

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA GENERADOR DE FORMAS
DE ONDAS DE TENSIÓN Y CORRIENTE PARA SISTEMAS TRIFÁSICOS**

**FREDY FERNEY ESTÉVEZ RODRÍGUEZ
ELKIN OMAR PEÑA RODRÍGUEZ**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y
TELECOMUNICACIONES
BUCARAMANGA**

2009

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA GENERADOR DE FORMAS
DE ONDAS DE TENSIÓN Y CORRIENTE PARA SISTEMAS TRIFÁSICOS**

Autores

**FREDY FERNEY ESTÉVEZ RODRÍGUEZ
ELKIN OMAR PEÑA RODRIGUEZ**

Tesis de grado para optar el título de
Ingeniero Electrónico

Director

MSc JAIME GUILLERMO BARRERO PÉREZ

Codirector

Ing. GABRIEL GONZÁLEZ SUA

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y
TELECOMUNICACIONES
BUCARAMANGA**

2009

DEDICATORIA

*A DIOS TODO PODEROSO POR DARME LA VIDA Y
PORTEGERME.*

*A MIS PADRES ANGEL MIGUEL Y MYRIAM POR SU
ESFUERZO, AMOR Y DEDICACION.*

A MI HERMANA CLAUDIA POR SU APOYO Y CARIÑO.

A MI FAMILIA POR SU COMPRESION COLABORACION.

*A MIS AMIGOS POR COMPARTIR MOMENTOS A LO
LARGO DE LA VIDA.*

FREDY

DEDICATORIA

A DIOS TODOPODEROSO POR DARME LA VIDA, LA FE, LA FORTALEZA Y LA ESPERANZA PARA CULMINAR ESTE PROYECTO, Y DARME FUERZAS EN LOS MOMENTOS DIFÍCILES DE CADA ETAPA DE MI VIDA.

A MIS PADRES RAMÓN PEÑA Y BETTY RODRÍGUEZ POR CREER EN MI, POR SU AMOR, PACIENCIA, APOYO INCONDICIONAL, POR SU ESFUERZO PARA HACER QUE ALCANCE TODAS MIS METAS, POR SER EJEMPLO DE SUPERACIÓN Y MI FUENTE DE INSPIRACIÓN.

A MIS HERMANOS MARTHA, SULAY, ADID E IBETH POR SU AMOR, APOYO, COMPRENSIÓN Y SER CÓMPLICES DURANTE TODA MI VIDA.

A MI SOBRINA ALEXANDRA POR SER LO MÁS HERMOSO QUE HA LLEGADO A MI VIDA.

A TODA MI FAMILIA POR SU AMOR, COMPRENSIÓN, COLABORACIÓN EN CADA UNA MIS PROYECTOS TRAZADOS.

A MIS AMIGOS POR COMPARTIR MOMENTOS DE ALEGRÍA Y TRISTEZA QUE DEJARON HUELLA EN NUESTRO PASO POR LA UNIVERSIDAD.

ELKIN

TABLA DE CONTENIDO

1.	INTRODUCCION	1
2.	FUNDAMENTOS TEÓRICOS.....	4
2.1.	DEFINICIONES Y TÉRMINOS	4
2.2.	SEÑALES DE TIPO TRIFÁSICO	5
2.2.1	ARMÓNICOS	7
2.2.2	ELEMENTOS GENERADORES DE ARMÓNICOS	9
2.3	PROCESADOR DIGITAL DE SEÑALES	10
2.4	CONVERSION DIGITAL-ANALÓGICO (DAC).....	12
2.4.1	COMPORTAMIENTO ESTÁTICO.....	13
2.4.2	COMPORTAMIENTO DINÁMICO.....	15
2.4.3	ARQUITECTURAS DE DAC	17
2.4.3.1.	ARQUITECTURA <i>RESISTOR LADDER</i>	17
2.4.3.2.	ARQUITECTURA <i>CURRENT STEERING</i>	18
2.4.3.3.	ARQUITECTURA <i>CHARGE REDISTRIBUTION</i>	19
2.4.3.4.	ARQUITECTURA SIGMA-DELTA.....	19
3.	HARDWARE	22
3.1.	TARJETA DE DESARROLLO.....	22
3.1.1.	DSP 56F8345.....	24
3.1.1.1.	ARQUITECTURA DE LA FAMILIA 56800E	24
3.1.1.2.	INFORMACIÓN DE ENERGÍA.....	25

3.1.1.3. ENTRADAS- SALIDAS DE PROPÓSITO GENERAL (GPIO)	26
3.1.1.4. JTAG / OnCE	26
3.1.1.5. CARACTERÍSTICAS DE MEMORIA	27
3.1.1.6. CIRCUITOS PERIFÉRICOS	28
3.1.2. PUERTO JTAG/ONCE.....	29
3.1.3. INTERFACE USB.....	35
3.1.4. PUERTO PARA COMUNICACION CON EL DAC	37
3.1.5. PUERTO PARA TECLADO.....	37
3.1.6. PUERTO PARA PANTALLA LCD	37
3.1.7. SISTEMA DE ALIMENTACIÓN.....	38

3.2.	INTERFAZ CON EL USUARIO	43
3.2.1.	PANTALLA GRAFICA DE CRISTAL LIQUIDO	43
3.2.2.	TECLADO 4X4	46
3.3.	CONVERSION DIGITAL-ANALÓGICA	47
3.3.1.	CONVERSOR DIGITAL-ANALÓGICO.....	48
3.3.1.1.	CONVERSOR DAC1220.....	50
3.3.1.1.1.	TEORÍA DE OPERACIÓN	51
3.3.1.1.2.	PROGRAMACIÓN	53
3.3.1.2.	CONVERSOR MCP4821	56
3.3.1.2.1.	TEORIA DE OPERACIÓN.....	58
3.3.2.3.	DAC 8568.....	60
3.3.1.3.1	TEORIA DE OPERACIÓN:.....	61
3.3.2.	ACONDICIONAMIENTO DE LA SEÑAL DE SALIDA.....	64
3.3.2.1.	FILTRO BUTTERWORTH.....	65
3.3.2.2.	SUMADOR INVERSOR CON GANANCIA	66
3.3.2.3.	AMPLIFICADOR OPERACIONAL LME49740	68
4.	SOFTWARE	70
4.1.	GENERACIÓN DE LA ONDA Y LOS ARMÓNICOS.....	71
4.2.	MENU DE INTERFAZ DE USUARIO	79
4.2.1.	MENÚ DE PARÁMETROS PRINCIPALES.....	81
4.2.2.	MENÚ DE PARÁMETROS DE LOS ARMÓNICOS	83
4.2.3.	MENÚ DE SELECCIÓN PARA LA EDICION DEL TIPO DE SALIDA	84
4.3.	ESTRUCTURA DE FUNCIONAMIENTO INTERNO.....	87
4.4.	APLICACIÓN MATLAB	92
5.	PRUEBAS Y RESULTADOS	95
6.	RECOMENDACIONES	108
7.	CONCLUSIONES	110
8.	BIBLIOGRAFÍA	112

LISTA DE FIGURAS

Figura 1: (a) Fuente trifásica de tensión y (b) representación fasorial.	6
Figura 2: Señales perturbadas por armónicos.....	8
Figura 3: Entrada/Salida de un conversor Digital-Analógico	12
Figura 4: Característica real de entrada/salida de un DAC	13
Figura 6: Error de ganancia de un DAC	15
Figura 7: Monotonicidad de un DAC.....	15
Figura 8: Tiempo de establecimiento.....	16
Figura 9: Salida <i>glitch</i> en un DAC.....	16
Figura 10: Diagrama de bloques de un DAC con arquitectura <i>Resistor Ladder</i>	18
Figura 11: Arquitectura <i>Current Steering</i>	19
Figura 12: Arquitectura <i>Charge Redistribution</i>	19
Figura 13: Conversor Sigma-Delta: (a) diagrama de bloques básico y (b) espectro de la señal en diferentes puntos.....	20
Figura 14: Filtro de interpolación: (a) Diagrama de bloques y (b) diferentes operaciones.....	21
Figura 15: diagrama de bloques del generador.....	22
Figura 16: Tarjeta de desarrollo 56F8345	23
Figura 17: Diagrama de bloques del DSP56F8345.....	25
Figura 18: Señales del DSP56F8345 identificadas por grupo funcional ...	30
Figura 19: Vista superior DSP 56F8345.....	31
Figura 20: Puerto JTAG/OnCE.....	31
Figura 21: Diagrama de bloques del puerto JTAG	32
Figura 22: del con 14 al DB25	34
Figura 23: Esquema de conexión de la interface de programación	34
Figura 24: Diagrama de pines del FT232R.....	35
Figura 25: Esquemático modulo USB.....	36
Figura 26: Puerto DAC	37
Figura 27: Puerto Teclado	37
Figura 28: Puerto Pantalla LCD.....	38
Figura 29: 7812PI regulador fijo 12V.....	39
Figura 30: Regulador LM1086	39
Figura 31: diagrama de pines del regulador TPS62046	40
Figura 32: Regulador Ajustable LM317	40
Figura 33: Regulador LT1026.....	41
Figura 34: Fuente de alimentación del generador	41
Figura 35: UA78L12 regulador fijo +12V	42
Figura 36: MC79L12 regulador fijo -12V	42
Figura 37: fuente alimentación del sistema	42
Figura 38: Pantalla grafica LCD Lumex de referencia LCM-S240128GSF43	
Figura 39: Diagrama de conexión de la pantalla LCD	46

Figura 40: Teclado matricial 4x4.....	47
Figura 41: Conexión del teclado al DSP.....	47
Figura 42: Tarjeta conversión digital analógica	48
Figura 43: Diagrama de bloques DAC1220.....	51
Figura 44: Esquemático DAC1220	53
Figura 45: Tarjeta elaborada para el DAC 1220.....	56
Figura 46: Diagrama de bloques DAC MCP4821.....	57
Figura 47: Esquemático DAC MCP4821	57
Figura 48: Tarjeta de pruebas del DAC MCP4821.....	60
Figura 49: DAC 8568.....	61
Figura 50: arquitectura del DAC	62
Figura 51: Secuencias para desactivar y activar referencia interna en modo estático.	63
Figura 52: Secuencias para desactivar y activar referencia interna en modo flexible	63
Figura 53: 32 bits de entrada de registro al DAC 8568	64
Figura 54: Diagrama de bloques acondicionamiento de la señal de salida.	65
Figura 55: (a) Esquemático filtro. Butterworth de segundo orden y (b) diagrama de Bode (verde) y margen de fase (roja).....	66
Figura 56: Sumador inversor con ganancia.....	67
Figura 57: Onda sinusoidal por cuadrante.....	73
Figura 58: Diezmado de la onda base para generar los armónicos de la señal.....	75
Figura 59: Ondas sinusoidales correspondientes a los armónicos de 60Hz, 1, 3, 5 y 7, con el 100%, 20%, 5% y 1% de magnitud respectivamente	78
Figura 60: Suma de las ondas anteriores.....	78
Figura 61: Espectro de la señal generada.....	79
Figura 62: Diagrama de flujo de la ejecución de un estado.....	82
Figura 63: Pantallazo de configuración de los parámetros principales.	83
Figura 64: Pantallazo de configuración de armónicos.....	84
Figura 65: Pantallazo de selección de parámetros de corriente.....	85
Figura 66: Pantallazo inicial.....	86
Figura 67: Pantallazo de las fases dibujadas una vez se invoca el comando de generar	87
Figura 68: Esquema de funcionamiento del programa.....	91
Figura 69: Aplicación creada en matlab	93
Figura 70: Filtro Butterworth simulado.....	95
Figura 71: Diagrama de bode filtro Butterworth.....	96
Figura 72: Circuito de acondicionamiento de señal.....	96
Figura 73: señales tomadas en diferentes puntos del circuito simulado ..	97
Figura 74: Diagrama de Bode del circuito completo.....	97
Figura 75: Señal de entrada y salida del filtro Butterworth.....	98
Figura 76: Señal de entrada y salida del circuito completo a 60 Hz.....	98

Figura 77: Señal de entrada y salida del circuito completo a 3kHz.	99
Figura 78: Ondas senoidales reconstruidas por el DAC MCP4821, (a) a 60Hz, (b) a 120Hz.	101
Figura 79: Señal contaminada con armónicos	102
Figura 80: Señal con el armónico 2 al 30 % y fase 230°.....	102
Figura 81: (a) señales de 60 HZ, (b) señales de 180 HZ	103
Figura 82: Forma de onda de la corriente absorbida por una lámpara fluorescente	104
Figura 83: (a) Señal cuadrada, (b) Señal diente de sierra.	105
Figura 84: Señales de prueba mostradas en pantalla.....	105
Figura 85: Generador de señales de formas de ondas de tensión y corriente para sistemas trifásicos.....	107

RESUMEN

TITULO

DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA GENERADOR DE FORMAS DE ONDAS DE TENSIÓN Y CORRIENTE PARA SISTEMAS TRIFÁSICOS¹.

AUTORES²: FREDY FERNEY ESTÉVEZ RODRÍGUEZ.
ELKIN OMAR PEÑA RODRÍGUEZ.

PALABRAS CLAVES: DSP, BEENS, DAC, arquitectura Delta-Sigma, SPI, RS-232.

DESCRIPCIÓN:

Este trabajo describe el diseño y la construcción de un dispositivo que genera funciones programables de tipo trifásico, similar a la presentes en una red eléctrica trifásica.

Durante el desarrollo de este proyecto se diseñaron y probaron los diferentes módulos que componen el prototipo final. La generación de las ondas se hizo digitalmente mediante la implementación de un Procesador Digital de Señales (DSP) 56F8345 de *Freescale*, asimismo se utilizó el DAC8568 para la conversión digital-analógica, un teclado y una pantalla LCD para interactuar con el usuario de manera que éste tenga el control en las acciones del equipo.

En el DSP 56F8345 reside un programa que permite generar y visualizar en la pantalla LCD las formas de onda de las tensiones y corrientes trifásicas que corresponden a los datos que el usuario introduce inicialmente por medio del teclado o a través del computador por medio de una interfaz gráfica desarrollada en Matlab³. Los datos que el usuario debe ingresar para realizar la generación de las ondas son: a) números de armónicos (máximo 50 armónicos), b) magnitud y fase de cada armónico, c) desfase entre cada una de las fases del sistema trifásico (a-b, a-c).

El sistema utiliza una tarjeta DAC para generar las salidas analógicas con una resolución teórica de 16 bits, la tarjeta por cada canal de salida tiene un filtro para eliminar las señales fuera de la banda de interés y un amplificador que se encarga de darle el nivel deseado a la señal.

Con éste generador se está ofreciendo una herramienta que complementa el aspecto educativo y de investigación, cualquier carga trifásica o monofásica, lineal o no lineal, se puede simular a partir de estas señales de tensión y corriente.

1 Proyecto de grado.

2 Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones. Director MPE. Jaime Barrero.

3 Licencia de tipo institucional propiedad de la UIS

ABSTRACT

TITLE:

DESIGN AND IMPLEMENTATION OF A GENERATOR WAVEFORMS FOR VOLTAGE AND CURRENT FOR THREE PHASE SYSTEMS*

AUTHORS:** FREDY FERNEY ESTÉVEZ RODRÍGUEZ.
ELKIN OMAR PEÑA RODRÍGUEZ

KEYWORDS: DSP, beens, DAC, Delta-Sigma architecture, SPI, RS-232.

DESCRIPTION:

This paper describes the design and construction of a device that generates three-phase programmable features of type similar to that present in a three-phase mains.

During the development of this project were designed and tested the various components of the final prototype. The generation of waves is made digitally by implementing Digital Signal Processor (DSP) from Freescale 56F8345 also was used for the DAC8568 digital-analog conversion, a keyboard and LCD display for user interaction so that he gets control of the team in action.

56F8345 DSP resides in a program that can generate and display on the LCD the waveforms of three phase voltages and currents that correspond to the data the user originally entered via the keyboard or through the computer via a graphical interface Matlab^{***} developed. The data that the user must enter to perform the generation of the waves are: a) the number of harmonics (maximum 50 harmonics), b) magnitude and phase of each harmonic, c) gap between each of the phases of three-phase system (a-b,a-c).

The DAC system uses a card to generate the analog outputs with a theoretical resolution of 16 bits, the card for each output channel has a filter to remove signals outside the band of interest and an amplifier that is responsible for giving the desired level to the signal. With this generator is offering a tool that complements the educational, research or any phase or single phase loads, linear or nonlinear, can be simulated using these voltage and current signals

* Degree Project.

** Physics Mechanical Engineering Faculty, Electric, Electronic and Telecommunications Engineer School. Director: MPE Jaime Barrero.

*** License institutional type property of UIS

1. INTRODUCCION

En los últimos tiempos los crecientes desarrollos tecnológicos han llevado al aumento en la utilización de cargas basadas en dispositivos electrónicos generadores de una gran cantidad de perturbaciones tales como: distorsión armónica, fluctuaciones rápidas de tensión, transitorios entre otros. Debido a esto, los usuarios en especial los del sector industrial, exigen a las empresas que les suministra la energía eléctrica una Calidad de Potencia Eléctrica que no les afecte sus procesos productivos; por ejemplo existen industrias las cuales cuentan con equipos que operan todo el año y cualquier perturbación que cause una parada no planificada se verá reflejada en la disminución de recursos esperados en un periodo determinado a causa de problemas en la Calidad de Potencia Eléctrica entregada por la empresa de servicio eléctrico.

Para estudiar la calidad de la potencia eléctrica (*Electric Power Quality* (EPQ)) en sistemas eléctricos de potencia se debe analizar una variedad de magnitudes como lo son: Tensión, Corriente, Frecuencia, Desequilibrio entre las N-fases, Armónicos, etc.

Una forma de realizar este estudio es generando señales perturbadas tal y como salen de la red eléctrica, (ondas distorsionadas por armónicos y desequilibrios entre las fases). El poder generar estas ondas ha tomado una gran importancia, porque por medio de ellas se puede estudiar el efecto que la falta de calidad tiene sobre los equipos y sistemas eléctricos y electrónicos, además que se utilizan en pruebas de fallos en sistemas eléctricos, en calibración de analizadores de potencia y equipos electrónicos de medida, tal como el que se está desarrollando en el proyecto de maestría “diseño e implementación de un prototipo de contador electrónico digital trifásico de energía eléctrica para control y monitorización remota” del cual hace parte este proyecto, y en el campo de la enseñanza e investigación simulando cargas eléctricas lineales o no lineales que se representan en una deformación o

comportamiento no sinusoidal de las ondas de tensión y corriente resultado de la superposición a la fundamental de 60 Hz en ondas de frecuencias múltiples.

Este proyecto plantea el diseño e implementación de un generador de señales analógicas programables, de tipo trifásico, que simulan las obtenidas de la red eléctrica a la salida de los transductores de tensión y corriente. El generador es controlado por medio de un DSP y debe generar simultáneamente las señales de tensión de cada fase y corrientes de línea correspondientes a los datos de magnitud y fase de los armónicos introducidos por el usuario. En este documento se encuentran los fundamentos teóricos sobre los cuales se basó y los resultados que se obtuvieron durante el desarrollo de este proyecto, con el fin de dar herramientas a quienes en un futuro pretendan desarrollar proyectos similares. El documento se encuentra dividido por capítulos. El segundo capítulo contiene toda la fundamentación teórica e información adicional que se requirió para el diseño e implementación del generador.

El tercer capítulo describe el hardware seleccionado y utilizado en la construcción del generador, se detallan la tarjeta diseñada, DSP, fuente de alimentación, interface de programación (JTAG), pantalla LCD, teclado, y conversor digital analógico.

En el cuarto capítulo se muestra el software desarrollado para la programación del DSP, los algoritmos utilizados para la generación de las ondas, la visualización en la pantalla, la recepción de datos por medio del teclado y la interfaz grafica desarrollada en MATLAB.

En el quinto capítulo se detallan las diferentes pruebas de laboratorio que se le realizaron al equipo en general, con el fin de comprobar el correcto funcionamiento del prototipo dentro de los márgenes establecidos en el plan de proyecto; aquí también se dan a conocer el diseño del PCB y el equipo final.

Por último, en el capítulo sexto y séptimo se presentan las conclusiones, observaciones y recomendaciones para el mejoramiento del equipo en futuros proyectos.

2. FUNDAMENTOS TEÓRICOS

En este capítulo se presentan al lector los principales términos y las definiciones de mayor interés utilizados en el desarrollo del proyecto con el fin de brindar una fácil comprensión y entendimiento de los procedimientos utilizados en el diseño e implementación del generador de formas de ondas de tensión y corriente para sistemas trifásicos.

2.1. DEFINICIONES Y TÉRMINOS

En esta parte se presentan algunas definiciones que se utilizan en el desarrollo del proyecto.

AMPLITUD: La altura máxima que alcanza una onda con respecto a su posición de equilibrio.

FASE: Es una medida de la diferencia de tiempo entre dos ondas senoidales. Aunque la fase es una diferencia verdadera de tiempo, siempre se mide en términos de ángulo, en grados o radianes. Eso es una normalización del tiempo que requiere un ciclo de la onda sin considerar su verdadero periodo de tiempo.

ARMÓNICOS³: Son tensiones o corrientes senoidales que corresponden a múltiplos enteros de la frecuencia normal de trabajo (en Colombia 60 Hz), conocida como fundamental. Se pueden caracterizar por tres valores: la magnitud, el ángulo de fase y la frecuencia.

3 PETIT J. & USTARIZA A. Análisis General de Armónicos: Revisión de las definiciones de Potencia

Universidad Industrial de Santander. Tesis de Grado. 1997.

DSP: Un procesador digital de señales (DSP, *Digital Signal Processor*) es un sistema basado en un procesador o microprocesador que posee un juego de instrucciones, un hardware y un software optimizados para aplicaciones que requieran operaciones numéricas a muy alta velocidad.

DAC: Es un dispositivo que convierte una señal de entrada digital a una señal analógica.

BEANS: Son componentes configurables (se puede reutilizar y pueden ser manipulados) que contienen subrutinas predefinidas que pueden ser utilizados por el Procesador Experto para la programación del DSP.

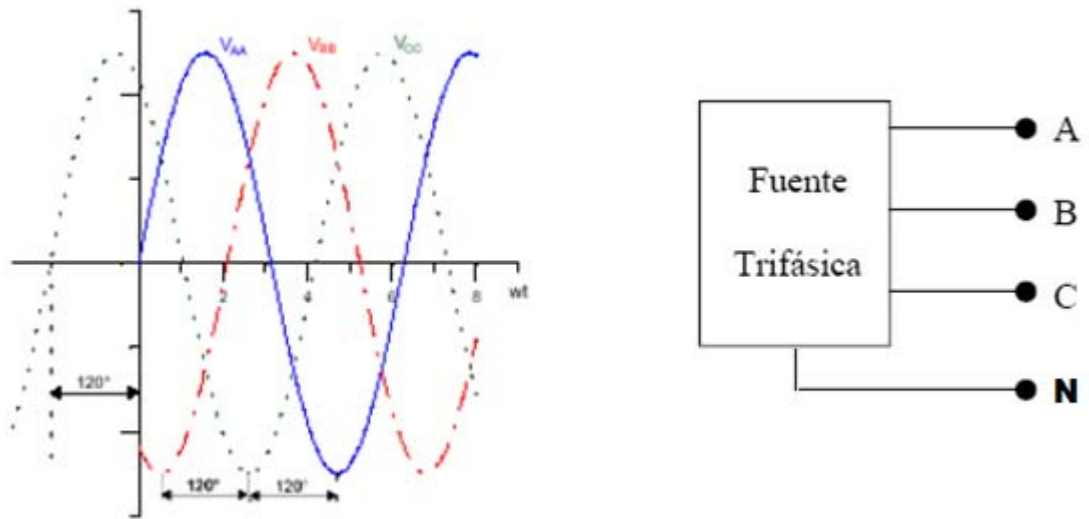
2.2. SEÑALES DE TIPO TRIFÁSICO

Gran parte de la generación, transmisión, distribución y utilización de la energía eléctrica se hace a través de sistemas polifásicos de 60 Hz; los sistemas polifásicos más conocidos y utilizados son los sistemas trifásicos por razones económicas y operativas.

Por ejemplo una fuente trifásica de tensión esta constituida por tres fuentes monofásicas de igual valor eficaz pero desfasadas 120° entre ellas. La figura 1 ilustra lo descrito.

La fuente tiene cuatro terminales, y las medidas demuestran que entre dos de esas tres terminales existen voltajes senoidales de igual magnitud.

Figura 1: (a) Fuente trifásica de tensión y (b) representación fasorial.

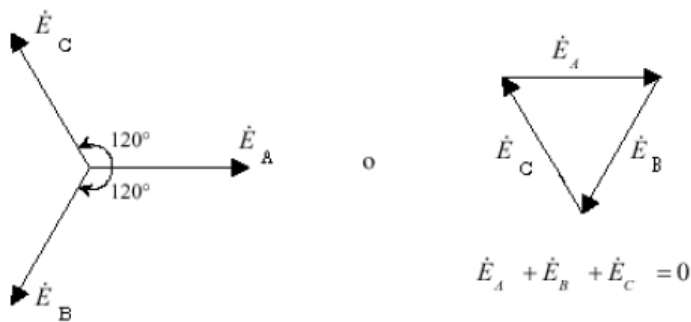


(a)

Analíticamente se puede expresar:

$$\begin{cases} e_A(t) = \sqrt{2}U \cos(\omega t) \\ e_B(t) = \sqrt{2}U \cos(\omega t - 2\pi/3) \\ e_C(t) = \sqrt{2}U \cos(\omega t + 2\pi/3) \end{cases}$$

Fasorialmente:



(b)

Fuente: Circuitos eléctricos trifásicos, Ing. Mauricio Riera – Ing. Luis García.

2.2.1 ARMÓNICOS

Hoy en día observamos que al medir la tensión en la salida de los generadores en la cabecera de los sistemas de potencia son puramente sinusoidales y equilibradas, pero, al momento de ir acercándose a las cargas finales la distorsión de estas señales se hace notable, esto se debe a que la corriente solicitada por las cargas finales no es realmente sinusoidal lo que produce la distorsión de las ondas de tensión.

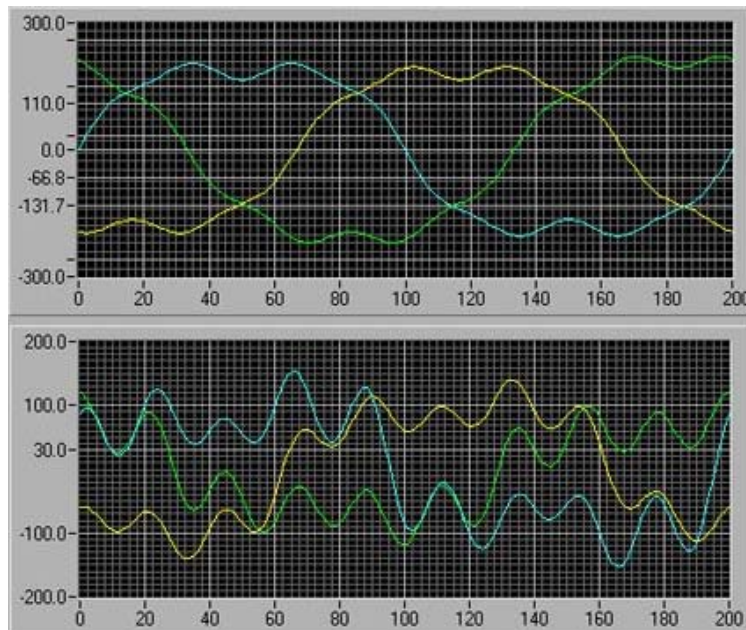
En muchas ocasiones las ondas de corrientes armónicas se consideran como aleatorias, pero en gran parte de los casos tienen un comportamiento eminentemente periódico, esto facilita su estudio al descomponerlas en armónicos mediante series de *Fourier*. Los diseños de los sistemas de potencia se hacen a una frecuencia fundamental y para que operen con formas de onda sinusoidal, por esta razón la circulación de estas corrientes armónicas afecta su funcionamiento.

La contaminación armónica de una red aún cuando no es causal de la interrupción de la energía, hace sentir marcadamente sus efectos en la calidad de la misma. La figura 2 muestra formas de ondas de tensión o corriente perturbadas por armónicos.

A continuación mencionamos algunos de los efectos que los armónicos producen sobre los sistemas de potencia y los equipos conectados a él:

- Al circular armónicos de corriente por las líneas de transporte y distribución produce caídas de tensión en las impedancias de estas.
- Producen un incremento en el efecto *Joule* en los conductores.

Figura 2: Señales perturbadas por armónicos.



Fuente: INVES POT

- En transformadores y reactancias, los armónicos de corriente aumentan las pérdidas en el cobre, mientras que los armónicos de tensión aumentan las pérdidas en el hierro. Ambas pérdidas producen calentamientos, disminuyendo la vida de los equipos y reduciendo la potencia útil del mismo.
- En motores y generadores las corrientes y tensiones armónicas aumentan las pérdidas de la máquina, también origina pares parásitos que se traducen en oscilaciones electromecánicas que reducen el par útil de la máquina.
- En los sistemas de protección causa efectos indeseados que origina disparos inesperados y retarda la actuación de dichas protecciones.
- Los armónicos de tensión y corriente generan errores en los equipos de medida y contadores de energía debido a que gran parte de estos equipos son diseñados para trabajar con ondas de tensión y corriente puramente sinusoidales.

- Los armónicos producen interferencia en los equipos de comunicación.

