If i...}
```

```

Else
  Begin
    Inicio[i] := Fin[i - 1];
    Fin[i] := i * MitAncMaxPul
  End; {End Else}
End; {End For i...}
End;

```

Procedure Despliegamiento;

var

n, i : integer;

Begin

-For n := 1 to NumArm do

  Begin

    Sumabn := 0;

    If n mod 2 = 1 then

      Begin

        For i := 1 to NumPulSem do

          Begin

            tau := Disparo[i] \* T;

            If i mod 2 = 1 then

              Begin

                bn[n] := cos(2 \* n \* Pi \* tau / T)

              End {End If i...}

            Else

              Begin

                bn[n] := -cos(2 \* n \* Pi \* tau / T)

              End; {End Else}

              Sumabn := Sumabn + bn[n];

            End; {End For i...}

          End; {End If n...}

          SigmaBn[n] := Sumabn \* 4 \* Amp / (n \* Pi);

        End; {End For n...}

    SumaTHD := 0;

    SumaDF1 := 0;

    SumaDF2 := 0;

    For n := 1 to NumArm do

      Begin

        If n > 1 then

          Begin

            SumaTHD := SumaTHD + Sqr(SigmaBn[n]);

            SumaDF1 := SumaDF1 + Sqr(SigmaBn[n] / n);

            SumaDF2 := SumaDF2 + Sqr(SigmaBn[n] / Sqr(n))

          End; {End If n...}

      End; {End For n...}

    THD := Sqrt(SumaTHD) / SigmaBn[1];

    DF1 := Sqrt(SumaDF1) / SigmaBn[1];

```

DF2 := Sqrt(SumaDF2) / SigmaBn[1];
Memo1.Lines.Add('Mod: ' + FormatFloat('000.00', IndMod * 100) + '% '
+ 'Bn(1): ' + FormatFloat('000.00', SigmaBn[1] * 100) + '% '
+ 'DF1: ' + FormatFloat('000.00', DF1 * 100) + '% '
+ 'DF2: ' + FormatFloat('000.00', DF2 * 100) + '% '
+ 'THD: ' + FormatFloat('000.00', THD * 100) + '%');
Memo1.Lines.Add("");
Memo1.Lines.Add('Angulos de disparo. ');
  For i := 1 to NumPulSem do
    Begin
      if i mod 3 = 0 then
        begin
          Memo1.Lines.Add('Disparo ' + FormatFloat('00', i - 2) + ' = ' +
FormatFloat('000.00', 360 * Disparo[i - 2]) + ' grados '
+ 'Disparo ' + FormatFloat('00', i - 1) + ' = ' + FormatFloat('000.00', 360 *
Disparo[i - 1]) + ' grados '
+ 'Disparo ' + FormatFloat('00', i) + ' = ' + FormatFloat('000.00', 360 * Disparo[i])
+ ' grados')
          end;
        if (i = NumPulSem) and (NumPulSem mod 3 = 2) then
          begin
            Memo1.Lines.Add('Disparo ' + FormatFloat('00', i - 1) + ' = ' +
FormatFloat('000.00', 360 * Disparo[i - 1]) + ' grados '
+ 'Disparo ' + FormatFloat('00', i) + ' = ' + FormatFloat('000.00', 360 * Disparo[i])
+ ' grados')
            end;
          if (i = NumPulSem) and (NumPulSem mod 3 = 1) then
            begin
              Memo1.Lines.Add('Disparo ' + FormatFloat('00', i) + ' = ' + FormatFloat('000.00',
360 * Disparo[i]) + ' grados')
              end;
            End; {End For i...}
          Memo1.Lines.Add("");
          Memo1.Lines.Add('Amplitud de los armónicos. ');
          For i := 1 to 24 do
            Begin
              if i mod 4 = 0 then
                begin
                  Memo1.Lines.Add('Bn(' + FormatFloat('00', 2 * (i - 3) + 1) + '): ' +
FormatFloat('000.00', Abs(SigmaBn[2 * (i - 3) + 1]) * 100) + '% '
+ 'Bn(' + FormatFloat('00', 2 * (i - 2) + 1) + '): ' + FormatFloat('000.00',
Abs(SigmaBn[2 * (i - 2) + 1]) * 100) + '% '
+ 'Bn(' + FormatFloat('00', 2 * (i - 1) + 1) + '): ' + FormatFloat('000.00',
Abs(SigmaBn[2 * (i - 1) + 1]) * 100) + '% '
+ 'Bn(' + FormatFloat('00', 2 * i + 1) + '): ' + FormatFloat('000.00', Abs(SigmaBn[2
* i + 1]) * 100) + '% '
)
                  end;
                end;
            end;

```

```
End; {End For i...}
Memo1.Lines.Add("")
End;

procedure Graficar1;
var
  Grafico : TBitmap;
  x, y, y2, armi, Num1, AmpMaxSal: integer;
  w, tiempo, H, phi, R1, R2, R3, R4, R5, R6, R7, SumaAICuadrado : real;
  Frec, L1, L2, L3, L4, L5, C1, C2, C3, C4, C5, THDSalida, y2real : real;
  Polilinea : Array [1..202] of TPoint;
begin
  Frec:= StrToFloat(Form4.Edit18.Text);
  R1 := StrToFloat(Form4.Edit1.Text);
  C1 := StrToFloat(Form4.Edit2.Text);
  R2 := StrToFloat(Form4.Edit3.Text);
  L1 := StrToFloat(Form4.Edit4.Text);
  R3 := StrToFloat(Form4.Edit5.Text);
  R4 := StrToFloat(Form4.Edit6.Text);
  I2 := StrToFloat(Form4.Edit7.Text);
  I3 := StrToFloat(Form4.Edit8.Text);
  C2 := StrToFloat(Form4.Edit9.Text);
  R5 := StrToFloat(Form4.Edit10.Text);
  L4 := StrToFloat(Form4.Edit11.Text);
  C3 := StrToFloat(Form4.Edit12.Text);
  R6 := StrToFloat(Form4.Edit13.Text);
  C4 := StrToFloat(Form4.Edit14.Text);
  L5 := StrToFloat(Form4.Edit15.Text);
  C5 := StrToFloat(Form4.Edit16.Text);
  R7 := StrToFloat(Form4.Edit17.Text);
  Grafico := TBitmap.Create;
  Grafico.Height := 500;
  Grafico.Width := 720;
  Grafico.Canvas.Pen.Color := clMaroon;
  for y := 0 to 15 do
    begin
      Grafico.Canvas.MoveTo(10 + y * 50, 50);
      Grafico.Canvas.LineTo(10 + y * 50, 450)
    end;
  for x := 1 to 9 do
    begin
      Grafico.Canvas.MoveTo(10, 50 * x);
      Grafico.Canvas.LineTo(710, 50 * x)
    end;
  Grafico.Canvas.Pen.Color := clBlue;
  Polilinea[1] := Point(10, 250);
  For Num1 := (NumPulSem * 8 + 2) to 202 do
```



```
Polilinea[Num1] := Point(710, 250);
Num1 := 2;
for x := 1 to NumPulSem do
begin
  if x mod 2 = 1 then
  begin
    Polilinea[Num1] := Point(10 + Trunc(Disparo[x] * 700), 250);
    Inc(Num1);
    Polilinea[Num1] := Point(10 + Trunc(Disparo[x] * 700), 150);
    Inc(Num1)
  end
  else
  begin
    Polilinea[Num1] := Point(10 + Trunc(Disparo[x] * 700), 150);
    Inc(Num1);
    Polilinea[Num1] := Point(10 + Trunc(Disparo[x] * 700), 250);
    Inc(Num1)
  end;
end;
for x := NumPulSem downto 1 do
begin
  if x mod 2 = 1 then
  begin
    Polilinea[Num1] := Point(360 - Trunc(Disparo[x] * 700), 150);
    Inc(Num1);
    Polilinea[Num1] := Point(360 - Trunc(Disparo[x] * 700), 250);
    Inc(Num1)
  end
  else
  begin
    Polilinea[Num1] := Point(360 - Trunc(Disparo[x] * 700), 250);
    Inc(Num1);
    Polilinea[Num1] := Point(360 - Trunc(Disparo[x] * 700), 150);
    Inc(Num1)
  end;
end;
for x := 1 to NumPulSem do
begin
  if x mod 2 = 1 then
  begin
    Polilinea[Num1] := Point(360 + Trunc(Disparo[x] * 700), 250);
    Inc(Num1);
    Polilinea[Num1] := Point(360 + Trunc(Disparo[x] * 700), 350);
    Inc(Num1)
  end
  else
  begin
```

```

    Polilinea[Num1] := Point(360 + Trunc(Disparo[x] * 700), 350);
    Inc(Num1);
    Polilinea[Num1] := Point(360 + Trunc(Disparo[x] * 700), 250);
    Inc(Num1)
end;
end;
for x := NumPulSem downto 1 do
begin
    if x mod 2 = 1 then
    begin
        Polilinea[Num1] := Point(710 - Trunc(Disparo[x] * 700), 350);
        Inc(Num1);
        Polilinea[Num1] := Point(710 - Trunc(Disparo[x] * 700), 250);
        Inc(Num1)
    end
    else
    begin
        Polilinea[Num1] := Point(710 - Trunc(Disparo[x] * 700), 250);
        Inc(Num1);
        Polilinea[Num1] := Point(710 - Trunc(Disparo[x] * 700), 350);
        Inc(Num1)
    end;
end;
end;
Grafico.Canvas.PolyLine(Polilinea);
AmpMaxSal := 0;    { Preiniciación de variables AmpMaxSal, }
SumaAlCuadrado := 0; { SumaAlCuadrado y THDSalida. Son variables }
THDSalida := 0;   { utilizadas como salidas en la gráfica. }
for x := 0 to 700 do
begin
    Grafico.Canvas.Pen.Color := clLime;
    w := 2 * pi * Frec;
    tiempo := x * 2 * pi / (w * 700);
    y := 0;
    y2real := 0;
    y2 := 0;
    For armi := 1 to NumArm do
    begin
        if armi mod 2 = 1 then
        begin
            if Form4.RadioGroup1.ItemIndex = 0 then
            begin
                H := R2 / Sqrt(Sqr(R1 * R2 * C1 * armi * w) + Sqr(R1 + R2));
                phi := -ArcTan((R1 * R2 * C1 * armi * w) / (R1 + R2))
            end;
        end;
        if Form4.RadioGroup1.ItemIndex = 1 then
        begin
            H := R3 / Sqrt(Sqr(L1 * armi * w) + Sqr(R3)).
        end;
    end;
end;
end;

```

