

DISEÑO E IMPLEMENTACION DE UN GENERADOR DE SEÑALES TRIFASICO

LUIS RAMON MERCHAN VILLALBA

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTA DE INGENIERIAS FISICO MECANICAS
ESCUELA DE INGENIERIAS ELECTRICA, ELECTRONICA Y
TELECOMUNICACIONES
BUCARAMANGA

2013

DISEÑO E IMPLEMENTACION DE UN GENERADOR DE SEÑALES TRIFASICO

Autor:

LUIS RAMON MERCHAN VILLALBA

Tesis de grado para optar por el título de
Ingeniero electrónico

Director

MSc. JAIME GUILLERMO BARRERO PEREZ

Codirector

Dr. GABRIEL ORODOÑEZ PLATA

UNIVERSIDAD INDUSTRIAL DE SANTANDER

FACULTA DE INGENIERIAS FISICO MECANICAS

ESCUELA DE INGENIERIAS ELECTRICA, ELECTRONICA Y

TELECOMUNICACIONES

BUCARAMANGA

2013

DEDICATORIA

Dedico este logro a mis padres, LUIS RAMON MERCHAN BARRAGAN y SARA INES VILLALBA TRILLOS, quienes mediante una gran labor de esfuerzo y dedicación lograron dejarme como legado el bien más preciado que puedo tener, la educación.

A mis hermanos MARIA ALEJANDRA MERCHAN VILLALBA y JHON EDGAR MERCHAN VILLALBA, por su apoyo incondicional.

A mi novia KARLY VANESSA PAJARO VARGAS, por ser ese soporte que siempre me alentó y por brindarme fuerzas en los momentos que más lo necesitaba.

A toda mis familiares quienes siempre me animaron al sentirse el orgullosos de poder tener el primer ingeniero de la familia.

A todos mis amigos, por su apoyo y motivación especialmente a CARLOS ARTURO RAMOS.

AGRADECIMIENTOS

Agradezco a mi director de trabajo de grado JAIME GUILLERO BARRERO PEREZ, quien me guio en la senda para poder llegar a feliz término con esta labor.

Al grupo de investigación en robótica aplicada ERA, especialmente a IVAN DARIO PEÑALOSA quien fue un gran apoyo.

A la UNIVERSIDAD INDUSTRIAL DE SANTANDER por brindarme la oportunidad de poder ser un profesional.

TABLA DE CONTENIDO

INTRODUCCION	18
1. FUNDAMENTOS TEORICOS.....	20
1.1. Microcontrolador.....	20
1.2. Memoria externa no volátil.....	21
1.2.1. Memoria Flash.....	21
1.2.2. Memoria eeprom.....	22
1.2.3. Diferencia entre las memorias eeprom y flash.....	22
1.3. Conversor digital analógico.....	23
1.3.1. Comportamiento estático.....	24
1.3.2. Comportamiento Dinámico.....	27
1.3.3. Arquitecturas DAC.....	29
1.3.3.1. Arquitectura resistor ladder.....	29
1.3.3.2. Arquitectura current steering.....	30
1.3.3.3. Arquitectura charge redistribution.....	31
1.3.3.4. Arquitectura sigma-delta.....	32
1.4. Sistemas trifásicos.....	34
1.5. Armónicos en un sistema eléctrico.....	36
2. DISEÑO DEL SISTEMA	39
2.1. Hardware.....	39
2.1.1. Dispositivo programable.....	39
2.1.2. Memoria externa.....	41
2.1.3. Interfaz de usuario local.....	43
2.1.3.1. Visualización.....	43
2.1.3.2. Ingreso de datos.....	45
2.1.4. Alimentación del sistema.....	47

2.1.5.	<i>Generador de señales</i>	49
2.1.5.1.	<i>Convertor digital analógico</i>	49
2.1.5.1.1.	<i>Circuito de compensación de referencia</i>	52
2.1.5.2.	<i>Circuito de referencia de tensión</i>	54
2.1.5.3.	<i>Etapas de filtrado</i>	55
2.2.	Software	61
2.2.1.	Creación de señales.....	61
2.2.1.1.	<i>Frecuencia de muestreo</i>	61
2.2.2.	Calculo, almacenamiento y generación de las señales	63
2.2.3.	Generación de señales en LabView	73
2.3.	Interfaz de usuario	77
2.3.1.	Ingreso de datos remota	77
2.3.2.	Ingreso de datos local	86
3.	PRUEBAS	90
3.1.	Distorsión Armónica.....	90
3.2.	Verificación amplitudes generadas	96
3.3.	Verificación fases generadas.....	101
3.4.	Verificación Señales Teóricas contra Experimentales	103
4.	CONCLUSIONES	109
5.	RECOMENDACIONES.....	110
	REFERENCIAS.....	111
	ANEXOS.....	112

LISTA DE FIGURAS

Figura 1 Grafica efecto de no linealidad DAC.....	25
Figura 2 Offset de un DAC.....	26
Figura 3 Error de ganancia en un DAC.....	26
Figura 4 Monotonicidad de un DAC.....	27
Figura 5 Tiempo de establecimiento DAC.....	28
Figura 6 Salida de glitch en un DAC.....	28
Figura 7 Arquitectura DAC <i>resistor ladder</i>	30
Figura 8. Arquitectura <i>current steering</i>	31
Figura 9 Arquitectura <i>Charge Redistribution</i>	32
Figura 10 Conversor sigma-delta: (a) diagrama de bloques, (b) espectro de la señal en diferentes puntos.....	33
Figura 11. Filtro de interpolación: (a) Diagrama de bloques, (b) diferentes operaciones. ..	34
Figura 12 Valores instantáneos de las tensiones. Fuente: Autor.....	35
Figura 13 Esquema alimentación de un generador trifásico en Y.....	35
Figura 14 (a) Cargas lineales (b) Cargas no lineales.....	36
Figura 15 (a) Señal Original, (b) 1 armónico, (c) 1+2 armónico, (d) 1+2+3 armónico, (e) 1+2+3+4 armónico, (f) 1+2+3+4+5 armónico. Fuente: Wikipedia series de Fourier.....	38
Figura 16 Arquitectura PIC18F4550.....	40
Figura 17 Trama de datos Escritura EEPROM.....	42
Figura 18 Trama de datos Lectura EEPROM.....	43
Figura 19 Pantalla Graficadora QY12864.....	44
Figura 20. Arquitectura pantalla grafica QY12864.....	45
Figura 21 Teclado Matricial.....	46
Figura 22 Teclado Analógico.....	47
Figura 23. Fuente de alimentación.....	48
Figura 24. Diagrama integrado DCP020505.....	48
Figura 25. Circuito DCP020505. Fuente: Autor.....	49
Figura 26 Arquitectura DAC8871.....	52
Figura 27 Trama escritura DAC8871.....	52
Figura 28 Corriente referencia vs código digital.....	53
Figura 29 Configuración buffer fuente dual +/- 10[V].....	53
Figura 30. Referencia positiva y negativa.....	55
Figura 31 Selección tipo de filtro.....	56
Figura 32 Parámetros Filtro activo.....	56
Figura 33. Selección Filtro.....	57

Figura 34 Selección topología filtro activo.	57
Figura 35 Filtro Activo más respuesta Magnitud y fase.	58
Figura 36 Filtro activo con valores normalizados.	59
Figura 37 Respuesta filtro activo valores normalizados.....	60
Figura 38. Señales de control DAC.	62
Figura 39. Diagrama de flujo generación señal.....	65
Figura 40. Desplazamiento señal generada.....	66
Figura 41. Diagrama de flujo cálculo de graficas de las señales.....	67
Figura 42. Trama de datos DAC.....	68
Figura 43. Trama de datos DAC y canales.	68
Figura 44. Diagrama de flujo, lectura de dato.	72
Figura 45. Matriz de parámetros.....	73
Figura 46. Diagrama generación de señales.	73
Figura 47. Desplazamiento de nivel.	74
Figura 48. Separación valor de 16 bits en bits.	74
Figura 49. Conversión arreglo de bits a número entero.....	75
Figura 50. Conversión numero entero a cadena, para transmisión.....	75
Figura 51. Puerto visa. a) Panel frontal b) Diagrama de bloques.....	75
Figura 52. Envío de datos puerto serial.....	76
Figura 53. Ingresando parámetro interfaz remota.....	78
Figura 54. Selección puerto serial.	78
Figura 55. Selección puerto RealTerm.....	79
Figura 56. Seleccionar archivo destino de datos recibidos.....	79
Figura 57. Recepción de datos habilitada.	80
Figura 58. Datos enviados.....	80
Figura 59. Recepción de datos.....	81
Figura 60. Habilitar comunicación USB.....	81
Figura 61. Sistema esperando enlace con el computador.....	82
Figura 62. Sistema enlazado con el computador.....	82
Figura 63. Asignación puerto COM al sistema.....	83
Figura 64. Selección puerto del sistema.	83
Figura 65. Selección archivo de datos.	84
Figura 66. Envío de datos PC-Sistema.	84
Figura 67. Sistema recibiendo datos.....	84
Figura 68. Sistema indicando que la transferencia ha sido completa.	85
Figura 69. Visualización señal generada.....	85
Figura 70. Modo adquisición de parámetros.....	86
Figura 71. Teclas de desplazamiento.....	87
Figura 72. Ingreso de dato parámetro.....	87
Figura 73. Visualización parámetro escrito.	87

Figura 74. Ingreso de valor 111,11.....	88
Figura 75. Mensaje al ingresar valor no valido.....	88
Figura 76. Espectro señal 60[Hz], cursor frecuencia.....	90
Figura 77. Espectro señal 60[Hz], cursor Amplitud.....	91
Figura 78. Espectro señal 600[Hz], cursor frecuencia.....	91
Figura 79. Espectro señal 600[Hz], cursor Amplitud.....	91
Figura 80. Espectro señal 1200[Hz], cursor frecuencia.....	92
Figura 81. Espectro señal 1200[Hz], cursor amplitud.....	92
Figura 82. Espectro señal 1800[Hz], cursor frecuencia.....	92
Figura 83. Espectro señal 1800[Hz], cursor Amplitud.....	93
Figura 84. Espectro señal 2400[Hz], cursor frecuencia.....	93
Figura 85. Espectro señal 2400[Hz], cursor amplitud.....	93
Figura 86. Espectro señal 3000[Hz], cursor frecuencia.....	94
Figura 87. Espectro señal 3000[Hz], cursor Amplitud.....	94
Figura 88. Señales de 10[V] y 1[V] de tensión pico 60[Hz].....	96
Figura 89. Señales de 10[V] y 0,1[V] de tensión pico 60[Hz].....	96
Figura 90. Señales de 10[V] y 0,05[V] de tensión pico 60[Hz].....	97
Figura 91. Señales de 10[V] y 0,01[V] de tensión pico 60[Hz].....	97
Figura 92. Señales de 10[V] y 0,005[V] de tensión pico 60[Hz].....	97
Figura 93. Señal con Amplitud de 10 [mV].....	98
Figura 94. Espectro señal de salida 10 [mV], cursor frecuencia.....	98
Figura 95. Espectro señal de salida 10 [mV], cursor Amplitud.....	99
Figura 96. Señal con Amplitud de 1 [mV].....	99
Figura 97. Espectro señal de salida 1 [mV], cursor frecuencia.....	99
Figura 98. Espectro señal de salida 1 [mV], cursor Amplitud.....	100
Figura 99. Señal de 60[Hz], desfase 45[°].....	101
Figura 100. Señal de 60[Hz], desfase de 90[°].....	101
Figura 101. Señal de 60[Hz], desfase de 180[°].....	102
Figura 102. Señal de 60[Hz], desfase de 270[°].....	102
Figura 103. Señal teórica 1, 2, 3, 4,5 armónicos temporal.....	103
Figura 104. Señal teórica 1, 2, 3, 4,5 armónicos frecuencia.....	103
Figura 105. Señal experimental 1, 2, 3, 4,5 armónicos temporal.....	104
Figura 106. Señal experimental 1, 2, 3, 4,5 armónicos frecuencia.....	104
Figura 107. Señal Real componentes 1 y 2.....	105
Figura 108. Señal Simulada componentes 1 y 2.....	105
Figura 109. Señal Real componentes 1, 2,3 ,4 y 5.....	106
Figura 110. Señal Simulada componentes 1, 2,3 ,4 y 5.....	106
Figura 111. Señal Real componentes 1, 2, 3, 4, 5 y 6.....	106
Figura 112. Señal Simulada componentes 1, 2, 3, 4, 5 y 6.....	107
Figura 113. Señal Real componentes 1, 2, 3, 4, 5, 6 y 7.....	107