2.2.2 ELEMENTOS GENERADORES DE ARMÓNICOS

El origen de los armónicos esta dado por la existencia de cargas no lineales conectadas a la red. Es bueno aclarar que la mayoría de las cargas reales son en mayor o menor medida, no lineales, aunque en muchos casos, existe un alto grado de homogeneidad entre la tensión y la corriente asociada, por lo que su característica de no lineal resulta despreciable. Sin embargo, existen otro grupo de cargas en la cuales predomina su característica no lineal, lo que las convierte en generadoras de armónicos. Las cargas no lineales pueden clasificarse en tres categorías básicamente:

- **Basadas en arcos y descargas eléctricas.**
 - ✓ Lámparas de descarga (fluorescentes, ahorradoras de energía, neón, vapores de sodio y mercurio, etc.)
 - ✓ Soldadores de arco
 - ✓ Hornos de arco
- **Basadas en inductancias saturables**
 - ✓ Transformadores
 - ✓ Motores
 - ✓ Reactancias para limitar los arcos de descarga
- **Electrónicas**
 - ✓ Rectificadores para cargas resistivas e inductivas
 - ✓ Fuentes de alimentación (aparatos electrónicos domésticos y de oficina, variadores de frecuencia, sistemas de alimentación ininterrumpida, lámparas electrónicas, etc.)

- ✓ Reguladores y recortadores (variador de velocidad de pequeños motores, ahorradores de energía para motores, reguladores de luz, etc.)
- ✓ Cargadores de baterías
- ✓ Ciclo convertidores,
- ✓ Convertidores continua-alterna sobre la red (energía solar, transmisión de energía en corriente continua).

2.3 PROCESADOR DIGITAL DE SEÑALES

Los procesadores digitales de señales (DSP) ganaron popularidad en la década de los sesentas con la introducción de la tecnología del estado sólido. Los primeros DSPs consistían en elementos de hardware donde cada uno de los elementos ejecutaba una operación específica para el procesamiento de las señales. Luego con los avances en la electrónica, especialmente en la tecnología de fabricación de circuitos integrados aparecen los procesadores de señales micro-programables, lo que permitió a los elementos de hardware estar compartidos en tiempo para implementar múltiples funciones de sistemas o ser reprogramados para una nueva función, lo que ha llevado a que hoy en día tengan un gran impacto en la industria y la sociedad.

Los procesadores digitales de señales (DSPs) se caracterizan por su alto *throughput* (cantidad de datos procesados en un tiempo determinado), bajo consumo de potencia y bajo costo. Estos dispositivos han traído nuevas herramientas de análisis al mundo de la electrónica, han aumentado la velocidad de análisis, y la resolución y precisión han mejorado significativamente, por su popularidad ha hecho que mejore la eficiencia en el manejo de datos y el cálculo de resultados y el desarrollo de rutinas de alta velocidad para que la información este lista en aplicaciones que requieren posterior análisis y visualización.

Hoy en día los equipos digitales han demostrado ser mejores y eficientes que los analógicos por su alto número de aplicaciones y de análisis. Estas ventajas se dan porque estos equipos digitales toman muestras de las señales, las almacenan y luego las procesan obteniendo resultados que un equipo analógico no podría.

La programación de la mayoría de los DSP se puede hacer en lenguaje C o en lenguaje ensamblador (*assembler*). Aunque escribir los programas en lenguaje C requiere menos esfuerzo, la eficiencia alcanzada es normalmente menor a la de los programas escritos en *assembler*, esta eficiencia hace referencia a que tenga pocas instrucciones o tan pocas instrucciones por ciclo como sea posible, haciendo uso máximo de los recursos del chip.

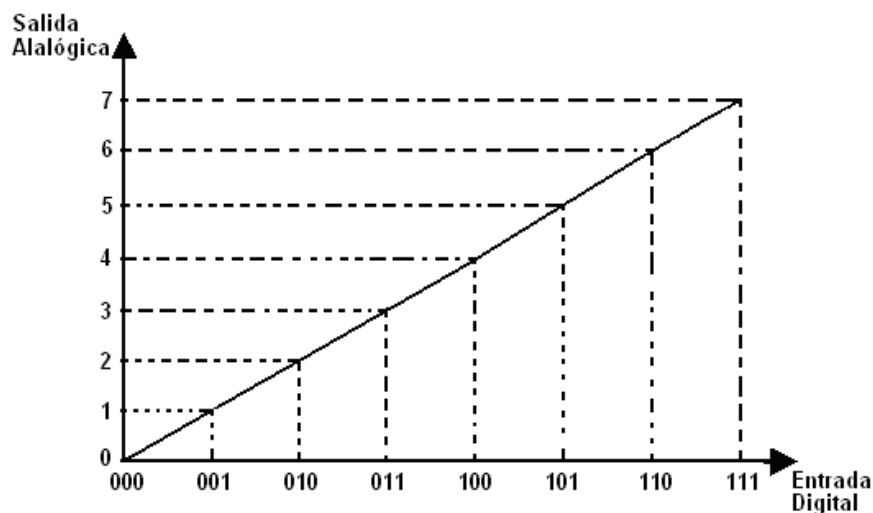
En la práctica, se comienza con la programación en C, analizando el comportamiento y la funcionalidad del algoritmo, luego si el proceso requerido no resulta usando el compilador optimizador de C, se identifican las porciones de tiempo desperdiciadas en el código C y se convierten a *assembler*, o el código entero se reescribe en *assembler*.

El diseño del generador de formas de ondas de tensión y corriente para sistemas trifásicos se realizó basado en un procesador digital de señales (DSP) de referencia 56F8345 de la familia de Motorola, que es el encargado de generar las señales con sus respectivos armónicos y desfases, además de administrar la interfaz con el usuario. Este DSP se escogió teniendo que es el mismo que utiliza el proyecto de maestría de cual hacemos parte pues la idea es que todo se maneja bajo un mismo procesador. El DSP se programó utilizando el lenguaje C sobre el software de programación *CodeWarrior*. Las características del DSP y del software de programación se describen en el capítulo 2 y 3 respectivamente.

2.4 CONVERTOR DIGITAL-ANALÓGICO (DAC)

Un conversor digital-analógico es un dispositivo que abre las puertas para conexión del dominio digital con el mundo analógico, se encarga de representar cierto número de salidas analógicas discretas para un número limitados de códigos digitales de entrada (figura 3), para cada uno de los cuales corresponde un único valor analógico.

Figura 3: Entrada/Salida de un conversor Digital-Analógico



Fuente: Los autores

Un conversor digital-analógico genera una señal continua en el tiempo a partir de muestras tomadas de una señal digital. Cada una de estas muestras se debe entregar al conversor a una misma frecuencia de muestreo que permita cumplir el teorema del muestreo para poder recuperar la señal original. La salida de un DAC puede ser una señal de corriente o tensión, pero en ambos casos se necesita de un *buffer* y un amplificador pues la impedancia de salida del DAC no permite conectar directamente a un transductor o a un dispositivo de instrumentación.

2.4.1 COMPORTAMIENTO ESTÁTICO

Característica De Entrada/Salida: La salida analógica ideal de un DAC se expresa en la siguiente ecuación:

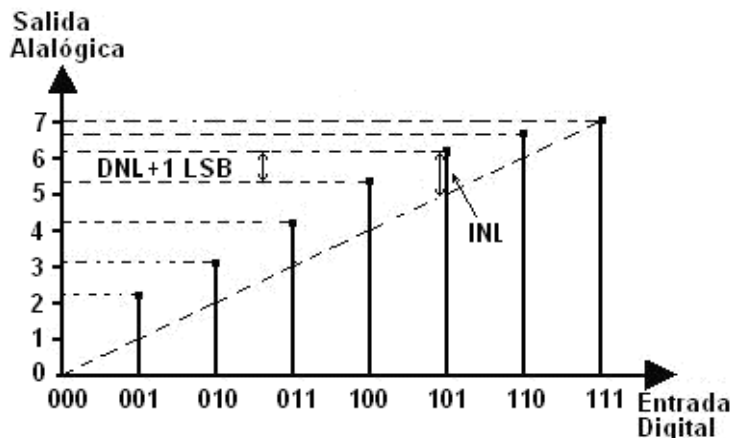
$$V_{out}(D_i) = V_{ref} \left(\frac{b_N}{2} + \frac{b_{N-1}}{2^2} + \dots + \frac{b_1}{2^N} \right) \quad (1.1)$$

Donde $\{D_i, i = 0, 1, \dots, 2^N - 1\}$ denota la palabra digital. El punto de referencia es usualmente considerado como el punto medio de rango de *full-scale* $-V_{ref}/2$ a $V_{ref}/2$.

No Linealidad Diferencial (DNL): El rango de salida de un DAC de N-bit es dividido en 2^N partes, y un cambio en un LSB (Bit menos significativo) en la palabra digital de entrada producirá un cambio de $V_{ref}/2^N$ en la salida analógica. La no linealidad diferencial es una medida de la desviación del cambio real a la salida de un DAC y el cambio ideal de voltaje al cambio de un LSB (figura 4) y está definida por:

$$DNL = \frac{V_{out}(D_{i+1}) - V_{out}(D_i) - V_{ref}/2^N}{V_{ref}/2^N}, \text{ para } i = 0, 1, \dots, 2^N - 2 \quad (1.2)$$

Figura 4: Característica real de entrada/salida de un DAC



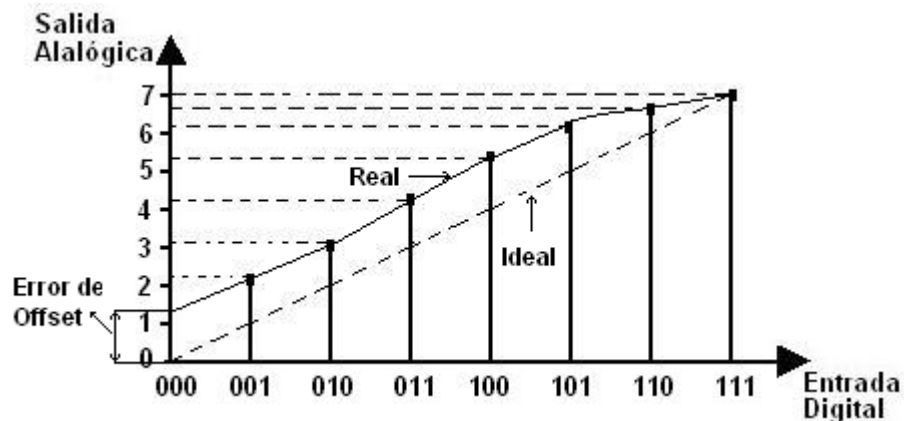
Fuente: NASSER KEHTARNAVAZ AND MANSOUR KERAMAT, DSP System Design Using The TMS320C6000. Prentice Hall 2001.

No Linealidad Integral (INL): En general la linealidad de un DAC puede ser especificada en términos de la no linealidad integral (INL). La INL es una medida de la desviación del voltaje de salida real del DAC de la línea ideal. Esta definida por:

$$INL_i = \frac{V_{out}(D_i) - i * V_{ref} / 2^N}{V_{ref} / 2^N} \quad (1.3)$$

Offset: La salida analógica del DAC debería ser 0 V cuando D=0. Sin embargo, como se muestra en la figura 5 existe un voltaje de *offset* similar al voltaje de *offset* que hay en un amplificador operacional.

Figura 5: Offset de un DAC



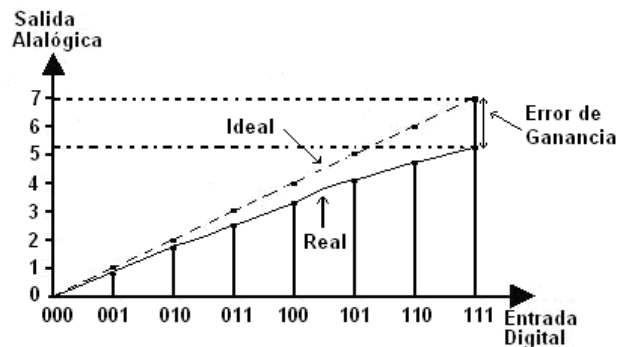
Fuente: NASSER KEHTARNAVAZ AND MANSOUR KERAMAT, DSP System Design Using The TMS320C6000. Prentice Hall 2001.

Error de Ganancia: Un error de ganancia existe cuando la pendiente de la función de transferencia del DAC es diferente a la pendiente de la función de transferencia ideal. Este error se ilustra en la figura 6.

Monotonicidad: La salida del DAC debe incrementarse durante todo su rango, cuando la palabra de entrada digital incremente. DNL debe ser superior

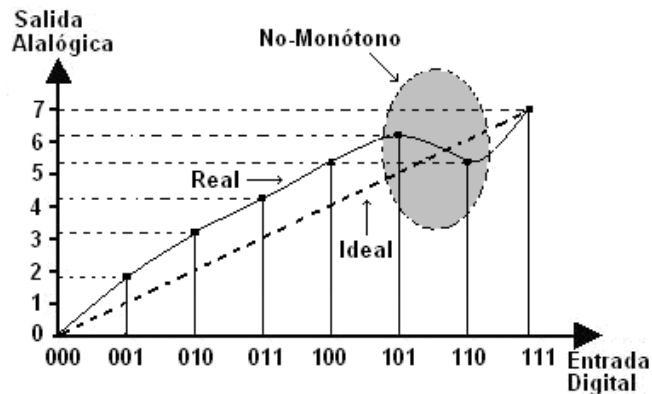
a -1 LSB para que el DAC sea monótono. La monotonicidad es importante en muchas aplicaciones particularmente en control digital. La figura 7 ilustra la característica no monótona de un DAC.

Figura 6: Error de ganancia de un DAC.



Fuente: NASSER KEHTARNAVAZ AND MANSOUR KERAMAT, DSP System Design Using The TMS320C6000. Prentice Hall 2001.

Figura 7: Monotonicidad de un DAC.



Fuente: NASSER KEHTARNAVAZ AND MANSOUR KERAMAT, DSP System Design Using The TMS320C6000. Prentice Hall 2001.

2.4.2 COMPORTAMIENTO DINÁMICO

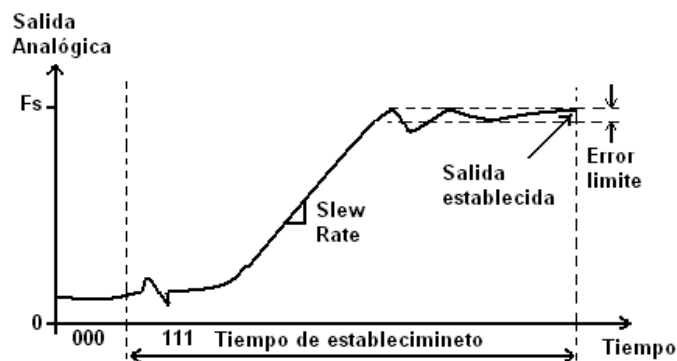
Tiempo De Establecimiento: Los cambios que experimenta la salida del DAC son producto de las variaciones de la palabra digital de entrada, estos cambios no ocurren instantáneamente, se necesita de un tiempo para que la señal se

establezca dentro de un error límite llamado tiempo de establecimiento (ver figura 8).

Latencia: Es el retardo total para obtener una salida valida después de que la palabra de entrada cambie.

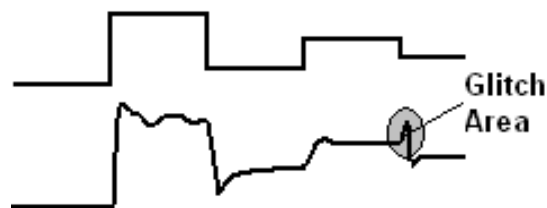
Área *Glitch*: El área máxima bajo el *glitch* de la salida cuando la palabra de entrada cambia es llamada área *glitch* (figura 9).

Figura 8: Tiempo de establecimiento



Fuente: NASSER KEHTARNAVAZ AND MANSOUR KERAMAT, DSP System Design Using The TMS320C6000. Prentice Hall 2001.

Figura 9: Salida *glitch* en un DAC



Fuente: NASSER KEHTARNAVAZ AND MANSOUR KERAMAT, DSP System Design Using The TMS320C6000. Prentice Hall 2001

Relación Señal a Ruido + Distorsión: Resolución es el término utilizado para describir el mínimo nivel de señal que un DAC puede resolver. El límite

fundamental de un DAC es gobernado por el ruido de cuantización. Si la palabra de entrada digital es de N-bits el mínimo paso que el DAC puede dar es $V_{ref}/2^N$. Si los voltajes de salida son reproducidos con este mínimo paso de incertidumbre, un DAC ideal debe tener una mínima relación señal-a-ruido de:

$$SNR = 10 \log \frac{(V_{ref}/2)^2 / 2}{\frac{1}{12} (V_{ref}/2^N)^2} = 10 \log \frac{3}{2} 2^{2N} = 6.02N + 1.76 (dB) \quad (1.4)$$

2.4.3 ARQUITECTURAS DE DAC

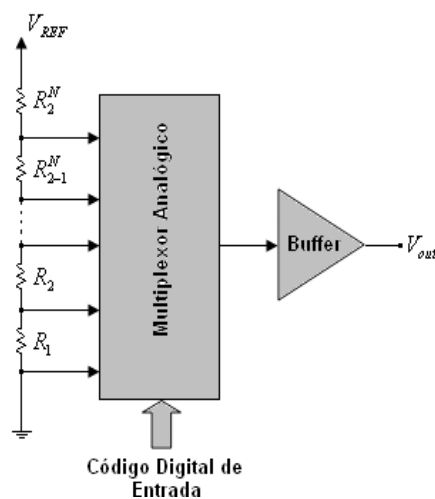
Un DAC convierte una secuencia de bits en niveles analógicos. Diferentes arquitecturas pueden realizar esta conversión con diferentes velocidades de muestreo y resolución. Los sistemas con DSP (procesador digital de señales) tienen diferentes necesidades que van desde altas velocidades (>40 MHz), baja resolución (<10 bits), como los sistemas de video, o una moderada velocidad (1-10 MHz), alta resolución (10-16 bits), como los sistemas DSL, y baja velocidad (< 50 MHz), y una muy alta resolución (>16 bits), tal como los sistemas de audio y video de alta calidad. A continuación se describen brevemente algunas de las arquitecturas encontradas en la literatura, mostrando algunas de sus características más relevantes, que permitieron la selección del DAC utilizado en el proyecto.

2.4.3.1. ARQUITECTURA RESISTOR LADDER

En la arquitectura *resistor ladder* el voltaje de referencia es dividido en $2^N - 1$ subvoltajes (figura 10). La diferencia entre dos voltajes subsecuentes es exactamente 1 LSB. Una red de conmutadores selecciona el voltaje de salida de una cadena de resistencias, basado en la palabra digital de entrada. Con el fin de evitar la carga de la serie de resistencias, se utiliza un *buffer* a la salida

del DAC, Usualmente el *buffer* de salida hace parte del DAC. La mayor ventaja de esta arquitectura es su simplicidad, también posee una característica monótonica. Sus desventajas son: a) necesita una combinación de $2^N - 1$ resistencias y b) requiere de un buffer de salida, que disminuye velocidad. Para un DAC de arquitectura *resistor ladder* de alta resolución se necesita un gran número de resistencias y conmutadores.

Figura 10: Diagrama de bloques de un DAC con arquitectura *Resistor Ladder*.



Fuente: NASSER KEHTARNAVAZ AND MANSOUR KERAMAT, DSP System Design Using The TMS320C6000. Prentice Hall 2001.

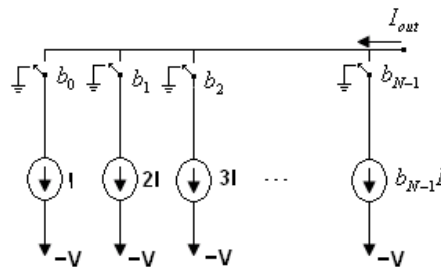
2.4.3.2. ARQUITECTURA *CURRENT STEERING*

En esta arquitectura una matriz de fuentes de corriente idénticas es utilizada para generar una salida de corriente correspondiente a la palabra digital de entrada, como se muestra en la figura 11. La mayor ventaja de ésta arquitectura es que puede manejar una resistencia de carga directamente, y puede convertir la salida de corriente en voltaje sin utilizar ningún *buffer* a la salida. La arquitectura *Current Steering* al igual que la anterior garantiza monotonidad. Además, permite grandes tasas de conversión. Una de las desventajas es la disipación de potencia estática en sus fuentes de corriente.

2.4.3.3. ARQUITECTURA CHARGE REDISTRIBUTION

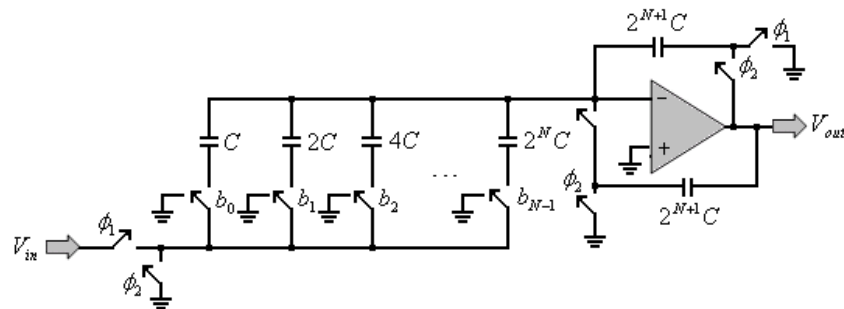
Este tipo de arquitectura utiliza una matriz de capacitores idénticos. Es la figura 12 se ilustra un circuito simple de esta arquitectura. La principal desventaja es que requiere de capacitores lineales lo que demanda una gran área en el chip. El rendimiento de esta arquitectura se ve limitado cuando hay una alta tasa de muestreo porque requiere de tener a la salida un *buffer* altamente lineal.

Figura 11: Arquitectura *Current Steering*.



Fuente: NASSER KEHTARNAVAZ AND MANSOUR KERAMAT, DSP System Design Using The TMS320C6000. Prentice Hall 2001.

Figura 12: Arquitectura *Charge Redistribution*.



Fuente: NASSER KEHTARNAVAZ AND MANSOUR KERAMAT, DSP System Design Using The TMS320C6000. Prentice Hall 2001.

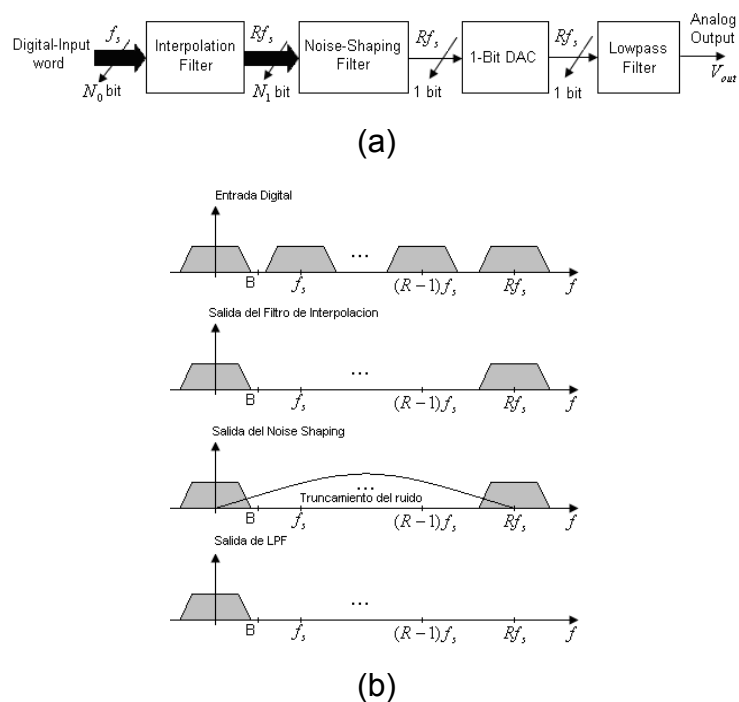
2.4.3.4. ARQUITECTURA SIGMA-DELTA

Las anteriores arquitecturas no están disponibles para una resolución alta (>14 bits). La arquitectura sigma-delta puede lograr alta resolución a costa de una

menor tasa de muestreo- es decir, altas tasas de sobremuestreo. En la figura 13 se muestra el diagrama de bloques básico de ésta arquitectura y el espectro de la señal en algunos puntos. El DAC sigma-delta consiste en un filtro digital de interpolación, un filtro *noise-shaping*, un conversor D/A de 1-bit y un filtro pasabajos a la salida (filtro de reconstrucción).

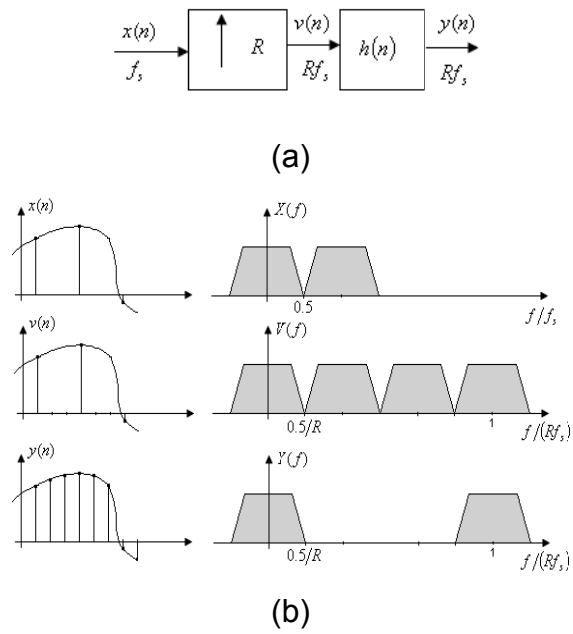
La función del filtro de interpolación es generar mas muestras entre muestras consecutivas (sobremuestreo). Éste aumenta la correlación entre muestras. En la figura 14 se muestra la operación del filtro de interpolación. Después de incrementar la rata de datos usando el filtro de interpolación, un modulador sigma-delta es usado para generar a 1-bit flujo de datos. El modulador empuja el ruido de cuantización a las frecuencias que están fuera de la banda de interés. La salida del modulador se le aplica a un filtro de decimación.

Figura 13: Conversor Sigma-Delta: (a) diagrama de bloques básico y (b) espectro de la señal en diferentes puntos.



Fuente: NASSER KEHTARNAVAZ AND MANSOUR KERAMAT, DSP System Design Using The TMS320C6000. Prentice Hall 2001.

Figura 14: Filtro de interpolación: (a) Diagrama de bloques y (b) diferentes operaciones.

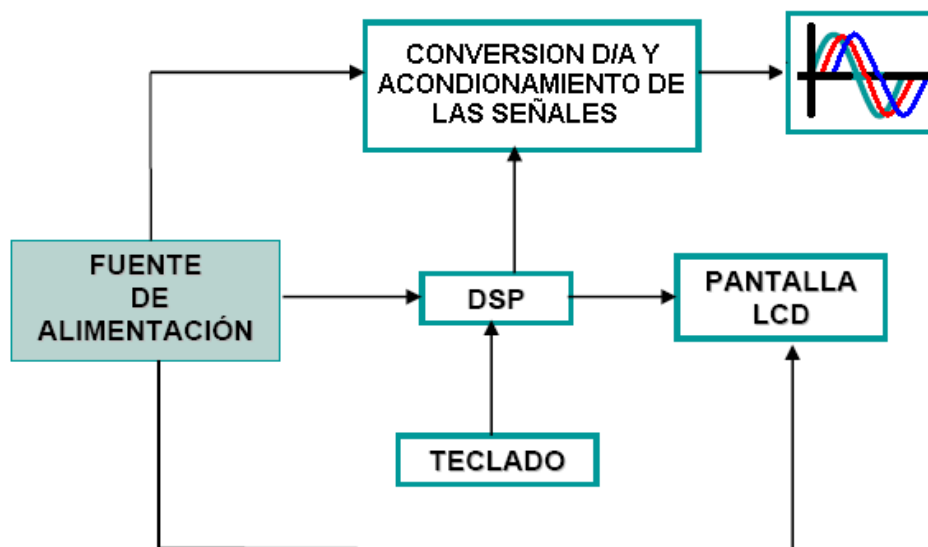


Fuente: NASSER KEHTARNAVAZ AND MANSOUR KERAMAT, DSP System Design Using The TMS320C6000. Prentice Hall 2001.

3. HARDWARE

El hardware implementado en este proyecto consiste en una tarjeta de desarrollo para el DSP 56F8345 de Motorola de propósito general que fue utilizado para la generación de las señales, una pantalla gráfica LCD para visualizar las señales generadas, un teclado que servirá de interfaz con el usuario que permite navegar por las diferentes opciones que presenta el menú visto en la pantalla e introducir datos; además de una etapa conversión digital analógica. A continuación se explicara más en detalle cada una de las partes que conforman el prototipo final. La figura 15 muestra un diagrama de bloques que describe el sistema final.

Figura 15: diagrama de bloques del generador



Fuente: autores del proyecto

3.1. TARJETA DE DESARROLLO

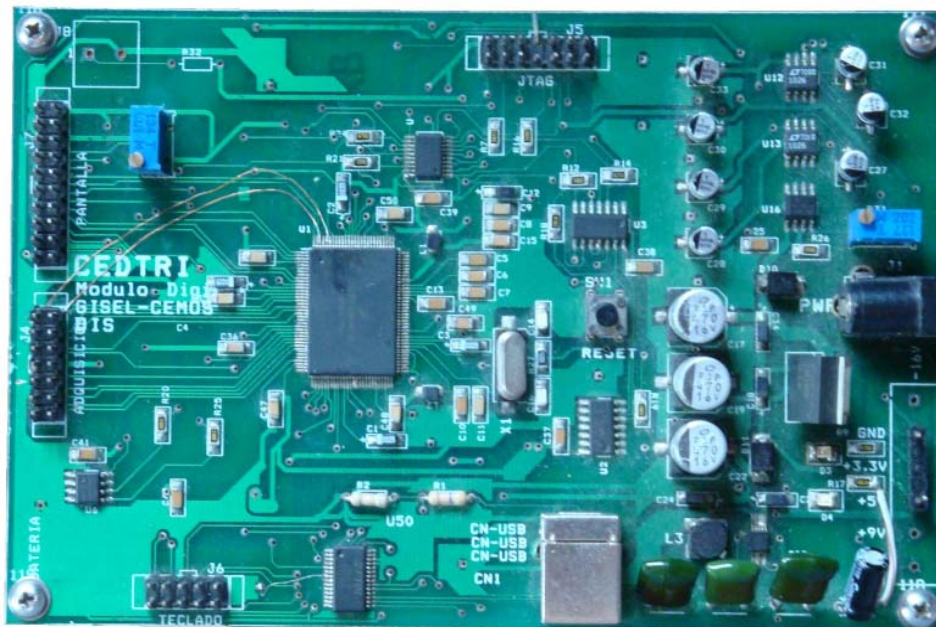
La tarjeta implementada en el prototipo final incluye el DSP56F8345 y está basada en la hoja de datos de la tarjeta de evaluación MC56F8367 que permite realizar aplicaciones propias por medio de la herramienta *codewarrior*. Esta parte del hardware fue implementada teniendo como base lo realizado por el

ingeniero Gabriel González y con la ayuda del manual de usuario de la misma. Se realizaron algunas modificaciones y se obviaron algunos componentes para facilitar el diseño y la implementación de los periféricos necesarios.