```

    phi := -ArcTan((L1 * armi * w) / R3)
  end;
  if Form4.RadioGroup1.ItemIndex = 2 then
    begin
      H := (L2 * armi * w) / Sqrt(Sqr(L2 * armi * w) + Sqr(R4));
      phi := (pi / 2) - ArcTan((L2 * armi * w) / R4)
    end;
  if Form4.RadioGroup1.ItemIndex = 3 then
    begin
      H := R5 / Sqrt(Sqr(L3 * armi * w - (1 / (C2 * armi * w))) + Sqr(R5));
      phi := -ArcTan((L3 * armi * w - (1 / (C2 * armi * w))) / R5)
    end;
  if Form4.RadioGroup1.ItemIndex = 4 then
    begin
      H := R6 / Sqrt(Sqr(R6 - L4 * R6 * C3 * Sqr(armi * w)) + Sqr(L4 * armi *
w));
      phi := -ArcTan((L4 * armi * w) / (R6 - L4 * R6 * C3 * Sqr(armi * w)))
    end;
  if Form4.RadioGroup1.ItemIndex = 5 then
    begin
      H := R7 / Sqrt(Sqr(L5 * armi * w - (1 / (C4 * armi * w))) + Sqr((R7 * C5 /
C4) + R7 - L5 * R7 * C5 * Sqr(armi * w)));
      phi := ArcTan((L5 * armi * w - (1 / (C4 * armi * w))) / ((R7 * C5 / C4) +
R7 - L5 * R7 * C5 * Sqr(armi * w)))
    end;
    y := y + Trunc(100 * SigmaBn[armi] * sin(armi * w * tiempo));
    y2real := y2real + (H * 100 * SigmaBn[armi] * sin(armi * w * tiempo + phi));
    If (x = 0) and (armi > 1) then SumaAlCuadrado := SumaAlCuadrado + Sqr(H *
SigmaBn[armi]); { Determinación }
    If (x = 1) and (armi = 1) then THDSalida := 100 * Sqrt(SumaAlCuadrado) /
(H * SigmaBn[1]) {de THDSalida. }
  end;
end;
y2 := Round(y2real);
If (AmpMaxSal - y2) < 0 then AmpMaxSal := y2; {Se determina la amplitud
máxima de la señal filtrada para mostrarla en la gráfica}
if x = 0 then Grafico.Canvas.MoveTo(x + 10, 250 - y2);
Grafico.Canvas.LineTo(x + 10, 250 - y2)
end; {for x := 0 to 700}
Grafico.Canvas.Font.Color := clMaroon;
Grafico.Canvas.Textout(10, 10, 'Frecuencia = ' + Form4.Edit18.Text + ' Hz');
{ Despliegamiento de Frecuencia}
Grafico.Canvas.Textout(160, 10, 'THD entrada = ' + FormatFloat('000.00', THD *
100) + '%'); {THD de entrada. THD de salida y }
Grafico.Canvas.Textout(310, 10, 'THD salida = ' + FormatFloat('000.00',
Abs(THDSalida)) + '%'); {Amplitud máxima de señal filtrada }

```

```

    Grafico.Canvas.Textout(460, 10, 'Amplitud máxima de salida = ' +
IntToStr(AmpMaxSal) + ' %'); {en la gráfica.      }
    Grafico.Canvas.Textout(410, 470, 'Amplitud de la señal de entrada = 100 % ');
    Grafico.Canvas.Textout(210, 470, 'Señal de salida: ');
    Grafico.Canvas.MoveTo(292, 477);
    Grafico.Canvas.LineTo(342, 477);
    Grafico.Canvas.Textout(10, 470, 'Señal de entrada: ');
    Grafico.Canvas.Pen.Color := ClBlue;
    Grafico.Canvas.MoveTo(100, 477);
    Grafico.Canvas.LineTo(150, 477);
    Form1.Image1.Picture.Graphic := Grafico;
    Grafico.Free
end;

```

```

procedure Graficar2;
var
    Grafico : TBitmap;
    x, y, y2, armi, Num1, AmpMaxSal : integer;
    w, tiempo, H, phi, R1, R2, R3, R4, R5, R6, R7, THDSalida : real;
    Frec, L1, L2, L3, L4, L5, C1, C2, C3, C4, C5, y2real, SumaAlCuadrado : real;
    Polilinea : Array [1..202] of TPoint;
begin
    Frec := StrToFloat(Form4.Edit18.Text);
    R1 := StrToFloat(Form4.Edit1.Text);
    C1 := StrToFloat(Form4.Edit2.Text);
    R2 := StrToFloat(Form4.Edit3.Text);
    L1 := StrToFloat(Form4.Edit4.Text);
    R3 := StrToFloat(Form4.Edit5.Text);
    R4 := StrToFloat(Form4.Edit6.Text);
    L2 := StrToFloat(Form4.Edit7.Text);
    L3 := StrToFloat(Form4.Edit8.Text);
    C2 := StrToFloat(Form4.Edit9.Text);
    R5 := StrToFloat(Form4.Edit10.Text);
    L4 := StrToFloat(Form4.Edit11.Text);
    C3 := StrToFloat(Form4.Edit12.Text);
    R6 := StrToFloat(Form4.Edit13.Text);
    C4 := StrToFloat(Form4.Edit14.Text);
    L5 := StrToFloat(Form4.Edit15.Text);
    C5 := StrToFloat(Form4.Edit16.Text);
    R7 := StrToFloat(Form4.Edit17.Text);
    Grafico := TBitmap.Create;
    Grafico.Height := 500;
    Grafico.Width := 720;
    Grafico.Canvas.Pen.Color := clMaroon;
    for y := 0 to 15 do
        begin
            Grafico.Canvas.MoveTo(10 + y * 50, 50);

```

```
Grafico.Canvas.LineTo(10 + y * 50, 450)
end;
for x := 1 to 9 do
begin
  Grafico.Canvas.MoveTo(10, 50 * x);
  Grafico.Canvas.LineTo(710, 50 * x)
end;
Grafico.Canvas.Pen.Color := clBlue;
Polilinea[1] := Point(10, 250);
For Num1 := (NumPulSem * 8) to 202 do
  Polilinea[Num1] := Point(710, 150);
Num1 := 2;
for x := 1 to NumPulSem do
begin
  if x mod 2 = 1 then
begin
  Polilinea[Num1] := Point(10 + Trunc(Disparo[x] * 700), 350);
  Inc(Num1);
  Polilinea[Num1] := Point(10 + Trunc(Disparo[x] * 700), 150);
  Inc(Num1)
end
else
begin
  Polilinea[Num1] := Point(10 + Trunc(Disparo[x] * 700), 150);
  Inc(Num1);
  Polilinea[Num1] := Point(10 + Trunc(Disparo[x] * 700), 350);
  Inc(Num1)
end;
end;
for x := NumPulSem downto 1 do
begin
  if x mod 2 = 1 then
begin
  Polilinea[Num1] := Point(360 - Trunc(Disparo[x] * 700), 150);
  Inc(Num1);
  Polilinea[Num1] := Point(360 - Trunc(Disparo[x] * 700), 350);
  Inc(Num1)
end
else
begin
  Polilinea[Num1] := Point(360 - Trunc(Disparo[x] * 700), 350);
  Inc(Num1);
  Polilinea[Num1] := Point(360 - Trunc(Disparo[x] * 700), 150);
  Inc(Num1)
end;
end;
for x := 1 to NumPulSem do
```

```

begin
  if x mod 2 = 1 then
    begin
      Polilinea[Num1] := Point(360 + Trunc(Disparo[x] * 700), 150);
      Inc(Num1);
      Polilinea[Num1] := Point(360 + Trunc(Disparo[x] * 700), 350);
      Inc(Num1)
    end
  else
    begin
      Polilinea[Num1] := Point(360 + Trunc(Disparo[x] * 700), 350);
      Inc(Num1);
      Polilinea[Num1] := Point(360 + Trunc(Disparo[x] * 700), 150);
      Inc(Num1)
    end;
  end;
end;
for x := NumPulSem downto 1 do
  begin
    if x mod 2 = 1 then
      begin
        Polilinea[Num1] := Point(710 - Trunc(Disparo[x] * 700), 350);
        Inc(Num1);
        Polilinea[Num1] := Point(710 - Trunc(Disparo[x] * 700), 150);
        Inc(Num1)
      end
    else
      begin
        Polilinea[Num1] := Point(710 - Trunc(Disparo[x] * 700), 150);
        Inc(Num1);
        Polilinea[Num1] := Point(710 - Trunc(Disparo[x] * 700), 350);
        Inc(Num1)
      end;
    end;
  end;
Grafico.Canvas.PolyLine(Polilinea);
AmpMaxSal := 0; { Preiniciación de variables AmpMaxSal. }
SumaAlCuadrado := 0; { SumaAlCuadrado y THDSalida. Son variables }
THDSalida := 0; { utilizadas como salidas en la gráfica. }
for x := 0 to 700 do
  begin
    Grafico.Canvas.Pen.Color := clLime;
    w := 2 * pi * Frec;
    tiempo := x * 2 * pi / (w * 700);
    y := 0;
    y2real := 0;
    y2 := 0;
    For armi := 1 to NumArm do
      begin

```

```

if armi mod 2 = 1 then
begin
  if Form4.RadioGroup1.ItemIndex = 0 then
  begin
    H := R2 / Sqrt(Sqr(R1 * R2 * C1 * armi * w) + Sqr(R1 + R2));
    phi := -ArcTan((R1 * R2 * C1 * armi * w) / (R1 + R2));
  end;
  if Form4.RadioGroup1.ItemIndex = 1 then
  begin
    H := R3 / Sqrt(Sqr(L1 * armi * w) + Sqr(R3));
    phi := -ArcTan((L1 * armi * w) / (R3));
  end;
  if Form4.RadioGroup1.ItemIndex = 2 then
  begin
    H := (L2 * armi * w) / Sqrt(Sqr(L2 * armi * w) + Sqr(R4));
    phi := (pi / 2) - ArcTan((L2 * armi * w) / (R4));
  end;
  if Form4.RadioGroup1.ItemIndex = 3 then
  begin
    H := R5 / Sqrt(Sqr(L3 * armi * w - (1 / (C2 * armi * w))) + Sqr(R5));
    phi := -ArcTan((L3 * armi * w - (1 / (C2 * armi * w))) / R5);
  end;
  if Form4.RadioGroup1.ItemIndex = 4 then
  begin
    H := R6 / Sqrt(Sqr(R6 - L4 * R6 * C3 * Sqr(armi * w)) + Sqr(L4 * armi *
w));
    phi := -ArcTan((L4 * armi * w) / (R6 - L4 * R6 * C3 * Sqr(armi * w)));
  end;
  if Form4.RadioGroup1.ItemIndex = 5 then
  begin
    H := R7 / Sqrt(Sqr(L5 * armi * w - (1 / (C4 * armi * w))) + Sqr((R7 * C5 /
C4) + R7 - L5 * R7 * C5 * Sqr(armi * w)));
    phi := ArcTan((L5 * armi * w - (1 / (C4 * armi * w))) / ((R7 * C5 / C4) +
R7 - L5 * R7 * C5 * Sqr(armi * w)));
  end;
  y := y + Trunc(100 * SigmaBn[armi] * sin(armi * w * tiempo));
  y2real := y2real + (H * 100 * SigmaBn[armi] * sin(armi * w * tiempo + phi));
  If (x = 0) and (armi > 1) then SumaAlCuadrado := SumaAlCuadrado + Sqr(H
* SigmaBn[armi]); { Determinación }
  If (x = 1) and (armi = 1) then THDSalida := 100 * Sqrt(SumaAlCuadrado) /
(H * SigmaBn[1]) {de THDSalida. };
end;
end;
y2 := Round(y2real);
If (AmpMaxSal - y2) < 0 then AmpMaxSal := y2; {Se determina la amplitud
máxima de la señal filtrada para mostrarla en la gráfica}
if x = 0 then Grafico.Canvas.MoveTo(x + 10, 250 - y2);

```