Figura 114. Señal Simulada componentes 1, 2, 3, 4, 5, 6 y 7. 107

LISTA DE ECUACIONES

Ecuación 1. Relación entrada salida DAC.....	24
Ecuación 2 No linealidad diferencia.....	25
Ecuación 3 No linealidad integral.	25
Ecuación 4 Relación señal a ruido.	29
Ecuación 5. Representación de la señal temporal $X(t)$ en series de Fourier.	37
Ecuación 6. Coeficientes de Fourier de la señal en el dominio del tiempo $X(t)$	37
Ecuación 7. Tensión Horizontal Teclado.	46
Ecuación 8. Tensión Vertical teclado.....	46

LISTA DE TABLAS

Tabla 1. Comparación protocolos seriales memorias.....	21
Tabla 2. Comparación DACS. Fuente: Autor.	50
Tabla 3. Comparación referencias de tensión.....	54
Tabla 4. Reemplazos de componentes filtro activo calculado.	58
Tabla 5. Posibles frecuencias de muestreo.....	63
Tabla 6. Ejemplo valores de una muestra.....	69
Tabla 7. Asignación de valores de una muestra a un vector de bytes.	70
Tabla 8. Magnitudes de armónicos diferentes frecuencias.....	95
Tabla 9. Distorsión armónica señales generadas.	95
Tabla 10. Magnitudes de armónicos diferentes frecuencias.....	100
Tabla 11. Distorsión armónica señales generadas.	100

LISTA DE ANEXOS

ANEXO a. Hoja de datos PIC18F4550.....	112
ANEXO b. Hoja de datos DCP020505	115
ANEXO c. Hoja de datos DAC8871	115
ANEXO d. Hoja de datos REF5010.....	122
ANEXO e. Diagrama de funcionamiento del generador de señales trifásico.....	124
ANEXO f. Esquema circuito de control generador de señales trifásico.....	125
ANEXO g. Esquema circuito canal conversor digital analógico completo.....	126
ANEXO h. Foto Proyecto	127

RESUMEN

TITULO

Diseño e implementación de un generador de señales trifásico¹

AUTOR: LUIS RAMON MERCHAN VILLALBA²

PALABRAS CLAVES: MICROCONTROLADOR, DAC, USB, SPI, EEPROM.

DESCRIPCION

Este trabajo describe el diseño e implementación de un dispositivo electrónico programable que genera 7 señales de salida analógicas.

Durante el desarrollo del proyecto se diseñaron los diferentes componentes que forman el prototipo final. El control y la generación de las ondas se realiza mediante el microcontrolador PIC18F4550 de la empresa microchip, el cual cuenta entre sus principales características con comunicación USB para la comunicación entre el computador y el dispositivo, además se utilizó el componente DAC8871 de Texas instruments para realizar la conversión de señales digitales a analógicas, una memoria externa EEPROM³ que utiliza el protocolo SPI para su funcionamiento donde se almacenan las señales generadas, una pantalla de cristal líquido y un teclado para interactuar con el dispositivo.

El microcontrolador posee un programa con dos modos de funcionamiento, uno es manual y otro es mediante el computador. En el modo manual, el usuario configura 7 señales, donde a cada una se le establece: el nivel de continua, las magnitudes y fases de 50 armónicos; con estos parámetros el microcontrolador genera las señales correspondientes; una vez que generadas, estas son almacenadas en una memoria externa no volátil, en la cual permanecerán estas señales hasta que quieran cambiarse por otras. Mediante el computador, se trabaja en un programa realizado con la herramienta de software LABVIEW, la cual cuenta con una interfaz gráfica, en la cual el usuario configura los mismos parámetros del modo manual; esta herramienta genera las señales y las envía al microcontrolador, el cual las almacena en una memoria externa. Una vez que se tienen los valores en la memoria externa, el microcontrolador accede a estos valores y los envía a la etapa de conversión digital analógica, la cual cuenta con una etapa de filtrado en la salida para eliminar componentes indeseadas.

¹ Trabajo de grado

² Facultad de ingenierías físico-mecánicas, escuela de ingenierías, eléctrica, electrónica y de telecomunicaciones. Director: MPE. Jaime Guillermo Barrero Pérez

³ Electrically Erasable Programmable Read-Only

ABSTRAC

TITLE

Design and implementation of a three-phase signals generator⁴.

AUTHOR: LUIS RAMÓN MERCHAN VILLALBA⁵

KEYWORDS: MICROCONTROLLER, DAC, USB, SPI, EEPROM.

DESCRIPTION:

This paper describes the design and implementation of a programmable electronic device that generates as an output seven analogic signals.

During the development of this paper the different components that form the final prototype were designed. The control and generation of waves is performed by the microcontroller PIC18F4550 from microchip company, which includes among its main features with USB communication for data transfer between the computer and device, it was also used the component DAC8871 of Texas Instrument, this one perform the conversion from digital to analogic signal, an EEPROM⁶ external memory that uses the SPI protocol, in which the signals generated are stored.

The microcontroller has a program with two modes of operation; one is manual and the other works through a computer. In the manual mode, the user sets 7 signals, where each one should have: The DC level, the magnitudes and phases of 50 harmonics, with these parameters the microcontroller generates the corresponding signals; once generated, these are stored in a non-volatile external memory, these signals will remain until the user wishes to change them. In the computer mode, the user interacts with a program developed with LabView software tool, which has a graphical interface in which the user sets the same parameters manually, this tool generates and sends signals to the microcontroller, which stores them in an external memory. Once the values are in the external memory, the microcontroller proceeds to access these values and send them to the analog digital conversion stage, which has a filter to remove undesired components in the output.

Besides generating signals corresponding to a three-phase system, the prototype can be used to generate any type of signal that considers the special characteristics of this generator.

⁴ Undergraduate Project

⁵ Physiomechanical Engineering College. Electronic Engineering School.

⁶ Electrically Erasable Programmable Read-Only.

INTRODUCCION

En el universo se presenta una gran cantidad de fenómenos físicos con los que se interactúan e influyen en el vivir, estos pueden ser representados mediante señales, las cuales muestran una medida cuantitativa del fenómeno a través de funciones continuas en el tiempo, con estas se pueden realizar análisis matemáticos para poder estudiar el comportamiento dichos fenómenos.

En la actualidad se analizan señales de tensión o corriente que representan fenómenos físicos, pues mediante el uso de transductores y sensores se pueden convertir medidas de presión, temperatura, radiación, peso, entre otros, a este tipo de señales, siendo de esta forma más factible poder analizar el comportamiento de cualquier medida que se requiera.

Al poder representar fenómenos físicos mediante señales eléctricas, se tiene que estos se pueden simular, al generar una señal que simbolice el fenómeno. Las señales de tensión pueden ser generadas mediante diversos métodos, teniendo la forma que se requiera para poder realizar pruebas en equipos, que responden ante estas, de manera que teniendo una forma de señal conocida se pueda corroborar un resultado esperado con el real, para así poder comprobar el buen funcionamiento estos.

Las entradas de los contadores digitales de energía eléctrica, están conformadas por 7 señales; tres de tensión y cuatro de corriente que representan un sistema eléctrico trifásico, no necesariamente de forma senoidales o tradicionales como la cuadrada, rampa y triangular. La mayoría de los generadores de señales existentes en el mercado crean estas formas de onda a una frecuencia determinada. También existe otro tipo de generador que produce una señal de salida, la cual puede ser creada por el usuario punto por punto⁷. Si se necesitan varias señales se requerirán varios generadores sincronizados entre sí, pues estos

⁷ Generador de funciones arbitrarias.

poseen habitualmente una sola salida o como máximo 2. Para el caso de un sistema eléctrico trifásico donde se busca simular señales con una frecuencia fundamental y afectada por armónicos, es necesario una herramienta en la que el usuario pueda generar estas señales de una manera cómoda, además de que tenga las salidas necesarias para la aplicación, que en el caso de simulación de cargas trifásicas no lineales serían 7.

Este proyecto, plantea el diseño e implementación de un generador de señales analógicas programables, las cuales pueden simular un sistema eléctrico trifásico al poder generar las señales presentes en este. A continuación se presentan los resultados obtenidos en el desarrollo del mismo.

1. FUNDAMENTOS TEORICOS

1.1. Microcontrolador

Un microcontrolador (abreviado μ C, UC o MCU) es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. Está compuesto de varios bloques funcionales, los cuales cumplen una tarea específica. Un microcontrolador incluye en su interior las tres principales unidades funcionales de una computadora:

- Unidad central de procesamiento.
- Memoria.
- Periféricos de entrada/salida.

Cuando es fabricado, el microcontrolador no contiene datos en la memoria de programa. Para que pueda controlar algún proceso es necesario generar o crear y luego grabar en dicha memoria o equivalente del microcontrolador algún programa, el cual puede ser escrito en lenguaje ensamblador, otro lenguaje para microcontroladores o algún lenguaje de alto nivel como C ,C++ entre otros con ayuda de compiladores.; sin embargo, para que el programa pueda ser grabado en la memoria del microcontrolador, debe ser codificado en sistema numérico hexadecimal que es finalmente el sistema que hace trabajar al microcontrolador cuando éste es alimentado con el voltaje adecuado y asociado a dispositivos analógicos y discretos para su funcionamiento.

Un microcontrolador difiere de una CPU normal, debido a que es más fácil convertirla en una computadora en funcionamiento, con un mínimo de chips externos de apoyo. La idea es que el chip se coloque en el dispositivo, enganchado a la fuente de energía y de información que necesite, y eso es todo.