Se realizó el montaje mostrado en la figura 16 donde se aprecian todos los componentes que conforman la tarjeta principal como son:

- DSP 56F8345
- Puerto JTAG (14 pines)
- Interface USB
- Puerto transmisión de datos al DAC (16 pines)
- Puerto para teclado (10 pines)
- Puerto para pantalla LCD (20 pines)
- Fuente de alimentación

Figura 16: Tarjeta de desarrollo 56F8345



Fuente: autores del proyecto

3.1.1. DSP 56F8345

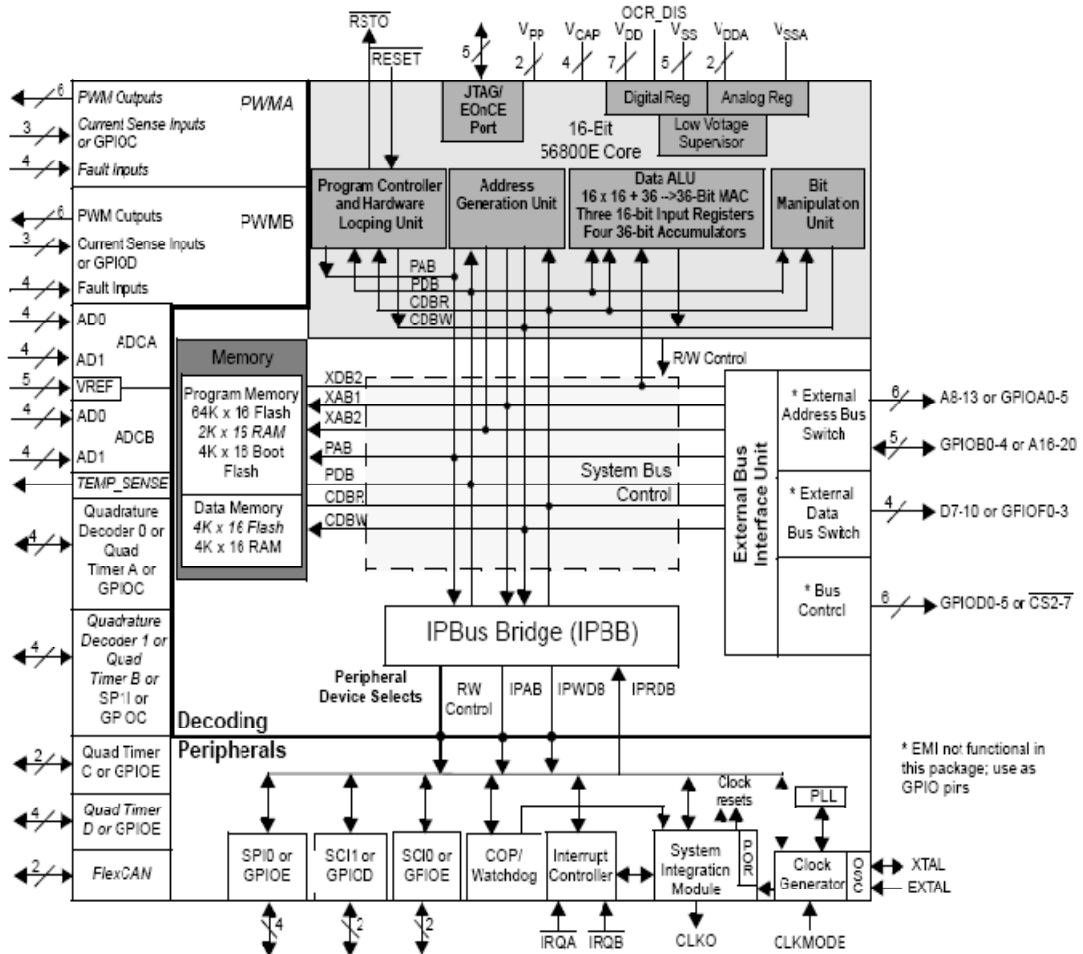
El Procesador de Señales Digitales DSP 56F8345 pertenece a la familia 56800E de *Freescale*, y en un solo chip combina el poder de procesamiento de un Procesador Digital de Señales (DSP) y la funcionalidad de un microcontrolador, con un juego flexible de periféricos para crear soluciones altamente rentables. Por su bajo costo, configuración flexible y código de programación compacto el 56F8345 es utilizado para muchas aplicaciones, sus periféricos son utilizados en control automático, control industrial, control de movimiento, etc. La arquitectura de los 56800E se compone de tres unidades de ejecución que operan en paralelo, permitiendo no más de seis operaciones por ciclo de instrucción. La ejecución del programa de soporte se da mediante memorias internas. Este DSP proporciona dos pines de interrupción externa y ofrece cuarenta y nueve pines de propósito general Input/Output (GPIO), dependiendo de la configuración periférica.

3.1.1.1. ARQUITECTURA DE LA FAMILIA 56800E

El DSP 56F8345 tiene una arquitectura ordenada en grupos funcionales como muestra la figura 17. A continuación se nombran algunas de las principales características de la familia 56800E:

- Procesa hasta 60 millones de instrucciones por segundo (MIPS), a una frecuencia de ejecución de 60 MHz.
- Procesador de señales digitales y microcontrolador funcionalmente unificados en un solo chip.
- Arquitectura en C eficiente.
- Cuatro acumuladores de 36 bits.
- Tres buses de dirección de 16 bits.
- Cuatro buses de datos de 16 bits.
- JTAG/OnCE interfaz de depuración y programación.

Figura 17: Diagrama de bloques del DSP56F8345



Fuente: Hoja de datos del fabricante.

3.1.1.2. INFORMACIÓN DE ENERGÍA

- Fabricado en tecnología CMOS de alta densidad, hasta 5V de tolerancia, entradas digitales compatibles con TTL.
- Regulador de voltaje de 3.3V a 2.6V para alimentar la memoria interna, puede ser deshabilitada.
- Reguladores internos para alimentar circuitos analógicos y digitales a bajo costo y bajo ruido.
- Modos de espera y detención disponibles.
- Manejo de potencia inteligente en los ADC's.

- Cada uno de los periféricos puede ser deshabilitado para ahorrar energía.

La tolerancia de 5V hace referencia a la capacidad de que un pin de entrada-salida construido en tecnología compatible de 3.3V pueda soportar una tensión de hasta 5.5V sin que el dispositivo vaya a sufrir algún daño. Un dispositivo de 3.3V puede soportar una tensión de $3.3V \pm 10\%$ sin que ocurra un efecto dañino, la capacidad de tolerancia de 5V brinda la posibilidad de ahorro de energía de 3.3V combinada con la capacidad de soportar niveles de 5V.

3.1.1.3. ENTRADAS- SALIDAS DE PROPÓSITO GENERAL (GPIO)

El DSP 56F8345 dispone de seis grupos de puertos GPIO, el ancho de cada puerto y su función periférica se muestran en la tabla 1. El ancho del puerto GPIO denota cuantos bits son implementados en cada uno de los registros del GPIO.

3.1.1.4. JTAG / OnCE

El DSP 56F8345 tiene la capacidad de prueba de tarjeta y programación del chip a través de dos módulos internos del integrado, el JTAG/OnCE. El JTAG es una recopilación de la norma IEEE 1149.1a.1993. El puerto JTAG/OnCE permite principalmente entrar a modo de depuración, salvar líneas de registro, leer desde un búfer trazado, escribir a memoria y leer de la memoria.

El modulo OnCE (*On chip Emulation*, Emulación sobre el chip) permite que el usuario pueda examinar registros, memoria y periféricos del chip, lo cual facilita el desempeño del “*hardware*” y el “*software*” del DSP. El modulo OnCE interactúa con el núcleo y sus periféricos de forma no intrusiva y se accede a través del puerto JTAG.

Tabla 1: Configuraciones de puertos GPIO del DSP56F8345.

GPIO Port	Port Width	Available Pins in 56F8345	Peripheral Function	Reset Function
A	14	6	6 pins - EMI Address pins - Can only be used as GPIO 8 pins - EMI Address pins - Not available in this package	EMI Address N/A
B	8	5	5 pins - EMI Address pins - Can only be used as GPIO 3 pins - EMI Address pins - Not available in this package	GPIO N/A
C	11	11	4 pins - DEC1 / TMRB / SPI1 4 pins - DEC0 / TMRA 3 pins - PWMA current sense	DEC1 / TMRB DEC0 / TMRA PWMA current sense
D	13	11	2 pins - EMI \overline{CSn} 4 pins - EMI \overline{CSn} - Can only be used as GPIO 2 pins - SCI1 2 pins - EMI \overline{CSn} - Not available in this package 3 pins - PWMB current sense	EMI Chip Selects EMI Chip Selects SCI1 N/A PWMB current sense
E	14	12	2 pins - SCIO 2 pins - EMI Address pins - Not available in this package 4 pins - SPI0 2 pins - TMRC 4 pins - TMRD	SCIO N/A SPI0 TMRC TMRD
F	16	4	4 pins - EMI Data - Can only be used as GPIO 12 pins - EMI Data - Not available in this package	EMI Data N/A

Fuente: Hoja de datos del fabricante

3.1.1.5. CARACTERÍSTICAS DE MEMORIA

A continuación se muestran las características de memoria del DSP 56F8345, el código de programación se almacena en la memoria *flash* programable de 128KB. La tabla 2 muestra la configuración de la memoria interna del DSP56F8345

- Arquitectura Harvard que permite hasta tres accesos de memoria simultáneos para programar y obtener datos de memoria.
- Protección de seguridad *flash*.
- Memoria interna de chip que incluye solución de flash de alto volumen.
- 128 KB de flash programable.
- 4KB de RAM programable.
- 8KB de flash para datos.
- 8 KB de flash para auto arranque.
- Capacidad de emulación de EEPROM.

- Las memorias operan a 60 MHz.

Tabla 2: Configuración de memoria interna del DSP56F8345

On-Chip Memory	56F8345	56F8145	Use Restrictions
Program Flash	128KB	128KB	Erase / Program via Flash interface unit and word writes to CDBW
Data Flash	8KB	—	Erase / Program via Flash interface unit and word writes to CDBW. Data Flash can be read via either CDBR or XDB2, but not by both simultaneously
Program RAM	4KB	—	None
Data RAM	8KB	8KB	None
Program Boot Flash	8KB	8KB	Erase / Program via Flash Interface unit and word writes to CDBW

Fuente: Hoja de datos del fabricante.

3.1.1.6. CIRCUITOS PERIFÉRICOS

El DSP56F8345 posee una buena cantidad de circuitos periféricos que le permite desarrollar su funcionalidad en adquisición de señales, control, comunicación, etc., con elementos externos. En la figura 18 se pueden ver los grupos funcionales del DSP, se aprecian las señales por su función principal y su función alterna entre paréntesis, son enumerados a continuación algunos de ellos:

- El DSP56F8345 cuenta con módulos PWM cada uno con seis salidas, tres entradas con sensores de corriente y cuatro entradas por defecto, permitiendo modo de alineado al centro y alineado al margen.
- Cuatro conversores analógico-digital de 12 bits, los cuales soportan cuatro conversiones simultáneas con cuadratura, entradas multiplexadas de 4 pines. Dos decodificadores de cuadratura de cuatro entradas.
- Un sensor de temperatura que puede ser conectado en tarjeta a las entradas del ADC para monitorear la temperatura del chip.

- Cuatro temporizadores de cuadratura dedicados de propósito general completando en general seis pines dedicados. Regulador opcional interno.
- Módulo FlexCAN con puerto de dos pines para transmitir y recibir.
- Dos interfaces de comunicación serial (SCIs) cada uno con dos pines (o cuatro líneas GPIO adicionales).
- Dos interfaces periféricas serial (SPIs) ambas con puerto configurables de cuatro pines (u ocho líneas adicionales de GPIO); SPI1 puede usarse también como decodificador de cuadratura.
- Dos pines de interrupción externa.
- 49 pines GPIO; 28 pines dedicados a GPIO.
- Pin de reset externo para reinicio de hardware.
- Pin externo de reset para reinicio del sistema.
- JTAG/OnCE para depurar en tiempo real, independiente de la velocidad del procesador.
- PLL basado en sintetizador de frecuencia, programable en software para el reloj del núcleo.

Además se muestra la figura 19 donde se presenta una vista superior del integrado con la función para cada uno de sus 128 pines.

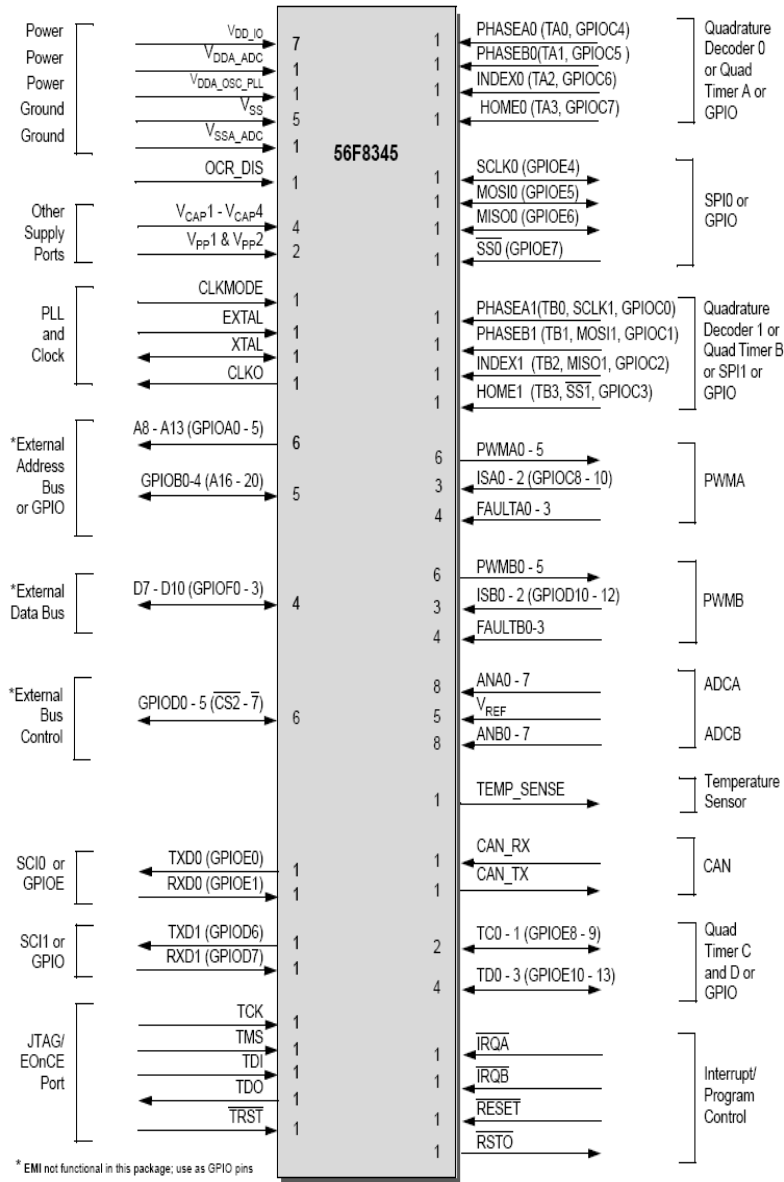
3.1.2. PUERTO JTAG/ONCE

El puerto JTAG/OnCE es básicamente un conector de 14 pines como el que se muestra en la figura 20, que le permite al usuario programar y depurar en tiempo real.

La figura 21 muestra la configuración del puerto JTAG (*Joint Action Group*). El JTAG es un puerto de acceso de prueba (TAP, *Test Access Port*) que brinda una forma de entrar en modo de depuración y acceder al controlador OnCE. Contiene cinco señales como se muestra en la tabla 3, el puerto OnCE tiene

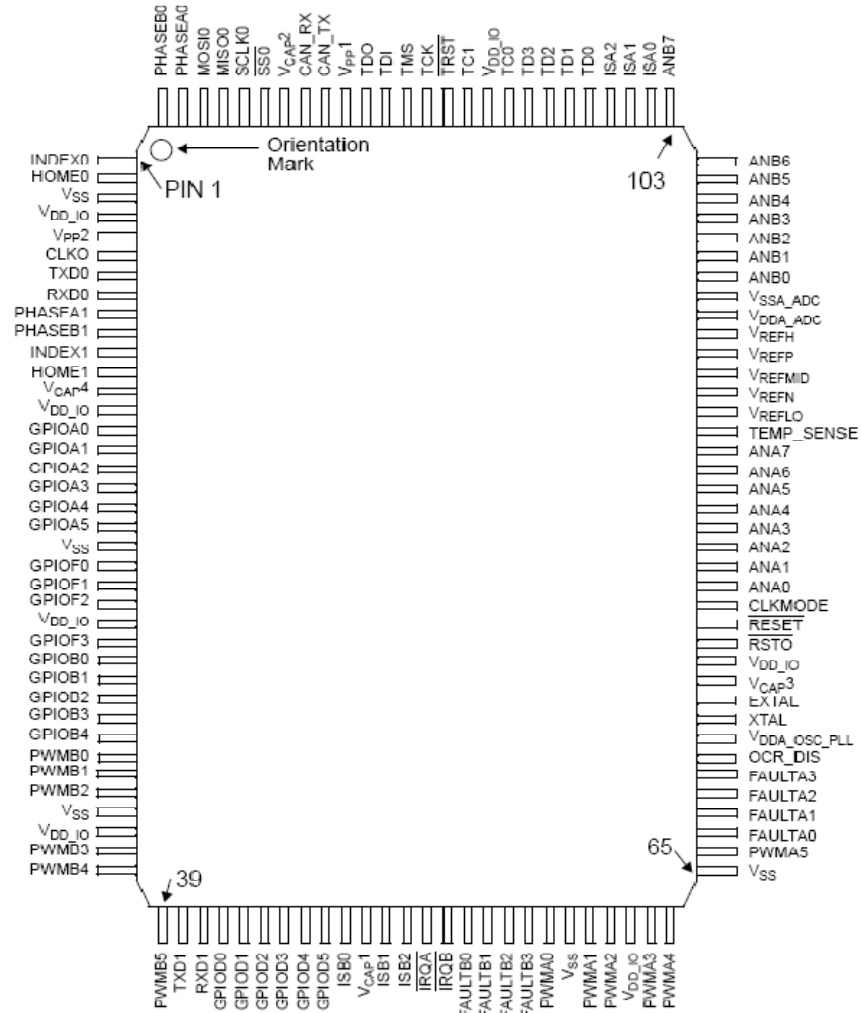
una sola señal DE para ser utilizada por el modulo OnCE. Las señales en negrita son activas bajas.

Figura 18: Señales del DSP56F8345 identificadas por grupo funcional



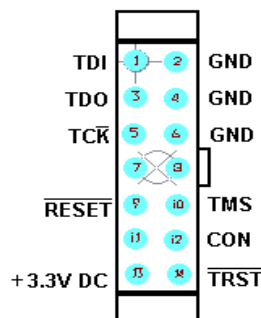
Fuente: Hoja de datos del fabricante.

Figura 19: Vista superior DSP 56F8345.



Fuente: hoja de datos del fabricante.

Figura 20: Puerto JTAG/OnCE

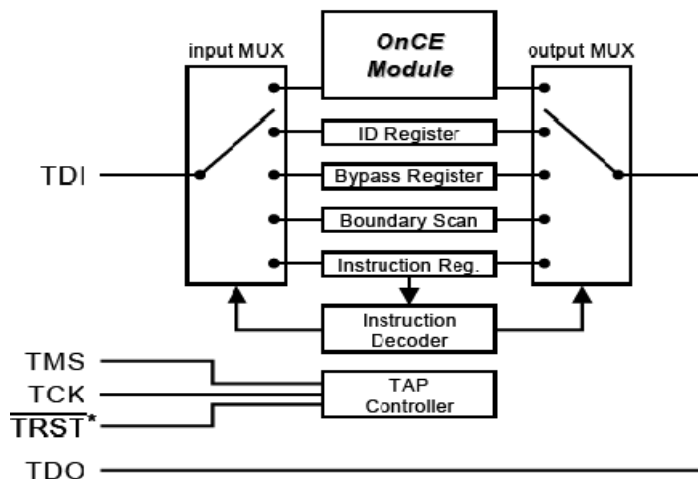


Fuente: autores del proyecto.

El JTAG maneja las recomendaciones del estándar IEEE 1149.1 1149.1^a-1993 *Standard Test Access Port and Boundary Scan Architecture*. (Estándar de acceso de prueba de Puerto y revisión de contorno de arquitectura), desarrollado para solucionar problemas asociados a circuitos de alta densidad, ya que a través de la máquina de estados que compone el TAP, accede y cambia el estado de cada uno de los pines del chip; la acción de prueba resulta muy lenta ya que una operación tan sencilla como escribir en una dirección de memoria puede tardar miles de ciclos de reloj, pero permite tener control total sobre la líneas del chip, teniendo esto una gran utilidad cuando el chip esta ya implementado en la tarjeta que lo contiene.

Para acceder al puerto JTAG es necesario disponer de una interfaz entre las cinco líneas del puerto del chip y el PC desde donde se entregará al chip los datos que se requieran permitiendo abordar la memoria *Flash* dentro del chip reprogramándola como se desee.

Figura 21: Diagrama de bloques del puerto JTAG.



Fuente: Programming On-Chip Flash Memories of 56F80x Devices Using the JTAG/OnCE Interface, Freescale Semiconductor.

Tabla 3: Descripción de los pines del JTAG/OnCE.

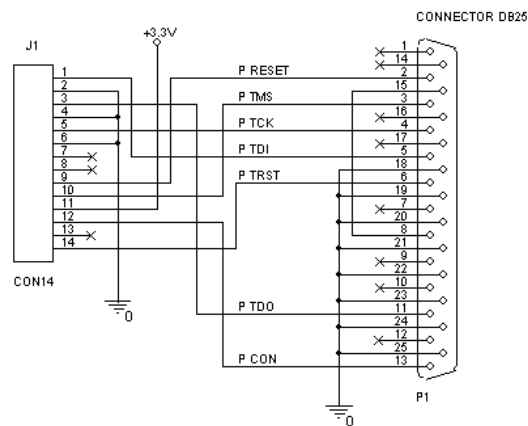
SEÑAL		DESCRIPCIÓN
DE	Evento de Depuración	Bidireccional. Como entrada permite inicializar el modo de depuración con un conversor de comando externo. Como salida informa al conversor externo que el chip ha entrado en modo de depuración.
TCK	Reloj de prueba	Entrada. Sincroniza la lógica de prueba.
TMS	Selección de modo prueba	Entrada. Secuencializa la máquina de estados del controlador de pruebas. Muestreado en el flanco de subida de TCK.
TDI	Datos de prueba de entrada	Salida. Recibe instrucciones de prueba seriales y datos. Muestreado en el flanco de subida de TCK
TDO	Datos de prueba de salida	Salida. Envía instrucciones de prueba y datos. Muestreado en el flanco de bajada de TCK.
TRST	Reset de prueba	Entrada. Inicializa el controlador de prueba.

Fuente: Diseño e implementación de un prototipo de contador de energía trifásico utilizando un procesador de señales digitales (DSP), Pérez Díaz Wilson Andrés, 2006.

La figura 23 muestra la interfaz de programación que se implementó, generalmente esta interfaz es sencilla siendo un circuito que transforma las cinco líneas del JTAG a pines de puerto paralelo; el “software” de programación de PC para el chip envía la información a este puerto y el circuito JTAG se

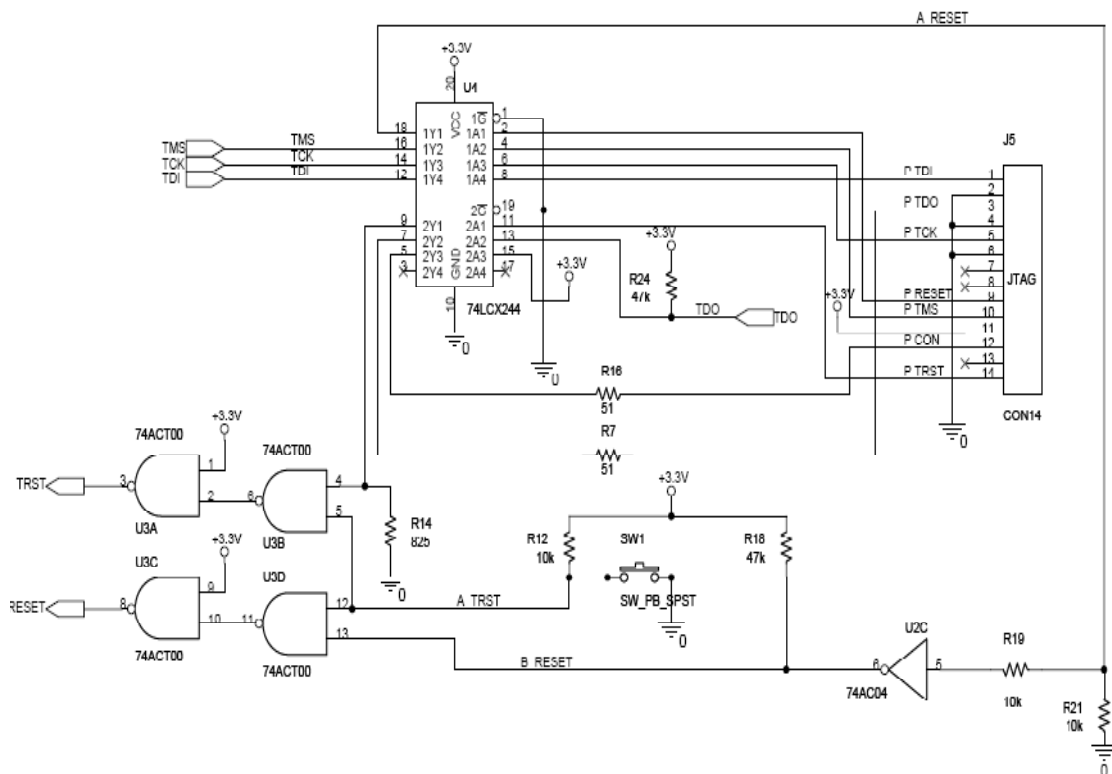
encarga de transmitirle a la tarjeta. El puerto JTAG permite programar el DSP56F8345 utilizando la herramienta de desarrollo *CodeWarrior* a través del puerto paralelo del PC. De éste conector sale el cable de programación que termina en un conector DB25 macho conectado de acuerdo a la figura 22.

Figura 22: del con 14 al DB25



Fuente: Autores del proyecto

Figura 23: Esquema de conexión de la interface de programación.



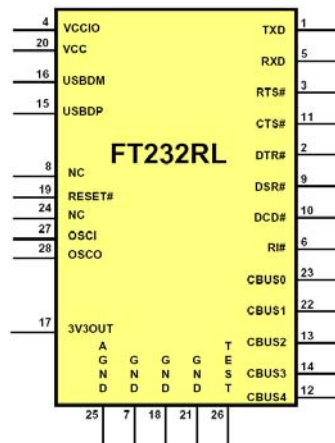
Fuente: Autores del Proyecto.

3.1.3. INTERFACE USB

Esta interface permite la comunicación del PC con el generador. Por medio de ésta se reciben datos desde una aplicación grafica realizada en Matlab, con el fin de lograr un manejo más amigable entre el operario y el generador; logrando ventajas tales como facilidad de conexión y velocidad entre otras.

En el proceso de comunicación con el PC se utiliza la interfaz UART (*Universal Synchronous Asynchronous Receiver Transmitter*) integrada en el modulo SCI del DSP compuesto de dos pines uno de trasmisión y uno de recepción, adicional a esto se agregaron dos pines de control para la recepción y transmisión mediante la configuración de pines GPIO. La conversión UART a USB es realizada mediante el integrado FT232R que se muestra en la figura 24, fabricado por FTDI (*Future Technology Devices International*), este dispositivo toma la trama RS232 y la transforma a una trama válida para el bus USB y a continuación la transmite al PC.

Figura 24: Diagrama de pines del FT232R



Fuente: hoja de datos de fabricante.

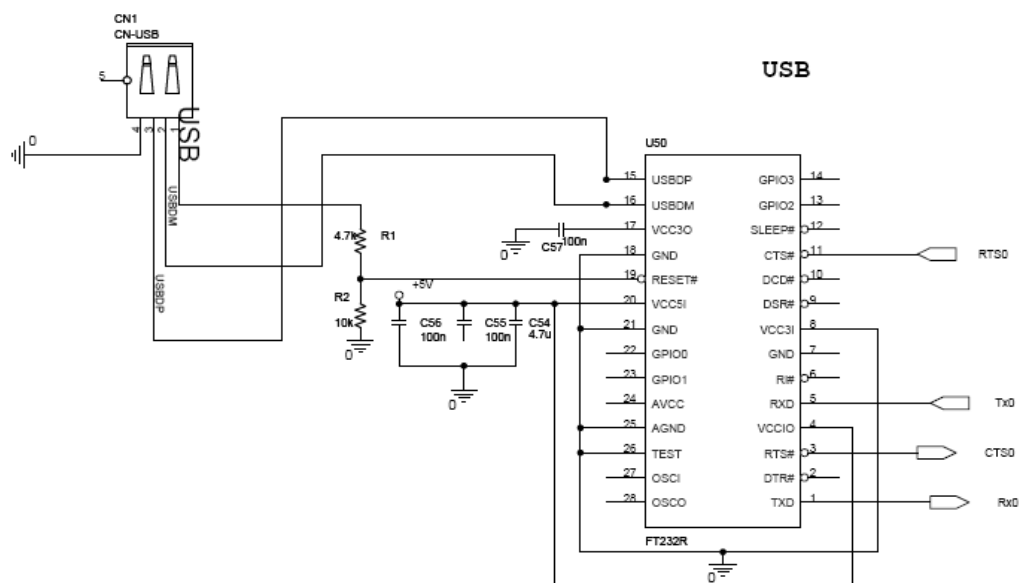
El FT232R permite una transformación transparente en ambas vías, pues en el PC crea un dispositivo serie virtual mediante la instalación del *driver* provisto por el fabricante del dispositivo. Una vez instalado el dispositivo puede ser

accedido por cualquier programa de aplicación de PC como si se tratara de cualquier puerto serie. El esquemático de la interface USB se muestra en la figura 25.