```

    Grafico.Canvas.LineTo(x + 10, 250 - y2)
    end; {for x := 0 to 700}
    Grafico.Canvas.Font.Color := clMaroon;
    Grafico.Canvas.Textout(10, 10, 'Frecuencia = ' + Form4.Edit18.Text + ' Hz');
    Grafico.Canvas.Textout(160, 10, 'THD entrada = ' + FormatFloat('000.00', THD *
100) + '%'); {THD de entrada, THD de salida y }
    Grafico.Canvas.Textout(310, 10, 'THD salida = ' + FormatFloat('000.00',
Abs(THDSalida)) + '%'); {Amplitud máxima de señal filtrada }
    Grafico.Canvas.Textout(460, 10, 'Amplitud máxima de salida = ' +
IntToStr(AmpMaxSal) + '%'); {en la gráfica. }
    Grafico.Canvas.Textout(410, 470, 'Amplitud de la señal de entrada = 100 % ');
    Grafico.Canvas.Textout(210, 470, 'Señal de salida: ');
    Grafico.Canvas.MoveTo(292, 477);
    Grafico.Canvas.LineTo(342, 477);
    Grafico.Canvas.Textout(10, 470, 'Señal de entrada: ');
    Grafico.Canvas.Pen.Color := clBlue;
    Grafico.Canvas.MoveTo(100, 477);
    Grafico.Canvas.LineTo(150, 477);
    Form1.Image1.Picture.Graphic := Grafico;
    Grafico.Free
end;

```

```
begin
```

```

If RadioGroup3.ItemIndex = 0 then Modula := 2;
If RadioGroup3.ItemIndex = 1 then Modula := 4;
If RadioGroup3.ItemIndex = 2 then Modula := 6;
If RadioGroup3.ItemIndex = 3 then Modula := 8;
If RadioGroup3.ItemIndex = 4 then Modula := 10;
If RadioGroup3.ItemIndex = 5 then Modula := 12;
If RadioGroup3.ItemIndex = 6 then Modula := 14;
If RadioGroup3.ItemIndex = 7 then Modula := 16;
If RadioGroup3.ItemIndex = 8 then Modula := 18;
If RadioGroup3.ItemIndex = 9 then Modula := 20;
Memo1.Clear;
NumPulSem := StrToInt(Edit1.Text);
If RadioGroup1.ItemIndex = 0 then
    Begin
        If RadioGroup2.ItemIndex = 0 then
            Begin
                For j := 1 to Paso do
                    Begin
                        IndMod := j / Paso;
                        AncMaxPul := 1 / (2 * NumPulSem);
                        AncPulMod := AncMaxPul * IndMod;
                        PriDis := (AncMaxPul - AncPulMod) / 2;
                        For i := 1 to NumPulSem do
                            Begin

```



```

If i mod 2 = 1 then
  Begin
    Disparo[i] := PriDis + (i div 2) * AncMaxPul;
  End {End If i...}
Else
  Begin
    Disparo[i] := Disparo[i - 1] + AncPulMod;
  End; {End Else}
End; {End For i...}
Desplegamiento;
if j = Modula then Graficar1
End; {End For j...}
End; {End if RadioGroup2...}
If RadioGroup2.ItemIndex = 1 then
  Begin
    For k := 1 to Paso do
      Begin
        Limites;
        For i := 2 to (NumPulSem + 1) do
          Begin
            j := -1;
            Repeat
              j := j + 1;
              If i mod 2 = 1 then
                Begin
                  tiempo := j * MitAncMaxPul / Iteracion;
                  y1 := tiempo / (Fin[i] - Inicio[i]);
                  y2 := IndMod * sin(2 * Pi * (Inicio[i] + tiempo));
                  Diferencia := Abs(y2 - y1);
                  If Diferencia < Error then
                    Begin
                      Disparo[i - 1] := Inicio[i] + tiempo
                    End; {End If Diferencia...}
                  End {End If i...}
                Else
                  Begin
                    tiempo := j * MitAncMaxPul / Iteracion;
                    y1 := tiempo / (Fin[i] - Inicio[i]);
                    y2 := IndMod * sin(2 * Pi * (Inicio[i] + tiempo));
                    Diferencia := Abs(y2 - (1 - y1));
                    If Diferencia < Error then
                      Begin
                        Disparo[i - 1] := Inicio[i] - tiempo
                      End; {End If Diferencia...}
                    End; {End Else}
                  Until (Diferencia < Error) or (j = Iteracion);
                End; {End For i...}
          End;
        End;
      End;
    End;
  End;

```

```

    Despliegamiento;
    if k = Modula then Graficar1
End; {End For k...}
End; {End If Seleccion...}
If RadioGroup2.ItemIndex = 2 then
Begin
  For k := 1 to Paso do
  Begin
    IndMod := k / Paso;
    MitAncMaxPul := T / (6 * (NumPulSem));
    For i := 1 to (NumPulSem) do
    Begin
      If i = 1 then
      Begin
        Inicio[i] := 0;
        Fin[i] := MitAncMaxPul
      End {End If i...}
      Else
      Begin
        Inicio[i] := Fin[i - 1];
        Fin[i] := i * MitAncMaxPul
      End; {End Else}
    End; {End For i...}
    For i := 2 to (NumPulSem) do
    Begin
      j := -1;
      Repeat
        j := j + 1;
        If i mod 2 = 1 then
        Begin
          tiempo := j * MitAncMaxPul / Iteracion;
          y1 := tiempo / (Fin[i] - Inicio[i]);
          y2 := IndMod * sin(2 * Pi * (Inicio[i] + tiempo));
          Diferencia := Abs(y2 - y1);
          If Diferencia < Error then
          Begin
            Disparo[i - 1] := Inicio[i] + tiempo
          End; {End If Diferencia...}
        End {End If i...}
      Else
      Begin
        tiempo := j * MitAncMaxPul / Iteracion;
        y1 := tiempo / (Fin[i] - Inicio[i]);
        y2 := IndMod * sin(2 * Pi * (Inicio[i] + tiempo));
        Diferencia := Abs(y2 - (1 - y1));
        If Diferencia < Error then
        Begin

```

```

        Disparo[i - 1] := Inicio[i] + tiempo
    End; {End If Diferencia...}
End; {End Else}
Until (Diferencia < Error) or (j = Iteracion);
End; {End For i...}
Disparo[NumPulSem] := T / 6;
Desplegamiento;
if k = Modula then Graficar1
End; {End For k...}
End; {End If Seleccion...}
If RadioGroup2.ItemIndex = 3 then
Begin
    For k := 1 to Paso do
    Begin
        Limites;
        For i := 2 to (NumPulSem + 1) do
        Begin
            j := -1;
            Repeat
                j := j + 1;
                If i mod 2 = 1 then
                Begin
                    tiempo := j * MitAncMaxPul / Iteracion;
                    y1 := tiempo / (Fin[i] - Inicio[i]);
                    y2 := 1.15 * sin(2 * Pi * (Inicio[i] + tiempo));
                    y3 := 0.19 * sin(6 * Pi * (Inicio[i] + tiempo));
                    y4 := IndMod * (y2 + y3);
                    Diferencia := Abs(y4 - y1);
                    If Diferencia < Error then
                    Begin
                        Disparo[i - 1] := Inicio[i] + tiempo
                    End; {End If Diferencia...}
                End {End If i...}
                Else
                Begin
                    tiempo := j * MitAncMaxPul / Iteracion;
                    y1 := tiempo / (Fin[i] - Inicio[i]);
                    y2 := 1.15 * sin(2 * Pi * (Inicio[i] + tiempo));
                    y3 := 0.19 * sin(6 * Pi * (Inicio[i] + tiempo));
                    y4 := IndMod * (y2 + y3);
                    Diferencia := Abs(y4 - (1 - y1));
                    If Diferencia < Error then
                    Begin
                        Disparo[i - 1] := Inicio[i] + tiempo
                    End; {End If Diferencia...}
                End; {End Else}
            Until (Diferencia < Error) or (j = Iteracion);
        End;
    End;
End;

```

```

    End; {End For i...}
  Desplegamiento;
  if k = Modula then Graficar1
End; {End For k...}
End; {End If Seleccion...}
If RadioGroup2.ItemIndex = 4 then
  Begin
    For k := 1 to Paso do
      Begin
        Limites;
        For i := 2 to (NumPulSem + 1) do
          Begin
            j := -1;
            Repeat
              j := j + 1;
              If i mod 2 = 1 then
                Begin
                  tiempo := j * MitAncMaxPul / Iteracion;
                  y1 := tiempo / (Fin[i] - Inicio[i]);
                  y2 := 1.15 * sin(2 * Pi * (Inicio[i] + tiempo));
                  y3 := 0.27 * sin(6 * Pi * (Inicio[i] + tiempo));
                  y4 := 0.029 * sin(18 * Pi * (Inicio[i] + tiempo));
                  y5 := IndMod * (y2 + y3 - y4);
                  Diferencia := Abs(y5 - y1);
                  If Diferencia < Error then
                    Begin
                      Disparo[i - 1] := Inicio[i] + tiempo
                    End; {End If Diferencia...}
                End {End If i...}
              Else
                Begin
                  tiempo := j * MitAncMaxPul / Iteracion;
                  y1 := tiempo / (Fin[i] - Inicio[i]);
                  y2 := 1.15 * sin(2 * Pi * (Inicio[i] + tiempo));
                  y3 := 0.27 * sin(6 * Pi * (Inicio[i] + tiempo));
                  y4 := 0.029 * sin(18 * Pi * (Inicio[i] + tiempo));
                  y5 := IndMod * (y2 + y3 - y4);
                  Diferencia := Abs(y5 - (1 - y1));
                  If Diferencia < Error then
                    Begin
                      Disparo[i - 1] := Inicio[i] + tiempo
                    End; {End If Diferencia...}
                End; {End Else}
              Until (Diferencia < Error) or (j = Iteracion);
            End; {End For i...}
          Desplegamiento;
        if k = Modula then Graficar1

```

```

End; {End For k...}
End; {End If Seleccion...}
If RadioGroup2.ItemIndex = 5 then
Begin
For k := 1 to Paso do
Begin
Limites;
For i := 2 to (NumPulSem + 1) do
Begin
j := -1;
Repeat
j := j + 1;
If i mod 2 = 1 then
Begin
tiempo := j * MitAncMaxPul / Iteracion;
y2 := (Inicio[i] + tiempo) * (4 * AmpTra) / T;
If y2 < IndMod then
Begin
y1 := tiempo / (Fin[i] - Inicio[i]);
Diferencia := Abs(y2 - y1);
End {End If y2...}
Else
Begin
y1 := tiempo / (Fin[i] - Inicio[i]);
y2 := IndMod;
Diferencia := Abs(y2 - y1);
End; {End Else}
If Diferencia < Error then
Begin
Disparo[i - 1] := Inicio[i] + tiempo
End; {End If Diferencia...}
End {End If i...}
Else
Begin
tiempo := j * MitAncMaxPul / Iteracion;
y2 := (Inicio[i] + tiempo) * (4 * AmpTra) / T;
If y2 < IndMod then
Begin
y1 := tiempo / (Fin[i] - Inicio[i]);
Diferencia := Abs(y2 - (1 - y1));
End {End If y2...}
Else
Begin
y1 := tiempo / (Fin[i] - Inicio[i]);
y2 := IndMod;
Diferencia := Abs(y2 - (1 - y1));
End; {End Else}

```