El microcontrolador es un sistema cerrado. Todas las partes están contenidas en su interior y sólo salen al exterior las líneas que gobiernan los periféricos.

En la práctica cada fabricante de microcontroladores oferta un elevado número de modelos diferentes, desde los más sencillos hasta los más poderosos. Es posible seleccionar la capacidad de las memorias, el número de líneas de E/S, la cantidad y potencia de los elementos auxiliares, la velocidad de funcionamiento, etc. Por todo ello, un aspecto muy destacado del diseño es la selección del microcontrolador a utilizar.

1.2. Memoria externa no volátil

En ocasiones, se tiene que las capacidades de almacenamiento de un microcontrolador no son las suficientes para algún tipo de aplicaciones, pues se tiene que estos manejan un almacenamiento de información del orden de Kbytes, por esto se hace necesaria la utilización de memorias externas, las cuales en la actualidad son generalmente de 2 tipos, EEPROM y FLASH.

Estos tipos de memorias permiten comunicación tipo serial y paralela. Entre los protocolos seriales que manejan están SPI, I2C y MICROWIRE, cuyas características más relevantes se presentan en la siguiente tabla:

Tabla 1. Comparación protocolos seriales memorias.

	SPI	I2C	Microwire
Canales	4	2	4
Tamaño palabra(bits)	8	8	8 o 16
Máximo reloj(Mhz)	2.1	0.4	2

Fuente: Autor

1.2.1. Memoria Flash

La memoria FLASH es similar a la EEPROM, es decir que se puede programar y borrar eléctricamente, son de alta densidad (gran capacidad de almacenamiento de bits). Alta densidad significa que se puede empaquetar en una pequeña

superficie del chip, gran cantidad de celdas, lo que implica que cuanto mayor sea la densidad, más bits se pueden almacenar en un chip de tamaño determinado. Sin embargo esta reúne algunas de las propiedades de las memorias anteriormente vistas, y se caracteriza por tener alta capacidad para almacenar información y es de fabricación sencilla, lo que permite fabricar modelos de capacidad equivalente a las EPROM a menor costo que las EEPROM.

1.2.2. Memoria eeprom

Actualmente estas memorias se construyen con transistores de tecnología MOS (Metal Oxide Silice) y MNOS (Metal Nitride-Oxide Silicon).

Las memorias EEPROM son memorias no volátiles y eléctricamente borrables a nivel de bytes. La posibilidad de programar y borrar las memorias a nivel de bytes supone una gran flexibilidad, pero también una celda de memoria más compleja. Además del transistor de puerta flotante anterior, es preciso un segundo transistor de selección. El tener 2 transistores por celda hace que las memorias EEPROM sean de baja densidad y mayor coste. La programación requiere de tiempos que oscilan entre 157 [µs] y 625 [µs] por byte. Además pueden ser borradas eléctricamente, y para reescribir arreglos de datos no se requiere de un previo borrado.

Este tipo de memoria puede ser leída indefinida veces, pero tiene entre 100.000 y 1'000.000 de ciclos de borrado/escritura.

1.2.3. Diferencia entre las memorias eeprom y flash

La diferencia de las memorias flash con las EEPROM reside en su velocidad: Son más rápidas en términos de programación y borrado.

Otra diferencia la encontramos en que en las EEPROM se puede borrar de forma selectiva cualquier byte, mientras que en las memorias FLASH sólo admite el borrado total de la misma.

Por otra parte estas memorias más baratas que las EEPROM, debido a que utilizan una tecnología más sencilla y se fabrican con grandes capacidades de almacenamiento.

1.3. Conversor digital analógico

El conversor digital analógico es un dispositivo que convierte señales del mundo digital al mundo analógico, donde un código puede ser transformado en un valor de tensión, ya que en el mundo real la interacción se realiza entre fenómenos físicos que siempre serán analógicos. Este procedimiento es muy necesario en un mundo actual donde la mayoría del procesamiento de señales se realiza en el dominio digital, donde se cuenta con altas velocidades de procesamiento además de contar con sistemas modificables mediante simples rutinas de software sin necesidad de cambios en hardware.

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 con el teorema de muestreo para poder recuperar la señal original. La salida de un DAC puede ser una señal de corriente o tensión, la cual puede necesitar o no de un buffer y un amplificador dependiendo del dispositivo, pues la impedancia de salida del DAC no permite conectar directamente cargas por su impedancia de salida en la mayoría de las arquitecturas.

Entre las características principales de un DAC que se deben tener en cuenta a la hora de seleccionar alguno, se tienen las siguientes:

- El número de bits, mediante el cual se tiene la resolución que puede brindar el dispositivo.
- Tiempo de establecimiento, el tiempo que demora en establecerse la salida una vez es escrito el código digital en el DAC.

- Rango de salida, especifica la excursión de tensión que puede presentar el dispositivo en su salida, siendo configurada mediante tensiones de referencia que especifican los límites indicados.
- Número de canales, indica la cantidad de salidas presentes en un DAC, las cuales pueden ser manejadas independientemente según el protocolo de comunicación que utilice el mismo.
- Errores, son componentes que alteran el funcionamiento ideal de un DAC.

Entre estas u otras características, algunas se basan en el comportamiento estático y otras en el dinámico del dispositivo.

1.3.1. Comportamiento estático

Característica de entrada/salida: La salida ideal de un DAC se expresa mediante la siguiente expresión:

Ecuación 1. Relación entrada salida DAC

$$V_{out}(D_i) = V_{ref} * \left(\frac{b_n}{2} + \frac{b_{(n-1)}}{2^2} + \dots + \frac{b_1}{2^n} \right)$$

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 la escala total

$-\frac{V_{ref}}{2}$ a $\frac{V_{ref}}{2}$. No linealidad

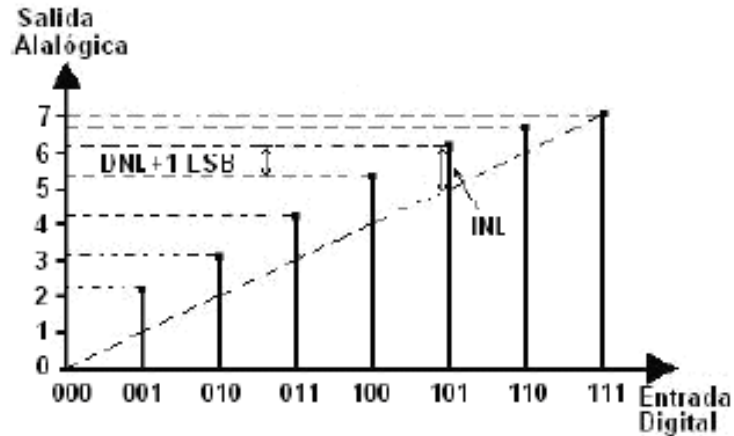
No linealidad diferencial (DNL): El rango de salida de un DAC de N bits es dividido en 2^N partes, y un cambio en un LSB⁸ en la palabra digital de entrada producirá un cambio de $\frac{V_{ref}}{2^N}$ en la salida analógica. La no linealidad diferencial es una medida de la desviación del cambio real a la salida del DAC y el cambio ideal de voltaje al cambio de una LSB (figura 3) está definido por:

⁸ Bit menos significativo

Ecuación 2 No linealidad diferencia.

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

Figura 1 Grafica efecto de no linealidad DAC.



Fuente: NASSER kehtarnavaz – 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. Está definida por:

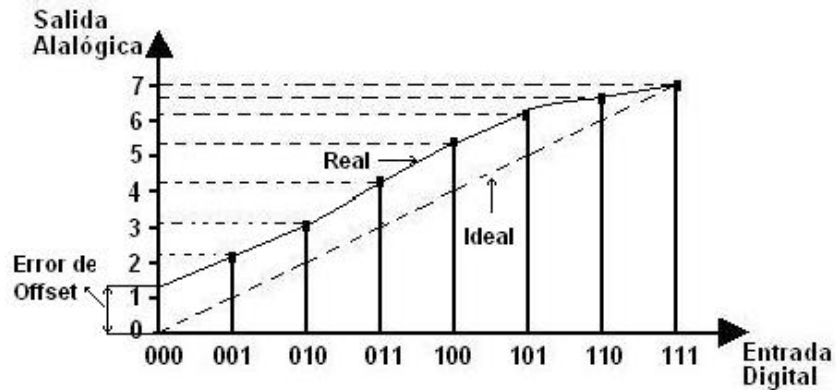
Ecuación 3 No linealidad integral.

$$INL = \frac{V_{out}(D_i) - i * \frac{V_{ref}}{2^N}}{\frac{V_{ref}}{2^N}}$$

Offset: La salida analógica del DAC debería ser 0 V cuando D=0. Sin embargo, como se muestra en la figura 4 existe un voltaje de *offset* similar al presente en un

amplificador operacional que desplaza la salida un nivel de tensión determinado constante.

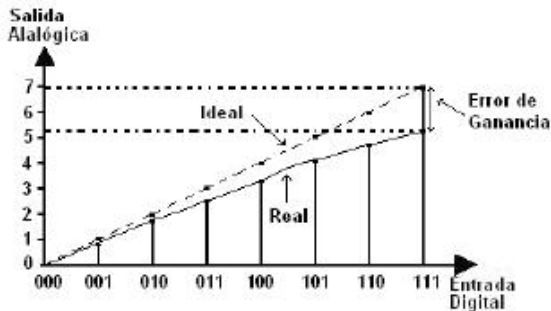
Figura 2 Offset de un DAC.



Fuente: NASSER kehtarnavaz – 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 5.

Figura 3 Error de ganancia en un DAC.

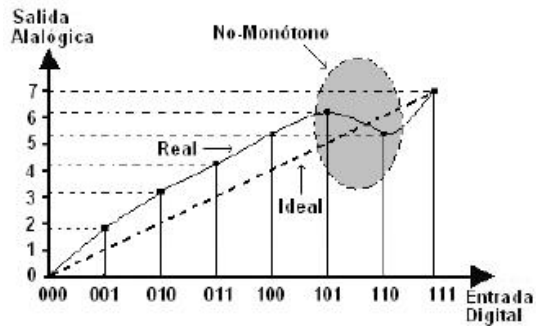


Fuente: NASSER kehtarnavaz – MANSOUR keramat. DSP System Design Using The TMS320C6000. Prentice Hall, 2001.

Monotonidad: La salida del DAC debe incrementarse durante todo su rango, cuando la palabra de entrada digital incrementa. DNL debe ser superior a -1 LSB para que el DAC sea monótono. La Monotonidad es importante en

muchas aplicaciones particularmente en control digital. La figura 6 ilustra la característica no monótona de un DAC.

Figura 4 Monotonicidad de un DAC.

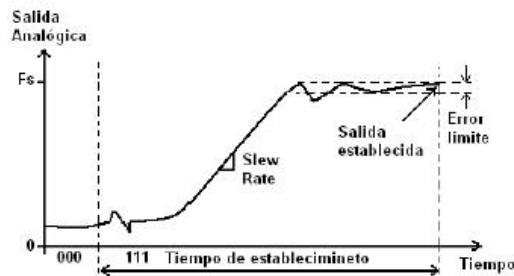


Fuente: NASSER kehtarnavaz – MANSOUR keramat. DSP System Design Using The TMS320C6000. Prentice Hall, 2001.