El controlador FT232R brinda las siguientes ventajas:

- EEPROM integrada y programable por medio del USB.
- A cada chip FT232R es asignado un único serial el cual es grabado en el dispositivo cuando es fabricado, este *id* no puede ser reprogramado.
- Circuito de reloj integrado.
- Disponibles dos tipos de drivers:
 - **Drivers VCP:** crean un puerto serial virtual para trabajar sobre aplicaciones que utilicen puertos serie para comunicaciones y programas tales como *Labview*, *Visual Studio*, o *C++ Dephi*.
 - **Drivers D2XX:** Permiten trabajar con las librerías dinámicas del sistema operativo (DLL) en conjunto con el kit de desarrollo de drivers (DDK) para desarrollar drivers para aplicaciones específicas.

Figura 25: Esquemático modulo USB

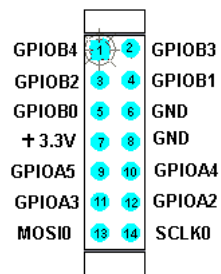


Fuente: Hoja de datos del FT232R

3.1.4. PUERTO PARA COMUNICACION CON EL DAC

Este puerto está compuesto por 16 pines como se muestra en la figura 26, de los cuales 11 pines vienen del DSP, que son los encargados de la transmisión de los datos para hacer la conversión, para la comunicación con el DAC se utilizo el modulo SPI de DSP lo que hizo sencilla la comunicación utilizando solo tres hilos (MOSI, SCLK y un GPIO)

Figura 26: Puerto DAC.

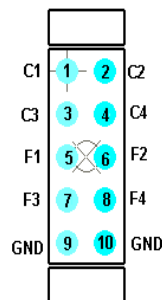


Fuente: Autores del proyecto

3.1.5. PUERTO PARA TECLADO

Este puerto costa de 10 pines que van directamente a un teclado matricial de 4x4, la figura 27 muestra como se distribuyeron los pines, la C indica la columna y la letra F la fila.

Figura 27: Puerto Teclado.

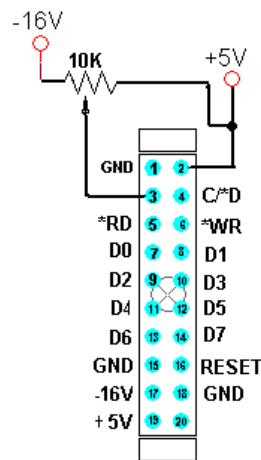


Fuente: Autores del proyecto

3.1.6. PUERTO PARA PANTALLA LCD

Este puerto provee los pines necesarios para el funcionamiento de la pantalla grafica utilizada en el prototipo final tal como se observa en la figura 28, incluye voltajes requeridos según la hoja de datos del fabricante de la pantalla así como los pines de comunicación con el DSP.

Figura 28: Puerto Pantalla LCD



Fuente: Autores del proyecto

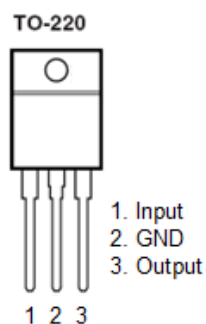
3.1.7. SISTEMA DE ALIMENTACIÓN

La alimentación del sistema en general se hace mediante un transformador de de dos devanados con tap central que suministra dos tensiones de salida de 12Vrms, estas tensiones se pasan a través de un puente rectificador obteniendo así una tensión de +12Vrms y otra de -12Vrms, a partir de estas tensiones se diseñaron fuentes reguladas que alimentan los diferentes integrados que conforman el dispositivo.

Para energizar la tarjeta de desarrollo se creó una fuente de +12V implementando el regulador 7812PI de KIA (figura 29), que toma como tensión de entrada los +12Vrms suministrados por el puente rectificador, este regulador se caracteriza por suministrar un voltaje fijo a la salida de +12V sin necesidad

de muchos componentes externos, con una corriente máxima de salida de 1A, además cuenta con un sistema interno que limita la corriente de corto circuito; a partir de esta tensión se diseñaron fuentes reguladas para cumplir con las necesidades de los diferentes circuitos integrados que conforman la tarjeta de desarrollo y la pantalla del sistema. Se crearon fuentes reguladas de 5.0 V, 3.3 V, y -16 V.

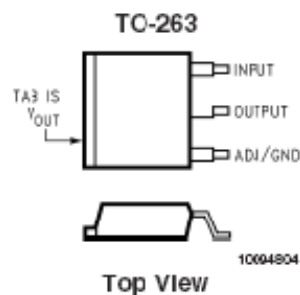
Figura 29: 7812PI regulador fijo 12V



Fuente: hoja datos del fabricante

Mediante la implementación del regulador LM1086 de *National Semiconductor* (figura 30), que es un regulador fijo, lineal tipo LDO (*Low Drop-Out Voltage*) y con una alta eficiencia se obtiene 5V que se encargan de alimentar la pantalla LCD y el circuito integrado FT232R encargado de la comunicación serial con el aplicación de MATLAB.

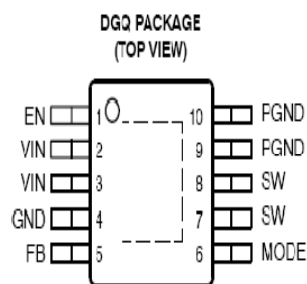
Figura 30: Regulador LM1086



Fuente: hoja de datos del fabricante

Para obtener la tensión de 3.3 V requerida por el DSP y los circuitos integrados LCX244, 74AC04 y 74AC00 que pertenecen a la interfaz programación se utilizó el regulador TPS62046 de *Texas Instruments* (figura 31), que tiene una eficiencia del 95% y permite un rango de tensión de entrada entre 2.5V y 6.0V con una corriente de salida hasta 1.2A., se tomó como tensión de entrada los 5V obtenidos del regulador LM1086.

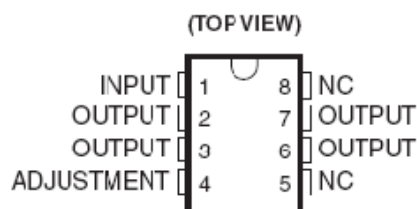
Figura 31: diagrama de pines del regulador TPS62046



Fuente: hoja de datos del fabricante.

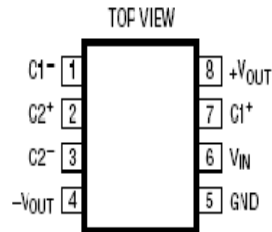
La pantalla LCD necesita una tensión de operación de -16V, que se obtienen mediante una configuración de reguladores, en primera medida se usa el regulador ajustable LM317 de *Texas Instruments* que se muestra en la figura 32, con este regulador se obtiene una tensión de 8V, luego este voltaje es doblado a +16V utilizando el regulador LT1026 de *Linear Technology* que se muestra en la figura 33 y luego utilizando el mismo regulador pero en otra configuración se invierte la polaridad y se obtiene -16V.

Figura 32: Regulador Ajustable LM317



Fuente: Hoja de datos del fabricante.

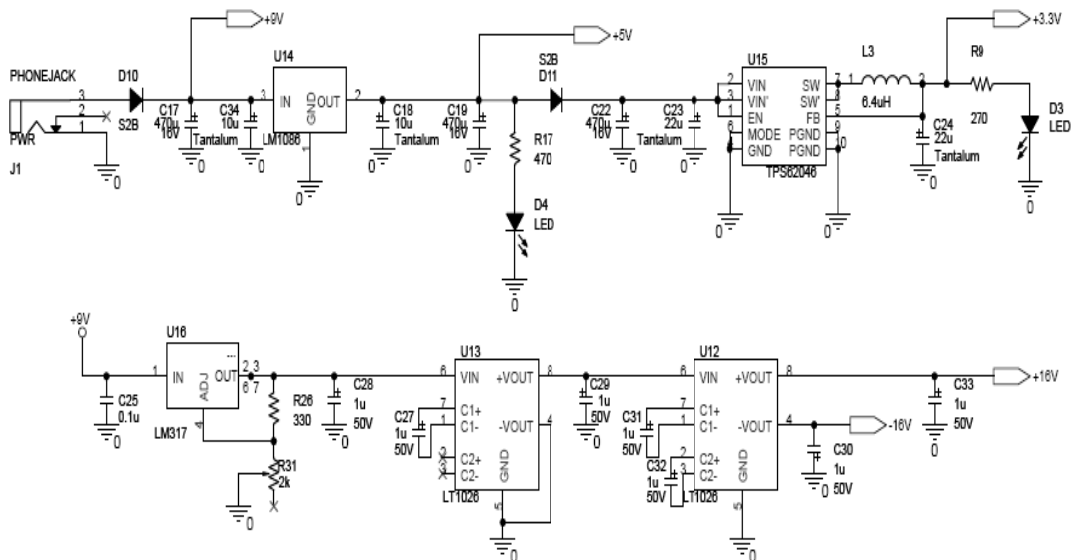
Figura 33: Regulador LT1026



Fuente: hoja de datos de fabricante

En la figura 34 se muestra el esquema general de las fuentes diseñadas en la tarjeta de desarrollo.

Figura 34: Fuente de alimentación del generador

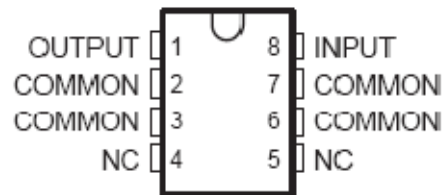


Fuente: autores del proyecto.

Para la tarjeta que se encarga de la conversión digital-analógica y del acondicionamiento de la señal de salida se crearon tres fuentes una de +12V utilizando el regulador UA78L12 (Figura 35), una de -12V mediante el regulador MC79L12 (Figura 36) de *Texas Instruments*, este es un regulador fijo de -12V, que no requiere componentes externos; estas dos fuentes se encargan de alimentar los amplificadores operacionales que se utilizaron para los filtros y los sumadores, y por ultimo una fuente de 5V que se obtuvo a través del regulador

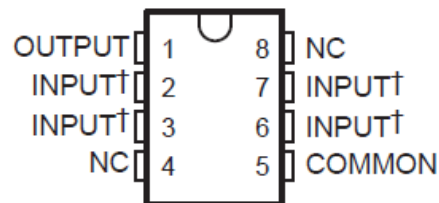
LM1086 (figura 30) que es utilizado por los sumadores para eliminar la componente de continua que traen las señales del DAC, además esta misma tensión alimenta el DAC. En la figura 37 muestra la fuente que alimenta el sistema.

Figura 35: UA78L12 regulador fijo +12V



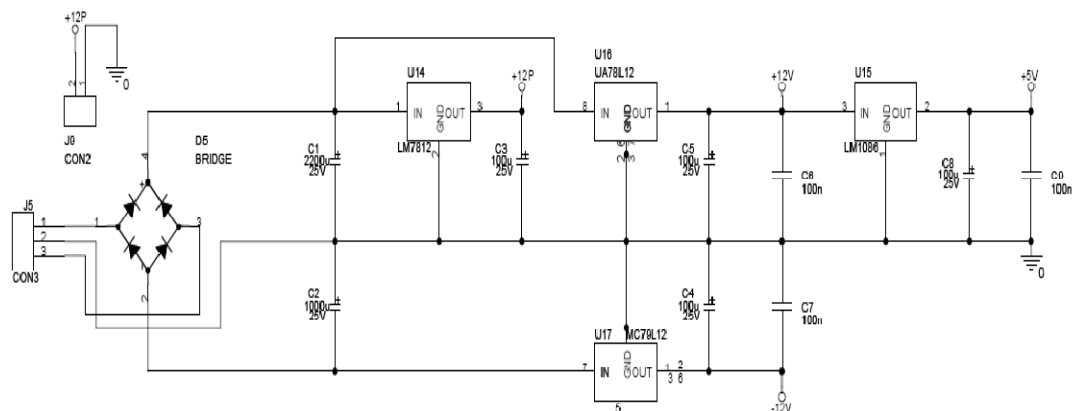
Fuente: hoja datos del fabricante

Figura 36: MC79L12 regulador fijo -12V



Fuente: hoja datos del fabricante

Figura 37: fuente alimentación del sistema



Fuente: autores del proyecto

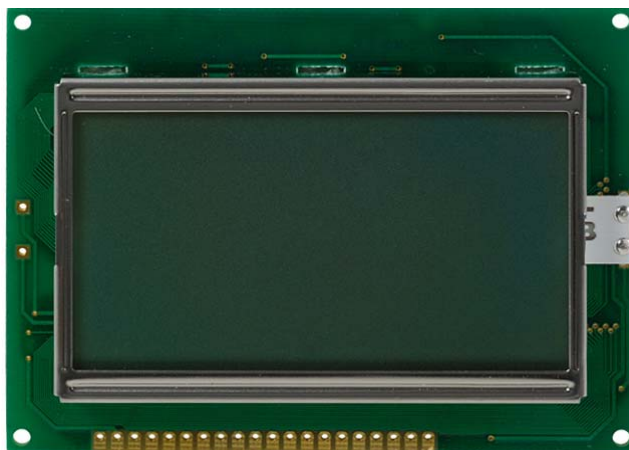
3.2. INTERFAZ CON EL USUARIO

El prototipo del generador de formas de ondas de tensión y corriente para sistemas trifásicos cuenta con una interfaz que permite al usuario interactuar con el sistema. Para la implementación de la interfaz se utilizó una pantalla gráfica de cristal líquido marca *Lumex* y un teclado matricial de 4x4.

3.2.1. PANTALLA GRAFICA DE CRISTAL LIQUIDO

La interfaz gráfica para visualizar el menú del sistema, los datos ingresados por el usuario, las señales generadas y las diferentes modificaciones que se le pueden efectuar se realizó mediante una pantalla de cristal líquido marca LUMEX referencia LCM-S240128GSF como la que se muestra en la figura 38, la cual cuenta con una resolución de 240x128 píxeles y posee una gran capacidad de combinar tanto textos como gráficos.

Figura 38: Pantalla gráfica LCD Lumex de referencia LCM-S240128GSF



Fuente: autores del proyecto

En la Tabla 4 y en la Tabla 5, se pueden observar algunas de las especificaciones de la pantalla LCM-S240128GSF, proporcionadas por el fabricante en la hoja de datos.

Tabla 4: Especificaciones de la pantalla LCD

Ítem	Especificación	Unidades
Tamaño de la pantalla LCD	264 x 183 x 12	mm
Área de visualización	114 x 64	mm
Tamaño del punto	0.4 x 0.4	mm
Formato de visualización	240 x 128	Puntos

Fuente: Hoja de datos del fabricante.

Tabla 5: Tensión de alimentación de la pantalla LCD a 25°

Ítem		Valor			Unidades
		Mínimo	Típico	Máximo	
Tensión de alimentación	Lógica	+4.75	5.0	5.25	V
	LCD	-15.5	-16	-16.5	V

Fuente: Hoja de datos del fabricante.

El manejo de la pantalla grafica se realiza mediante el controlador T6963C de Toshiba, que permite su direccionamiento, acceso y entrada de información fácilmente. Algunas de las características que posee el controlador son:

- Posee un bus de datos paralelo de 8 bits y líneas de control para escritura o lectura.
- Cuenta con un generador interno de caracteres, con 128 caracteres predeterminados.
- Puede ser utilizado en texto, gráfico y una combinación ambos modos.

- Tiene una amplia gama de formatos para la pantalla LCD permitiendo la selección de diversas combinaciones a través del sistema de entradas programables.

El modulo compuesto por la pantalla y su controlador utiliza 20 pines como interfaz de conexión. Cuatro de estos pines se utilizan para alimentación y el resto son utilizados como terminales de entrada/salida (datos o comandos). En la tabla 6 se observan cada uno de los terminales con su respectiva función.

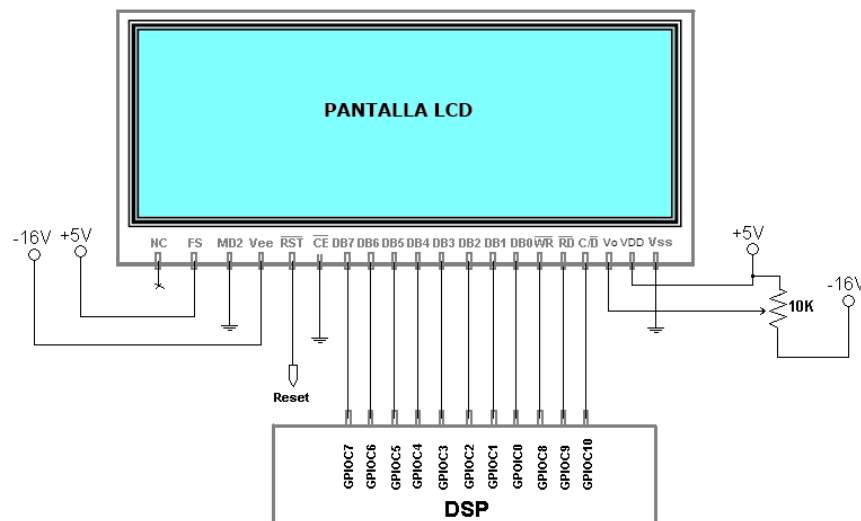
Tabla 6: Terminales de la pantalla LCD

No PIN	SIMBOLO	NIVEL	FUNCION
1	V_{ss}	0V	Tierra.
2	V_{DD}	+5V	Tensión de alimentación para la lógica.
3	V_o	-	Tensión de operación de la LCD (Variable).
4	C/\overline{D}	H/L	Señal para escoger el tipo de dato
5	\overline{RD}	L	Señal de lectura.
6	\overline{WR}	L	Señal de escritura.
7-14	$DB0 - DB7$	H/L	Línea de bus de datos.
15	\overline{CE}	L	Señal de selección del chip.
16	\overline{RST}	L	Señal de reset.
17	V_{ee}	-16V	Tensión de alimentación para LCD.
18	$MD2$	H/L	# columnas: "H": 32 columnas, "L": 40 columnas
19	FS	H/L	Tamaño del carácter: "H": 6x8 , "L":8x8
20	$N.C.$		

Fuente: hoja de datos del fabricante

En la figura 39 se observa el diagrama de la conexión del DSP con la pantalla gráfica, cada pin de la pantalla tiene una numeración que coincide con la de la tabla anterior. Nótese que el terminal 3 va conectado a la tensión de operación de la LCD la cual se gradúa por medio de un *trimmer* de $10K\Omega$, conectado entre +5V y -16V, la variación de esta tensión de operación V_0 permite manejar el contraste de la pantalla. El terminal de reset es activado mediante un pulsador. Los terminales 7-14 (DB0-DB7) de la pantalla, corresponden a las entradas y salidas de datos. Las señales que llegan a estas terminales son generados por el DSP y pueden ser comandos o datos.

Figura 39: Diagrama de conexión de la pantalla LCD



Fuente: Autores del proyecto.

3.2.2. TECLADO 4X4

Con el fin de brindar al usuario una interacción con las diferentes opciones que ofrece el sistema, éste cuenta con un teclado matricial 4x4 como el mostrado en la figura 40, que se utiliza para desplazarse y elegir las diferentes funciones que tiene el sistema y para ingresar los datos al mismo.

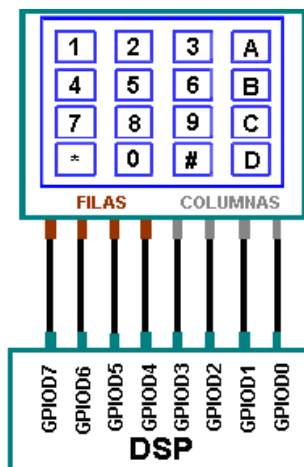
Figura 40: Teclado matricial 4x4



Fuente: Autores del proyecto

El diagrama de la conexión del teclado con el DSP se muestra en la figura 41. El problema de los rebotes se maneja mediante retardos en la programación del DSP.

Figura 41: Conexión del teclado al DSP

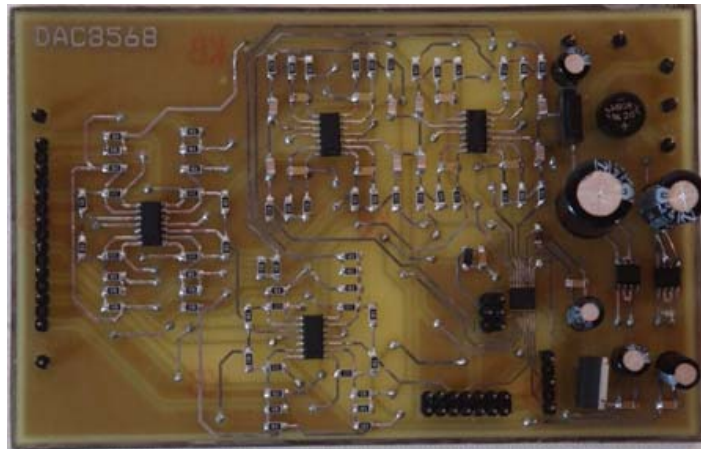


Fuente: Autores del proyecto

3.3. CONVERSION DIGITAL-ANALÓGICA

En esta etapa del proyecto se diseñó una tarjeta que se encarga de hacer la conversión digital- analógica y el acondicionamiento de la señal reconstruida, en la figura 42 se observa la tarjeta diseñada. A continuación se describen cada uno de las partes que componen esta etapa del prototipo.

Figura 42: Tarjeta conversión digital analógica



Fuente: los autores del proyecto

3.3.1. CONVERTOR DIGITAL-ANALÓGICO

En el capítulo 1 se describieron algunas arquitecturas de DAC's y se nombraron unas de sus principales características, basados en esta información y con el fin de realizar una buena conversión se analizaron y probaron varios convertidores de diferentes características tales como: arquitectura, frecuencia de muestreo, número de canales, consumo de potencia, interfaz de comunicación, simplicidad a la hora de hacer el montaje, programación, etc.

Algunas características de los DAC's seleccionados se describen en la tabla 7.

Tabla 7: DAC's escogidos para el diseño

	Arquitectura	# de Bits	Frecuencia de Reloj (f_{sck})	# de canales	Alimentación	Interfaz de comunicación
AD1934	Sigma-delta (Σ - Δ)	24	Hasta 27.6 MHz	8	3.3 V	I2S

DAC1220	Delta-sigma (Δ - Σ)	20	2.5 MHz	1	5V	SPI
MCP4821	<i>Resistor Ladder</i>	12	20 MHz	1	5V	SPI
DAC8568	<i>Resistor Ladder</i>	16	Hasta 50 MHz	8	5V	SPI

Fuente: autores del proyecto.

En primera instancia se intento realizar la conversión digital-analógica utilizando el conversor AD1934, pues su resolución de 24 bits y sus ocho canales de salida lo hacían ver como una excelente opción, éste tipo de conversores son utilizados en sistemas de audio para autos, teatros en casa, procesadores de efecto de audio digital, etc. Para estas aplicaciones *Philips Semiconductors* desarrollo un protocolo de comunicación llamado I2S (por sus siglas en ingles *inter-IC-sound*) que busca estandarizar las estructuras de comunicación. Como el DSP escogido para este proyecto no cuenta con un modulo I2S dentro de su gama de periféricos, se necesitaría emular este protocolo que sumado a la compleja programación comparada con los otros conversores demandaría de un mayor tiempo de diseño obteniendo resultados a los que se podrían lograr con un conversor de menor resolución con una implementación más sencilla.

Por las razones mencionadas anteriormente se descartó este conversor, se buscaron conversores con alta resolución y comunicación SPI pues el DSP cuenta con este módulo y se haría más fácil la comunicación, entonces se utilizó el DAC1220 que es un conversor de 20 bits de resolución de arquitectura sigma-delta con protocolo de comunicación SPI, logrando obtener buenos resultados, pero aun así sin cumplir uno de los objetivos trazados en el desarrollo del proyecto como es el de la cantidad de armónicos, puesto que

este conversor cuenta con bajo tiempo de establecimiento (*Time settling 2ms*). Más adelante se describen las principales características de este conversor.

Luego de los intentos fallidos con los anteriores conversores se implementó el DAC MCP4821 de 12 bits, un conversor que por sus características que se describirán mas adelante puede cumplir a cabalidad con los objetivos propuestos, además es de fácil adquisición pues se encuentra en nuestro mercado local, la gran desventaja de este conversor es que solo tiene un canal lo que demandaría un conversor por cada canal con un total de 7 integrados traduciéndose en un hardware de gran tamaño.

Las tarjetas de pruebas elaboradas para el DAC 1220 y el MCP4821 se dejan como material de apoyo para los laboratorios de sistemas digitales.

Finalmente se adquirió el conversor DAC8568 de *Texas Instruments*, en este conversor se reúnen las características más sobresalientes de los demás conversores estudiados, como por ejemplo: comunicación SPI, número de canales, buen tiempo de establecimiento, alta resolución, fácil programación e implementación que hace que su montaje sea sencillo, este conversor fue el que finalmente se implementó, en el capítulo de pruebas y resultados se describe el comportamiento de este conversor y sus alcances. A continuación se hace un resumen de las principales características de los DAC's mencionados anteriormente.

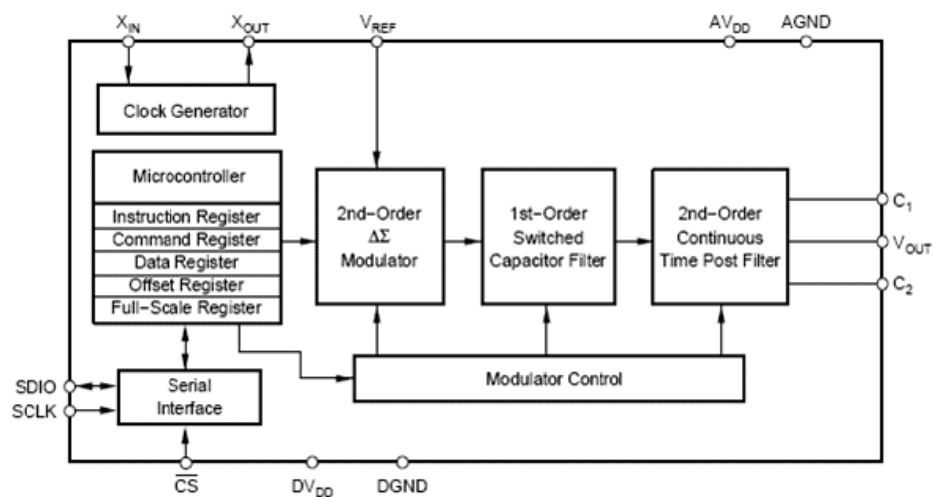
3.3.1.1. CONVERSOR DAC1220

El DAC1220 de *Texas Instruments* es un conversor digital-analógico de 20 bits, basado en tecnología delta-sigma que se seleccionó por su alta resolución, interfaz de comunicación, lo sencillo de su configuración, bajo consumo de potencia y por su facilidad para implementar múltiples conversores.

3.3.1.1.1. TEORÍA DE OPERACIÓN

El núcleo del DAC1220 (figura 43) está compuesto por un filtro de interpolación y un modulador sigma-delta de segundo orden. La salida del modulador es pasada por un filtro de capacitor-conmutado de primer orden, en serie con un filtro de tiempo-continuo de segundo orden que genera la tensión de salida.

Figura 43: Diagrama de bloques DAC1220.



Fuente: Hoja de datos del fabricante.

Para aumentar el tiempo de establecimiento el DAC1220 puede ajustar la frecuencia de corte de su filtro. Además posee un sistema de auto calibración que mide la salida del DAC y calcula las constantes de calibración apropiadas para la ganancia y el *offset*, la salida cambia durante la calibración pero puede ser desconectada durante el proceso. El DAC1220 puede ser puesto en modo de reposo, en donde el consumo de energía se reduce aproximadamente a 0.45 mW, en este modo la salida del DAC es desconectada.

El DAC1220 es controlado mediante una interfaz serial síncrona utilizando dos o tres líneas y se puede operar de forma bidireccional o unidireccional.

El rango de salida de DAC1220 es nominalmente de 0V a 2xVREF, donde VREF es el voltaje de referencia de entrada y está diseñado para 2.5V. El DAC1220 se energiza con dos fuentes una digital y una analógica ambas de 5V. Algunas de las características eléctricas más importantes del DAC1220 están en la tabla 8.