```
For k := 1 to Paso do
  Begin
    IndMod := k / Paso;
    AncPen := T / (6 * NumPulSem - 9);
    For i := 1 to (NumPulSem - 2) do
      Begin
        If i = 1 then
          Begin
            Inicio[i] := AncPen / 2;
            Fin[i] := AncPen + AncPen / 2
          End {End If i...}
        Else
          Begin
            Inicio[i] := Fin[i - 1];
            Fin[i] := Fin[i - 1] + AncPen
          End; {End Else}
        End; {End For i...}
      Disparo[1] := 0;
      For i := 1 to (NumPulSem - 2) do
        Begin
          j := -1;
          Repeat
            j := j + 1;
            If i mod 2 = 1 then
              Begin
                tiempo := j * AncPen / Iteracion2;
                y1 := ((2 * tiempo) / (Fin[i] - Inicio[i])) - 1;
                y2 := IndMod * sin(2 * Pi * (Inicio[i] + tiempo));
                Diferencia := Abs(y2 - y1);
                If Diferencia < Error then
                  Begin
                    Disparo[i + 1] := Inicio[i] + tiempo
                  End; {End If Diferencia...}
                End {End If i...}
              Else
                Begin
                  tiempo := j * AncPen / Iteracion2;
                  y1 := ((2 * tiempo) / (Fin[i] - Inicio[i])) - 1;
                  y2 := IndMod * sin(2 * Pi * (Inicio[i] + tiempo));
                  Diferencia := Abs(y2 + y1);
                  If Diferencia < Error then
                    Begin
                      Disparo[i + 1] := Inicio[i] + tiempo
                    End; {End If Diferencia...}
                  End; {End Else}
                Until (Diferencia < Error) or (j = Iteracion2);
              End; {End For i...}
```

```

Disparo[NumPulSem] := T / 6;
Desplegamiento2;
if k = Modula then Graficar2
End; {End For k...}
End; {End If Seleccion...}
If RadioGroup2.ItemIndex = 3 then
Begin
For k := 1 to Paso do
Begin
Limites2;
Disparo[1] := 0;
For i := 1 to (NumPulSem - 1) do
Begin
j := -1;
Repeat
j := j + 1;
If i mod 2 = 1 then
Begin
tiempo := j * AncPen / Iteracion2;
y1 := ((2 * tiempo) / (Fin[i] - Inicio[i])) - 1;
y2 := 1.15 * sin(2 * Pi * (Inicio[i] + tiempo));
y3 := 0.19 * sin(6 * Pi * (Inicio[i] + tiempo));
y4 := IndMod * (y2 + y3);
Diferencia := Abs(y4 - y1);
If Diferencia < Error then
Begin
Disparo[i + 1] := Inicio[i] + tiempo
End; {End If Diferencia...}
End {End If i...}
Else
Begin
tiempo := j * AncPen / Iteracion2;
y1 := ((2 * tiempo) / (Fin[i] - Inicio[i])) - 1;
y2 := 1.15 * sin(2 * Pi * (Inicio[i] + tiempo));
y3 := 0.19 * sin(6 * Pi * (Inicio[i] + tiempo));
y4 := IndMod * (y2 + y3);
Diferencia := Abs(y4 + y1);
If Diferencia < Error then
Begin
Disparo[i + 1] := Inicio[i] + tiempo
End; {End If Diferencia...}
End; {End Else}
Until (Diferencia < Error) or (j = Iteracion2);
End; {End For i...}
Desplegamiento2;
if k = Modula then Graficar2
End; {End For k...}

```

```

End; {End If Seleccion...}
If RadioGroup2.ItemIndex = 4 then
Begin
  For k := 1 to Paso do
  Begin
    Limites2;
    Disparo[1] := 0;
    For i := 1 to (NumPulSem - 1) do
    Begin
      j := -1;
      Repeat
      j := j + 1;
      If i mod 2 = 1 then
      Begin
        tiempo := j * AncPen / Iteracion2;
        y1 := ((2 * tiempo) / (Fin[i] - Inicio[i])) - 1;
        y2 := 1.15 * sin(2 * Pi * (Inicio[i] + tiempo));
        y3 := 0.27 * sin(6 * Pi * (Inicio[i] + tiempo));
        y4 := 0.029 * sin(18 * Pi * (Inicio[i] + tiempo));
        y5 := IndMod * (y2 + y3 - y4);
        Diferencia := Abs(y5 - y1);
        If Diferencia < Error then
        Begin
          Disparo[i + 1] := Inicio[i] + tiempo
        End; {End If Diferencia...}
      End {End If i...}
      Else
      Begin
        tiempo := j * AncPen / Iteracion2;
        y1 := ((2 * tiempo) / (Fin[i] - Inicio[i])) - 1;
        y2 := 1.15 * sin(2 * Pi * (Inicio[i] + tiempo));
        y3 := 0.27 * sin(6 * Pi * (Inicio[i] + tiempo));
        y4 := 0.029 * sin(18 * Pi * (Inicio[i] + tiempo));
        y5 := IndMod * (y2 + y3 - y4);
        Diferencia := Abs(y5 + y1);
        If Diferencia < Error then
        Begin
          Disparo[i + 1] := Inicio[i] + tiempo
        End; {End If Diferencia...}
      End; {End Else}
      Until (Diferencia < Error) or (j = Iteracion2);
    End; {End For i...}
  Desplegamiento2;
  if k = Modula then Graficar2
  End; {End For k...}
  End; {End If Seleccion...}
End; {End If RadioGroup2.ItemIndex = 4 then}

```



```

Begin
  For k := 1 to Paso do
    Begin
      Limites2;
      Disparo[1] := 0;
      For i := 1 to (NumPulSem - 1) do
        Begin
          j := -1;
          Repeat
            j := j + 1;
            If i mod 2 = 1 then
              Begin
                tiempo := j * AncPen / Iteracion2;
                y2 := (Inicio[i] + tiempo) * 6 / T;
                If y2 < IndMod then
                  Begin
                    y1 := ((2 * tiempo) / (Fin[i] - Inicio[i])) - 1;
                    Diferencia := Abs(y2 - y1);
                  End {End If y2...}
                Else
                  Begin
                    y1 := ((2 * tiempo) / (Fin[i] - Inicio[i])) - 1;
                    y2 := IndMod;
                    Diferencia := Abs(y2 - y1);
                  End; {End Else}
                If Diferencia < Error then
                  Begin
                    Disparo[i + 1] := Inicio[i] + tiempo
                  End; {End If Diferencia...}
                End {End If i...}
            Else
              Begin
                tiempo := j * AncPen / Iteracion2;
                y2 := (Inicio[i] + tiempo) * 6 / T;
                If y2 < IndMod then
                  Begin
                    y1 := ((2 * tiempo) / (Fin[i] - Inicio[i])) - 1;
                    Diferencia := Abs(y2 + y1);
                  End {End If y2...}
                Else
                  Begin
                    y1 := ((2 * tiempo) / (Fin[i] - Inicio[i])) - 1;
                    y2 := IndMod;
                    Diferencia := Abs(y2 + y1);
                  End; {End Else}
                If Diferencia < Error then
                  Begin

```

```

        Disparo[i + 1] := Inicio[i] + tiempo
    End; {End If Diferencia...}
    End; {End Else}
    Until (Diferencia < Error) or (j = Iteracion2);
    End; {End For i...}
    Desplegamiento2;
    if k = Modula then Graficar2
    End; {End For k...}
    End; {End If Seleccion...}
    End; {If RadioGroup1.ItemIndex = 1}
end;
```

```

procedure TForm2.Filtro1Click(Sender: TObject);
begin
    Form4.Show
end;
```

end.

```

unit Unit3;
```

```

interface
```

```

uses
```

```

    Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
    Menus, ExtCtrls, StdCtrls, Grids;
```

```

type
```

```

TForm3 = class(TForm)
    Panel1: TPanel;
    MainMenu1: TMainMenu;
    Opciones1: TMenuItem;
    Modulacin1: TMenuItem;
    Salir1: TMenuItem;
    RadioGroup1: TRadioGroup;
    Label1: TLabel;
    Memo1: TMemo;
    Button1: TButton;
    Edit1: TEdit;
    Label2: TLabel;
    StringGrid1: TStringGrid;
    Label3: TLabel;
    StringGrid2: TStringGrid;
    Button2: TButton;
    Button3: TButton;
    Label4: TLabel;
    Salir2: TMenuItem;
```

---

```
Filtro1: TMenuItem;
procedure Modulacin1Click(Sender: TObject);
procedure Salir1Click(Sender: TObject);
procedure Salir2Click(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure StringGrid1Click(Sender: TObject);
procedure StringGrid2Click(Sender: TObject);
procedure Filtro1Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form3: TForm3;

implementation

uses Unit2, Unit1, Unit4;

{$R *.DFM}

procedure TForm3.Modulacin1Click(Sender: TObject);
begin
  Form2.Show
end;

procedure TForm3.Salir1Click(Sender: TObject);
begin
  Form3.Hide
end;

procedure TForm3.Salir2Click(Sender: TObject);
begin
  Application.Terminate
end;

procedure TForm3.Button3Click(Sender: TObject);
var
  i: integer;
begin
  Edit1.Clear;
  Memo1.Clear;
  with StringGrid1 do
```

---

```

for i := 0 to ColCount - 1 do
  StringGrid1.Cells[i, 0] := "";
with StringGrid2 do
  for i := 0 to ColCount - 1 do
    StringGrid2.Cells[i, 0] := "";
end;

```

```

procedure TForm3.Button2Click(Sender: TObject);

```

```

type

```

```

  arreglo = array [1 .. 5000] of real;
  matriz = array [1.. 25, 1.. 25] of real;
  vector = array [1.. 25] of real;

```

```

var

```

```

  Sumabn, DetMatDer, y2real: real;
  tau, THD, SumaTHD, DF1, SumaDF1, H, phi, R1, R2, R3, R4, R5, R6, R7 : real;
  Frec, L1, L2, L3, L4, L5, C1, C2, C3, C4, C5, THDSalida : real;
  DF2, SumaDF2, error1, error2, tiempo, w, SumaAlCuadrado : real;
  j, i, RenCol, ciclo, x, y, signo, y2, Num1, AmpMaxSal : integer;
  ba, SigmaBn : arreglo;
  arm, VecSus, AngDisRad, AngDisRadi, DetMatSus: vector;
  m, m1: matriz;
  grafico : TBitmap;
  Polilinea : Array [1..202] of TPoint;

```

```

const

```

```

  T = 1;
  Amp = 1;
  NumArm = 5000;

```

```

Function Determinante(mat: matriz; rc: integer): real;

```

```

Var

```

```

  inicio, a, b, c: integer;
  auxiliar, DetMat: real;

```

```

Begin

```

```

  inicio:= 1;
  For a:= 1 to rc - 1 do
    Begin
      inicio:= inicio + 1;
      For b:= inicio to rc do
        Begin
          auxiliar:= mat[b, a];
          For c:= 1 to rc do
            Begin
              mat[b, c]:= mat[b, c] - mat[a, c] * auxiliar / mat[a, a];
            End; {End For c...}
          End; {End For b...}
        End; {End For a...}
      DetMat:= 1;
    End;

```

```

For b:= 1 to rc do
  Begin
    DetMat:= DetMat * mat[b, b];
  End; {End For b...}
Determinante:= DetMat
End;

Procedure Angulos;
var
  i, g, h, j : integer;
Begin
  RenCol := StrToInt(Edit1.Text);
  For i := 1 to RenCol do
    Begin
      For g:= 1 to RenCol do
        Begin
          For h:= 1 to RenCol do
            Begin
              m1[g, h]:= m[g, h];
            End; {End For h...}
          End; {End For g...}
        End
      For j:= 1 to RenCol do
        Begin
          m1[j, i]:= VecSus[j];
        End; {End For j...}
      DetMatSus[i]:= Determinante(m1, rencol)
    End; {End For i...}
  DetMatDer:= Determinante(m, rencol);
  error1:= 0;
  error2:= 0;
  For i:= 1 to rencol do
    Begin
      error1:= error1 + Abs(DetMatSus[i] / DetMatDer);
      error2:= error2 + Abs(AngDisRad[i]);
      AngDisRad[i]:= AngDisRad[i] + (DetMatSus[i] / DetMatDer);
      If (AngDisRad[i] - AngDisRad[i - 1]) < 0.001 then Error2:= 1.6 * RenCol;
    End; {End For i...}
  End;