1.3.2. 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 7).

Figura 5 Tiempo de establecimiento DAC.

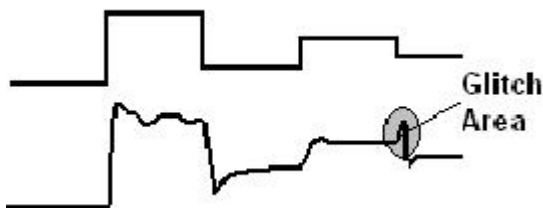


Fuente: NASSER kehtarnavaz – MANSOUR keramat. DSP System Design Using The TMS320C6000. Prentice Hall, 2001.

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 8).

Figura 6 Salida de glitch en un DAC.



Fuente: NASSER kehtarnavaz – 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 $\frac{V_{ref}}{2^N}$. 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:

Ecuación 4 Relación señal a ruido.

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

1.3.3. Arquitecturas 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.

1.3.3.1. *Arquitectura resistor ladder*

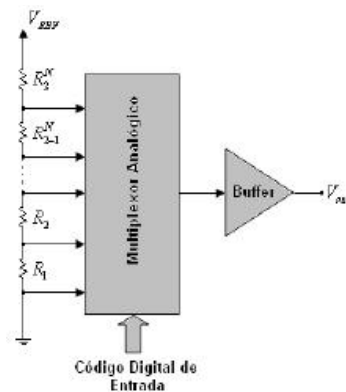
En la arquitectura resistor *ladder* el voltaje de referencia es dividido en $2^N - 1$ subvoltajes (figura 9). La diferencia entre dos voltajes subsecuentes es exactamente 1 LSB. Una red de conmutadores selecciona el voltaje de salida de una cadena de resistencias, basado en la palabra digital de entrada. 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 monotónica. Sus desventajas son:

- a) Necesita una combinación de $2^N - 1$ resistencias.
- 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 7 Arquitectura DAC *resistor ladder*.



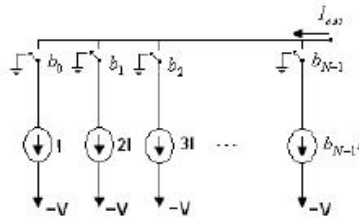
Fuente: NASSER kehtarnavaz – MANSOUR keramat. DSP System Design Using The TMS320C6000. Prentice Hall, 2001.

1.3.3.2. *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 Monotonicidad. Además, permite grandes tasas de conversión. Una de las desventajas es la disipación de potencia estática en sus fuentes de corriente.

Figura 8. Arquitectura *current steering*.

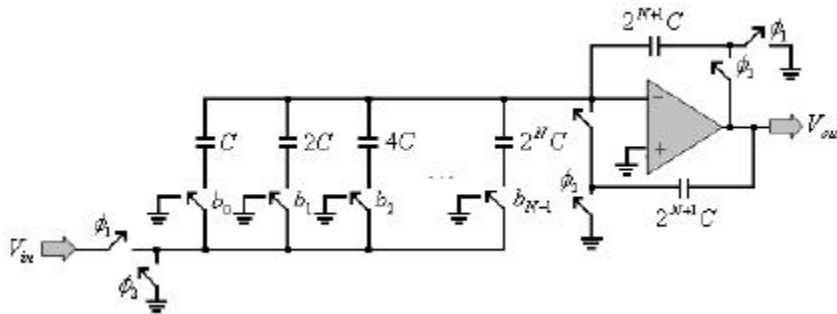


Fuente: NASSER kehtarnavaz – MANSOUR keramat. DSP System Design Using The TMS320C6000. Prentice Hall, 2001.

1.3.3.3. *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 9 Arquitectura *Charge Redistribution*.



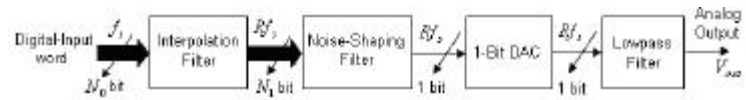
Fuente: NASSER kehtarnavaz – MANSOUR keramat. DSP System Design Using The TMS320C6000. Prentice Hall, 2001.

1.3.3.4. *Arquitectura sigma-delta*

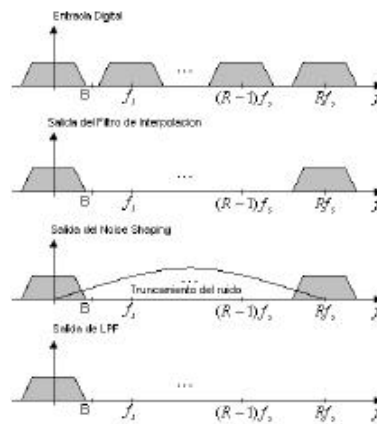
Las anteriores arquitecturas no están disponibles para una resolución alta (>14 bits). La arquitectura sigma-delta puede lograr alta resolución a costa de una menor tasa de muestreo- es decir, altas tasas de sobre muestreo. En la figura 12 se muestra el diagrama de bloques básico de ésta arquitectura 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 pasa bajos a la salida (filtro de reconstrucción).

La función del filtro de interpolación es generar más muestras entre muestras consecutivas (sobre muestreo). Éste aumenta la correlación entre muestras. En la figura 13 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 diezmo.

Figura 10 Conversor sigma-delta: (a) diagrama de bloques, (b) espectro de la señal en diferentes puntos.



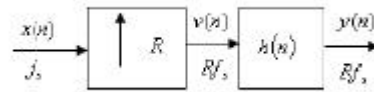
(a)



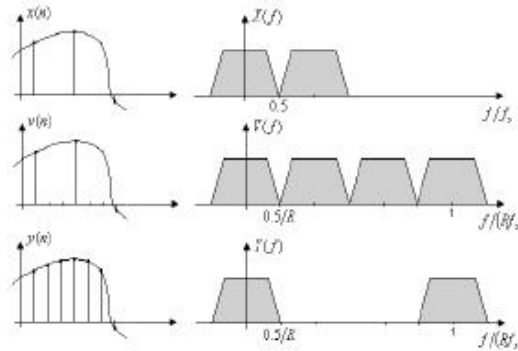
(b)

Fuente: NASSER kehtarnavaz – MANSOUR keramat. DSP System Design Using The TMS320C6000. Prentice Hall, 2001.

Figura 11. Filtro de interpolación: (a) Diagrama de bloques, (b) diferentes operaciones.



(a)



(b)

Fuente: NASSER kehtarnavaz – MANSOUR keramat. DSP System Design Using The TMS320C6000. Prentice Hall, 2001.

1.4. Sistemas trifásicos

Gran parte de la generación, transmisión, distribución y utilización de la energía eléctrica se hace a través de sistemas polifásicos de 60 [Hz] y 50[Hz]; los sistemas polifásicos más conocidos y utilizados son los sistemas trifásicos por razones económicas y operativas.

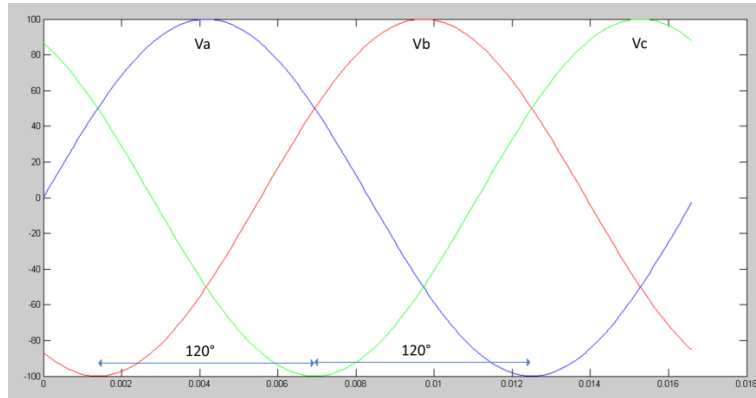
En la figura 14 se tiene la representación temporal de las señales de tensión en un sistema trifásico, estas se representan mediante las siguientes expresiones matemáticas

$$ea(t) = U * \cos(w * t)$$

$$eb(t) = U * \cos(w * t - 2\pi/3)$$

$$ec(t) = U * \cos(w * t + 2\pi/3)$$

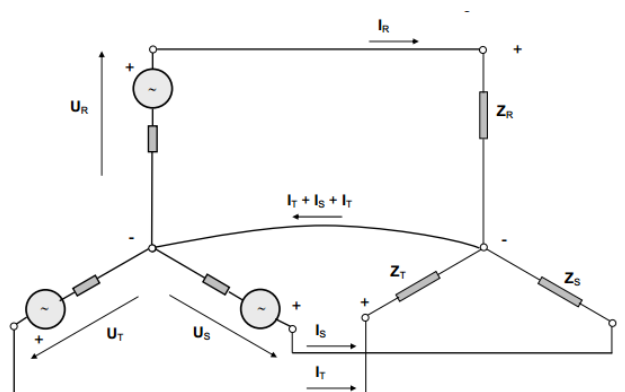
Figura 12 Valores instantáneos de las tensiones. Fuente: Autor.



Fuente: Autor.

En un sistema eléctrico trifásico con carga en Y, se tiene para cada tensión de fase una corriente de fase, estas tres corrientes de fase retornar al generador mediante el terminal de neutro, en el cual las corrientes de fase se suman, el sistema completo se representa en la figura 15. De esta forma un sistema trifásico se representa mediante 7 señales, donde 3 son de tensión y 4 de corriente.

Figura 13 Esquema alimentación de un generador trifásico en Y.

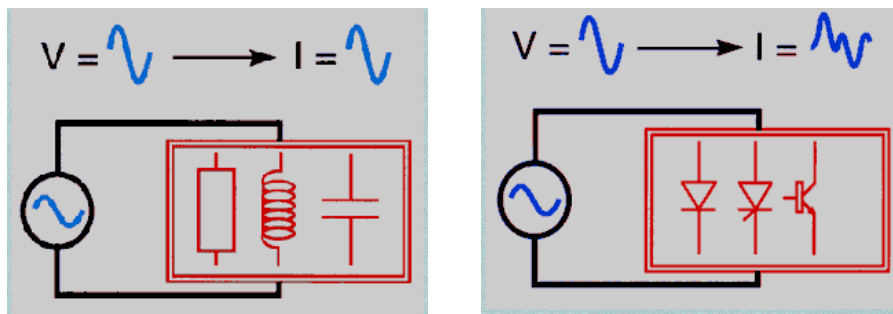


Fuente: sistemas de corriente alterna trifásicos Ing. Julio Álvarez.

1.5. Armónicos en un sistema eléctrico

En general los armónicos son producidos por cargas no lineales que a pesar de ser alimentadas con una tensión senoidal, absorben una intensidad no senoidal. Para simplificar se considera que las cargas no lineales se comportan como fuentes de intensidad que inyectan armónicos en la red. Las cargas armónicas no lineales más comunes son las que se encuentran en los receptores alimentados por electrónica de potencia, tales como variadores de velocidad, rectificadores, convertidores, etc. Otro tipo de cargas que inyectan armónicos son reactancias saturables, equipos de soldadura, hornas de arco, etc. El resto de cargas tienen un comportamiento lineal y no generan armónicos: inductancias, resistencias y condensadores. También se generan armónicos al existir tensiones de alimentación que no son completamente senoidales.