Tabla 8: Características Eléctricas de DAC1220

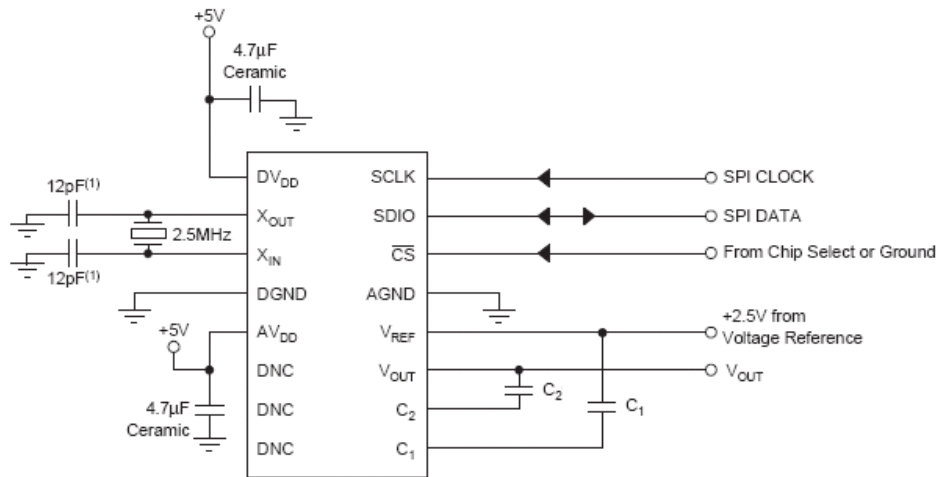
PARAMETROS	CONDICIONES	DAC1220			UNIDADES
		MIN	TYP	MAX	
Exactitud					
Monotonicidad	20 BITS	20			Bits
PSRR	DC, dB = - $20\log(\Delta V_{OUT}/\Delta V_{DD})$		60		dB
Salida analogical					
Voltaje de salida		0		2xVREF	V
Corriente de salida				0,5	mA
Capacitancia de carga			500		pF
Rendimiento dinámico					
Tiempo de establecimiento	20 bits a $\pm 0,003\%$		15		ms
Alimentación					
Fuente de poder		4,75		5,25	V
Corriente análoga	20 bits		460		μ A
Corriente digital	20 bits		140		μ A
Disipación de potencia	20 bits		3		mW
	Modo de reposo		0,45		mW

Fuente: Adaptado de la hoja de datos del fabricante

En la figura 44 se muestra el esquema que se implemento para cada uno de los conversores. El DAC 1220 posee un filtro de tiempo continuo que necesita dos capacitores externos apra trabajar eneste caso que los conversores

operaron a 20 bits el fabricante recomienda para C1 un valor de 10nF y para C2 un valor de 3.3nF.

Figura 44: Esquemático DAC1220



Fuente: Hoja de datos del fabricante

3.3.1.1.2. PROGRAMACIÓN

Comandos: La comunicación con el DAC1220 se hace mediante comandos, a través del acceso de sus registros. Los comandos consisten en un byte de comando seguido de uno, dos o tres bytes de datos. Los bytes de datos pueden ser enviados al DAC1220 o leídos del DAC1220, dependiendo si es un comando de lectura o escritura. El formato del byte de comando se muestra en la tabla 9 y su descripción en la tabla 10.

Registros: Existen cuatro registros en el DAC1220 como se muestran en la tabla 11, el registro de datos de entrada (DIR) y los registros de calibración (OCR y FCR) tienen un ancho de 24 bits cada uno y el registro de comando (CMR) tiene un ancho de 16 bits.

Tabla 9: Formato del byte de comando

7	6	5	4	3	2	1	0
R/W	MB			0	ADR		

Fuente: Hoja de datos del fabricante.

Tabla 10: Bits del byte de comando

BIT(S)	NAME	VALUE	DESCRIPTION
7	R/W	0	Write to register map
		1	Read from register map
6-5	MB		Number of bytes to read or write
		00b	1 byte
		01b	2 bytes
		10b	3 bytes
		11b	Reserved; do not use
3-0	ADR	0-15	Start address in register map

Fuente: Hoja de datos del fabricante.

Tabla 11: Mapa del registro de memoria.

ADDRESS	CONTENT
0	DIR byte 2 (MSB)
1	DIR byte 1
2	DIR byte 0 (LSB)
3	Reserved
4	CMR byte 1 (MSB)
5	CMR byte 0 (LSB)
6	Reserved
7	Reserved
8	OCR byte 2 (MSB)
9	OCR byte 1
10	OCR byte 0 (LSB)
11	Reserved
12	FCR byte 2 (MSB)
13	FCR byte 1
14	FCR byte 0 (LSB)
15	Reserved

Fuente: Hoja de datos del fabricante.

El Comando de Registro (CMR) contiene la configuración de bits del DAC1220 como se muestra en la tabla 12, si el lector quiere saber la descripción

completa de los bits en el registro de comando puede remitirse a la hoja de datos del convertor.

El registro de dato de entrada (DIR) determina el voltaje de salida cuando el DAC está en modo normal. En modo de reposo escribir en este registro no tiene efecto en la salida, pero el valor es guardado y llega a ser efectivo inmediatamente después que el DAC pase a modo normal. Después de reiniciar el DIR contiene cero.

El registro de calibración de offset (OCR) contiene un valor de 24 bits en complemento a dos, este valor es sumado al valor en DIR antes de la conversión por el DAC. En modo de reposo escribir en este registro no tiene efecto en la salida, el valor es guardado y se hace efectivo inmediatamente después que el DAC entra en modo normal. Después de reiniciar OCR contiene cero.

Tabla 12: Registro de comando.

15	14	13	12	11	10	9	8
ADPT	CALPIN	Reserved	Reserved	Reserved	Reserved	CRST	Reserved
R/W-0	R/W-0	R-1 ⁽¹⁾	R-0	R-1	R-0	R/W-0	R-0
7	6	5	4	3	2	1	0
RES	CLR	DF	DISF	BD	MSB	MD	
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-10b	

Fuente: Hoja de datos del fabricante.

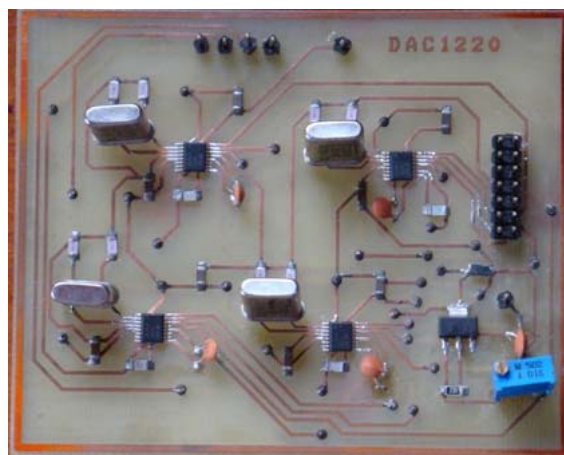
El registro de calibración de escala completa (FSR) guarda la constante de ganancia de calibración. El contenido de DIR es ajustado multiplicándolo por este valor antes de que el DAC haga la conversión. En modo de reposo escribir en este registro no tiene efecto en la salida, el valor guardado en FSR y se hace efectivo inmediatamente después que el DAC entra en modo normal. Después de reiniciar FSR contiene 800000h.

En la figura 45 se muestra la tarjeta implementada del DAC1220 con la cual se realizaron las pruebas a este convertor.

3.3.1.2. CONVERTOR MCP4821

El DAC MCP4821 de *Microchip* es un convertor digital-analógico de 12 bits de resolución que pertenece a una familia de convertidores de alta precisión y bajo ruido. Este DAC está diseñado bajo la arquitectura *resistor ladder* descrita en el capítulo 1, con grandes ventajas como bajo error DNL y un bajo tiempo de establecimiento, además cuenta con amplificadores de salida rail-to-rail, voltaje de referencia interno, manejo de apagado y reinicio de los circuitos, protocolo de comunicación SPI, el cual es requerido para facilitar la comunicación con el DSP escogido y aumentar el número de armónicos y cumplir con el objetivo trazado.

Figura 45: Tarjeta elaborada para el DAC 1220



Fuente: autores del proyecto

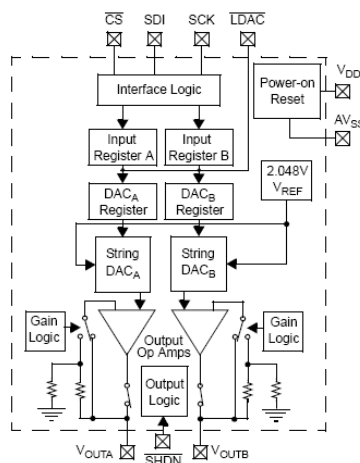
Los códigos de este DAC son completamente binarios, con un voltaje de salida ideal regido por la ecuación 2.1, donde G es la ganancia seleccionada ($1x$ o

2x), D_N representa el valor de entrada digital y n representa el numero de bits de resolución (n=12).

$$V_{OUT} = \frac{2.048V * G * D_N}{2^n} \quad (3.1)$$

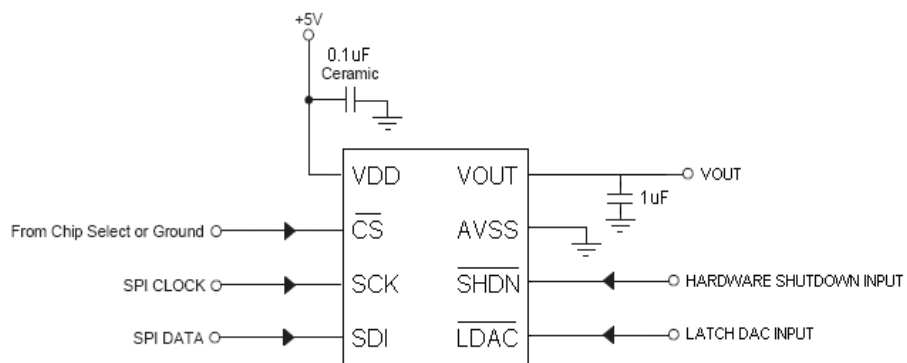
En la figura 46 se muestra el diagrama de bloques general del convertor digital-analógico MCP4821, y en la figura 47 se observa el esquemático implementado para cada uno de los DACs utilizados.

Figura 46: Diagrama de bloques DAC MCP4821



Fuente: Hoja de datos del fabricante.

Figura 47: Esquemático DAC MCP4821.



Fuente: los autores.

3.3.1.2.1. TEORIA DE OPERACIÓN:

Amplificadores de Salida: El DAC tiene a la salida un amplificador CMOS de precisión y baja potencia. Este amplificador da un bajo voltaje de *offset* y bajo ruido. La etapa de salida permite que el DAC funcione con tensiones de salida cercanos a los de la fuente de alimentación. Además de la capacidad de manejar cargas resistivas, el amplificador también puede manejar cargas capacitivas sin grandes oscilaciones.

Bloque de Ganancia Programable: El amplificador de salida rail-to-rail tiene ganancia configurable, permitiendo una óptima salida a plena escala para diferentes voltajes de entradas. El amplificador de salida tiene para seleccionar dos ganancias, una ganancia de 1V/V ($\overline{GA}=1$) y una ganancia de 2V/V ($\overline{GA}=0$). El rango de salida idealmente va de 0.0V a $(4095/4096)*2.048V$ con $G=1$, y de 0.0V a $(4095/4096)*4.096V$ con $G=2$, esta ganancia es la que viene por defecto.

Voltaje de Referencia: El DAC MCP4821 utiliza un voltaje de referencia interno de 2.048V. El voltaje de referencia tiene un bajo coeficiente de temperatura y una característica de bajo ruido.

Circuito de Encendido y Reinicio (POR): Este circuito asegura que el DAC este encendido con $\overline{SHDN}=0$ (alta impedancia). Las salidas seguirán en alta impedancia hasta que un comando válido de escritura se realice en cualquiera de los registros del CAD y el pin LDAC cumpla con el mínimo nivel de entrada. Si el voltaje de alimentación es menor que el umbral de POR ($V_{POR} = 2.0V$), el DAC mantendrá su estado de reinicio. El DAC permanecerá en este estado hasta que $V_{DD} > V_{POR}$ y posteriormente reciba un comando de escritura.

Modo de Parada: El modo de parada se puede activar mediante el uso hardware o comandos de software. Durante el modo de parada la corriente de alimentación es aislada del resto de circuitos internos. La interfaz serial permanece activa lo que permite escribir comandos para llevar el DAC al modo de parada.

Interfaz Serial: El convertor MCP4821 está diseñado para comunicarse directamente con puerto SPI. Los datos y comandos son enviados a través del pin SDI, los datos son registrados en el flanco de subida de SCK. La comunicación es unidireccional, así los datos no pueden ser leídos por dispositivos fuera del DAC.

El pin \overline{SC} debe permanecer en bajo mientras dura la escritura de un comando. Un comando contiene 16 bits y son usados para la configuración, control y datos de enganche. En la tabla 13 detallan los registros de entrada usados para configurar y cargar los registros del DAC.

Tabla 13: registros de entrada para configurar y cargar los registros al DAC_A y DAC_B

Upper Half:							
W-x	W-x	W-x	W-0	W-x	W-x	W-x	W-x
$\overline{A/B}$	—	\overline{GA}	\overline{SHDN}	D11	D10	D9	D8
bit 15						bit 8	
Lower Half:							
W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x
D7	D6	D5	D4	D3	D2	D1	D0
bit 7						bit 0	

Fuente: hoja de datos del fabricante

Bit 15 $\overline{A/B}$: bit de selección DAC_A o DAC_B .

1= escribe en el DAC_A

0= escribe en el DAC_B

Bit 14 no interesa.

Bit 13 \overline{GA} : bit de selección de ganancia de salida.

$$1 = 1x(V_{OUT} = V_{REF} * D / 4096)$$

$$0 = 2x(V_{OUT} = 2 * V_{REF} * D / 4096)$$

Bit 12 \overline{SHDN} : bit de control de salida apagada.

1= bit de control de salida apagada.

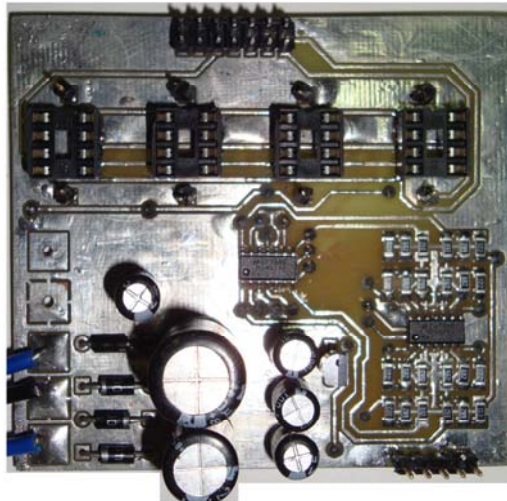
0= buffer de salida habilitado, salida en alta impedancia.

Bit 11-0 D11:D0: bits de datos del DAC

Numero "D que establece en valor de la salida. Esta comprendido entre 0 y 4095.

En la figura 48 se muestra la tarjeta con la que se realizaron algunas pruebas de funcionamiento al DAC MCP4821

Figura 48: Tarjeta de pruebas del DAC MCP4821



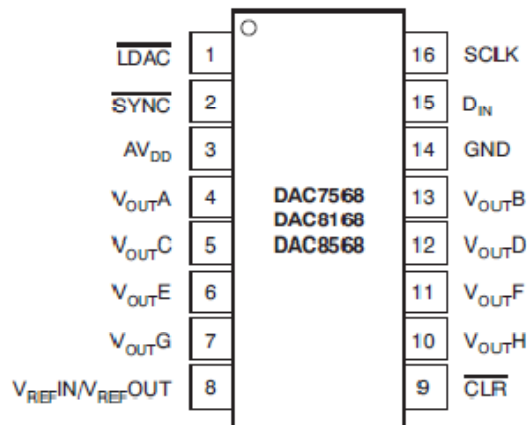
Fuente: autores del proyecto

3.3.2.3. DAC 8568

El DAC 8568 de *Texas instruments* este es un conversor digital – analógico de 16 bits de resolución, de ocho canales, con características como bajo consumo de potencia, amplificadores de salida *rail-to-rail*, voltaje de referencia interno,

manejo de apagado y reinicio de los circuitos, protocolo de comunicación SPI, el cual es requerido para facilitar la comunicación con el DSP escogido; se utilizo el chip de 16 pines mostrado en la figura 49.

Figura 49: DAC 8568



Fuente: hoja de datos del fabricante

3.3.1.3.1 TEORIA DE OPERACIÓN:

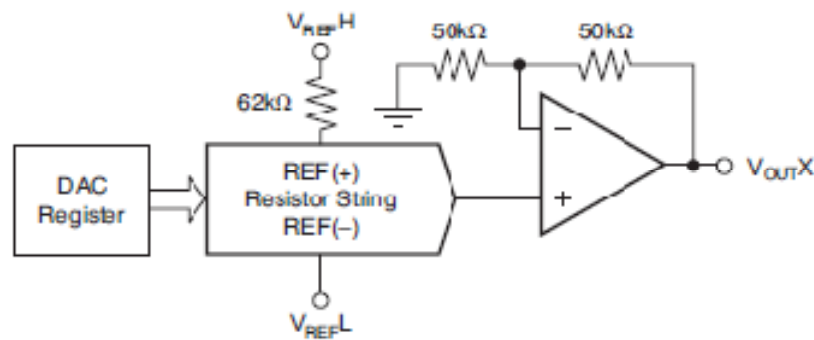
La arquitectura del DAC 8568 mostrada en la figura 50 consiste de una cadena de ocho DACs, cada uno seguido por un buffer de amplificación a la salida. El dispositivo incluye un voltaje de referencia interno de 2.5V. Los códigos de este DAC son binarios, con un voltaje de salida ideal regido por la ecuación 3.2, donde D_{IN} es el equivalente decimal del código binario que se carga en el registro del DAC, el rango para el DAC 8568 va de 0 a 65535, n es 16 (resolución del DAC), y gain es la ganancia 1 para A/B grados o 2 para C/D.

$$V_{OUT} = \left[\frac{D_{IN}}{2^n} \right] \times V_{REF} \times GAIN \quad (3.2)$$

CADENA RESISTORES: Es una cadena simple de resistores cada una de valor R. El código cargado en el registro del DAC determina en cual nodo de la cadena el voltaje es activado para ser alimentada la salida del amplificador

mediante el cierre de uno de los interruptores conectando la cadena al amplificador. Es monotónico porque es una cadena de resistores.

Figura 50: arquitectura del DAC.



Fuente: hoja de datos del fabricante

AMPLIFICADORES DE SALIDA: El buffer amplificador a la salida es capaz de generar voltajes *rail to rail* a la salida, dando un rango máximo a la salida de 0 a AV_{DD} . Es capaz de manejar una carga de 2K en paralelo con 3000pF a tierra. El típico *slew rate* es 0.75V/us con un tiempo de establecimiento de 5 us con la salida sin carga.

REFERENCIA INTERNA: El DAC 8568 posee una referencia interna de 2.5V que se encuentra desactivada por defecto. Externamente esta referencia se encuentra disponible en el pin 8 del chip mostrado en la figura 42. esta referencia puede ser activada o desactivada mediante un comando en serie que requiere escribir una secuencia de 32 bits; durante el tiempo que esta referencia este desactivada el DAC funciona normalmente usando una referencia externa. Existen dos modos, estático y flexible que permiten la comunicación con esta referencia interna, en la figura 51 y 52 respectivamente se observa la secuencia utilizada por los dos modos. En la implementación que se realizo se utilizo una referencia externa de 5V con el fin de obtener un mayor rango de salida.

Figura 51: Secuencias para desactivar y activar referencia interna en modo estático.

SECUENCIA PARA ACTIVAR LA REFERENCIA INTERNA (Internal Reference Powered On—08000001h)																																			
DB31				DB27				DB23				DB19				DB4				DB0															
0	X	X	X	C3	C2	C1	C0	A3	A2	A1	A0	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	F3	F2	F1	F0				
0	X	X	X	1	0	0	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	1			
-- Prefix Bits --				-- Control Bits --				-- Address Bits --				----- Data Bits -----																-- Feature Bits --							

SECUENCIA PARA DESACTIVAR LA REFERENCIA INTERNA (Internal Reference Powered On—08000000h)																																			
DB31				DB27				DB23				DB19				DB4				DB0															
0	X	X	X	C3	C2	C1	C0	A3	A2	A1	A0	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	F3	F2	F1	F0				
0	X	X	X	1	0	0	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	0
-- Prefix Bits --				-- Control Bits --				-- Address Bits --				----- Data Bits -----																-- Feature Bits --							

Fuente: hoja de datos del fabricante.

Figura 52: Secuencias para desactivar y activar referencia interna en modo flexible.

SECUENCIA PARA ACTIVAR LA REFERENCIA INTERNA (Internal Reference Powered On—09080000h)																																			
DB31				DB27				DB23				DB19				DB4				DB0															
0	X	X	X	C3	C2	C1	C0	A3	A2	A1	A0	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	F3	F2	F1	F0				
0	X	X	X	1	0	0	1	X	X	X	X	1	0	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
-- Prefix Bits --				-- Control Bits --				-- Address Bits --				----- Data Bits -----																-- Feature Bits --							

SECUENCIA PARA ACTIVAR LA REFERENCIA INTERNA (Internal Reference Always Powered On—090A0000h)																																			
DB31				DB27				DB23				DB19				DB4				DB0															
0	X	X	X	C3	C2	C1	C0	A3	A2	A1	A0	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	F3	F2	F1	F0				
0	X	X	X	1	0	0	1	X	X	X	X	1	0	1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
-- Prefix Bits --				-- Control Bits --				-- Address Bits --				----- Data Bits -----																-- Feature Bits --							

SECUENCIA PARA DESACTIVAR REFERENCIA INTERNA (Internal Reference Always Powered Down—090C0000h)																																			
DB31				DB27				DB23				DB19				DB4				DB0															
0	X	X	X	C3	C2	C1	C0	A3	A2	A1	A0	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	F3	F2	F1	F0				
0	X	X	X	1	0	0	1	X	X	X	X	1	1	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
-- Prefix Bits --				-- Control Bits --				-- Address Bits --				----- Data Bits -----																-- Feature Bits --							

SECUENCIA PARA ALTERNAR DE MODO FLEXIBLE A MODO ESTATICO PARA REFERENCIA INTERNA (Internal Reference Always Powered Down—09000000h)																																			
DB31				DB27				DB23				DB19				DB4				DB0															
0	X	X	X	C3	C2	C1	C0	A3	A2	A1	A0	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	F3	F2	F1	F0				
0	X	X	X	1	0	0	1	X	X	X	X	0	0	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
-- Prefix Bits --				-- Control Bits --				-- Address Bits --				----- Data Bits -----																-- Feature Bits --							

Fuente: adaptado de hoja de datos de fabricante.

INTERFAZ SERIAL: El DAC tiene una interfaz serial de 3 pines (SYNC, SCLK, y DIN, ver figura 49) que son compatibles con QSPI, y SPI, este ultimo utilizado para nuestro proyecto para lograr la comunicación con el DSP escogido.

El registro de desplazamiento del DAC8568 es de 32 bits de ancho, que consisten en 4 bits de inicio (DB31 a DB28), 4 bits de control (DB27 a DB24) 16 bits de datos (DB23 a DB4) y 4 bits de característica; una representación de esta trama puede ser vista en la figura 46. Cuando se escribe el registro al DAC (transferencia de datos) los bits de característica son ignorados por el DAC y deben ser tratados como bits no importantes. Todos los 32 bits de datos se cargan en el DAC bajo el control del reloj de entrada serial SCLK.

Figura 53: 32 bits de entrada de registro al DAC 8568

Table 7. DAC8568 Data Input Register Format

DB31				DB27				DB23				DB19								DB4				DB0							
0	x	x	x	C3	C2	C1	C0	A3	A2	A1	A0	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	F3	F2	F1	F0
- Prefix Bits -				- Control Bits -				Address Bits				----- Data Bits -----																Feature Bits			

Fuente: hoja de datos del fabricante.

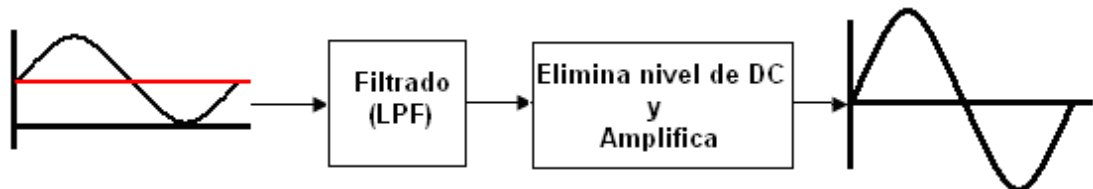
DB31 (MSB) es el primer bit que se carga en el registro de desplazamiento del DAC y debe ser siempre ajustado a 0. Seguido por el resto de los 32-bit de la palabra patrón alineado a la izquierda. Esta configuración significa que los primeros 32 bits de datos son normalmente enganchados en el registro de desplazamiento y cualquier bit adicional es ignorado.

3.3.2. ACONDICIONAMIENTO DE LA SEÑAL DE SALIDA

La señal reconstruida por el DAC 8568 es una señal que tiene una amplitud máxima de 5 Vpp, además cuenta con un nivel de continua de 2.5V, con el propósito de que esta señal de salida cumpla con los requerimientos (rango máximo de salida $\pm 5V$ y frecuencia máxima de interés 3KHz) propuestos en el

proyecto es necesario pasarla por una serie de bloques (Figura 54) que se encargan de acondicionarla para obtener la señal deseada.

Figura 54: Diagrama de bloques acondicionamiento de la señal de salida.



Fuente: los autores.

3.3.2.1. FILTRO BUTTERWORTH

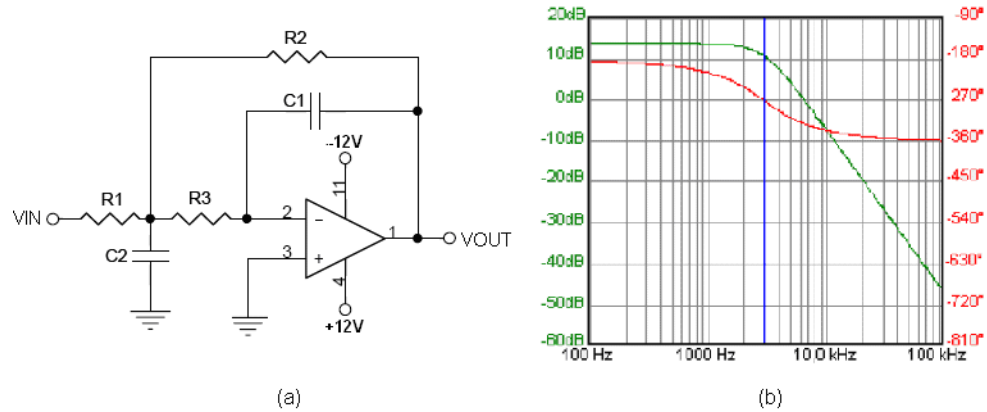
Después que el DAC hace la conversión es necesario filtrar estas señales, con el fin de eliminar las frecuencias que están fuera de la banda de interés. Se diseñó un filtro activo Butterworth MFB⁴ de segundo orden (figura 55) utilizando el programa *FilterPro* de *Texas Instruments*.

El filtro diseñado tiene las siguientes características:

- Ganancia 1 V/V
- Frecuencia de corte 3KHz
- 2 polos
- Inversor

⁴ Múltiple realimentación.

Figura 55: (a) Esquemático filtro. Butterworth de segundo orden y (b) diagrama de Bode (verde) y margen de fase (roja)



Fuente: los autores del proyecto

En la tabla 13 se muestran los valores de resistencias y capacitores arrojados por el programa *FilterPro* y los valores utilizados en la implementación del filtro.

Tabla 13: valores de resistencias y capacitores del filtro

	C1	C2	R1	R2	R3
<i>FilterPro</i>	1nF	10nF	8.45KΩ	8.45KΩ	33.2KΩ
Implementados	1nF	10nF	8.2KΩ	8.2KΩ	33KΩ

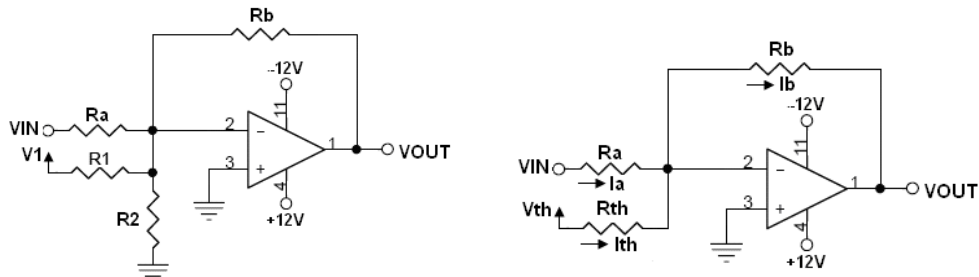
Fuente: los autores

En el capítulo de pruebas se expone el comportamiento ideal mediante la simulación en orcad y el comportamiento real del filtro mediante pruebas hechas en el laboratorio.

3.3.2.2. SUMADOR INVERSOR CON GANANCIA

Después que la señal es filtrada es necesario quitarle el nivel de DC y darle la amplificación que se desea, esto se hace mediante la implementación de un sumador inversor con ganancia, como el que se muestra en la figura 56.

Figura 56: Sumador inversor con ganancia



Fuente: los autores del proyecto

Después que la señal pasa por el filtro se invierte su polaridad por la característica inversora del filtro, basándose en esto se diseño el sumador que debe contar con las siguientes características:

- Sumar un nivel de continua de 2.5V.
- Ganancia de 4V/V.

A continuación se describe el diseño del sumador basados en el circuito descrito en la figura 51.

$$i_a + i_{th} = i_b \quad (3.3)$$

$$\frac{v_{in}}{R_a} + \frac{V_{th}}{R_{th}} = -\frac{V_{out}}{R_b} \quad (3.4)$$

$$\text{Si } R_a = R_{th} \quad V_{out} = -\frac{R_b}{R_a}(V_{IN} + V_{th}) \quad (3.5)$$

Tomando como referencia las anteriores ecuaciones se realizo el diseño.

$$\frac{R_b}{R_a} = 4 \quad (3.6)$$

Se selecciona arbitrariamente un valor para R_a .

$$R_a = 5K\Omega$$

$$R_b = 20K\Omega$$

Ahora se encuentran los valores de R_1 y R_2 , con $V_1=5V$.

$$R_a = R_{th} = \frac{R_1 * R_2}{R_1 + R_2} = 5k\Omega \quad (3.7)$$

$$V_{th} = V_R = \frac{V_1 * R_1}{R_1 + R_2} = 2.5V \quad (3.8)$$

Al resolver las ecuaciones 5 y 6 con $V_1= 5V$ se obtuvieron los siguientes valores:

$$R_1 = 10K\Omega, R_2 = 10K\Omega$$

3.3.2.3. AMPLIFICADOR OPERACIONAL LME49740

Tanto el filtro como el sumador con ganancia se diseñaron implementando el amplificador operacional LME49740, que se caracteriza por tener una baja distorsión, bajo ruido, alto *slew rate*, baja densidad de voltaje de ruido, un gran rango dinámico maximizado por su etapa de salida, también posee un excelente CMRR y PSRR que dan al amplificador un extraordinario desempeño en DC y además asegura el manejo de cargas difíciles sin ningún problema.