Procedure Errores;
Begin
  RenCol := StrToInt(Edit1.Text);
  If ((error1 / RenCol) < 0.00001) then
    Begin
      If RadioGroup1.ItemIndex = 0 then
        begin
          Memo1.Lines.Add("");
        end
    end

```

```

    Memo1.Lines.Add('Sistema unipolar convergente. Angulos determinados.')
end;
If RadioGroup1.ItemIndex = 1 then
begin
    Memo1.Lines.Add("");
    Memo1.Lines.Add('Sistema bipolar convergente. Angulos determinados.')
end;
End {End If...}
Else
Begin
    Memo1.Clear;
    Memo1.Lines.Add('Sistema no convergente. Error en los ángulos determinados.')
End; {End Else}
End;

Procedure ImprimirAngulos;
var
    i, n : integer;
Begin
    RenCol := StrToInt(Edit1.Text);
    Memo1.Lines.Add('Angulos de disparo:');
    For i: 1 to rencol do
        Begin
            if i mod 3 = 0 then
                begin
                    Memo1.Lines.Add('Disparo ' + FormatFloat('00', i - 2) + ' = ' + FormatFloat('000.00',
180 * AngDisRad[i - 2] / pi) + ' grados '
                    + 'Disparo ' + FormatFloat('00', i - 1) + ' = ' + FormatFloat('000.00', 180 *
AngDisRad[i - 1] / pi) + ' grados '
                    + 'Disparo ' + FormatFloat('00', i) + ' = ' + FormatFloat('000.00', 180 * AngDisRad[i]
/ pi) + ' grados')
                end;
            if (i - RenCol) and (RenCol mod 3 = 2) then
                begin
                    Memo1.Lines.Add('Disparo ' + FormatFloat('00', i - 1) + ' = ' + FormatFloat('000.00',
180 * AngDisRad[i - 1] / pi) + ' grados '
                    + 'Disparo ' + FormatFloat('00', i) + ' = ' + FormatFloat('000.00', 180 * AngDisRad[i]
/ pi) + ' grados')
                end;
            if (i = RenCol) and (RenCol mod 3 = 1) then
                begin
                    Memo1.Lines.Add('Disparo ' + FormatFloat('00', i) + ' = ' + FormatFloat('000.00',
180 * AngDisRad[i] / pi) + ' grados')
                end;
            End;
        Memo1.Lines.Add("");
        If (RadioGroup1.ItemIndex = 0) and ((error1 / RenCol) < 0.00001) then

```

```

Begin
For n := 1 to NumArm do
  Begin
    Sumabn := 0;
    If n mod 2 = 1 then
      Begin
        For i := 1 to rencol do
          Begin
            tau := AngDisRad[i] * T / (2 * Pi);
            If i mod 2 = 1 then
              Begin
                bn[n] := cos(2 * n * Pi * tau / T)
              End {End If i...}
            Else
              Begin
                bn[n] := -cos(2 * n * Pi * tau / T)
              End; {End Else}
            Sumabn := Sumabn + bn[n];
          End; {End For i...}
        End; {End If n...}
        SigmaBn[n] := Sumabn * 4 * Amp / (n * Pi);
      End; {End For n...}
    SumaTHD := 0;
    SumaDF1 := 0;
    SumaDF2 := 0;
    For n := 1 to NumArm do
      Begin
        If n > 1 then
          Begin
            SumaTHD := SumaTHD + Sqr(SigmaBn[n]);
            SumaDF1 := SumaDF1 + Sqr(SigmaBn[n] / n);
            SumaDF2 := SumaDF2 + Sqr(SigmaBn[n] / Sqr(n))
          End; {End If n...}
        End; {End For n...}
    THD := Sqrt(SumaTHD) / SigmaBn[1];
    DF1 := Sqrt(SumaDF1) / SigmaBn[1];
    DF2 := Sqrt(SumaDF2) / SigmaBn[1];
    Memol.Lines.Add('Figuras de mérito:');
    Memol.Lines.Add('Bn(1): ' + FormatFloat('000.00', SigmaBn[1] * 100) + '%'
    + 'DF1: ' + FormatFloat('000.00', DF1 * 100) + '%'
    + 'DF2: ' + FormatFloat('000.00', DF2 * 100) + '%'
    + 'THD: ' + FormatFloat('000.00', THD * 100) + '%');
    Memol.Lines.Add("");
    Memol.Lines.Add('Amplitud de los armónicos:');
    For i := 1 to 24 do
      Begin
        if i mod 4 = 0 then

```

```

begin
  Memo1.Lines.Add('Bn(' + FormatFloat('00', 2 * (i - 3) + 1) + ');' +
FormatFloat('000.00', Abs(SigmaBn[2 * (i - 3) + 1]) * 100) + '%
  + 'Bn(' + FormatFloat('00', 2 * (i - 2) + 1) + ');' + FormatFloat('000.00',
Abs(SigmaBn[2 * (i - 2) + 1]) * 100) + '%
  + 'Bn(' + FormatFloat('00', 2 * (i - 1) + 1) + ');' + FormatFloat('000.00',
Abs(SigmaBn[2 * (i - 1) + 1]) * 100) + '%
  + 'Bn(' + FormatFloat('00', 2 * i + 1) + ');' + FormatFloat('000.00', Abs(SigmaBn[2
* i + 1]) * 100) + '%
  end;
End; {End For i...}
End;
-- If (RadioGroup1.ItemIndex = 1) and ((error1 / RenCol) < 0.00001) then
Begin
  AngDisRad1[1] := 0;
  For n := 1 to rencol do
    Begin
      AngDisRad1[n + 1] := AngDisRad[n]
    End;
  For n := 1 to NumArm do
    Begin
      Sumabn := 0;
      if n mod 2 = 1 then
        Begin
          For i := 1 to rencol + 1 do
            Begin
              tau := AngDisRad1[i] * T / (2 * Pi);
              If i mod 2 = 1 then
                Begin
                  If i = 1 then
                    Begin
                      bn[n] := 1
                    End {End If i...};
                  else
                    Begin
                      bn[n] := 2 * cos(2 * n * Pi * tau / T)
                    End {End Else};
                End {End If i...}
              Else
                Begin
                  bn[n] := -2 * cos(2 * n * Pi * tau / T)
                End; {End Else}
              Sumabn := Sumabn + bn[n];
            End; {End For i...}
          End; {End If n...}
          SigmaBn[n] := Sumabn * 4 * Amp / (n * Pi);
        End; {End For n...}

```



```

SumaTHD := 0;
SumaDF1 := 0;
SumaDF2 := 0;
For n := 1 to NumArm do
  Begin
    If n > 1 then
      Begin
        SumaTHD := SumaTHD + Sqr(SigmaBn[n]);
        SumaDF1 := SumaDF1 + Sqr(SigmaBn[n] / n);
        SumaDF2 := SumaDF2 + Sqr(SigmaBn[n] / Sqr(n))
      End; {End If n...}
    End; {End For n...}
  THD := Sqrt(SumaTHD) / Abs(SigmaBn[1]);
  DF1 := Sqrt(SumaDF1) / Abs(SigmaBn[1]);
  DF2 := Sqrt(SumaDF2) / Abs(SigmaBn[1]);
  Memo1.Lines.Add('Figuras de mérito:');
  Memo1.Lines.Add('Bn(1): ' + FormatFloat('000.00', Abs(SigmaBn[1]) * 100) + '%
+ 'DF1: ' + FormatFloat('000.00', DF1 * 100) + '%
+ 'DF2: ' + FormatFloat('000.00', DF2 * 100) + '%
+ 'THD: ' + FormatFloat('000.00', THD * 100) + '%');
  Memo1.Lines.Add("");
  Memo1.Lines.Add('Amplitud de los armónicos. ');
  For i := 1 to 24 do
    Begin
      if i mod 4 = 0 then
        begin
          Memo1.Lines.Add('Bn(' + FormatFloat('00', 2 * (i - 3) + 1) + '); ' +
FormatFloat('000.00', Abs(SigmaBn[2 * (i - 3) + 1]) * 100) + '%
+ 'Bn(' + FormatFloat('00', 2 * (i - 2) + 1) + '); ' + FormatFloat('000.00',
Abs(SigmaBn[2 * (i - 2) + 1]) * 100) + '%
+ 'Bn(' + FormatFloat('00', 2 * (i - 1) + 1) + '); ' + FormatFloat('000.00',
Abs(SigmaBn[2 * (i - 1) + 1]) * 100) + '%
+ 'Bn(' + FormatFloat('00', 2 * i + 1) + '); ' + FormatFloat('000.00', Abs(SigmaBn[2
* i + 1]) * 100) + '%
          end;
        End; {End For i...}
      End;
    End;
  begin
  Frec := StrToFloat(Form4.Edit18.Text);
  R1 := StrToFloat(Form4.Edit1.Text);
  C1 := StrToFloat(Form4.Edit2.Text);
  R2 := StrToFloat(Form4.Edit3.Text);
  L1 := StrToFloat(Form4.Edit4.Text);
  R3 := StrToFloat(Form4.Edit5.Text);

```

```

R4 := StrToFloat(Form4.Edit6.Text);
L2 := StrToFloat(Form4.Edit7.Text);
L3 := StrToFloat(Form4.Edit8.Text);
C2 := StrToFloat(Form4.Edit9.Text);
R5 := StrToFloat(Form4.Edit10.Text);
L4 := StrToFloat(Form4.Edit11.Text);
C3 := StrToFloat(Form4.Edit12.Text);
R6 := StrToFloat(Form4.Edit13.Text);
C4 := StrToFloat(Form4.Edit14.Text);
L5 := StrToFloat(Form4.Edit15.Text);
C5 := StrToFloat(Form4.Edit16.Text);
R7 := StrToFloat(Form4.Edit17.Text);
Memo1.Clear;
RenCol := StrToInt(Edit1.Text);
With StringGrid1 do
  For i:= 0 to ColCount - 1 do
    Begin
      arm[i + 1] := StrToFloat(Cells[i, 0])
    End; {End For i...}
With StringGrid2 do
  For i:= 0 to ColCount - 1 do
    Begin
      AngDisRad[i + 1] := StrToFloat(Cells[i, 0]) * pi / 180
    End; {End For i...}
ciclo := 0;
If RadioGroup1.ItemIndex = 0 then
  Begin
    Repeat
      For i:= 1 to RenCol do
        Begin
          VecSus[i] := 0;
          For j:= 1 to RenCol do
            Begin
              If j mod 2 = 1 then
                Begin
                  m[i, j] := - SIN(arm[i] * AngDisRad[j]);
                  VecSus[i] := VecSus[i] - (COS(arm[i] * AngDisRad[j])) * arm[i]
                End
              Else
                Begin
                  m[i, j] := SIN(arm[i] * AngDisRad[j]);
                  VecSus[i] := VecSus[i] + (COS(arm[i] * AngDisRad[j])) * arm[i]
                End;
            End; {End For j...}
          End; {End For i...}
        Angulos;
        ciclo := ciclo + 1;

```