Figura 14 (a) Cargas lineales (b) Cargas no lineales.



(a)

(b)

Fuente: Autor.

Estas componentes indeseadas en un sistema eléctrico por efecto de no linealidades alteran el funcionamiento del sistema, especialmente la medición realizada sobre las señales que presenta este mismo.

El análisis de estas componentes armónicas se realiza mediante una herramienta matemática conocida como series de Fourier, la cual indica que cualquier señal

periódica puede ser descompuesta como una suma infinita de senos y cosenos, cuyas frecuencias son múltiplos de una frecuencia fundamental [2]. La serie de Fourier de la señal $x(t)$ se expresa mediante la ecuación 5:

Ecuación 5. Representación de la señal temporal $X(t)$ en series de Fourier.

$$x(t) = \sum_{k=-\infty}^{\infty} C_k e^{i2\pi k f_0 t}$$

Dónde:

$f_0 = \frac{1}{T_p}$, Frecuencia fundamental.

C_k = Coeficientes de la serie de Fourier compleja de $x(t)$.

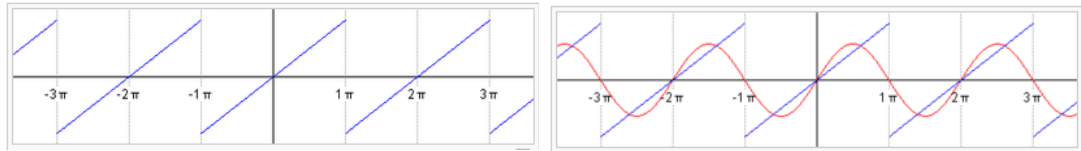
Los coeficientes de la serie de Fourier, se hallan mediante la ecuación 6.

Ecuación 6. Coeficientes de Fourier de la señal en el dominio del tiempo $X(t)$.

$$C_k = \frac{1}{T_p} \int_0^{T_p} x(t) e^{-i2\pi k f_0 t} dt$$

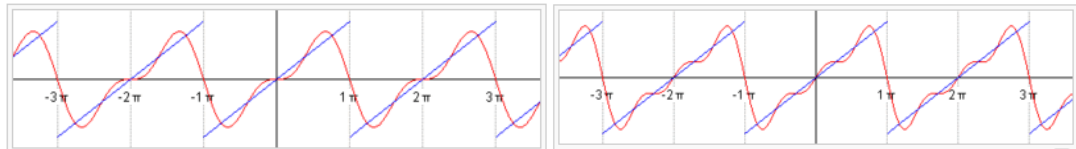
El paso del dominio frecuencial al temporal, mediante series de Fourier, se realiza por medio de la ecuación 5, de la cual se induce que para poder reconstruir la señal original se necesitan todos los armónicos para que la reconstrucción sea perfecta. Como se presenta en la figura 15, entre más armónicos se tengan, la reconstrucción va a ser cada vez más exacta.

Figura 15 (a) Señal Original, (b) 1 armónico, (c) 1+2 armónico, (d) 1+2+3 armónico, (e) 1+2+3+4 armónico, (f) 1+2+3+4+5 armónico. Fuente: Wikipedia series de Fourier.



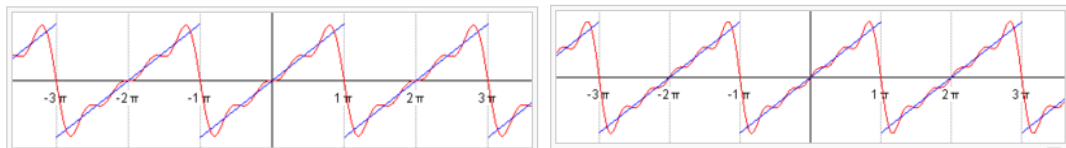
(a)

(b)



(c)

(d)



(e)

(f)

Fuente: Autor.

2. DISEÑO DEL SISTEMA

En este capítulo, se abordara el tema del diseño y selección de los componentes necesarios para cada etapa del proyecto.

2.1. Hardware

Los esquemas implementados para el desarrollo del proyecto se presentan en los anexos E y F, donde el primero, es el que presenta los componentes necesarios para la etapa de control del sistema completo.

El anexo F, presenta la circuitería necesaria para hacer funcionar un canal de conversión digital analógica, los 6 canales restantes poseen los mismos componentes.

2.1.1. Dispositivo programable

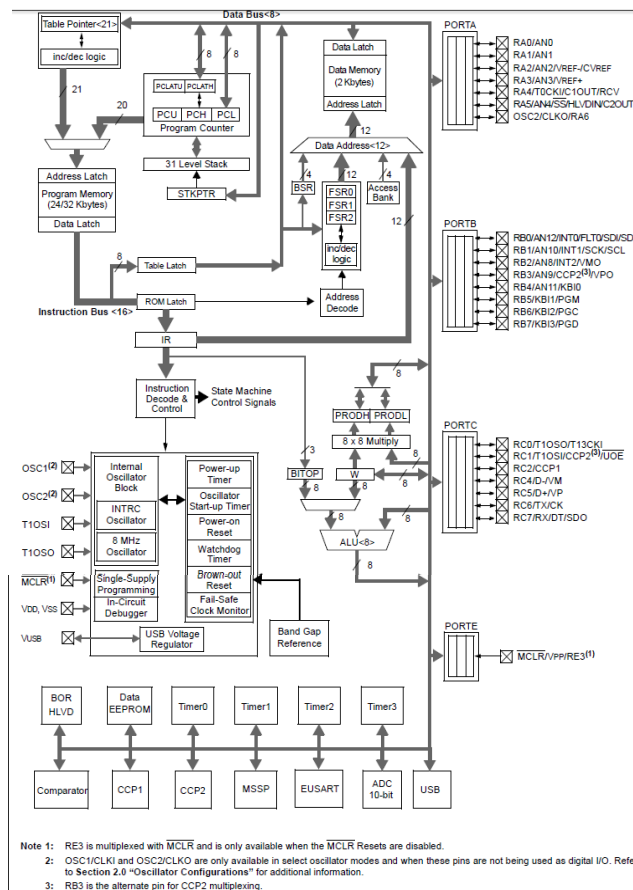
Como dispositivo programable se decidió utilizar un microcontrolador, debido a que la capacidad de cálculo del sistema no es tan relevante, pues la generación de señales no se realiza en tiempo real, más si lo es, el control del sistema en sí y el manejo de las diferentes interfaces de usuario con el mismo, lo que hace que mediante este, se alcancen los resultados esperados de la manera más fácil.

Para esto se utilizó el microcontrolador de la casa Microchip PIC18F4550 cuya arquitectura de muestra en la figura 18, el cual cuenta con las siguientes características:

- Frecuencia de operación de hasta 48 Mhz con velocidad de CPU de hasta 12MIPS.
- Memoria de programa FLASH de 32 Kbytes.
- Memoria ram de 2 Kbytes.

- Puerto de comunicación USART, SPI e I2C.
- 35 GPIO.
- Multiplicador por hardware 8x8.
- Voltaje de operación entre 2[V] y 5[V].
- Comunicación USB de baja velocidad (1.5Mb/s) y alta velocidad (12 Mb/s).
- 10 canales ADC de 10 bits.
- 2 temporizadores de 16 bits y 1 de 8 bits.

Figura 16 Arquitectura PIC18F4550.



Fuente Anexo [A]

Mediante este dispositivo, se manejó la comunicación USB con el computador directamente, sin necesidad de componentes externos; la generación de las señales, además de administrar los datos en una memoria externa que es

manejada mediante el protocolo serial SPI, el manejo de una pantalla de cristal líquido para la visualización y un teclado para la interfaz de usuario.

2.1.2. Memoria externa

Como memoria externa, se necesita una que sea de tipo no volátil, de tal manera que los datos de las señales que sean almacenados en estas, estén disponibles las veces que se requieran, así el sistema quede sin energía.

Entre los tipos de memorias con posible uso para este trabajo se encuentran las EEPROM y las FLASH, con sus respectivos protocolos de comunicación seriales.

Para este proyecto se necesita una capacidad de almacenamiento de aproximadamente 5 kbytes, con una lectura de datos lo según la capacidad del microcontrolador (aproximadamente 10 Mhz). Por esto, se decide trabajar con una memoria tipo EEPROM, pues estas se consiguen de capacidades mucho menores que las flash (del orden de Mbytes), lo que indica menos bytes para el direccionamiento de un dato. Como protocolo de comunicación se tiene como opción el I2C y el SPI pues el microcontrolador tiene posee estos protocolos como periféricos; se decide por este segundo ya que alcanza mayores velocidades.

Para el sistema, se decidió trabajar con una memoria EEPROM que maneja el protocolo SPI, la 25LC256, la cual presenta entre sus características principales:

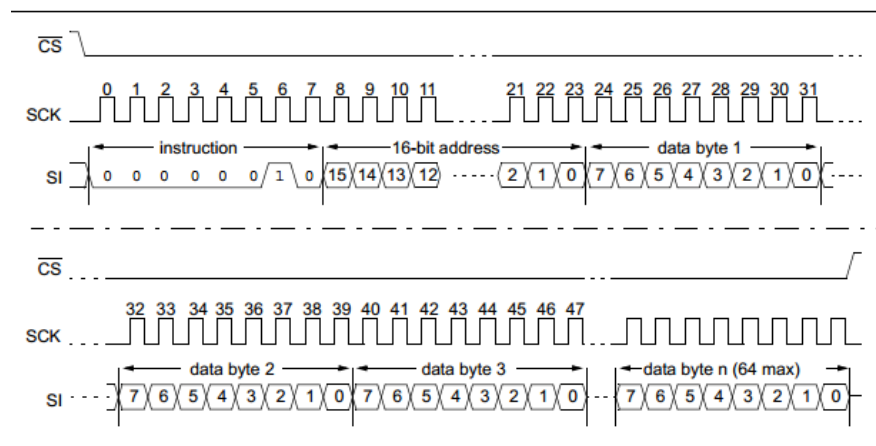
- Máximo reloj 10 Mhz.
- Organización de 32768x8 bits.
- Páginas de 64 bytes.
- Tiempo de borrado/escritura máximo 5 [ms].
- 1.000.000 de ciclos borrado/escritura.
- Retención >200 años.

- Protección EDS⁹ >400 [V].

El control de la memoria se realiza mediante 3 bytes de forma serial en la línea de datos, donde se tiene que el primero indica la operación que se quiere realizar y los dos siguientes dicen la dirección a la que se quiere acceder.

La escritura de este tipo de memoria se realiza según la trama de datos que se muestra en la figura 19, donde se tiene que esta se realiza por bloques de máximo 64 bytes en una secuencia, y el tiempo entre escritura puede ser de hasta 5 [ms].

Figura 17 Trama de datos Escritura EEPROM.