Mezclando los avances en los procesos tecnológicos con el estado del arte en el diseño de circuitos, el amplificador operacional LME49740 entrega un altísimo desempeño en sus aplicaciones como son amplificación de audio de alta calidad, pre-amplificadores de alta fidelidad, audio profesional de alto rendimiento, ecualizadores de alta fidelidad y filtro separadores de frecuencia y

para filtros activos de alta fidelidad. Además de todas las características expuestas anteriormente este amplificador operacional tiene una protección contra corto circuito a la salida que permite cumplir con una de las especificaciones que debe tener el prototipo final.

En la tabla 14 se muestran algunas de las principales características del amplificador operacional LME49740.

Tabla 14: características del LME49740

Power Supply Voltage Range	$\pm 2.5V$ to $\pm 17V$
THD+N ($A_V = 1$, $V_{OUT} = 3V_{RMS}$, $f_{IN} = 1kHz$)	0.00003% (typ)
RL = 2K Ω	0.00003% (typ)
RL = 600 Ω	
Input Noise Density	2.7nV/ \sqrt{Hz} (typ)
Slew Rate	$\pm 20V/\mu s$ (typ)
Gain Bandwidth Product	55MHz (typ)
Open Loop Gain (RL = 600 Ω)	140dB (typ)
Input Bias Current	10nA (typ)
Input Offset Voltage	0.1mV (typ)
DC Gain Linearity Error	0.000009%
CMRR	120dB
PSRR	120dB

Fuente: Hoja de datos del fabricante.

4. SOFTWARE

El prototipo del sistema generador de formas de ondas de tensión y corriente para sistemas trifásicos cuenta con una gran aplicación de software, encargada de toda la configuración y programación del procesador digital de señales (DSP).

El programa almacenado en el DSP se encarga de calcular los valores numéricos de las fases que deben ser enviados al DAC, se encarga de dibujar e implementar una interface con el usuario (UI) que permite conectarse y controlar un display gráfico, manejar un teclado matricial, y establecer una comunicación remota para la configuración del sistema sin uso de la UI proporcionada.

La programación del DSP se hace mediante la utilización de un software de desarrollo. El paquete *CodeWarrior™ Development Studio for Freescale™ Digital Signal Controllers V8.0*, es empleado para este fin. Este paquete de *software* permite realizar funciones de compilación de programa, simulación en circuito y descarga del programa desde el PC al DSP.

Además brinda una herramienta llamada *UNIS Processor Expert™*, la cual permite un rápido y fácil desarrollo de programas y aplicaciones utilizando unos componentes denominados *embedded beans™* en donde se encapsulan diferentes elementos de la CPU, los periféricos del DSP y otras funciones del software. Cada uno de los beans puede ser configurado por el usuario según sus necesidades. *UNIS Processor Expert™* genera automáticamente el código C de los diferentes *beans* utilizados.

4.1. GENERACIÓN DE LA ONDA Y LOS ARMÓNICOS

La generación de ondas sinusoidales es común implementarla por medio de tres métodos:

- Series de potencia
- Osciladores biestables (con matrices de estados)
- Tablas

De estos tres mecanismos, el primero posee el inconveniente de requerir, cuando se usa la expansión de Taylor, una cantidad considerable de cálculos matemáticos que suelen ser implementados por medio de notación flotante. El segundo mecanismo posee una implementación rápida y consume un mínimo de memoria tanto de programa como de datos, sin embargo posee el inconveniente de presentar un pequeño error en cada iteración, ya que es un algoritmo iterativo, y este error se integra. Aunque existen variantes que afrontan y corrigen este error, la característica adicional de solo generar valores sinusoidales consecutivos o en forma secuencial complican el diseño del proyecto. Finalmente el tercer método consiste en almacenar una tabla de valores correspondientes a una onda sinusoidal. Esta tabla es diezmada o interpolada (usualmente de forma lineal) para obtener una resolución diferente a la presentada por la tabla en sí. La principal ventaja de este método es su rapidez, en especial si se obvia la interpolación de la tabla. Esto puede ser realizado ajustando la cantidad de coeficientes de forma tal que coincida con la frecuencia de salida deseada según la frecuencia de muestreo seleccionada. Aunque este es el mecanismo más rápido de generación de una onda sinusoidal, usualmente requiere de una considerable cantidad de memoria ROM en caso de que se desee una buena resolución. Este mecanismo permite el cálculo aleatorio de cualquier valor sinusoidal.

La generación de las tres fases con sus respectivos armónicos es realizada por medio de una tabla de 400 muestras de una onda sinusoidal. Sin embargo, estas 400 muestras son generadas con una tabla de 100 muestras y una función que se encarga de generar las 400 en base a estas 100. La rutina utilizada para realizar este proceso se observa en el código de programa 1. Esta rutina se encuentra almacenada en la librería *sintable.h*.

```
uint32 sint(uint16 *muestra)
{
    if(*muestra>sinSize-1)
        *muestra -= sinSize;    // hace que sea periódica, le resta a
                                // 401, 400 para que de 1

    // *muestra = (*muestra) % sinSize;

    if(*muestra<(sinSize + 1))    // primer cuadrante (0 - 90 grados)
        return(sint[*muestra]);
    if(*muestra<(2*sinSize + 1)) // segundo cuadrante (90 - 180
    grados)
        return(sint[(2*sinSize) - (*muestra)]);
    if(*muestra<(3*sinSize + 1))
        return(-sint[*muestra] - (2*sinSize));
    return(-sint[(sinSize) - (*muestra)]);
}
```

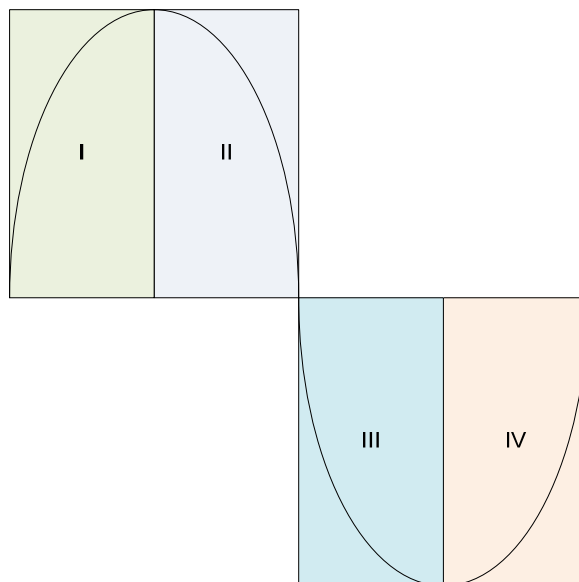
Código de programa 1: Rutina de transformación de una tabla de 100 muestras en una de 400 muestras.

Como se observa en el código de programa 1, el procedimiento consiste en determinar en qué cuadrante se encuentra el ángulo de interés y posteriormente se realiza un reflejo tanto en magnitud como en “tiempo” de la tabla de 100 muestras de forma tal que el resultado sea equivalente.

En la figura 57 se observa una onda sinusoidal dividida en cuatro cuadrantes. Nótese que el cuadrante II es una versión reflejada del cuadrante I, y por ende equivalente a $180^\circ - \theta$. De igual forma el cuadrante III es similar al I, simplemente con valores negativos, y el cuadrante IV es equivalente al cuadrante II, pero con valores negativos. La rutina `sinf` se implemento para que tuviera esto en cuenta y permite solo almacenar el cuadrante I equivalente a los primeros 90° .

Los valores almacenados de la onda sinusoidal se representan en formato *fixed*. En este formato se utilizan números de punto fijo, simplemente que los pesos del mismo son similares a los que se muestran en la tabla 15. Para este caso, la resolución utilizada para representar los valores sinusoidales fue de 24 bits.

Figura 57: Onda sinusoidal por cuadrante



Fuente: autores del proyecto

Tabla 15. Valores equivalentes de un número con representación binaria y fixed para el caso de 8 bits – se asume una representación sin signo.

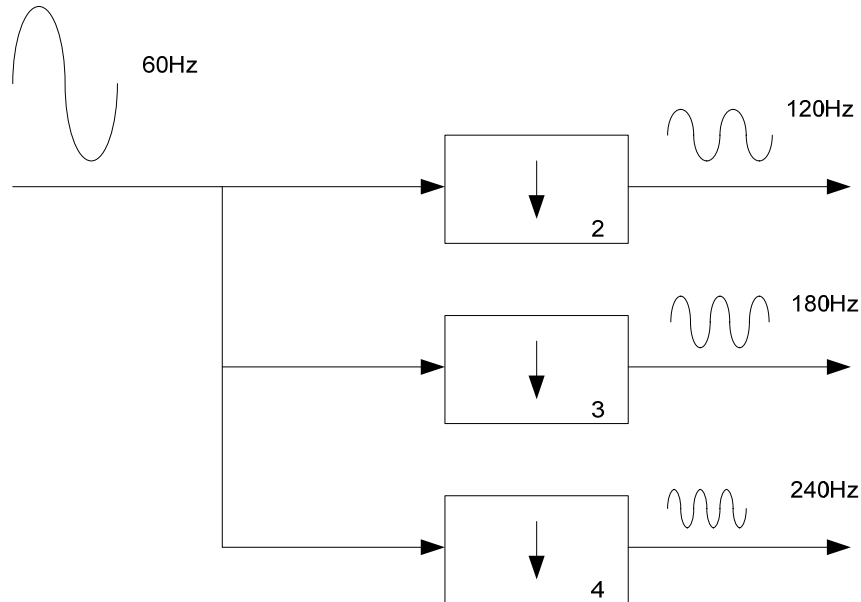
Bit	Binario	Fixed
7	128	0.5
6	64	0.25
5	32	0.125
4	16	0.0625
3	8	0.03125
2	4	0.015625
1	2	0.0078125
0	1	0.00390625

Fuente: autores del proyecto

Una vez se tiene la onda sinusoidal base de 60Hz, lo cual implicaría que una frecuencia de muestreo de 6kHz tendría una resolución de 100 muestras por ciclo, se procede a generar los armónicos pares e impares de la señal. Para generar dichos armónicos se tiene que para una misma frecuencia de muestreo o de refresco en este caso, realizar un diezmado de 2 nos generará una señal de 120Hz a la salida. De igual forma un diezmado de 3 genera una señal de 180Hz a la salida, y así sucesivamente. En la figura 58 se muestra una representación de esto.

La generación final de la señal y sus armónicos es realizada por medio de la suma de la señal fundamental y cada uno de sus armónicos, escalados a la magnitud determinada por el usuario. Es de observarse de igual forma que debido a la multiplicidad existente, la onda resultante en la duración de tiempo de un periodo de la onda fundamental, es periódica en este mismo valor de tiempo, por lo que solo es necesario generar 400 muestras, en el caso en particular de la implementación realizada, para poder representar la onda fundamental y todos sus armónicos. Este efecto se observa visualmente en las figuras 59-61.

Figura 58. Diezmado de la onda base para generar los armónicos de la señal



Fuente: autores del proyecto

Para implementar las fases con sus distintos armónicos, el dispositivo configura inicialmente los parámetros de cada fase y posteriormente realiza la generación matemática de estos valores. La generación de la onda de cada fase se realiza por medio del envío a los DACs de los valores almacenados en tres vectores de 400 posiciones representantes de cada una de las fases. La rutina matemática de generación calcula la onda correspondiente y la almacena en cada uno de estos vectores. Los campos que se utilizan en la fórmula de generación de la onda se relacionan en la tabla 16. Nótese que cada registro es indiferente de si es la fundamental o un armónico. Esto se determina por medio del parámetro *freq*, que para el caso de la fundamental sería 1.

Tabla 16: Registros asociados a la estructura de la onda y sus armónicos para cada fase.

Registro/Campo	Descripción
faseAsoc	Determina a que fase corresponde el registro, ya sea A,

	B o C
amplitud	Amplitud del correspondiente armónico. Puede ser un valor entre 0x00 y 0x10
fase	Valor de ángulo de la fase. Este es el valor absoluto, por lo que cada vez que se especifique una fase asociada, y un ángulo de desfase en relación a esta fase asociada, el ángulo de fase absoluto y general es calculado y almacenado en este campo.
freq	Corresponde a la frecuencia (múltiplo de 60Hz) o armónico de la señal. Se utiliza particularmente como factor de diezmado para generar las diferentes frecuencias.

La estructura computacional en C utilizada se relaciona en el código de programa 2. Aquí se observa la definición de la estructura, y la declaración de la misma en memoria. La constante *NUMTOTALARMON* corresponde a 50. De igual forma se observan los vectores “separados” para almacenar las ondas calculadas correspondientes a cada fase. El tamaño de cada uno es de 32 bits como un vector de 400 posiciones.

```
//      datos hacia el dac
uint32 datosFaseA[sinSize];
uint32 datosFaseB[sinSize];
uint32 datosFaseC[sinSize];
uint32 datosFase0[sinSize];
```

```
struct caracteristicasOnda
{
    uint16 fase;
    uint8 amplitud;
    uint8 freq;
```

```

        uint8 faseAsoc;
};
struct característicasOnda fases[NUMTOTALARMON];

```

Código de programa 2: Estructura y declaración de las fases.

El código de programa 3 muestra la función utilizada para generar matemáticamente los armónicos y almacenarlos en los vectores de salida. Como se puede observar se calcula para la cantidad de fases “activadas” la onda por medio de la suma individual de cada uno de los armónicos especificados para la duración de la onda fundamental, en este caso, 400 muestras por el periodo de muestreo o refresco. Nótese que se realiza un escalamiento para evitar el desbordamiento de la variable. El primer segmento del código se encarga de determinar de forma absoluta el ángulo de desfase para cada armónico.

```

//    fases iniciales para los armonicos
for(i=3;i<=NUMTOTALARMON;i++)
    fases[i].fase = (fases[fases[i].faseAsoc/* freq fundamental 0..3 fases a,b
o c*/].fase*fases[i].freq) % sinSize;           //    0
//    Genera un ciclo
for(i=0;i<sinSize;i++)
{
    datosFaseA[i] = 0;
    datosFaseB[i] = 0;
    datosFaseC[i] = 0;

//    genera cada uno de los armonicos
for(j=0;j<totalNumArmon;j++)
{
    if(fases[j].faseAsoc == FASEA)

```

```

        datosFaseA[i] += (fases[j].amplitud*sinf(&fases[j].fase)) >>
4;

        if(fases[j].faseAsoc == FASEB)
            datosFaseB[i] += (fases[j].amplitud*sinf(&fases[j].fase)) >>
4;

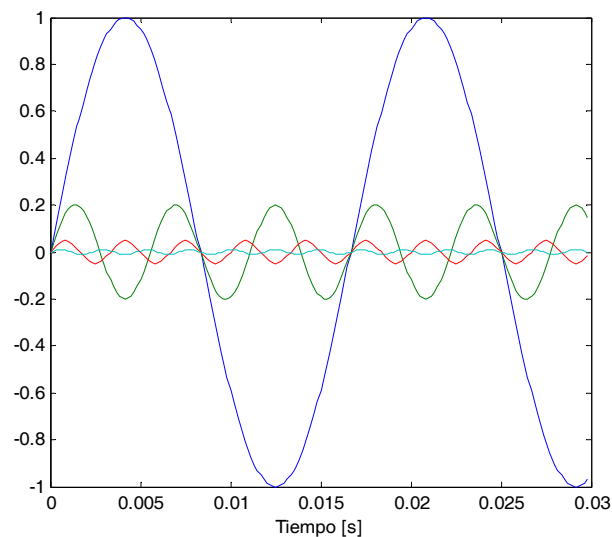
        if(fases[j].faseAsoc == FASEC)
            datosFaseC[i] += (fases[j].amplitud*sinf(&fases[j].fase)) >>
4;

        fases[j].fase += fases[j].freq;
    }
}

```

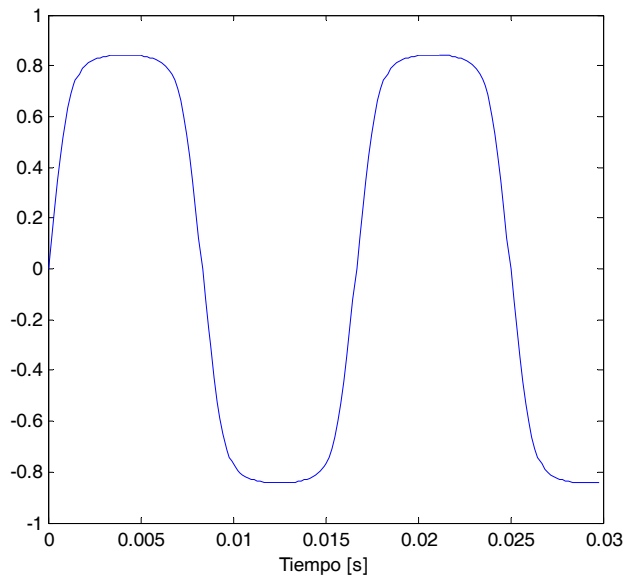
Código de programa 3: Cálculo de las formas de onda de cada fase.

Figura 59: Ondas sinusoidales correspondientes a los armónicos de 60Hz, 1, 3, 5 y 7, con el 100%, 20%, 5% y 1% de magnitud respectivamente



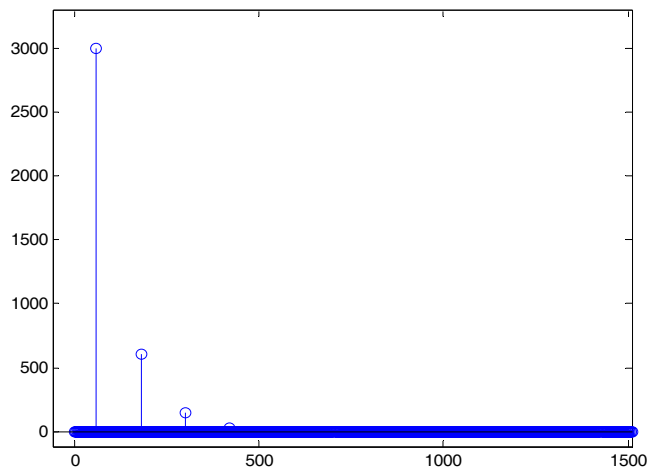
Fuente: autores del proyecto

Figura 60: Suma de las ondas anteriores



Fuente: autores del proyecto

Figura 61: Espectro de la señal generada



Fuente: autores del proyecto.

4.2. MENU DE INTERFAZ DE USUARIO

La interfaz de usuario del dispositivo se realiza por medio de la pantalla gráfica. Se tienen varios “pantallazos” que son dibujados gracias a funciones gráficas

incluidas en las librerías de dibujo. La interface es realizada por medio de mensajes de texto, valores numéricos, botones y gráficos de las ondas.

Además de esto, se utiliza un cursor de texto para indicar los campos bajo edición. Para controlar este proceso e ingresar los valores se utiliza un teclado alfanumérico que se configura como ingreso de valores, dos teclas de desplazamiento adelante-abajo o anterior-siguiente (teclas C y D) para navegar a través de la forma mostrada en el pantallazo y finalmente una tecla que hace las veces de *enter* (tecla *), utilizada para finalizar la edición de un campo numérico o para presionar uno de los botones que desencadenan una acción en la interfaz del usuario.

Las funciones de dibujo fueron implementadas en dos librerías: *lcd.h* y *lcd2.h*. La primera contiene todas las funciones de bajo nivel requeridas para poderse comunicar con el LCD. La segunda contiene las funciones de alto nivel requeridas para dibujar bitmaps, pixels, líneas, círculos, rectángulos, etc. Además, esta última librería también contiene el código necesario para dibujar botones, incluyendo el efecto de presionarlos. Funciones para escribir texto también están incluidas en *lcd2.h*. En la tabla 16 se relacionan las funciones contenidas en *lcd2.h*

La navegación a través de cada uno de los campos se describe en los siguientes numerales, y en general es resumida en la figura 62. Para cada uno de los menús, se determina en cada instante de refresco si es necesario dibujar la forma, usualmente solo es realizado cuando se cambia de una forma a la otra.

Tabla 17: Funciones de alto nivel de dibujo en *lcd2.h*

Función	Descripción
inicializarLCD	Inicializa el LCD en el modo gráfico, limpia la pantalla y las páginas de

	memoria gráficas y de texto
limpiarLCD	Limpia la pantalla
escribirChar	Escribe un carácter en la línea y columna deseadas
escribirString	Escribe una cadena de caracteres en la línea y columna deseados
dibujarPixel	Dibuja un único pixel en pantalla en las coordenadas especificadas por el usuario
invertirPixel	Genera un NOT o invierte el color del pixel en la pantalla
dibujarRectángulo	Dibuja un rectángulo en las coordenadas especificadas
dibujarCírculo	Dibuja un círculo en las coordenadas especificadas
dibujarBitmap	Dibuja un icono es las coordenadas especificadas. El icono está almacenado como una variable de 8bits, y se debe de indicar la dirección en donde reside en memoria, así como su tamaño.
seleccionarBoton	Rellena el dibujo de un botón para sugerir que este fue presionado
dibujarBoton	Dibuja un botón como un conjunto de rectángulos con un mensaje
posicionarCursor	Localiza el cursor de texto en la línea y columna deseada
seleccionarEdit	Selecciona el campo de edición (mostrado por medio de < >)
ingresarEdit	Escribe el valor determinado por el usuario en el campo de edición
escribirFVal	Escribe un valor en formato decimal

Fuente: autores del proyecto

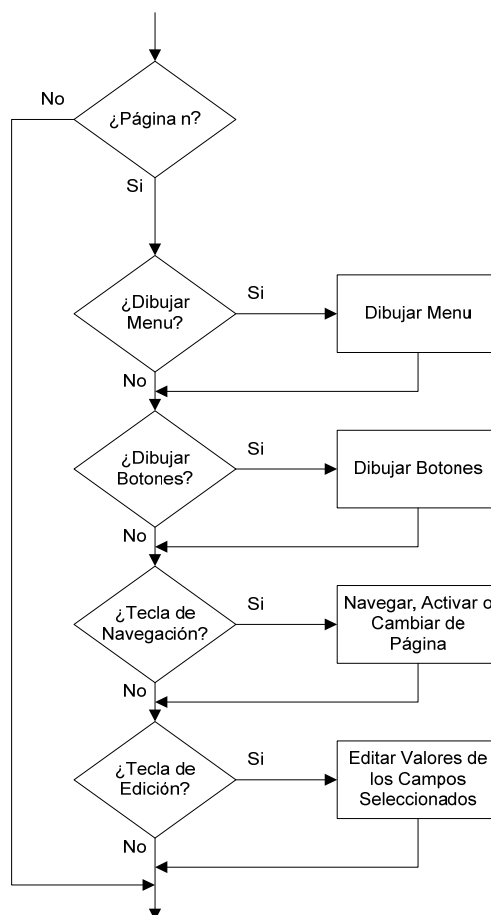
De igual manera se determina si es necesario dibujar los botones, ya sea como presionado o para des-presionarlo. Dependiendo si se detecto la presencia de una tecla presionada o de un comando remoto, se procede a determinar si es necesario cambiar de forma, editar un campo numérico, modificar el parámetro o navegar en la forma. Finalmente se modifican numéricamente los parámetros si se modifico exitosamente alguno. Nótese que cada uno de estos estados es ejecutado indiferentemente de si el anterior requirió de tiempo de cómputo o no.

4.2.1. MENÚ DE PARÁMETROS PRINCIPALES

Esta forma (menú) se encarga de configurar los parámetros asociados con cada una de las fases. De esta forma es posible determinar el ángulo de desfase en relación a la fase A para las fases B y C, en caso de que se desee

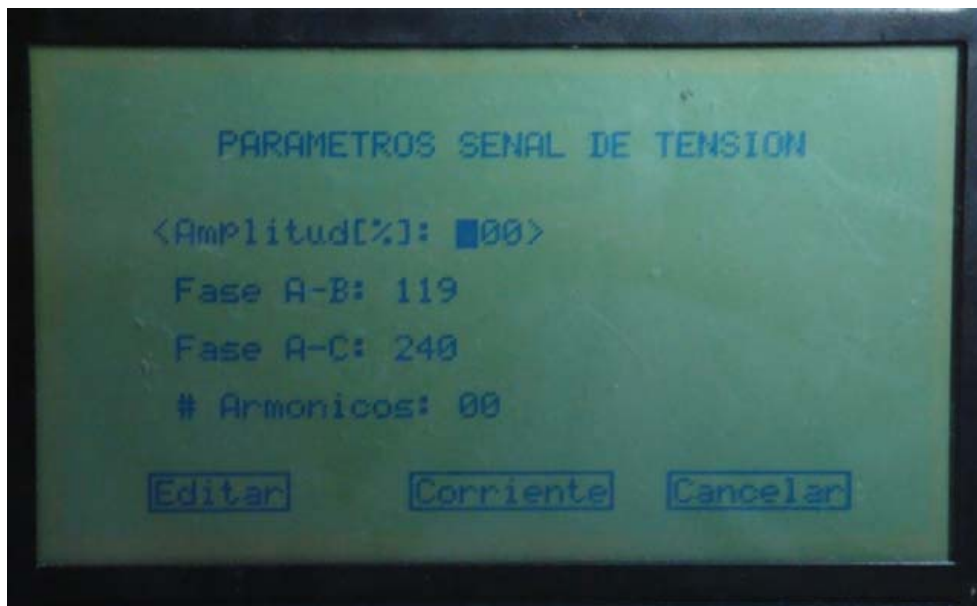
un valor distinto de 120° y 240°. De igual forma se especifican la cantidad de armónicos y el valor de la amplitud de las fundamentales. La figura 63 muestra un pantallazo de esta forma. Esta forma posee los botones de editar, generar y cancelar. El botón de editar es utilizado para especificar los armónicos y por ende ir a la forma o menú de los armónicos. El botón de generar procede a graficar y generar por medio del DAC la configuración de onda para cada fase que fue especificada. Finalmente el botón de cancelar se encarga de cancelar la generación actual de ya sea una onda de tensión o una onda de corriente y regresa al menú anterior encargado de seleccionar esto.

Figura 62: Diagrama de flujo de la ejecución de un estado



Fuente: autores del proyecto

Figura 63. Pantallazo de configuración de los parámetros principales.



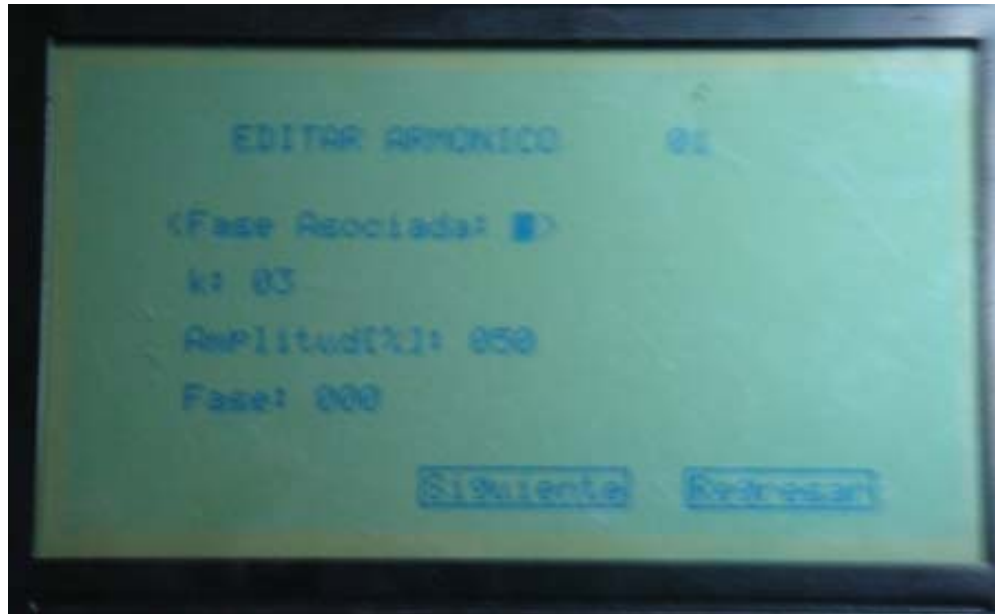
Fuente: Autores del proyecto

4.2.2. MENÚ DE PARÁMETROS DE LOS ARMÓNICOS

Esta forma es utilizada para especificar las características de los armónicos. Es independiente de la fase, por lo que indicar a cual fase se asocia el armónico es necesario, y es el último campo que debe ser indicado. Además de esto, se debe indicar el múltiplo, es decir, si es el armónico 2, 3, 4, etc. Además de esto, la amplitud que este posee. Es posible de igual forma especificar un ángulo de fase para dicho armónico, teniéndose en cuenta que esta fase es relativa a el ángulo de fase que ya posee la fase editada en relación a la fase A.

Las posibles acciones además de configurar los parámetros son configurar otro armónico, acción que se realiza seleccionando “siguiente”, o finalizar dicho proceso por medio de la opción de regresar. Nótese que sin importar cuantos armónicos hayan sido editados, estos se habilitan en la forma de configuración principal por medio de la especificación de armónicos a generar. Los valores entregados se almacenan y pueden ser editados posteriormente. En la figura 64 se puede apreciar gráficamente este menú.

Figura 64: Pantallazo de configuración de armónicos.