```

Until (ciclo = 25) or ((error1 / RenCol) < 0.00001) or ((error2 / Rencol) > 1.58);
ImprimirAngulos;
Errores;
if ((error1 / RenCol) < 0.00001) then
  begin {if ((error1 / RenCol) < 0.00001)}
    Grafico := TBitmap.Create;
    Grafico.Height := 500;
    Grafico.Width := 720;
    Grafico.Canvas.Pen.Color := clMaroon;
    for y := 0 to 15 do
      begin
        Grafico.Canvas.MoveTo(10 + y * 50, 50);
        Grafico.Canvas.LineTo(10 + y * 50, 450)
      end;
    for x := 1 to 9 do
      begin
        Grafico.Canvas.MoveTo(10, 50 * x);
        Grafico.Canvas.LineTo(710, 50 * x)
      end;
    Grafico.Canvas.Pen.Color := clBlue;
    Polilinea[1] := Point(10, 250);
    For Num1 := (RenCol * 8 + 2) to 202 do
      Polilinea[Num1] := Point(710, 250);
    Num1 := 2;
    for x := 1 to RenCol do
      begin
        if x mod 2 = 1 then
          begin
            Polilinea[Num1] := Point(10 + Trunc(AngDisRad[x] * 350 / pi), 250);
            Inc(Num1);
            Polilinea[Num1] := Point(10 + Trunc(AngDisRad[x] * 350 / pi), 150);
            Inc(Num1)
          end
        else
          begin
            Polilinea[Num1] := Point(10 + Trunc(AngDisRad[x] * 350 / pi), 150);
            Inc(Num1);
            Polilinea[Num1] := Point(10 + Trunc(AngDisRad[x] * 350 / pi), 250);
            Inc(Num1)
          end;
        end;
      end;
    for x := RenCol downto 1 do
      begin
        if x mod 2 = 1 then
          begin
            Polilinea[Num1] := Point(360 - Trunc(AngDisRad[x] * 350 / pi), 150);
            Inc(Num1);
          end;
        end;
      end;

```

```

    Polilinea[Num1] := Point(360 - Trunc(AngDisRad[x] * 350 / pi), 250);
    Inc(Num1)
end
else
begin
    Polilinea[Num1] := Point(360 - Trunc(AngDisRad[x] * 350 / pi), 250);
    Inc(Num1);
    Polilinea[Num1] := Point(360 - Trunc(AngDisRad[x] * 350 / pi), 150);
    Inc(Num1)
end;
end;
for x := 1 to RenCol do
begin
    if x mod 2 = 1 then
    begin
        Polilinea[Num1] := Point(360 + Trunc(AngDisRad[x] * 350 / pi), 250);
        Inc(Num1);
        Polilinea[Num1] := Point(360 + Trunc(AngDisRad[x] * 350 / pi), 350);
        Inc(Num1)
    end
    else
    begin
        Polilinea[Num1] := Point(360 + Trunc(AngDisRad[x] * 350 / pi), 350);
        Inc(Num1);
        Polilinea[Num1] := Point(360 + Trunc(AngDisRad[x] * 350 / pi), 250);
        Inc(Num1)
    end;
end;
for x := RenCol downto 1 do
begin
    if x mod 2 = 1 then
    begin
        Polilinea[Num1] := Point(710 - Trunc(AngDisRad[x] * 350 / pi), 350);
        Inc(Num1);
        Polilinea[Num1] := Point(710 - Trunc(AngDisRad[x] * 350 / pi), 250);
        Inc(Num1)
    end
    else
    begin
        Polilinea[Num1] := Point(710 - Trunc(AngDisRad[x] * 350 / pi), 250);
        Inc(Num1);
        Polilinea[Num1] := Point(710 - Trunc(AngDisRad[x] * 350 / pi), 350);
        Inc(Num1)
    end;
end;
end;
Grafico.Canvas.PolyLine(Polilinea);
AmpMaxSal := 0; { Preiniciación de variables AmpMaxSal, }

```

```

SumaAlCuadrado := 0; { SumaAlCuadrado y THDSalida. Son variables }
THDSalida := 0; { utilizadas como salidas en la gráfica. }
For x := 0 to 700 do
begin {For x := 0 to 700}
  Grafico.Canvas.Pen.Color := clLime;
  w := 2 * pi * Frec;
  tiempo := x * 2 * pi / ( w * 700);
  y := 0;
  y2real := 0;
  y2 := 0;
  For i := 1 to NumArm do
  begin
    if i mod 2 = 1 then
    begin
      if Form4.RadioGroup1.ItemIndex = 0 then
      begin
        H := R2 / Sqrt(Sqr(R1 * R2 * C1 * i * w) + Sqr(R1 + R2));
        phi := -ArcTan((R1 * R2 * C1 * i * w) / (R1 + R2))
      end;
      if Form4.RadioGroup1.ItemIndex = 1 then
      begin
        H := R3 / Sqrt(Sqr(L1 * i * w) + Sqr(R3));
        phi := -ArcTan((L1 * i * w) / R3)
      end;
      if Form4.RadioGroup1.ItemIndex = 2 then
      begin
        H := (L2 * i * w) / Sqrt(Sqr(L2 * i * w) + Sqr(R4));
        phi := (pi / 2) - ArcTan((L2 * i * w) / R4)
      end;
      if Form4.RadioGroup1.ItemIndex = 3 then
      begin
        H := R5 / Sqrt(Sqr(L3 * i * w - (1 / (C2 * i * w))) + Sqr(R5));
        phi := -ArcTan((L3 * i * w - (1 / (C2 * i * w))) / R5)
      end;
      if Form4.RadioGroup1.ItemIndex = 4 then
      begin
        H := R6 / Sqrt(Sqr(R6 - L4 * R6 * C3 * Sqr(i * w)) + Sqr(L4 * i * w));
        phi := -ArcTan((L4 * i * w) / (R6 - L4 * R6 * C3 * Sqr(i * w)))
      end;
      if Form4.RadioGroup1.ItemIndex = 5 then
      begin
        H := R7 / Sqrt(Sqr(L5 * i * w - (1 / (C4 * i * w))) + Sqr((R7 * C5 / C4) +
R7 - L5 * R7 * C5 * Sqr(i * w)));
        phi := ArcTan((L5 * i * w - (1 / (C4 * i * w))) / ((R7 * C5 / C4) + R7 - L5
* R7 * C5 * Sqr(i * w)))
      end;
      y := y + Trunc(100 * SigmaBn[i] * sin(i * w * tiempo));
    end;
  end;
end;

```

```

        y2real:= y2real + (H * 100 * SigmaBn[i] * sin(i * w * tiempo + phi));
        If (x = 0) and (i > 1) then SumaAlCuadrado := SumaAlCuadrado + Sqr(H *
SigmaBn[i]); { Determinación }
        If (x = 1) and (i = 1) then THDSalida := 100 * Sqrt(SumaAlCuadrado) / (H *
SigmaBn[1]) {de THDSalida. }
    end;
    end;
    y2 := Round(y2real);
    If (AmpMaxSal - y2) < 0 then AmpMaxSal := y2; {Se determina la amplitud
máxima de la señal filtrada para mostrarla en la gráfica}
    if x = 0 then Grafico.Canvas.MoveTo(x + 10, 250 - y2);
    Grafico.Canvas.LineTo(x + 10, 250)
    end; {For x := 0 to 700}
    Grafico.Canvas.Font.Color := clMaroon;
    Grafico.Canvas.Textout(10, 10, 'Frecuencia = ' + Form4.Edit18.Text + ' Hz');
    Grafico.Canvas.Textout(160, 10, 'THD entrada = ' + FormatFloat('000.00', THD *
100) + '%'); {THD de entrada, THD de salida y }
    Grafico.Canvas.Textout(310, 10, 'THD salida = ' + FormatFloat('000.00',
Abs(THDSalida)) + '%'); {Amplitud máxima de señal filtrada }
    Grafico.Canvas.Textout(460, 10, 'Amplitud máxima de salida ' +
IntToStr(AmpMaxSal) + '%'); {en la gráfica. }
    Grafico.Canvas.Textout(410, 470, 'Amplitud de la señal de entrada - 100 %');
    Grafico.Canvas.Textout(210, 470, 'Señal de salida: ');
    Grafico.Canvas.MoveTo(292, 477);
    Grafico.Canvas.LineTo(342, 477);
    Grafico.Canvas.Textout(10, 470, 'Señal de entrada: ');
    Grafico.Canvas.Pen.Color := clBlue;
    Grafico.Canvas.MoveTo(100, 477);
    Grafico.Canvas.LineTo(150, 477);
    Form1.Image1.Picture.Graphic := Grafico;
    Grafico.Free
    end; {if ((error1 / RenCol) < 0.00001)}
end. {End if RadioGroup1.ItemIndex = 0...}
If RadioGroup1.ItemIndex = 1 then
Begin
Repeat
For i:= 1 to RenCol do
Begin
VecSus[i]:= - 1 / (2 * arm[i]);
For j:= 1 to RenCol do
Begin
If j mod 2 = 1 then
Begin
m[i, j]:= SIN(arm[i] * AngDisRad[j]);
VecSus[i]:= VecSus[i] + (COS(arm[i] * AngDisRad[j])) / arm[i]
End
Else

```

```

Begin
  m[i, j]:= - SIN(arm[i] * AngDisRad[j]);
  VecSus[i]:= VecSus[i] - (COS(arm[i] * AngDisRad[j])) / arm[i]
End;
End; {End For j...}
End; {End For i...}
Angulos;
ciclo:= ciclo + 1;
Until (ciclo = 25) or ((error1 / RenCol) < 0.00001) or ((error2 / Rencol) > 1.58);
ImprimirAngulos;
Errores;
If ((error1 / RenCol) < 0.00001) then
begin {if ((error1 / RenCol) < 0.00001)}
  Grafico := TBitmap.Create;
  Grafico.Height := 500;
  Grafico.Width := 720;
  Grafico.Canvas.Pen.Color := clMaroon;
  for y := 0 to 15 do
  begin
    Grafico.Canvas.MoveTo(10 + y * 50, 50);
    Grafico.Canvas.LineTo(10 + y * 50, 450)
  end;
  for x := 1 to 9 do
  begin
    Grafico.Canvas.MoveTo(10, 50 * x);
    Grafico.Canvas.LineTo(710, 50 * x)
  end;
  Grafico.Canvas.Pen.Color := clBlue;
  For Num1 := (RenCol * 8) to 202 do
    Polilinea[Num1] := Point(710, 250);
  Num1 := 1;
  for x := 1 to RenCol + 1 do
  begin
    if x mod 2 = 1 then
    begin
      Polilinea[Num1] := Point(10 + Trunc(AngDisRad1[x] * 350 / pi), 350);
      Inc(Num1);
      Polilinea[Num1] := Point(10 + Trunc(AngDisRad1[x] * 350 / pi), 150);
      Inc(Num1)
    end
    else
    begin
      Polilinea[Num1] := Point(10 + Trunc(AngDisRad1[x] * 350 / pi), 150);
      Inc(Num1);
      Polilinea[Num1] := Point(10 + Trunc(AngDisRad1[x] * 350 / pi), 350);
      Inc(Num1)
    end;
  end;
end;

```