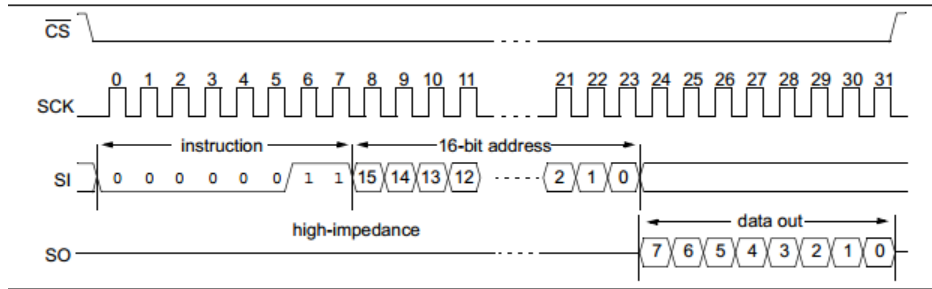


Fuente: Anexo [B]

La lectura, se realiza según lo indica la figura 20 , donde se tiene que después de decirle a la memoria que lea, esta lo sigue ejecutando continuamente hasta que la señal de habilitación vuelva a alto, haciendo esto posible que se pueda leer la cantidad de datos completos que tenga la memoria de ser así requerido.

⁹ Electrostatic Discharge.

Figura 18 Trama de datos Lectura EEPROM.



Fuente: Anexo [B]

2.1.3. Interfaz de usuario local

La interfaz de usuario comprende los componentes mediante los cuales el usuario interactúa directamente con el sistema, esto comprende la visualización e ingreso de datos de manera local.

2.1.3.1. Visualización

La visualización del sistema será realizado mediante una pantalla de cristal líquido grafica con retroiluminación LED, QY12864 que se muestra e la figura 4, la cual es manejada por un chipset KS0108B.

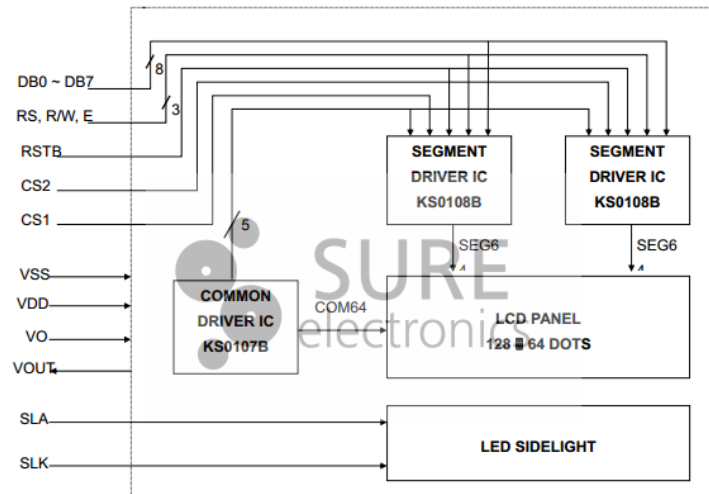
Figura 19 Pantalla Graficadora QY12864.



Fuente: www.miniintebox.com

Esta pantalla maneja una resolución de 128x64 pixeles y es monocromática, la arquitectura de la pantalla se muestra en la figura 22, donde se tiene que para el control son necesarias 8 líneas de datos (DB0-DB7) y 6 de control (RS, RW, E, RSTB, CS1, CS2), que son manejadas por GPIOs del microcontrolador, los demás pines son de alimentación y manejo del brillo y contraste. Su arquitectura se muestra en la figura 5.

Figura 20. Arquitectura pantalla grafica QY12864.

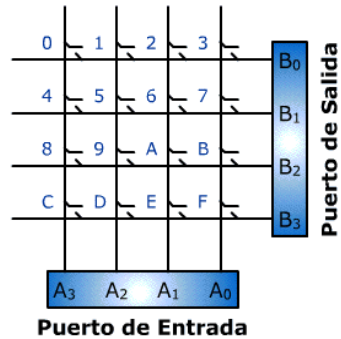


Fuente: www.sureelectronics.com

2.1.3.2. Ingreso de datos

Para el ingreso de datos, se decidió manejar otro tipo de teclado al matricial, diferente al tradicional que se muestra en la figura 21, debido a que este hace necesario la utilización de más puertos E/S del microcontrolador.

Figura 21 Teclado Matricial.



Fuente: www.virtual.unal.edu.co

Mediante el teclado que se muestra en el esquema de la figura 22, se tiene un circuito que entrega un nivel de tensión vertical y horizontal mediante el divisor de tensión que trabaja en función al pulsador que se active , de esta manera se tendrán las siguientes tensiones según la fila y columna seleccionada;

Ecuación 7. Tensión Horizontal Teclado.

$$V(x) = VCC * \frac{1000}{1000 * nC^{10}}$$

Ecuación 8. Tensión Vertical teclado.

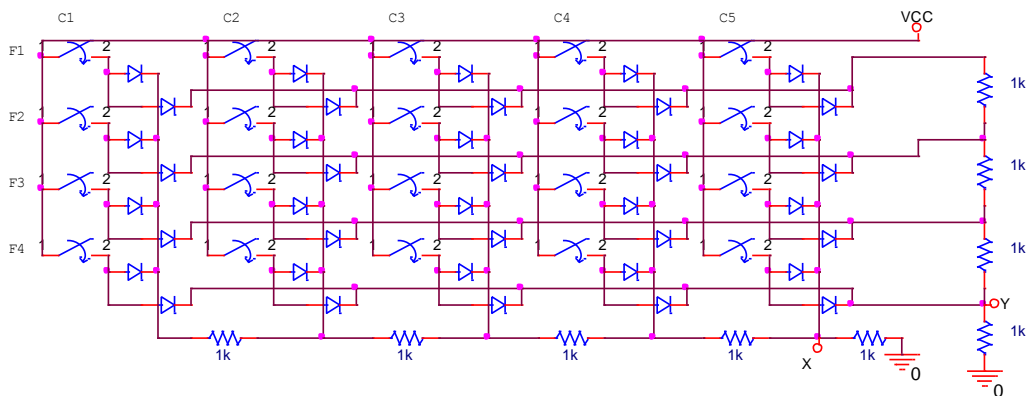
$$V(y) = VCC * \frac{1000}{1000 * nF^{11}}$$

Este método hace necesario la utilización de dos canales del conversor analógico digital para procesar la información de las coordenadas del pulsador presionado y según saber que tecla fue. Este conversor no necesita de una gran exactitud, por lo cual se utiliza el conversor analógico digital del microcontrolador con una resolución de 8 bits.

¹⁰ Numero de columna.

¹¹ Numero de fila.

Figura 22 Teclado Analógico.

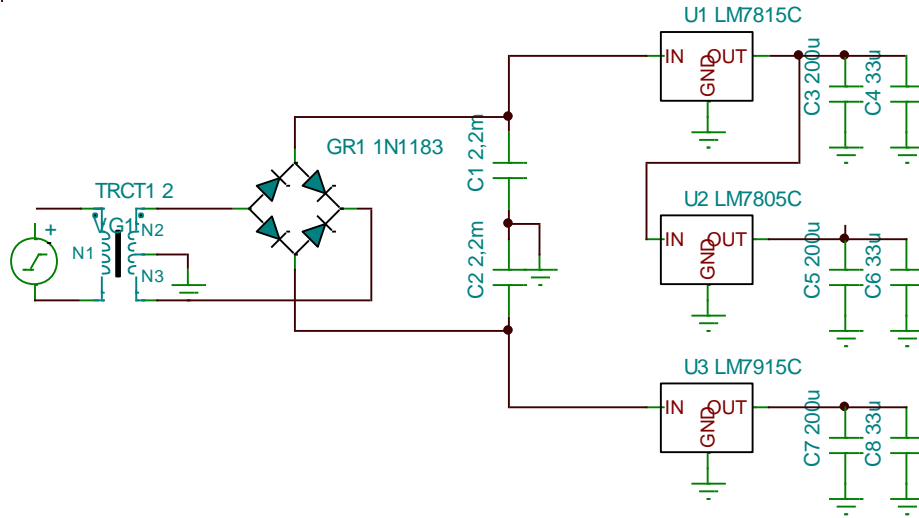


Fuente: Autor.

2.1.4. Alimentación del sistema

Para la alimentación del sistema se decide diseñar una fuente sencilla debido a que los requerimientos de potencia no son altos, más si lo es buscar una buena estabilidad. Por ello, se deciden utilizar reguladores lineales para establecer los niveles de salida necesarios, que son de 15[V], -15[V] y 5[V], el esquema queda como se muestra en la figura 25, donde se tiene que los condensadores C1, C2, C3, C5 y C7 son electrolíticos, los otros son cerámicos. Estos reguladores, presentan buena regulación de carga y de línea para valores bajos de corriente, siendo estas de aproximadamente 5mV y 4mV respectivamente, para cargas inferiores a 500 [mA], además de poseer un ruido en la salida de aproximadamente 100 [μ V].

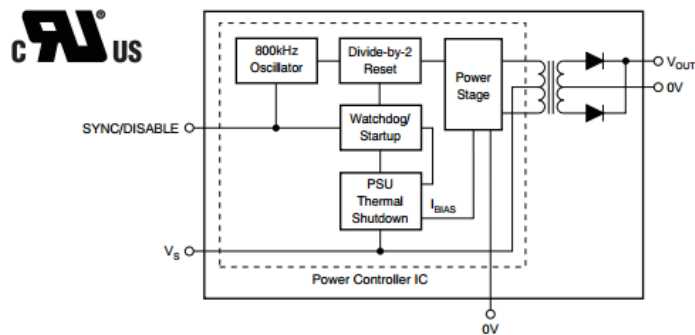
Figura 23. Fuente de alimentación.



Fuente: Autor.

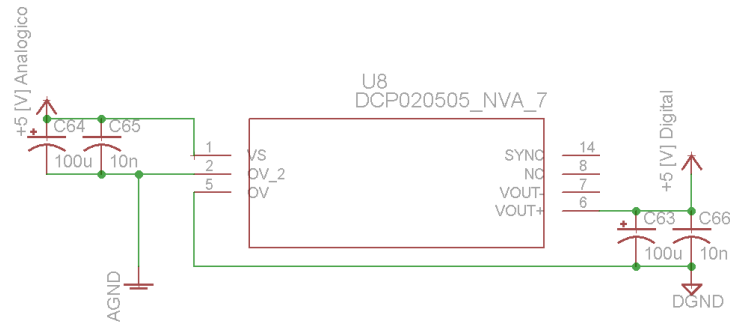
Para mejor estabilidad del sistema analógico, se opta por trabajar con la alimentación del sistema digital aislada del sistema analógico, por ello se hace uso del integrado DCP020505, mediante el cual al tener una entrada de 5[V], genera una misma tensión totalmente aislada, este dispositivo es capaz de manejar una potencia de 2 [W] con una eficiencia del 89%.

Figura 24. Diagrama integrado DCP020505.



Fuente: Anexo B.

Figura 25. Circuito DCP020505. Fuente: Autor.