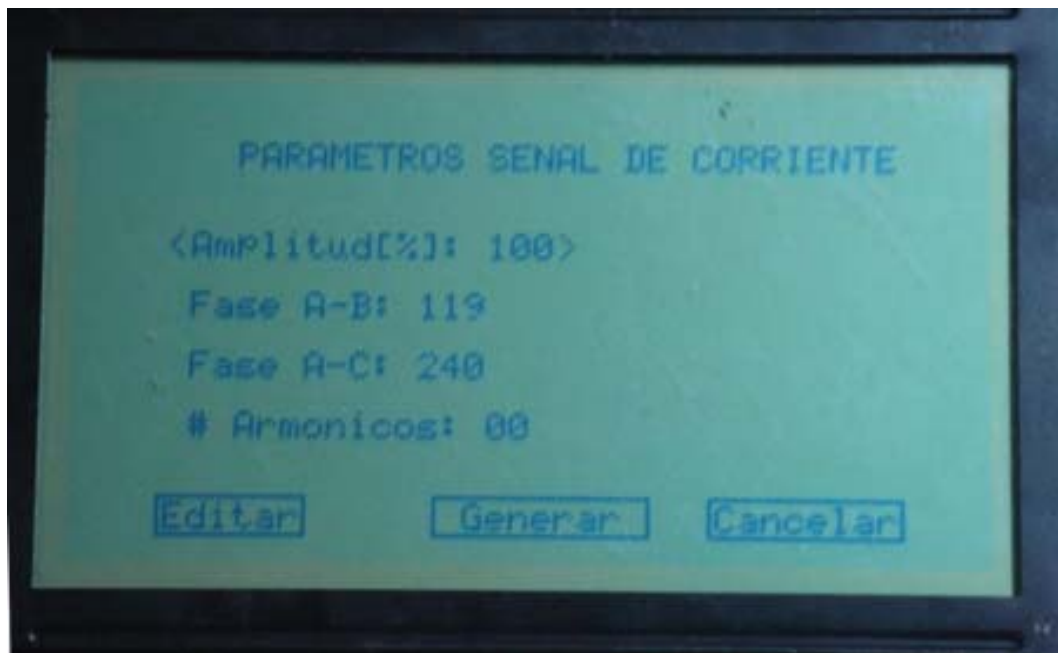


Fuente: autores del proyecto

4.2.3. MENÚ DE SELECCIÓN PARA LA EDICION DEL TIPO DE SALIDA

Esta forma es sencilla, y se encarga de determinar si la onda a editar y generar corresponde a las salidas de corriente o a las de tensión. Este menú solo se encarga de modificar la variable global *modoTensionCorriente*, encargada de especificar el modo de trabajo. Esto se puede apreciar en la figura 65, ahí se está en la edición de parámetros de tensión pero está el botón de selección de corriente.

Figura 65: Pantallazo de selección de parámetros de corriente

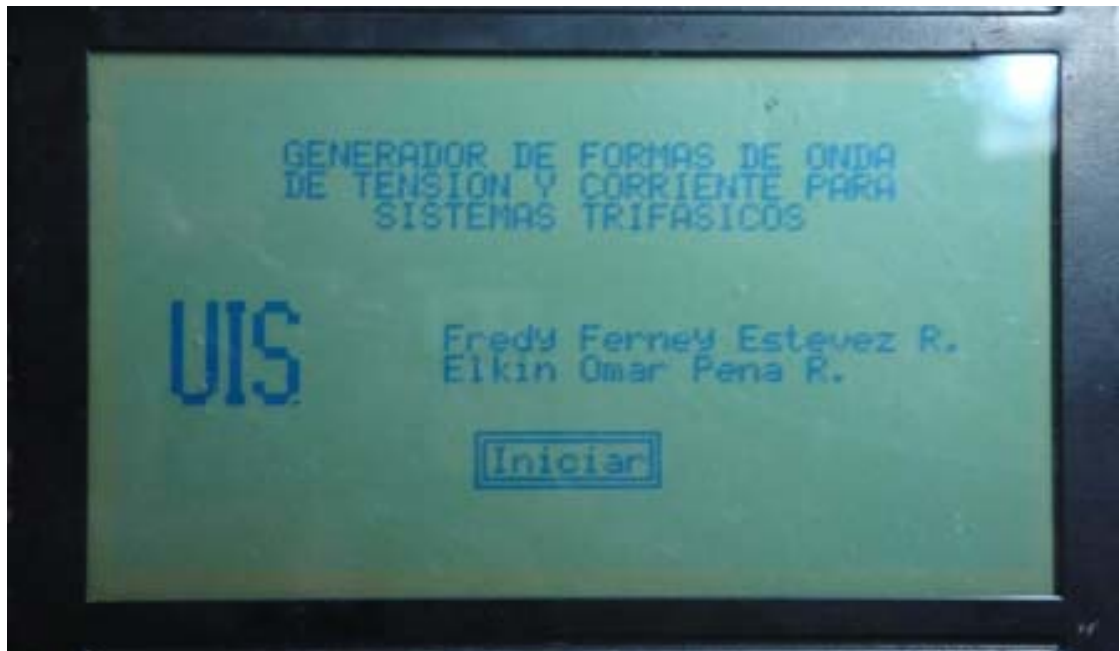


Fuente: autores del proyecto

Splash

Este es el primer pantallazo y muestra el logo de la UIS, el título de la aplicación y sus autores. Hace uso de la función *dibujarBitmap* para dibujar el logo de la UIS almacenado en memoria ROM. Este pantallazo solo es mostrado una vez cada vez que se energiza el dispositivo y es mostrado en la figura 66.

Figura 66: Pantallazo inicial.

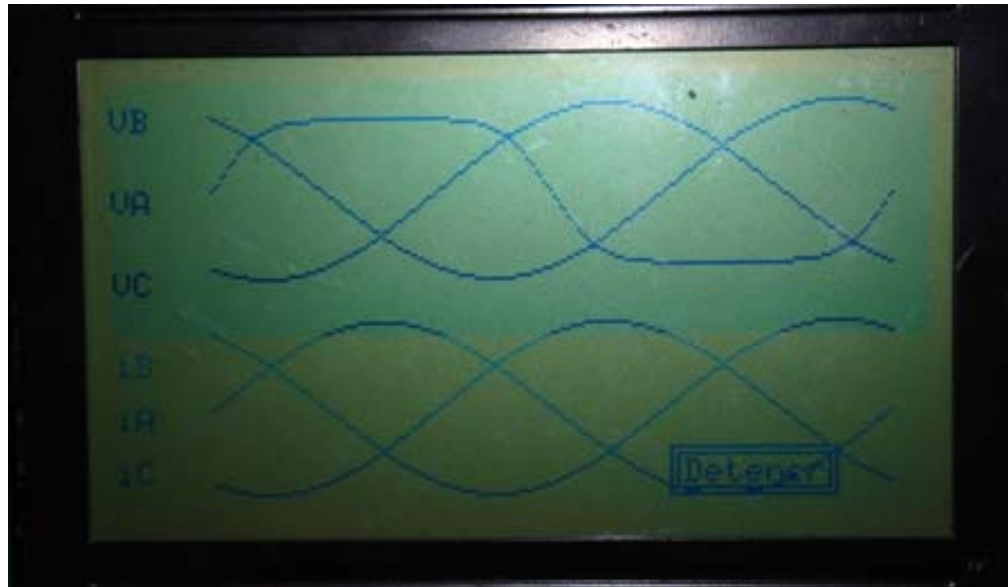


Fuente: autores del proyecto

Generación de la Onda

En este “pantallazo”, el dispositivo despliega visualmente seis señales tal y como serán generadas por medio de la tarjeta de DSP y el DAC por medio de una señal de voltaje y corriente. La figura 67 muestra una foto del pantallazo como es visto en el LCD.

Figura 67: Pantallazo de las fases dibujadas una vez se invoca el comando de generar.



Fuente: autores del proyecto.

4.3. ESTRUCTURA DE FUNCIONAMIENTO INTERNO

El DSP maneja dos tareas principales: interface con el usuario y generación matemática y física de la señal. Para realizar esto de forma adecuada, la programación fue implementada por medio de varios estados y del uso de rutinas de atención a la interrupción. Las rutinas de atención a la interrupción se encargan de manejar la base de tiempos utilizados para el refresco de los estados de funcionamiento de la aplicación, así como también de la base de tiempo utilizada como frecuencia de refresco para generar la señal; de igual forma maneja los comandos remotos enviados a través del puerto serial.

Los códigos asociados con las rutinas de atención a la interrupción son cortos. El código de programa 4 muestra la rutina de atención a la interrupción asociada con la base de tiempo de la frecuencia de refresco. Como se observó en el numeral 4.1, la generación de la onda se realiza a partir de una tabla base de 100 muestras equivalentes a 90° , que es utilizada por medio de la función

sinf para generar una onda sinusoidal de 400 muestras. Los parámetros correspondientes a la fase y armónicos utiliza esta tabla para generar 400 muestras para cada una de las salidas. Del listado se observa que si la bandera *generarOnda* esta activa, el puntero a este vector de ondas de salida es incrementado progresivamente y reiniciado a la posición inicial una vez se detecta el desbordamiento. Por lo tanto, esta función es llamada cada T_s , es decir, con la frecuencia correspondiente a la frecuencia de muestreo/refresco.

```
void fs_OnInterrupt(void)
{
    tic++;

    if(generarOnda)
    {
        //      Envia datos al DAC

        datosFaseIndice++;

        if(datosFaseIndice>400-1)
            datosFaseIndice = 0;
    }
}
```

Código de programa 4: Rutina de base de tiempo para la frecuencia de refresco.

```
void refrescar_OnInterrupt(void)
{
    refrescar++;
}
```

Código de programa 5: Rutina de base de tiempo para los estados y la interface de usuario

El código de programa 5 se encarga de actualizar la bandera utilizada para refrescar los estados y cambiar los estados. Una utilidad importante es como mecanismo antirebote, ya que establece el tiempo en el cual el teclado es encuestado. De igual forma resulta útil para generar el efecto de botón presionado, ya que funciona como tiempo de espera entre el botón dibujado como presionado y el botón dibujado como no presionado, creando la ilusión de la animación. Por último también permite la implementación del tiempo de cancelación de mensaje (*timeout*) para el puerto serial, utilizado para determinar que un mensaje fue transmitido incorrectamente y por ende cancelarlo. Esto último se describirá posteriormente.

Un esquema general de funcionamiento de la rutina principal se observa en la figura 68; Aquí se observa que una vez iniciada la ejecución se procede a inicializar los parámetros de sistema, que en este caso corresponden a inicializar las tres fases a ondas sinusoidales puras, con 120° de desfase entre cada una y sin ningún armónico. El código por medio del cual se realiza esto se observa en el código de programa 6. Las estructuras observadas aquí fueron descritas anteriormente.

```
armonicoActual = 0;
```

```
    fases[armonicoActual].faseAsoc = FASEA;  
    fases[armonicoActual].amplitud = 0x10;  
    fases[armonicoActual].fase = 0; // 0  
    fases[armonicoActual].freq = 1;  
    armonicoActual++;  
  
    fases[armonicoActual].faseAsoc = FASEB;
```

```

fases[armonicoActual].amplitud = 0x10;
fases[armonicoActual].fase = sintSize + (sintSize/3);           //      120
fases[armonicoActual].freq = 1;
armonicoActual++;

fases[armonicoActual].faseAsoc = FASEC;
fases[armonicoActual].amplitud = 0x10;
fases[armonicoActual].fase = 3*sintSize - (sintSize/3);       //
240
fases[armonicoActual].freq = 1;
armonicoActual++;

totalNumArmon = 3;

//      Valores Predeterminados
for(ii=3;ii<=NUMTOTALARMON-3;ii+=3)
{
    fases[ii+0].faseAsoc = FASEA;
    fases[ii+0].amplitud = (uint8)(fases[ii+0-3].amplitud >> 1);
    fases[ii+0].freq = (uint8)(fases[ii+0-3].freq + 2);

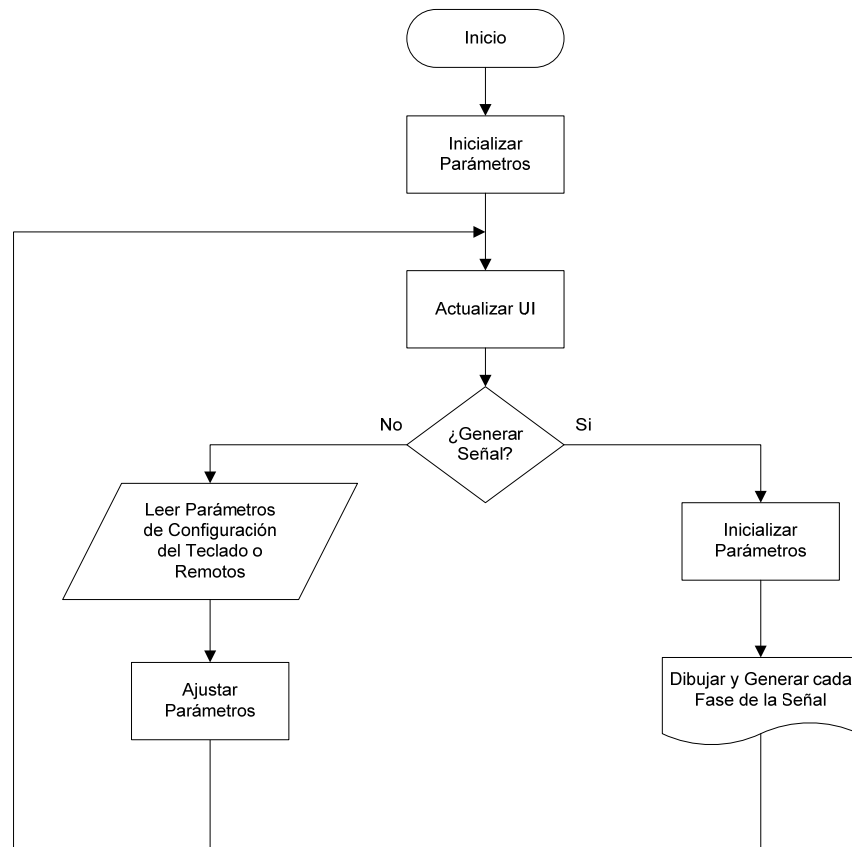
    fases[ii+1].faseAsoc = FASEB;
    fases[ii+1].amplitud = (uint8)(fases[ii+1-3].amplitud >> 1);
    fases[ii+1].freq = (uint8)(fases[ii+1-3].freq + 2);

    fases[ii+2].faseAsoc = FASEC;
    fases[ii+2].amplitud = (uint8)(fases[ii+2-3].amplitud >> 1);
    fases[ii+2].freq = (uint8)(fases[ii+2-3].freq + 2);
}

```

Código de programa 6: Código de inicialización

Figura 68: Esquema de funcionamiento del programa



Fuente: autores del proyecto

En general nótese que por más que de que se generan armónicos iniciales correspondientes a los armónicos de múltiplos impares, con magnitudes del 50%, 25%, 12.5%, etc., estos se encuentran desactivados gracias a la inicialización en 3 de la variable totalNumArmon. La rutina de inicialización ajusta la magnitud al 100% y la fase al ángulo adecuado.

Una vez realizada la inicialización de cada una de las fases, se procede a generar la interface gráfica de usuario (GUI), o en general la UI. La interface de

usuario representa el estado en el cual se encuentra en ejecución el programa, y puede ser uno de los siguientes:

- Página principal de selección entre corriente o tensión
- Página de parámetros principales de corriente o tensión
- Página de especificación de armónicos de corriente o tensión
- Generación de las diferentes fases

Cada uno de estos estados tiene asociado un menú, que es dibujado por medio de los mecanismos sugeridos anteriormente. El estado correspondiente a la generación realiza un dibujo de la señal con sus tres fases y sus respectivos armónicos tal como serán generados; y finalmente genera la señal de salida. El funcionamiento de cada una de las páginas fue indicado en el numeral anterior.

4.4. APLICACIÓN MATLAB

Se realizó una aplicación en matlab amigable que simula el funcionamiento del dispositivo y además permite la manipulación del mismo para la edición de los armónicos.

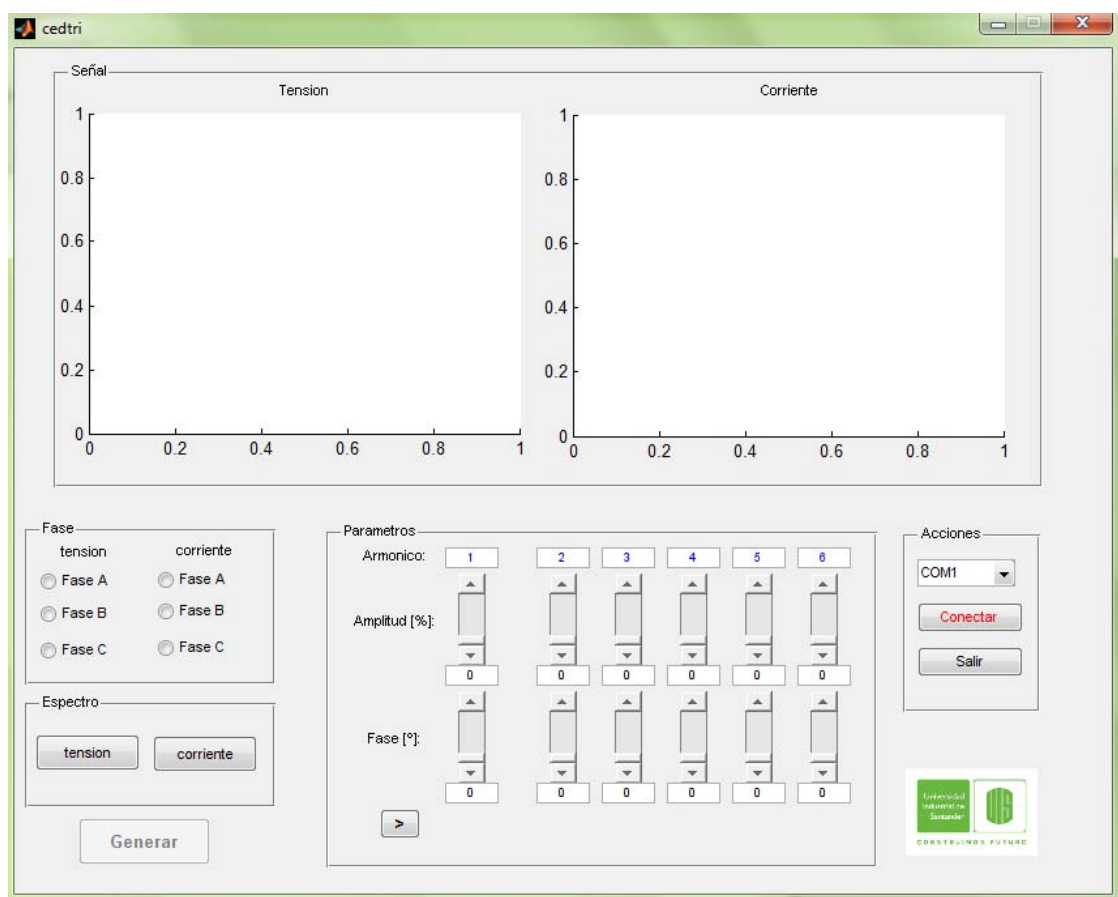
Se crearon botones, barras de selección, ejes para gráficos que conforman la interfaz grafica creada, se encuentra dividida en cinco partes que pueden ser vistas en la figura 69.

ACCIONES: En esta parte de la aplicación se encuentran la selección del puerto en el cual esta conectado el dispositivo, un botón de conectar o desconectar según sea el caso y de salir para cerrar completamente la aplicación.

FASE: Podemos observar dos columnas una para tensión y otra para corriente cada una con la opción de seleccionar la fase a, b o c, se puede seleccionar solo una fase a la vez y a la cual se desea editar los diferentes parámetros.

PARAMETROS: en similitud a los que podemos editar con el teclado en el dispositivo, aca encontramos barras que nos facilitan esta edición. Podemos modificar la amplitud y fase de la señal fundamental, además de la amplitud y fase de cada armónico. Se observan inicialmente los 5 primeros armónicos y usando la flecha ubicada en la parte inferior izquierda avanzamos las “paginas” hasta localizar el armonico deseado.

Figura 69: Aplicación creada en matlab



Fuente: autores del proyecto

SEÑAL: la aplicación así como el dispositivo están en capacidad de generar las tres señales de tensión y tres corrientes de línea para ver su representación grafica se crearon dos ejes donde observamos las señales generadas ya sean de tensión o corriente. Además se podrá observar en los mismos ejes el espectro de la señal según sea lo escogido en el modulo de espectro.

ESPECTRO: Acá encontramos dos botones llamados tensión o corriente que al oprimir una vez se hayan generado las señales invocaran una rutina aeu dibujara los espectros de las señales en los ejes creados en señal.

GENERAR: Este botón se encarga de llamar la rutina creada para la generación de las señales con los distintos armónicos que el usuario haya escogido modificar para posteriormente ser dibujada en los ejes creados en el modulo de señal.

5. PRUEBAS Y RESULTADOS

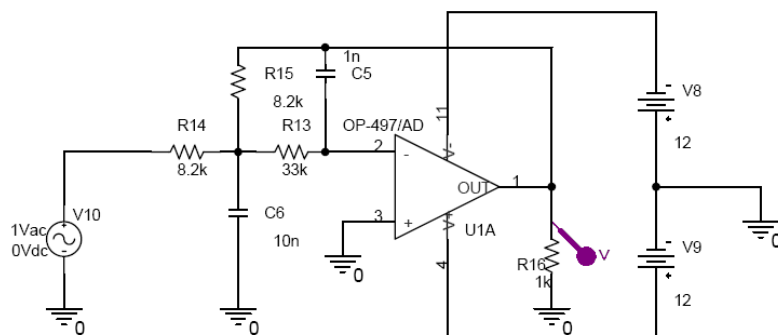
Como parte final del proceso que conlleva el diseño e implementación de un equipo están las pruebas que certifican el verdadero funcionamiento del equipo, y entregan una medida de los alcances obtenidos con respecto a los objetivos propuestos al inicio del proyecto y dan una visión de la utilidad que brindará el equipo en diferentes campos.

Las pruebas hechas al generador de formas de ondas de tensión y corriente para sistemas trifásicos se describen en este capítulo, además de los resultados obtenidos. Antes de describir las pruebas realizadas al equipo completo se detallarán algunas simulaciones y pruebas hechas a la etapa de acondicionamiento de la señal.

Etapa de acondicionamiento de la señal de salida

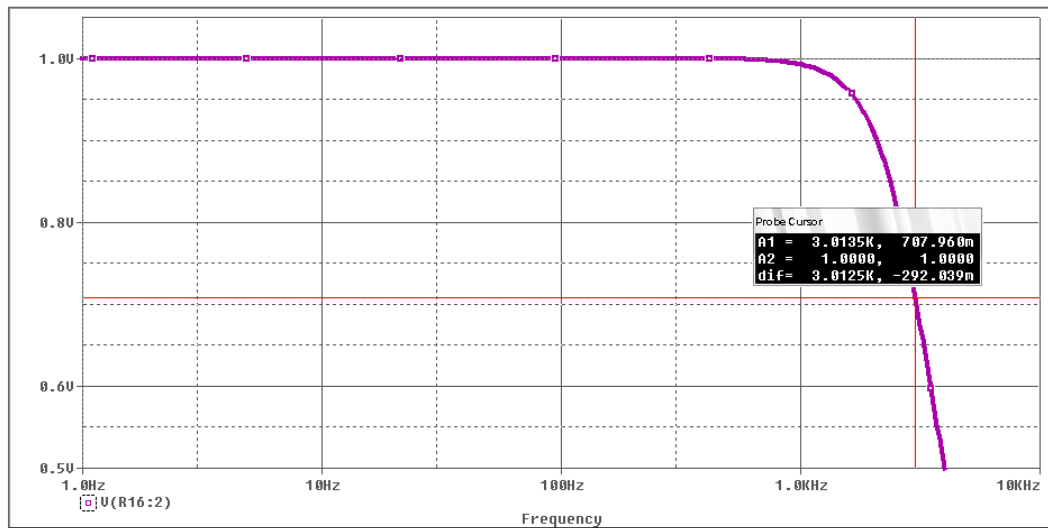
Primero se realizó una simulación en orcad del filtro Butterworth, utilizando el modelo del amplificador LME4970 brindado por el fabricante, con los valores de resistencias y condensadores implementados en la tarjeta final, a continuación se muestran las gráficas del circuito simulado (figura 70) y el diagrama de Bode dado por orcad (figura 71).

Figura 70: Filtro Butterworth simulado



Fuente: autores del proyecto

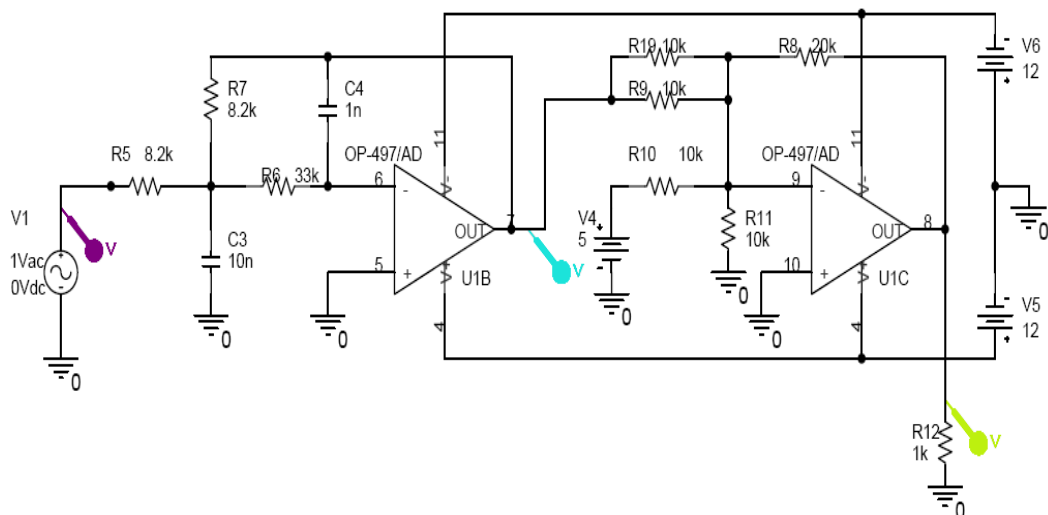
Figura 71: Diagrama de bode filtro Butterworth



Fuente: autores del proyecto.

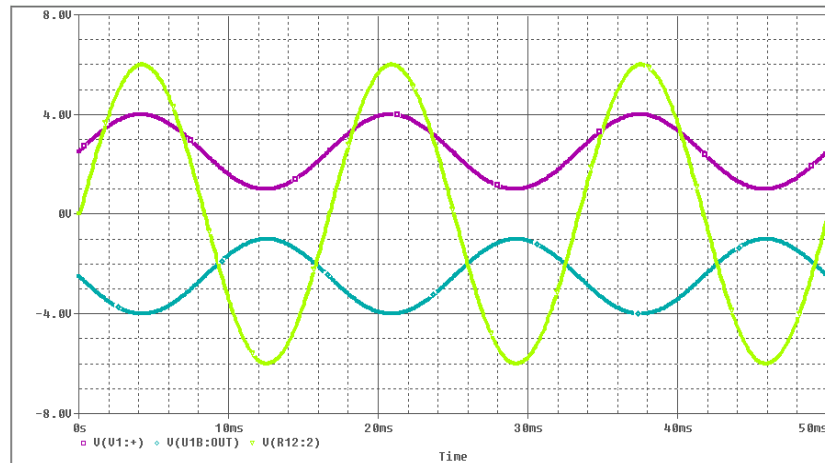
La siguiente simulación que se realizó fue la del circuito de acondicionamiento de señal completo, es decir al filtro Butterworth seguido del sumador inversor con ganancia 4, en la grafica 72 muestra el esquema del circuito completo.

Figura 72: Circuito de acondicionamiento de señal



Fuente: Los autores del proyecto.

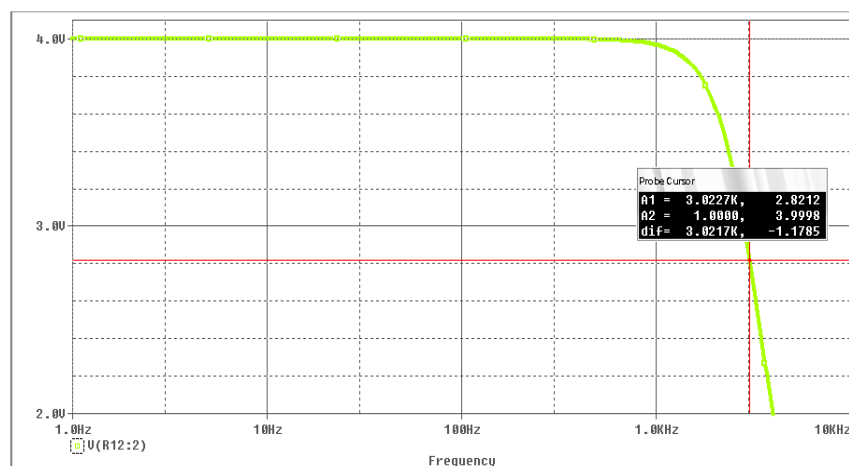
Figura 73: señales tomadas en diferentes puntos del circuito simulado.



Fuente: Los autores del proyecto.

La señal de color morado corresponde a la señal de entrada con una amplitud de 3Vp-p y centrada en 2.5V, esta señal corresponde a la reconstruida por el DAC8568, la señal de color azul es la salida del filtro pasabajos, por su característica inversora el filtro invierte la señal obteniéndose una señal de 3Vp-p centrada en -2.5V y por ultimo esta la señal verde que es la señal resultante con una amplitud de 12Vp-p, producto de la ganancia de 4 con la que fue diseñado el sumador inversor con ganancia, además de esto la componente de continua es eliminada y la señal es nuevamente invertida.

Figura 74: Diagrama de Bode del circuito completo.