```
end;
for x := RenCol + 1 downto 1 do
begin
  if x mod 2 = 1 then
  begin
    Polilinea[Num1] := Point(360 - Trunc(AngDisRad1[x] * 350 / pi), 150);
    Inc(Num1);
    Polilinea[Num1] := Point(360 - Trunc(AngDisRad1[x] * 350 / pi), 350);
    Inc(Num1)
  end
  else
  begin
    Polilinea[Num1] := Point(360 - Trunc(AngDisRad1[x] * 350 / pi), 350);
    Inc(Num1);
    Polilinea[Num1] := Point(360 - Trunc(AngDisRad1[x] * 350 / pi), 150);
    Inc(Num1)
  end;
end;
for x := 1 to RenCol + 1 do
begin
  if x mod 2 = 1 then
  begin
    Polilinea[Num1] := Point(360 + Trunc(AngDisRad1[x] * 350 / pi), 150);
    Inc(Num1);
    Polilinea[Num1] := Point(360 + Trunc(AngDisRad1[x] * 350 / pi), 350);
    Inc(Num1)
  end
  else
  begin
    Polilinea[Num1] := Point(360 + Trunc(AngDisRad1[x] * 350 / pi), 350);
    Inc(Num1);
    Polilinea[Num1] := Point(360 + Trunc(AngDisRad1[x] * 350 / pi), 150);
    Inc(Num1)
  end;
end;
for x := RenCol + 1 downto 1 do
begin
  if x mod 2 = 1 then
  begin
    Polilinea[Num1] := Point(710 - Trunc(AngDisRad1[x] * 350 / pi), 350);
    Inc(Num1);
    Polilinea[Num1] := Point(710 - Trunc(AngDisRad1[x] * 350 / pi), 150);
    Inc(Num1)
  end
  else
  begin
    Polilinea[Num1] := Point(710 - Trunc(AngDisRad1[x] * 350 / pi), 150);
```



```

    Inc(Num1);
    Polilinea[Num1] := Point(710 - Trunc(AngDisRad1[x] * 350 / pi), 350);
    Inc(Num1)
end;
end;
Grafico.Canvas.PolyLine(Polilinea);
AmpMaxSal := 0;    { Preiniciación de variables AmpMaxSal, }
SumaAlCuadrado := 0; { SumaAlCuadrado y THDSalida. Son variables }
THDSalida := 0;    { utilizadas como salidas en la gráfica. }
For x := 0 to 700 do
begin {For x := 0 to 700}
    Grafico.Canvas.Pen.Color := clLime;
    w := 2 * pi * Frec;
    tiempo := x * 2 * pi / (w * 700);
    y := 0;
    y2real := 0;
    y2 := 0;
    For i := 1 to NumArm do
    begin
        if i mod 2 = 1 then
        begin
            if Form4.RadioGroup1.ItemIndex = 0 then
            begin
                H := R2 / Sqrt(Sqr(R1 * R2 * C1 * i * w) + Sqr(R1 + R2));
                phi := -ArcTan((R1 * R2 * C1 * i * w) / (R1 + R2))
            end;
            if Form4.RadioGroup1.ItemIndex = 1 then
            begin
                H := R3 / Sqrt(Sqr(L1 * i * w) + Sqr(R3));
                phi := -ArcTan((L1 * i * w) / R3)
            end;
            if Form4.RadioGroup1.ItemIndex = 2 then
            begin
                H := (L2 * i * w) / Sqrt(Sqr(L2 * i * w) + Sqr(R4));
                phi := (pi / 2) - ArcTan((L2 * i * w) / R4)
            end;
            if Form4.RadioGroup1.ItemIndex = 3 then
            begin
                H := R5 / Sqrt(Sqr(L3 * i * w - (1 / (C2 * i * w))) + Sqr(R5));
                phi := -ArcTan((L3 * i * w - (1 / (C2 * i * w))) / R5)
            end;
            if Form4.RadioGroup1.ItemIndex = 4 then
            begin
                H := R6 / Sqrt(Sqr(R6 - L4 * R6 * C3 * Sqr(i * w)) + Sqr(L4 * i * w));
                phi := -ArcTan((L4 * i * w) / (R6 - L4 * R6 * C3 * Sqr(i * w)))
            end;
            if Form4.RadioGroup1.ItemIndex = 5 then

```

```

begin
  H := R7 / Sqrt(Sqr(L5 * i * w - (1 / (C4 * i * w))) + Sqr((R7 * C5 / C4) +
R7 - L5 * R7 * C5 * Sqr(i * w)));
  phi := ArcTan((L5 * i * w - (1 / (C4 * i * w))) / ((R7 * C5 / C4) + R7 - L5
* R7 * C5 * Sqr(i * w)))
  end;
  y := y + Trunc(100 * SigmaBn[i] * sin(i * w * tiempo));
  y2real := y2real + (H * 100 * SigmaBn[i] * sin(i * w * tiempo + phi));
  If (x = 0) and (i > 1) then SumaAlCuadrado := SumaAlCuadrado + Sqr(H *
SigmaBn[i]); { Determinación }
  If (x = 1) and (i = 1) then THDSalida := 100 * Sqrt(SumaAlCuadrado) / (H *
SigmaBn[1]) {de THDSalida. }
  end;
end;
end;
y2 := Round(y2real);
If (AmpMaxSal - y2) < 0 then AmpMaxSal := y2; {Se determina la amplitud
máxima de la señal filtrada para mostrarla en la gráfica}
if x = 0 then Grafico.Canvas.MoveTo(x + 10, 250 - y2);
Grafico.Canvas.LineTo(x + 10, 250 - y2)
end; {For x := 0 to 700}
Grafico.Canvas.Font.Color := clMaroon;
Grafico.Canvas.Textout(10, 10, 'Frecuencia --' + Form4.Edit18.Text + ' Hz');
Grafico.Canvas.Textout(160, 10, 'THD entrada --' + FormatFloat('000.00', THD *
100) + '%'); {THD de entrada, THD de salida y }
Grafico.Canvas.Textout(310, 10, 'THD salida = ' + FormatFloat('000.00',
Abs(THDSalida)) + '%'); {Amplitud máxima de señal filtrada }
Grafico.Canvas.Textout(460, 10, 'Amplitud máxima de salida = ' +
IntToStr(AmpMaxSal) + '%'); {en la gráfica. }
Grafico.Canvas.Textout(410, 470, 'Amplitud de la señal de entrada = 100 % ');
Grafico.Canvas.Textout(210, 470, 'Señal de salida: ');
Grafico.Canvas.MoveTo(292, 477);
Grafico.Canvas.LineTo(342, 477);
Grafico.Canvas.Textout(10, 470, 'Señal de entrada: ');
Grafico.Canvas.Pen.Color := clBlue;
Grafico.Canvas.MoveTo(100, 477);
Grafico.Canvas.LineTo(150, 477);
Form1.Image1.Picture.Graphic := Grafico;
Grafico.Free
end; {if ((error1 / RenCol) < 0.00001)}
end; {End If RadioGroup1.ItemIndex = 1...}
end;

procedure TForm3.Button1Click(Sender: TObject);
begin
  Form1.Show
end;

```

---

```
procedure TForm3.StringGrid1Click(Sender: TObject);
begin
  StringGrid1.ColCount := StrToInt(Edit1.Text)
end;

procedure TForm3.StringGrid2Click(Sender: TObject);
begin
  StringGrid2.ColCount := StrToInt(Edit1.Text)
end;

procedure TForm3.Filtro1Click(Sender: TObject);
begin
  Form4.Show
end;

end.

unit Unit4;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  StdCtrls, ExtCtrls;

type
  TForm4 = class(TForm)
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Image1: TImage;
    Edit4: TEdit;
    Edit5: TEdit;
    Edit6: TEdit;
    Label4: TLabel;
    Label5: TLabel;
    Label6: TLabel;
    Image2: TImage;
    RadioGroup1: TRadioGroup;
    Label7: TLabel;
    Label8: TLabel;
    Label9: TLabel;
    Image3: TImage;
    Edit7: TEdit;
```

---

```

Edit8: TEdit;
Edit9: TEdit;
Label10: TLabel;
Label11: TLabel;
Label12: TLabel;
Image4: TImage;
Edit10: TEdit;
Edit11: TEdit;
Edit12: TEdit;
Image5: TImage;
Label13: TLabel;
Edit13: TEdit;
Label14: TLabel;
Label15: TLabel;
Edit14: TEdit;
Edit15: TEdit;
Image6: TImage;
Label16: TLabel;
Edit16: TEdit;
Label17: TLabel;
Edit17: TEdit;
Button1: TButton;
Label18: TLabel;
Edit18: TEdit;
procedure Button1Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form4: TForm4;

implementation

{$R *.DFM}

procedure TForm4.Button1Click(Sender: TObject);
begin
  Form4.Hide;
end;

end.

unit Unit5;

```

---

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
ExtCtrls, StdCtrls;

type

TForm5 = class(TForm)

  Image1: TImage;

  Label1: TLabel;

  Label2: TLabel;

  Label3: TLabel;

  Panel1: TPanel;

private

  { Private declarations }

public

  { Public declarations }

end;

var

  Form5: TForm5;

implementation

  {\$R \*.DFM}

end.

### A.3. Código fuente del programa Memoria

```
unit Unit1;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
StdCtrls, Grids;
```

```
type
```

```
TForm1 = class(TForm)  
  GroupBox1: TGroupBox;  
  Label1: TLabel;  
  Label2: TLabel;  
  Edit1: TEdit;  
  StringGrid1: TStringGrid;  
  GroupBox2: TGroupBox;  
  Label3: TLabel;  
  Label4: TLabel;  
  Edit2: TEdit;  
  StringGrid2: TStringGrid;  
  GroupBox3: TGroupBox;  
  Label5: TLabel;  
  Label6: TLabel;  
  Edit3: TEdit;  
  StringGrid3: TStringGrid;  
  GroupBox4: TGroupBox;  
  Label7: TLabel;  
  Label8: TLabel;  
  Edit4: TEdit;  
  StringGrid4: TStringGrid;  
  GroupBox5: TGroupBox;  
  Label9: TLabel;  
  Label10: TLabel;  
  Edit5: TEdit;  
  StringGrid5: TStringGrid;  
  GroupBox6: TGroupBox;  
  Label11: TLabel;  
  Label12: TLabel;  
  Edit6: TEdit;  
  StringGrid6: TStringGrid;  
  GroupBox7: TGroupBox;  
  Label13: TLabel;  
  Label14: TLabel;  
  Edit7: TEdit;  
  StringGrid7: TStringGrid;
```

---

```
GroupBox8: TGroupBox;
Label15: TLabel;
Label16: TLabel;
Edit8: TEdit;
StringGrid8: TStringGrid;
Label17: TLabel;
Edit9: TEdit;
Button1: TButton;
Button2: TButton;
procedure StringGrid1Click(Sender: TObject);
procedure StringGrid2Click(Sender: TObject);
procedure StringGrid3Click(Sender: TObject);
procedure StringGrid4Click(Sender: TObject);
procedure StringGrid5Click(Sender: TObject);
procedure StringGrid6Click(Sender: TObject);
procedure StringGrid7Click(Sender: TObject);
procedure StringGrid8Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form1: TForm1;

implementation

{$R *.DFM}

procedure TForm1.StringGrid1Click(Sender: TObject);
begin
  StringGrid1.ColCount := StrToInt(Edit1.Text)
end;

procedure TForm1.StringGrid2Click(Sender: TObject);
begin
  StringGrid2.ColCount := StrToInt(Edit2.Text)
end;

procedure TForm1.StringGrid3Click(Sender: TObject);
begin
  StringGrid3.ColCount := StrToInt(Edit3.Text)
end;
```

---

```

procedure TForm1.StringGrid4Click(Sender: TObject);
begin
  StringGrid4.ColCount := StrToInt(Edit4.Text)
end;

```

```

procedure TForm1.StringGrid5Click(Sender: TObject);
begin
  StringGrid5.ColCount := StrToInt(Edit5.Text)
end;

```