Fuente: Autor.

2.1.5. Generador de señales

Para la generación de las señales, se trabajara mediante un conversor digital analógico, al cual se le entregan código digitales y los convierte en una señal analógica, la cual es reconstruida mediante muestras a una frecuencia específica. Esta etapa además del conversor necesita de otros componentes para su perfecto funcionamiento, como lo es el circuito que entrega la tensión de referencia a este y un buffer que ayuda a estabilizarla, además como a la salida del sistema se tiene una señal reconstruida en muestras discretas, de hace necesario realizar de una etapa de filtrado para eliminar componentes frecuenciales no deseadas, que en este caso serán frecuencias mayores a 3 [Khz].

2.1.5.1. Conversor digital analógico

Dado que se busca obtener una alta resolución en la salida para tener una mejor calidad de la señal generada, se decidió trabajar con una resolución de 16 bits como primer parámetro de la selección del dispositivo. Además en el mercado se encuentran DAC de varios canales, en los cuales se realiza un multiplexado para obtener la salida requerida, pero este procedimiento hace que la salida de datos

sea más lenta y que se produzca un pequeño desfase entre ellas, por esto, se decidió implementar 7 DACS de un solo canal.

A continuación, se presenta una tabla comparativa de algunos DAC'S de varias casas fabricantes, donde se analizan las características más importantes pertinentes para el desarrollo del este proyecto, como base, se tomaran las peores valores de cada elemento y no los valores típicos

Tabla 2. Comparación DACS. Fuente: Autor.

P.N	Bits	INL (LSB)	Vout	SETTLING TIME [μs]	BIPOLAR ZERO ERROR [LSB]	OUTPUT NOISE [nV/\sqrt{Hz}]	DIGITAL FEEDTHROUGH [nV-s]	PRECIO [US]
DAC8 871	16	+/-1	+/-18	1	1	10nV	0.2	9
MAX5 216	16	+/-1.2	5	14	1	73	0.5	9.92
LTC16 55	16	+/-8	5	20	3	280	0.3	10.3
DAC8 580	16	+/-1	+/-5	0.65	5	25	0.5	5.6

Fuente: Autor.

Según las necesidades presentes, se decide utilizar el DAC8871, pues tiene entre sus principales características distintivas una mayor excursión de salida, que permite cumplir la necesidad de un nivel de DC de hasta 5[V], logrando tener una salida de una tensión mayor a esta, la cual será para este trabajo de +/-10[V], que se puede lograr mediante el DAC8871 sin necesidad de una acondicionamiento de señal, que sería necesario si el dispositivo no tuviera una salida dentro de estos valores, además el DAC8871 presenta una arquitectura *resistor ladder* que se muestra en el anexo C.

Con una excursión de salida de +/-10, el sistema presenta la siguiente resolución

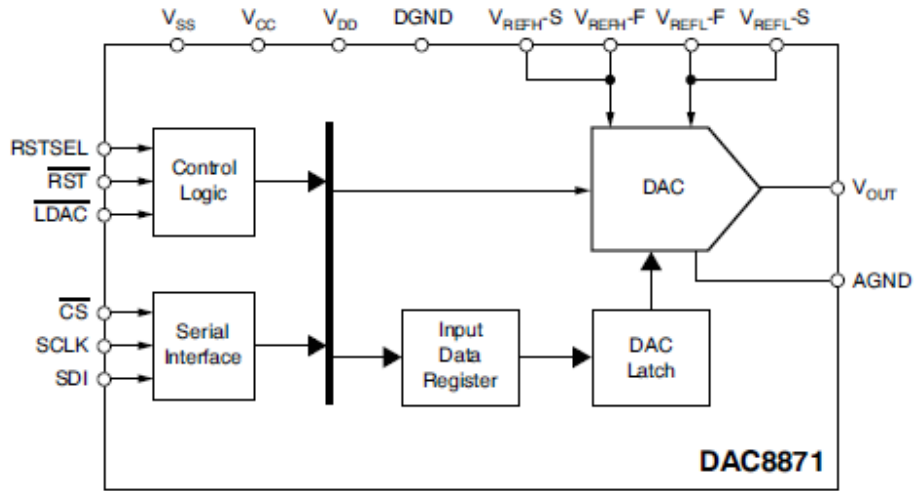
$$V_{lsb} = \frac{10 - (-10)}{2^{16}} = 305.15 [\mu V]$$

El DAC8871 se maneja mediante el protocolo serial SPI de 16 bits, por lo cual se hace necesario de solamente 3 líneas de datos para su manejo que son SDI, /CS y SCK. La salida requerida se selecciona mediante una trama de 2 bytes, que representa un código de 16 bits, mediante este se tiene una tensión según la siguiente expresión:

$$V_{out} = \frac{V_{refH} - V_{refL}}{65535} * \text{codigo} + V_{refL}$$

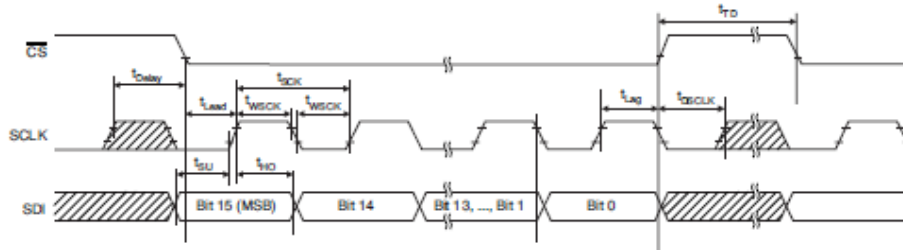
Para este caso, se tiene que $V_{refH} = 10 [V]$ y $V_{refL} = -10 [V]$.

Figura 26 Arquitectura DAC8871.



Fuente: Anexo D.

Figura 27 Trama escritura DAC8871.



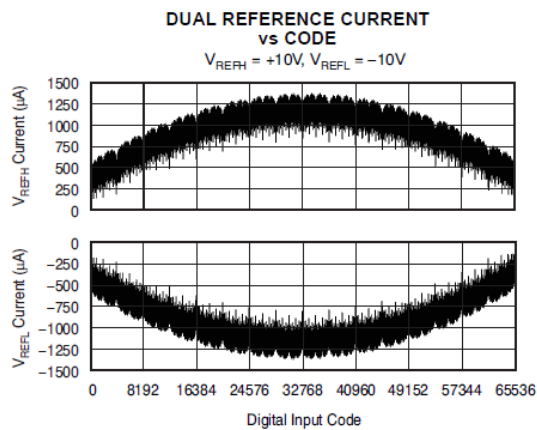
Fuente: Anexo D.

2.1.5.1.1. Circuito de compensación de referencia

Para este convertor Digital Analógico, las entradas V_{refH} y V_{refL} , se comportan como una carga variable según el código, como lo indica la figura 30, esto hace necesario que la fuente de referencia pueda soportar estas variaciones de carga

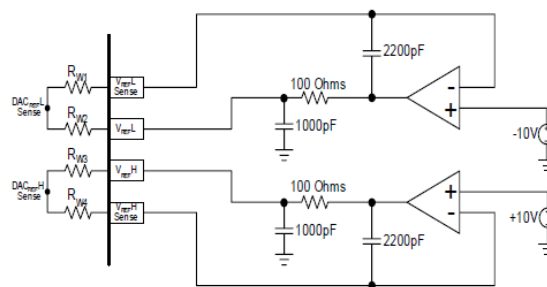
sin que afecte el nivel de tensión y esta a su vez pueda interferir en la precisión de la salida. El DAC8871 cuenta con una de conexión de referencia (force) y de sensado (sense) que minimiza los errores internos provocados por la corriente de referencia y su impedancia. La solución a este inconveniente es analizada [3] y plantean como solución el esquema que se muestra en la figura 31, el cual trabaja como un buffer para la referencia de tensión, haciendo más estable la tensión que aparecen en los terminales VrefH y VrefL.

Figura 28 Corriente referencia vs código digital.



Fuente: Anexo D.

Figura 29 Configuración buffer fuente dual +/- 10[V].



Fuente: PARGUAIN, Joselito Building a Stable DAC External Reference Circuit SLAA172 marzo 2003.

2.1.5.2. Circuito de referencia de tensión

Para el desempeño del sistema de conversión de datos además de tener en cuenta las características del convertor digital analógico, también se deben tener en cuenta que este trabaja con una tensión de referencia, la cual debe presentar ciertas características como bajo ruido y estabilidad para que el dato que se tenga a la salida, sea tal cual se necesita, debido a esto la selección del circuito de referencia es tan importante como la selección del mismo convertor.

El sistema necesitara de dos referencias de tensión, una con un valor de 10[V] y otra de -10[V], para lo cual, se buscara solamente la de 10[V], y con un circuito adicional, se obtendrá la de -10[V].

Tabla 3. Comparación referencias de tensión.

	Error Inicial	Tempco [ppm/°C]	Estabilidad a largo plazo [ppm/1000h]	Ruido salida (0.1-10Hz) [μ Vp-p]	Regulación de línea [ppm/V]	Precio
REF02	0.2%	15	50	10	60	5
REF5010	0.05%	2.5	5	30	1	3
MAX6173	0.05%	3	50	18	10	6.3
LTC1021	0.05%	5	7	4	10	6

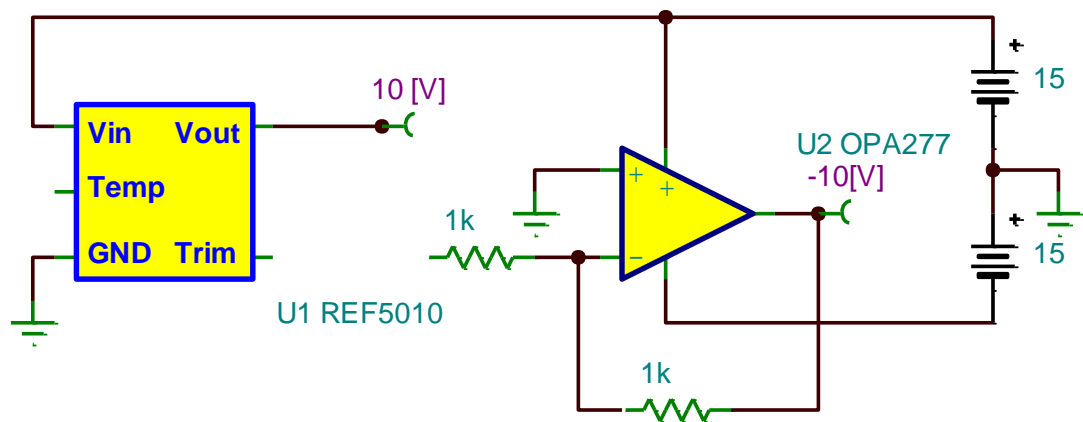
Fuente: Autor.

Como se tiene según estos los parámetros presentados en la tabla anterior, se utilizara el REF5010 como generador de la señal de referencia de tensión.