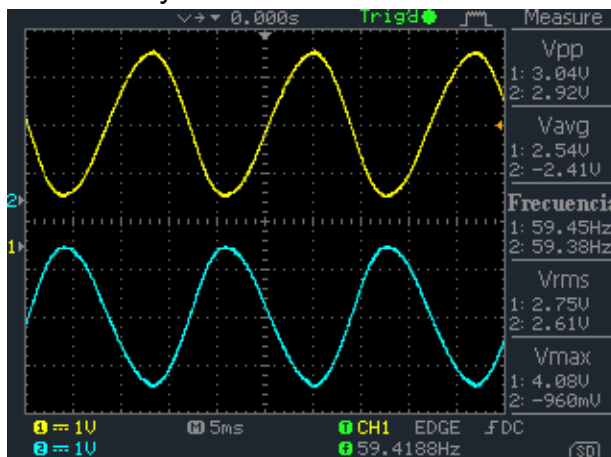


Fuente: Los autores del proyecto

En la figura 74 se muestra el diagrama de Bode del circuito completo, se observa claramente una ganancia de 4[V/V] y una frecuencia de corte de 3.0227 KHz.

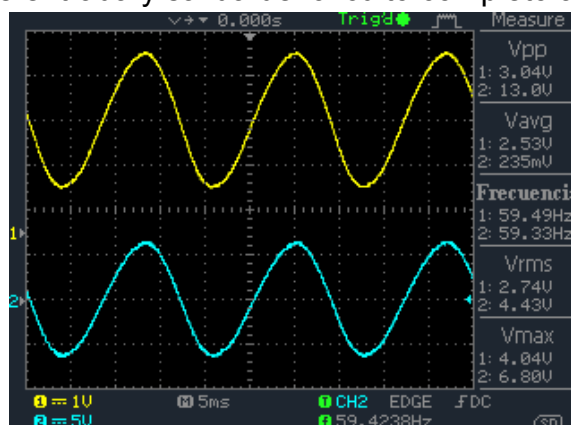
Para contrastar los resultados obtenidos en las simulaciones se hicieron pruebas en el laboratorio al circuito de acondicionamiento de señal implementado en la tarjeta final. En la figura 75 se observa la señal de entrada del circuito (amarilla) y la señal de salida del filtro (azul). La figura 76 muestra la señal de entrada y la señal de salida del circuito completo.

Figura 75: Señal de entrada y salida del filtro Butterworth.



Fuente: Los autores del proyecto.

Figura 76: Señal de entrada y salida del circuito completo a 60 Hz.



Fuente: los autores del proyecto.

Como se observa en las figuras 75 y 76 las señales son similares a las obtenidas en las simulaciones, en la tabla 18 se muestran los datos simulados y obtenidos en el laboratorio.

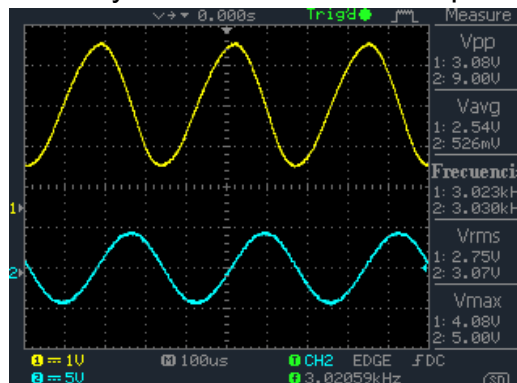
Tabla 18: Datos obtenidos de la simulacion y de las pruebas de laboratorio.

	Datos	Amplitud (Vp-p)	Frecuencia (Hz)	Nivel de DC (V)	Ganancia (V/V)
Señal de entrada	simulado	3	60	2.5	//
	prácticos	3.04	59.45	2.54	//
Salida del filtro	simulado	3	60	-2.5	1
	prácticos	2.92	59.38	-2.41	1.041
Señal de salida	simulado	12	60	0	4
	prácticos	13	59.33	0.235	4.276

Fuente: los autores del proyecto.

Para comprobar que el circuito implementado efectivamente tiene una frecuencia de corte de 3kHz la prueba que se realizó fue inyectar la misma señal de 3Vp-p que se utilizó para la anteriores pruebas pero aumentando la frecuencia a 3kHz y observar si a la salida la señal se reduce en un factor de $1/\sqrt{2}$. La figura 77 muestra los resultados obtenidos.

Figura 77: Señal de entrada y salida del circuito completo a 3kHz.



Fuente: los autores del proyecto.

En la tabla 19 se muestran los resultados de esta prueba, de esta tabla se observa que los datos obtenidos en la simulación son muy similares a los obtenidos en el laboratorio.

Tabla 19: Datos obtenidos de la simulación y de pruebas del laboratorio a 3khz.

		Amplitud (V)	Frecuencia (kHz)
Señal de entrada	Simulado	3	3
	Prácticos	3.08	3,023
Señal de salida	Simulado	8.485	3
	Prácticos	9	3,03

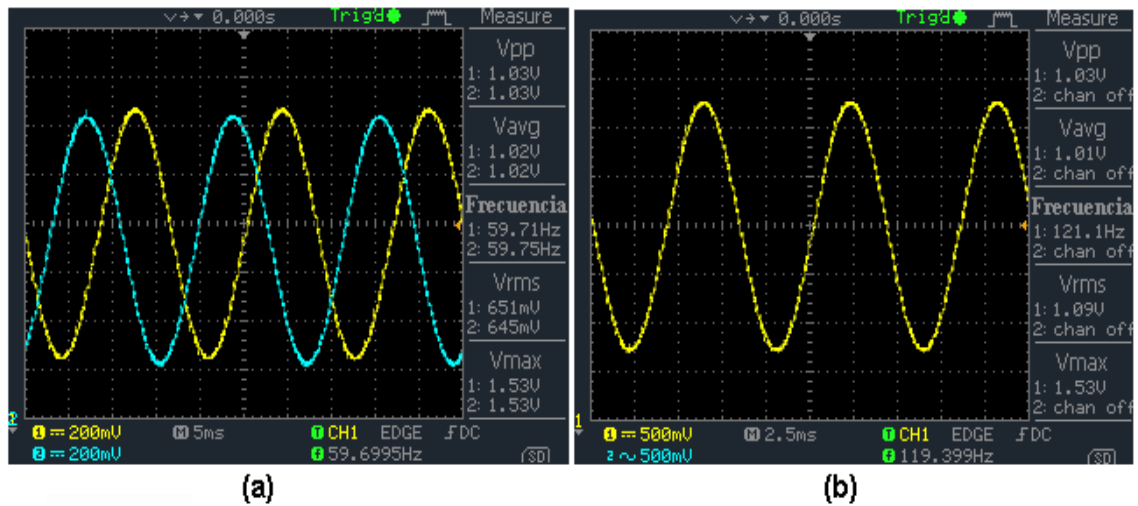
Fuente: Los autores del proyecto.

PRUEBAS REALIZADAS CON EL GENERADOR SE SEÑALES

A continuación se muestran los resultados obtenidos generando señales puras y contaminadas con armónicos, las pruebas se realizaron utilizando las tarjetas hechas con el DAC-MCP4821 y con el DAC8568.

Primero utilizando la tarjeta con el DAC MCP4821 se generaron una señales senoidales puras de 60Hz y 120 Hz como se muestra en la figura 78, con una amplitud pico-pico de 1V y centrada en 1 V, esta tarjeta no cuenta con la etapa de acondicionamiento de señal.

Figura 78: Ondas senoidales reconstruidas por el DAC MCP4821, (a) a 60Hz, (b) a 120Hz.



Fuente. Los autores del proyecto

En la grafica 79 se observa una onda generada con tres armónicos con los datos mostrados en la tabla 20.

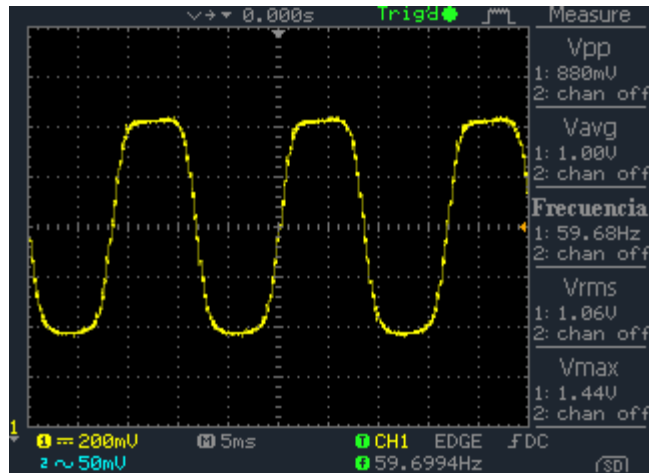
Tabla 20: Armónicos, magnitud y fase de generada

# de armónico	Magnitud [%]	Fase (0°-360°)
3	20	0
5	5	0
7	1	0

Fuente: Los autores del proyecto.

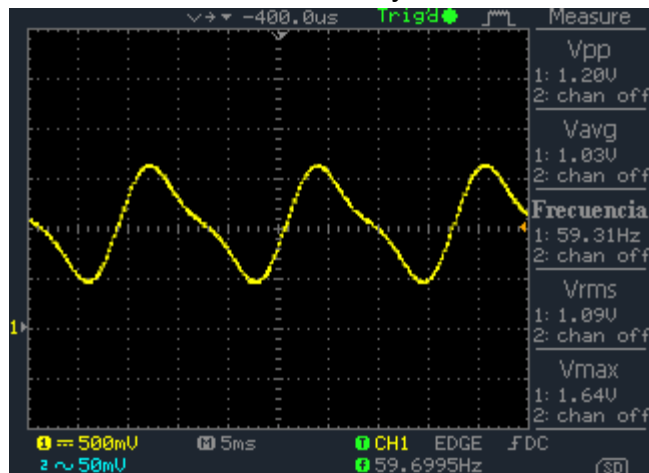
En la figura 80 se muestra una señal generada con el armónico 2 con una magnitud del 30% y una fase de cero grados.

Figura 79: Señal contaminada con armónicos



Fuente: Los autores

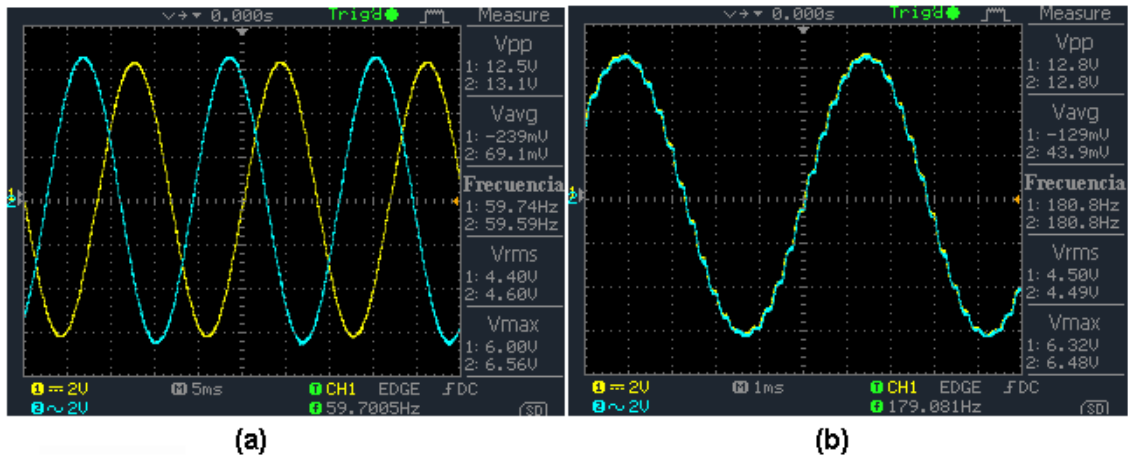
Figura 80: Señal con el armónico 2 al 30 % y fase 230°.



Fuente: Los autores

Las siguientes pruebas fueron hechas utilizando la tarjeta DAC8568, primero se generaron señales senoidales puras de 60 Hz y de 180 Hz como las que se muestran en la figura 81.

Figura 81: (a) señales de 60 HZ, (b) señales de 180 HZ



Fuente: los autores.

La corriente absorbida por una lámpara fluorescente equipada con reactancias electrónicas contiene los armónicos descritos en la tabla 21 y en la figura 82 se muestra su forma de onda.

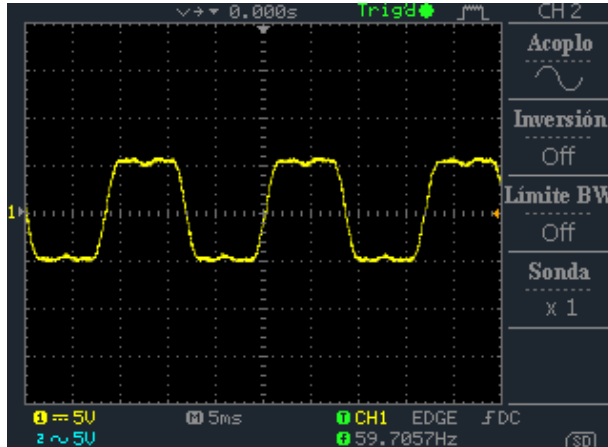
Tabla 21: Armónicos contenidos en la corriente absorbida por una lámpara fluorescente.

# de armónico	Magnitud [%]
3	29
5	9
7	5

Fuente: Conviviendo con los armónicos, Información Técnica, CYDESA

En la tabla 22 se muestran los valores de magnitud, fase y el orden de los armónicos utilizados para generar una onda cuadrada y una onda diente de sierra, en la figura 83 se observan las formas de ondas obtenidas para esta prueba.

Figura 82: Forma de onda de la corriente absorbida por una lámpara fluorescente



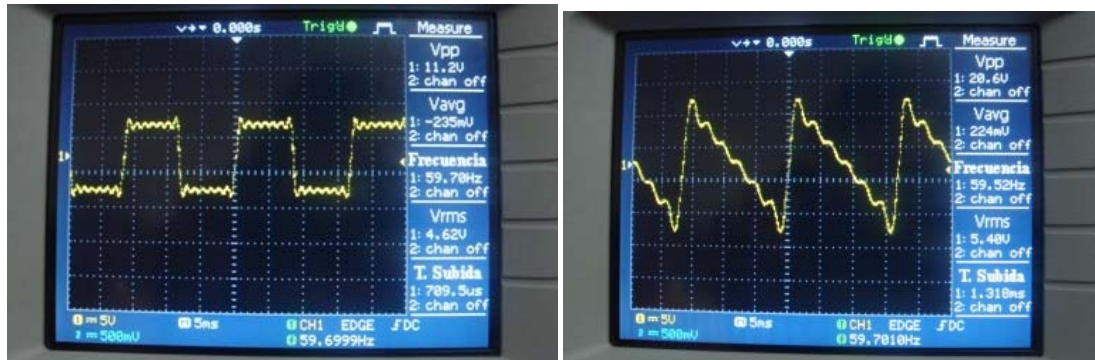
Fuente: los autores

Tabla 22: Orden del armónico, magnitud y fase para generar una señal cuadrada y una señal diente de sierra.

Onda cuadrada			Onda diente de sierra		
# de armónico	Magnitud [%]	Fase	# de armónico	Magnitud [%]	Fase
3	33	0	2	50	0
5	20	0	3	33	0
7	14	0	4	25	0
9	11	0	5	20	0
11	9	0	6	17	0

Fuente: los autores del proyecto

Figura 83: (a) Señal cuadrada, (b) Señal diente de sierra.



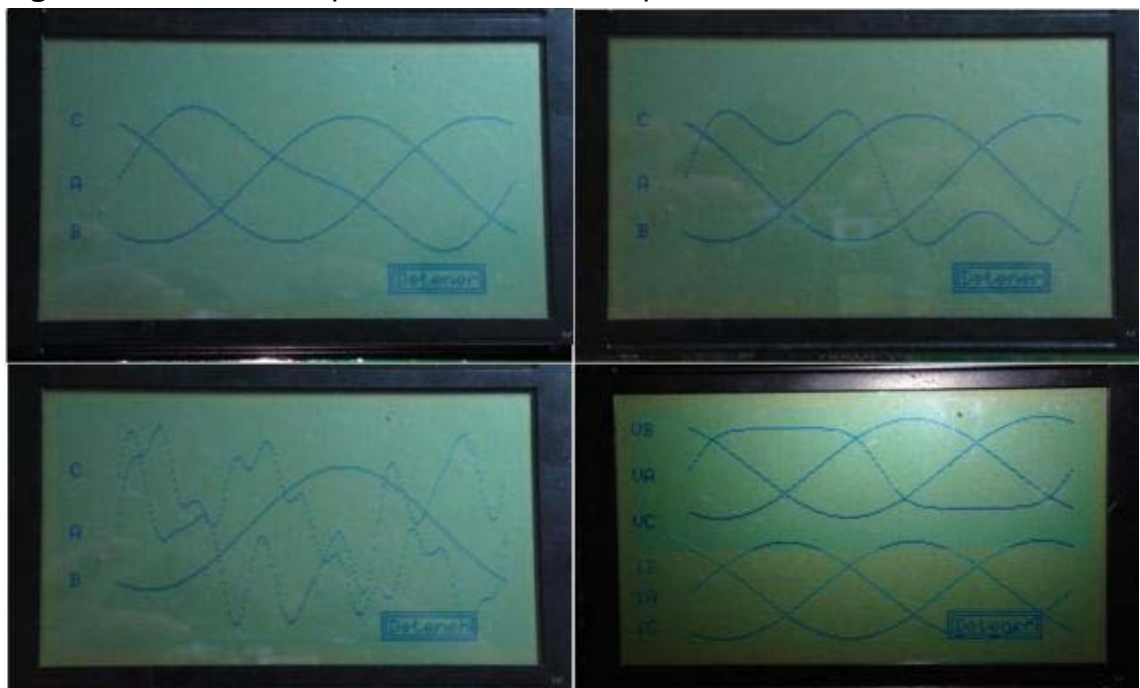
a)

b)

Fuente: los autores del proyecto

En la figura 84 se muestran algunas fotos tomadas a la pantalla del equipo donde se muestran diferentes señales generadas.

Figura 84: Señales de prueba mostradas en pantalla



Fuente: los autores del proyecto

Corriente de las señales de salida

Para medir la corriente que suministra el equipo se puso una carga resistiva a la salida y se genero una señal senoidal con la amplitud máxima que el equipo da (6Vp), se tomo el valor de la tensión en la carga y se hallo el valor de la corriente, esta prueba se realizo con varias cargas en, la tabla 22 se muestran los datos obtenidos.

Como se observa en la tabla poniendo una carga resistiva de 986Ω se obtuvo una corriente de 6,45 mA, cumpliendo y superando el objetivo de generar una señal con una potencia máxima de 25mW.

Tabla 23: Corriente a la salida con diferentes cargas resistivas.

Vout (Vp)	R(k Ω)	Tensión en R (V)	I=V/R (mA)
6.2	0,568	6,38	11,23
	0,986	6,36	6,45
	1,954	6,36	3,25
	5,02	6,30	1,25
	7,45	6,32	0,85
	8,1	6,32	0,78
	11,7	6,26	0,54
	50,2	6,34	0,13

Fuente: los autores de proyecto.

En la figura 85 se muestra una foto del generador señales de formas de onda de tensión y corriente para sistemas trifásicos.

Figura 85: Generador de señales de formas de ondas de tensión y corriente para sistemas trifásicos.



Fuente: los autores del proyecto

6. RECOMENDACIONES

Con el propósito de obtener un equipo optimizado y mejorado en el desarrollo de proyectos futuros tomando como base el nuestro los autores damos algunas recomendaciones.

- Para evitar el uso de dispositivos externos como el FT232R que permite la comunicación USB, el procesador también debe tener este modulo con el fin de alcanzar mayor velocidad de transmisión y simplicidad de hardware.
- Dentro de las funciones que tiene la interfaz en MATLAB adicionar la obtención de magnitud y fase de los armónicos de señales de ondas triangulares y ondas cuadradas con el fin de enviarlas al equipo para que sean generadas.
- Para que el equipo funcione el cable de programación debe estar conectado al PC, esto hace que el prototipo no sea totalmente independiente, para mejorar esto se debe desconectar la interfaz de programación (JTAG) del resto del circuito cuando no se necesite⁵.
- Adicional a las funciones del equipo se puede implementar la visualización en pantalla del contenido espectral de las señales generadas.
- Brindar la posibilidad de cambiar la frecuencia fundamental con la que se generan las ondas, en este proyecto está por defecto 60Hz.

⁵ Para una mejor comprensión ver 56F8323 Evaluation Module User Manual.

- Visualización independiente de cada una de las señales trifásicas y a su vez ajustar el número de periodos que se desean observar.
- Partiendo de la necesidad de que no se pierdan datos almacenados ante un corte de energía se recomienda implementar un mecanismo de respaldo como puede ser una batería que alimente el dispositivo, con lo que también se lograría mayor independencia del mismo. Otra opción sería la implementación de una memoria externa EEPROM que almacene los datos ingresados.

7. CONCLUSIONES

- Se construyó un dispositivo versátil, capaz de generar las señales de tensión de fase o corrientes de línea balanceadas o desbalanceadas correspondientes a los datos de magnitud y fase y número de armónicos introducidos por el usuario utilizando el teclado o a través de la interfaz creada en MATLAB.
- Se implementó una tarjeta de desarrollo para el DSP 56F8345 de Motorola que fue acondicionada a los periféricos y características requeridas para el dispositivo planteado en este proyecto que incluían la pantalla LCD, puerto de comunicación USB, teclado e interfaz de programación
- Se elaboró un algoritmo en *code warrior* para la programación que controla el DSP y sus periféricos, que genera y visualiza en pantalla las formas de onda trifásicas y contiene las funciones necesarias para que el usuario del dispositivo navegue a través de los diferentes menús y escoja el número de armónicos, los desfases y la clase de señales que quiera generar.
- La comunicación con el PC se hace a través de un puerto USB por el cual se reciben las órdenes y los datos enviados desde la interface grafica creada en Matlab, amigable para el usuario y que facilita el manejo del dispositivo.
- Las salidas del generador cuentan con una protección contra corto circuito gracias a que los amplificadores de salida cuentan con esta característica.
- Se implementó el DAC 8568 de 16 bits de resolución, que cuenta con comunicación SPI que facilitó la comunicación con el DSP y gracias a sus ocho canales se generaron las siete señales mediante un hardware de fácil implementación y una configuración sencilla.

- Bajo las condiciones con las que fue configurado las salidas del DAC y la etapa de acondicionamiento se obtiene a la salida del generador una señal mínima de xxx y una señal máxima de $+ / - 6,2 \text{ V}$.
- El generador está en capacidad de generar ondas periódicas conocidas como onda cuadrada y diente de sierra.

8. BIBLIOGRAFÍA

[1] HERNÁNDEZ, Hugo – CARVAJAL, Wilmar – ROA, Elkim. Conversores: Digital Analógico – Analógico Digital una perspectiva de diseño; División de publicaciones. UIS, 2009.

[2] HERNÁNDEZ HERRERA, Hugo Daniel. Diseño de un Conversor Digital Analógico Integrado para un Transmisor de RF en Tecnología CMOS. Trabajo de Grado. UIS, 2005.

[3] NASSER Kehtarnavaz - MANSOUR Keramat. DSP System Design Using the TMS320C6000. Prentice Hall, 2001.

[4] PETIT J. & USTARIZ A. Análisis General de Armónicos: Revisión de las definiciones de Potencia. Trabajo de Grado. UIS, 1997.

[5] MONTAÑO ASQUERINO, Juan Carlos - LÓPEZ OJEDA, Antonio - CASTILLO TRUJILLO, Luis Francisco. Generador de Funciones Trifásicas. CSIC, Universidad de Sevilla. *Nº registro*: P200202110, Oficina Española de Patentes y Marcas (13-9-02).

[6] OPPENHEIM, Alan V – WILLSKY, Alan S. Señales y Sistemas. Segunda Edición. Prentice Hall, 1997.

[7] SEDRA, Adel – SMITH, Kenneth. Circuitos Microelectrónicos. Cuarta Edición. Oxford University Press, 1999

[8] COUGHLIN, Robert F – DRISCOLL, Frederick F. Circuitos Integrales Lineales y Amplificadores Operacionales. Segunda Edición. Prentice Hall, 1987.

[9] PÉREZ DÍAZ, Wilson Andrés. Diseño e implementación de un prototipo de contador de energía trifásico utilizando un procesador de señales digitales (DSP). Trabajo de Grado. UIS, 2006.

[10] Programming On-Chip Flash Memories of 56F80x Devices Using the JTAG/OnCE Interface, Freescale Semiconductor.

[11]

ANEXO A

MANUAL DE USUARIO DEL GENERADOR

ESPECIFICACIONES

Tensión de alimentación: 9v

Número de salidas: 7

Amplitud máxima de salida del generador: 12 Vpp

Corriente de salida con carga de 1K Ω : 6.45 mA

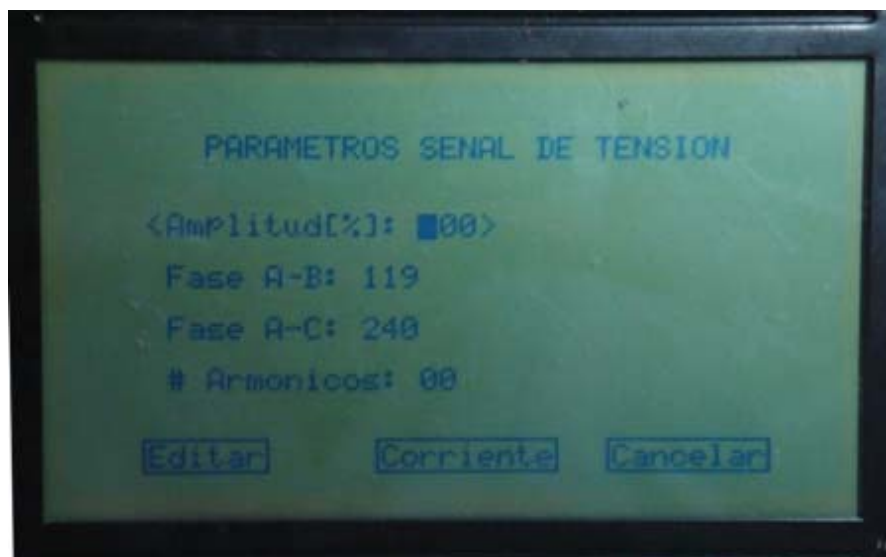
No máximo de armónicos por fase: 50

1. Se energiza el dispositivo de acuerdo a la especificación dada.
2. Después de encendido el dispositivo se observara en la pantalla LCD, el nombre del dispositivo, el logo de la UIS, el nombre de los autores y el botón de iniciar que debe ser seleccionado con el botón de enter del teclado.

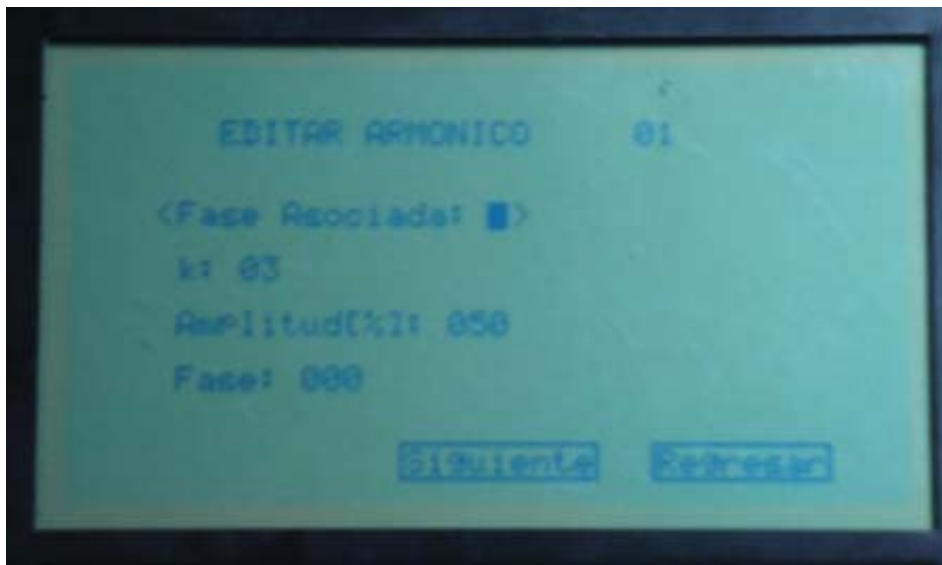


3. Después de oprimir el botón iniciar la pantalla LCD nos muestra los parámetros de la señal de tensión que pueden ser manipulados desplazándose con las flechas del teclado y modificar sus valores. Nos

encontramos entonces con la amplitud de la señal fundamental en porcentaje de la amplitud máxima del generador, el desfase entre A-B, y A-C y el número total de armónicos que queramos generar. En la parte inferior aparecen 3 botones: editar, corriente y cancelar. Si escogemos editar pasaremos a la pantalla de editar armónicos. Al escoger corriente pasaremos a la pantalla de parámetros de las señales de corriente donde se encuentra el botón de generar las señales.



4. Una vez estamos en la pantalla de editar armónicos, manipularemos la fase asociada A, B, ó C con el botón #1 según corresponda, moviéndonos con las flechas encontramos k que es el número del armónico a editar que para las especificaciones del dispositivo varía entre 1 y 50, la amplitud en porcentaje y fase del mismo. Encontramos también los botones de siguiente y regresar, siguiente nos habilitará la pantalla para editar el armónico numero dos y así sucesivamente dependiendo del número de armónicos que se escoja en el numeral 3. Se debe después de editar el ultimo armónico dar siguiente para guardar bien los datos y después en regresar para volver a la pantalla de parámetros de señal de tensión.



5. Después de editar los armónicos para la señal de tensión o de no querer hacerlo y generar senoidales puras ingresamos con el botón corriente a la pantalla de parámetros de corriente. Encontraremos los parámetros a modificar si así lo queremos, podemos modificar la amplitud (%) de la señal fundamental, moviéndonos con las flechas encontramos el desfase entre A-B y A-C, y el número total de armónicos que deseemos generar. Encontraremos también tres botones, el de editar que nos llevará a la pantalla de edición de armónicos si es necesario, el de cancelar, y el botón de generar para ver en pantalla las señales generadas.