```

procedure TForm1.StringGrid6Click(Sender: TObject);
begin
  StringGrid6.ColCount := StrToInt(Edit6.Text)
end;

```

```

procedure TForm1.StringGrid7Click(Sender: TObject);
begin
  StringGrid7.ColCount := StrToInt(Edit7.Text)
end;

```

```

procedure TForm1.StringGrid8Click(Sender: TObject);
begin
  StringGrid8.ColCount := StrToInt(Edit8.Text)
end;

```

```

procedure TForm1.Button2Click(Sender: TObject);
{ Este procedimiento genera el contenido de las localidades de una memoria }
{ UVEPROM a partir de los ángulos de disparo proporcionados. La memoria }
{ es grabada con un patrón de conmutación para activar los transistores }
{ de un inversor de puente completo. Memoria: NMC27C16B: 8 x 2048      }
type
  arreglo1 = array [1..44] of integer,
  arreglo2 = array [0..2047] of string [4];
  arreglo3 = array [1..40] of real;
var
  k : integer;
  Archivo : TextFile;
  ConLocMem0, ConLocMem1 : arreglo2;
  ConLocMem2, ConLocMem3 : arreglo2;
  CuatroBits1, CuatroBits2 : arreglo2;
  angulos : arreglo3;

```

```

procedure GenConLocMem (NumPulSem : integer, AngDis : arreglo3; var ConLocMem :
arreglo2);
{ LocMemDis es un arreglo que se utiliza para almacenar las localidades }
{ de memoria que contienen las marcas de encendido y apagado. ConLocMem es }
{ un arreglo donde se almacena el contenido de cada localidad de memoria. }

```



```

var
  contador, i, j : integer;
  LocMemDis : arreglo1;
begin {procedure GenConLocMem}
  { Los siguientes 11 renglones determinan las localidades de memoria que }
  { contienen marcas de encendido y apagado y las almacenan en el arreglo }
  { LocMemDis. }
  LocMemDis[1] := 0;
  LocMemDis[NumPulSem * 2 + 2] := 2047;
  for i := 1 to NumPulSem do
    begin
      LocMemDis[i + 1] := round(AngDis[i] * 11.3778);
      LocMemDis[NumPulSem * 2 + 2 - i] := round((180 - AngDis[i]) * 11.3778);
    end;
  { Los siguientes 13 renglones determinan el contenido binario de las 2048 }
  { direcciones de memoria de la NMC27C17B, asignando 1 ó 0 a cada localidad. }
  contador := -1;
  for i := 1 to (NumPulSem * 2 + 1) do
    begin
      for j := LocMemDis[i] to LocMemDis[i + 1] - 1 do
        begin
          contador := contador + 1;
          if i mod 2 = 1 then
            ConLocMem[contador] := '0'
          else
            ConLocMem[contador] := '1';
          end;
        end;
      end;
    ConLocMem[2047] := ConLocMem[0];
  end; {procedure GenConLocMem}

procedure ConBinHex (CB1 : arreglo2; var CB2: arreglo2);
{ Este procedimiento convierte binario de a hexadecimal. El arreglo }
{ de entrada contiene 2048 palabras de cuatro bits que se convierten a }
{ hexadecimal por comparación. }
var
  c : integer;
begin {procedure ConBinHex}
  for c := 0 to 2047 do
    begin
      if CB1[c] = '0000' then CB2[c] := '0';
      if CB1[c] = '0001' then CB2[c] := '1';
      if CB1[c] = '0010' then CB2[c] := '2';
      if CB1[c] = '0011' then CB2[c] := '3';
      if CB1[c] = '0100' then CB2[c] := '4';
      if CB1[c] = '0101' then CB2[c] := '5';
      if CB1[c] = '0110' then CB2[c] := '6';
    end;
  end;

```

```

    if CB1[c] = '0111' then CB2[c] := '7';
    if CB1[c] = '1000' then CB2[c] := '8';
    if CB1[c] = '1001' then CB2[c] := '9';
    if CB1[c] = '1010' then CB2[c] := 'A';
    if CB1[c] = '1011' then CB2[c] := 'B';
    if CB1[c] = '1100' then CB2[c] := 'C';
    if CB1[c] = '1101' then CB2[c] := 'D';
    if CB1[c] = '1110' then CB2[c] := 'E';
    if CB1[c] = '1111' then CB2[c] := 'F';
  end;
end; {procedure ConBinHex}

begin {Programa principal: procedimiento TForm1.Button2Click}
  { Los siguientes 34 renglones almacenan las marcas de encendido y
  { apagado en cada uno de los ocho bits de las 2048 palabras que tiene
  { la memoria. Los bits son divididos en dos palabras de cuatro bits.
  { CuatroBits1 y CuatroBits2, para su conversión a hexadecimal. }
  With StringGrid8 do
    begin
      For k := 0 to ColCount - 1 do
        angulos[k + 1] := StrToFloat(Cells[k, 0])
      end;
    GenConLocMem (StringGrid8.ColCount, angulos, ConLocMem0);
  With StringGrid7 do
    begin
      For k := 0 to ColCount - 1 do
        angulos[k + 1] := StrToFloat(Cells[k, 0])
      end;
    GenConLocMem (StringGrid7.ColCount, angulos, ConLocMem1);
  With StringGrid6 do
    begin
      For k := 0 to ColCount - 1 do
        angulos[k + 1] := StrToFloat(Cells[k, 0])
      end;
    GenConLocMem (StringGrid6.ColCount, angulos, ConLocMem2);
  With StringGrid5 do
    begin
      For k := 0 to ColCount - 1 do
        angulos[k + 1] := StrToFloat(Cells[k, 0])
      end;
    GenConLocMem (StringGrid5.ColCount, angulos, ConLocMem3);
  for k := 0 to 2047 do
    begin
      CuatroBits1[k] := ConLocMem0[k] + ConLocMem1[k] +
        ConLocMem2[k] + ConLocMem3[k]
    end;
  With StringGrid4 do

```

```

begin
  For k := 0 to ColCount - 1 do
    angulos[k + 1] := StrToFloat(Cells[k, 0])
  end;
GenConLocMem (StringGrid4.ColCount, angulos, ConLocMem0);
With StringGrid3 do
  begin
    For k := 0 to ColCount - 1 do
      angulos[k + 1] := StrToFloat(Cells[k, 0])
    end;
GenConLocMem (StringGrid3.ColCount, angulos, ConLocMem1);
With StringGrid2 do
  begin
    For k := 0 to ColCount - 1 do
      angulos[k + 1] := StrToFloat(Cells[k, 0])
    end;
GenConLocMem (StringGrid2.ColCount, angulos, ConLocMem2);
With StringGrid1 do
  begin
    For k := 0 to ColCount - 1 do
      angulos[k + 1] := StrToFloat(Cells[k, 0])
    end;
GenConLocMem (StringGrid1.ColCount, angulos, ConLocMem3);
for k := 0 to 2047 do
  begin
    CuatroBits2[k] := ConLocMem0[k] + ConLocMem1[k] +
      ConLocMem2[k] + ConLocMem3[k]
  end;
{ Los siguientes 2 renglones realizan la conversi3n a hexadecimal de }
{ la informaci3n. }
  ConBinHex (CuatroBits1, CuatroBits1);
  ConBinHex (CuatroBits2, CuatroBits2);
{ Los siguientes 8 renglones guardan la informaci3n obtenida en el }
{ archivo de texto Memoria.txt. }
  AssignFile (Archivo, Edit9.Text);
  rewrite (Archivo);
  writeln (Archivo, '      ORG $00');
  For k := 0 to 2047 do
    begin
      writeln (Archivo, '      db $', CuatroBits1[k], CuatroBits2[k],
        '      ; Localidad de memoria: ', k: 4);
    end;
  CloseFile (archivo)
end; { Programa principal: procedimiento TForm1.Button2Click }

procedure TForm1.Button1Click(Sender: TObject);
var

```

```
contador1 : integer;
begin
  Edit1.Clear;
  Edit2.Clear;
  Edit3.Clear;
  Edit4.Clear;
  Edit5.Clear;
  Edit6.Clear;
  Edit7.Clear;
  Edit8.Clear;
  Edit9.Clear;
  With StringGrid1 do
    begin
      for contador1 := 0 to ColCount do
        Cells[Contador1, 0] := "
      end;
    With StringGrid2 do
      begin
        for contador1 := 0 to ColCount do
          Cells[Contador1, 0] := "
        end;
      With StringGrid3 do
        begin
          for contador1 := 0 to ColCount do
            Cells[Contador1, 0] := "
          end;
        With StringGrid4 do
          begin
            for contador1 := 0 to ColCount do
              Cells[Contador1, 0] := "
            end;
          With StringGrid5 do
            begin
              for contador1 := 0 to ColCount do
                Cells[Contador1, 0] := "
              end;
            With StringGrid6 do
              begin
                for contador1 := 0 to ColCount do
                  Cells[Contador1, 0] := "
                end;
              With StringGrid7 do
                begin
                  for contador1 := 0 to ColCount do
                    Cells[Contador1, 0] := "
                  end;
                With StringGrid8 do
```

```
begin
  for contador1 := 0 to ColCount do
    Cells[Contador1, 0] := "
  end;
end;
end.
```

---

## **REFERENCIAS**

---

- 
- [1] M. A. Boost and P. D. Ziogas, "State-of-the-Art Carrier PWM Techniques: A Critical Evaluation", *IEEE, Transactions on Industry Applications*, vol. 24, No. 2, pp. 271-280, March / April 1988.
- [2] M. J. Fisher, *Power Electronics*. Boston: PWS-KENT, 1991.
- [3] W. A. Granville, P. F. Smith y W. R. Longley. *Cálculo Diferencial e Integral*. México: UTHEA, 1972.
- [4] D. W. Hart, *Introduction to Power Electronics*. New Jersey: Prentice Hall, 1997.
- [5] J. G. Kassakian, M. F. Schlecht and G. C. Verghese. *Principles of Power Electronics*. New York: Addison-Wesley, 1991.
- [6] N. Mohan, T. M. Undeland and W. P. Robbins, *Power Electronics. Principles and Applications*. New York: McGraw-Hill, 1995.
- [7] P. H. Nayak and R. G. Hoft, "Optimizing the PWM Waveform of a Thyristor Inverter", *IEEE, Transactions on Industry Applications*, vol. IA-11, No. 5, pp. 526-530, September / October 1975.
- [8] H. S. Patel and R. G. Hoft, "Generalized Techniques of Harmonic Elimination and Voltage Control in Thyristor Inverters: Part I-Harmonic Elimination", *IEEE, Transactions on Industry Applications*, vol. IA-9, No. 3, pp. 310-317. May / June 1973.
- [9] M. H. Rashid, *Power Electronics. Devices. Drivers and Applications*. New Jersey: Prentice Hall, 1993.
- [10] K. Taniguchi and H. Irie, "Trapezoidal Modulating Signal for Three-Phase PWM inverter", *IEEE, Transactions on Industrial Electronics*, vol. IE-33, No. 2, pp. 193-200, May 1986.
- [11] J. Vithayathil. *Power Electronics. Principles and Applications*. New York: McGraw-Hill, 1995.
-

- [12] B. W. Williams. *Power Electronics. Devices, Drivers, Applications and Passive Components*. New York: MacGraw-Hill, 1992.
- [13] D. G. Zill, *Ecuaciones Diferenciales con Aplicaciones*. México: Grupo Editorial Iberoamericano, 1982.