Ahora para sacar la tensión de referencia negativa, se utilizara un inversor con ganancia unitaria (ver ANEXO D). Para este caso se hace necesario de un amplificador operacional de precisión que tenga un entre sus características principales bajo ruido y nivel *offset* de salida, este debe ser menor que la resolución que maneja convertor para que este parámetro no modifique la salida. Según estas indicaciones, se realizó una búsqueda correspondiente en diferentes

casas fabricantes mediante sus bases de datos, y se seleccionó el amplificador operacional OPA277, el cual posee un *offset* de máximo 20[μ V], una variación de ± 0.15 [μ V/ $^{\circ}$ C] y un nivel de ruido de 8 [n V/ \sqrt{Hz}], cabe resaltar que el nivel de *offset* es 30 veces menor a la resolución del circuito. El circuito a implementar queda como se muestra en la figura 32, para el sistema, se van a implementar tres etapas de estas, de modo que a dos les corresponda la carga a cada una le corresponda la carga de 2 conversores digital analógico, la tercera etapa, tendrá a su cargo los restantes 3 conversores.

Figura 30. Referencia positiva y negativa.



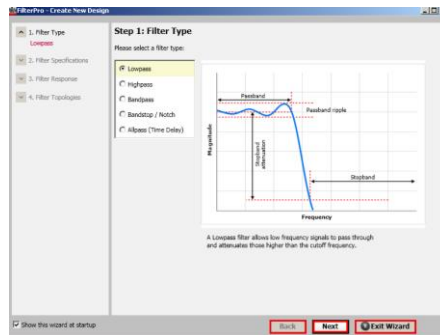
Fuente: Autor.

2.1.5.3. Etapa de filtrado

Para la elaboración de la etapa de filtrado, se utilizó la herramienta de software FILTER PRO de la empresa Texas instrument, esta herramienta, facilita el diseño de filtros activos mediante una interfaz amigable y eficiente.

Para el diseño, primero se selecciona el tipo de filtro que se necesita, para este caso, será un pasa bajas.

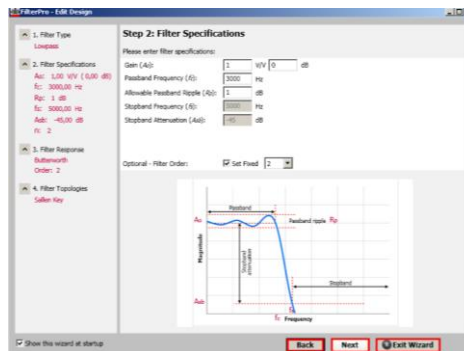
Figura 31 Selección tipo de filtro.



Fuente: Autor.

Luego se selecciona los parámetros de este, como la ganancia, la frecuencia de corte y el máximo rizado permitido, además permite seleccionar el orden del filtro que se quiera utilizar, en nuestro caso, tendremos una frecuencia de corte de 3000 [Khz], y un orden de 2.

Figura 32 Parámetros Filtro activo.



Fuente: Autor.

Posteriormente el halla los diferentes tipos de filtros, que pueden ser *bessel*, *chevichev*, *butterworth* entre otros que se puedan utilizar, mostrando sus respectivas respuestas tanto en magnitud como fase. Se selecciona el filtro tipo *butterworth*.

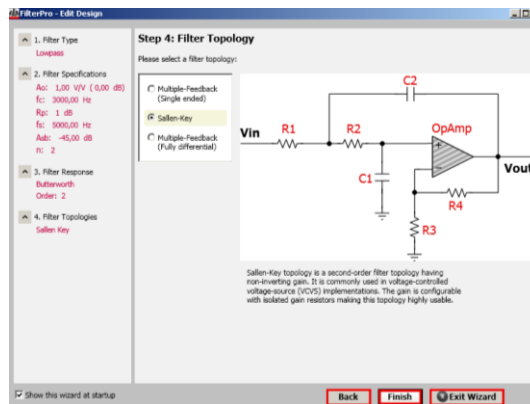
Figura 33. Selección Filtro.



Fuente: Autor.

Seguido a esto, se muestran las posibles topologías a implementar, el presenta 3, *Multiple-Feedback(single-ended)*, *sallen-key* y *Multiple-feedback(Fully differential)* todas presentan el mismo comportamiento. Se utilizara la *Sallen-key* pues es la que requiere de menos componentes para su implementación.

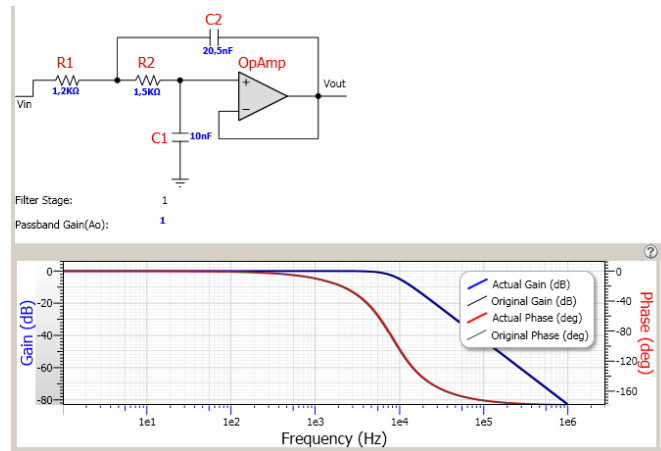
Figura 34 Selección topología filtro activo.



Fuente: Autor.

Finalmente se tiene el resultado mediante un circuito que presenta la topología a implementar con su respectivo comportamiento.

Figura 35 Filtro Activo más respuesta Magnitud y fase.



Fuente: Autor.

Como elemento activo del filtro, se decide utilizar el amplificador operacional OPA277, por las características ya mencionadas, además de tener un ancho de banda de 1 [Mhz].

A la hora de implementar se decide buscar los valores normalizados de los componentes que más se acerquen a estos valores, siendo estos:

Tabla 4. Reemplazos de componentes filtro activo calculado.

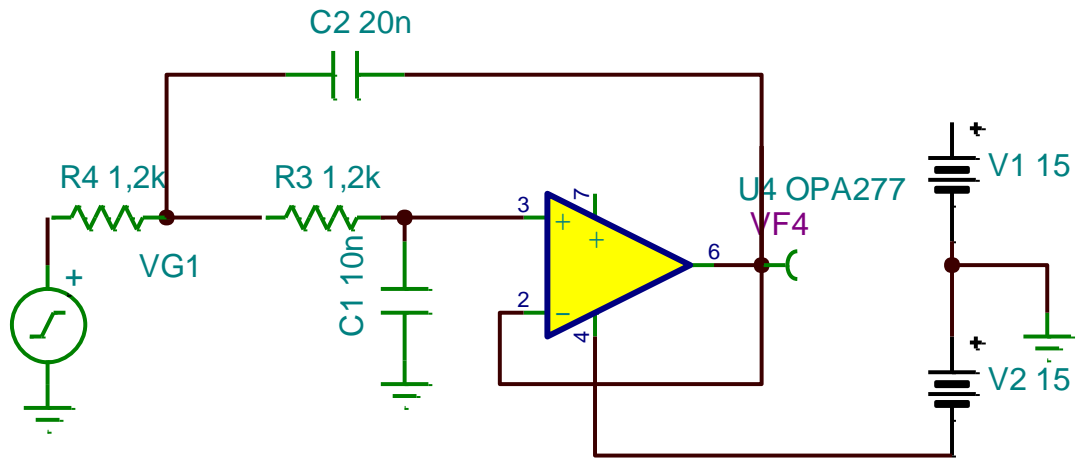
Componente	Valor Requerido	Valor normalizado
R1	1.2 [KΩ]	1.2 [KΩ]
R2	1.5 [KΩ]	1.2[KΩ]
C1	10 [nF]	10[nF]
C2	20.5 [nF]	20[nF]

Fuente: Autor.

Con estos nuevos valores se procede a realizar la simulación del filtro, mediante la herramienta de software TINA¹², en su versión gratuita. La simulación se realiza mediante el esquema presente en la figura 38.

¹² <http://www.ti.com/tool/tina-ti>

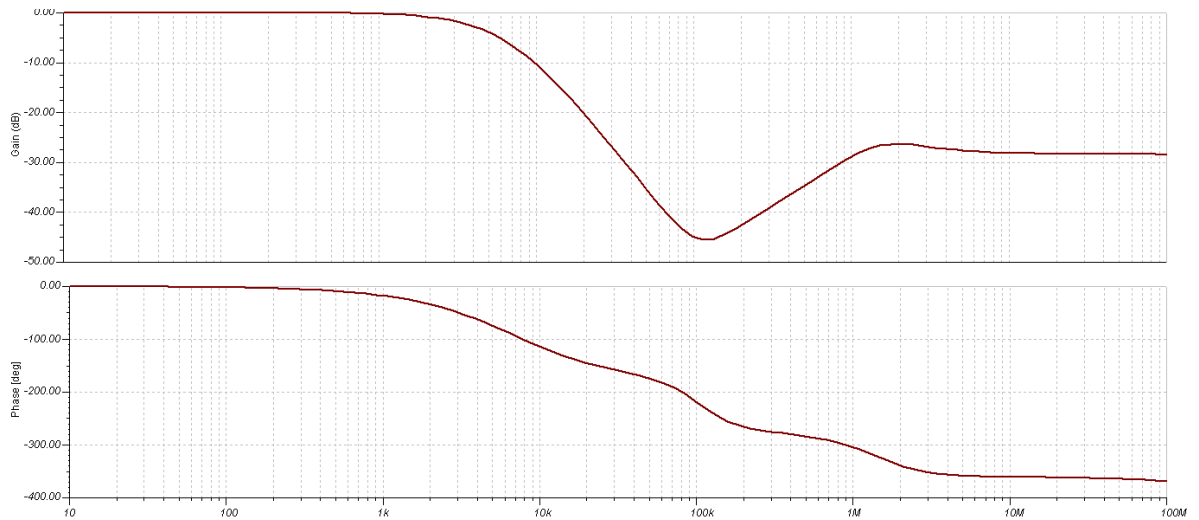
Figura 36 Filtro activo con valores normalizados.



Fuente: Autor.

La respuesta tanto en magnitud como en frecuencia, se presenta en la figura 39, donde se tiene que al realizar las modificaciones mencionadas, la respuesta no cambia drásticamente, por lo cual se decide realizar los cambios.

Figura 37 Respuesta filtro activo valores normalizados.



Fuente: Autor.

La etapa de filtrado, además de suavizar la salida del sistema reconstruida al acotar el ancho de banda, también ejerce una muy importante función al eliminar las señales de glitch, que afectan negativamente en la calidad de la señal generada.

2.2. Software

A continuación se presentan las componentes lógicas del sistema para su funcionamiento; el diagrama de flujo total del sistema, se presenta en el anexo E.

2.2.1. Creación de señales

2.2.1.1. Frecuencia de muestreo

Para la generación de las señales, en primera instancia se debe definir la frecuencia de muestreo a la cual se deben reconstruir las señales, estos análisis se realizan en base a la frecuencia máxima presente en el sistema, que en este caso será de 3 [Khz].

Según el criterio de Nyquist, una señal debe ser reconstruida como mínimo a una frecuencia que corresponda con la siguiente relación:

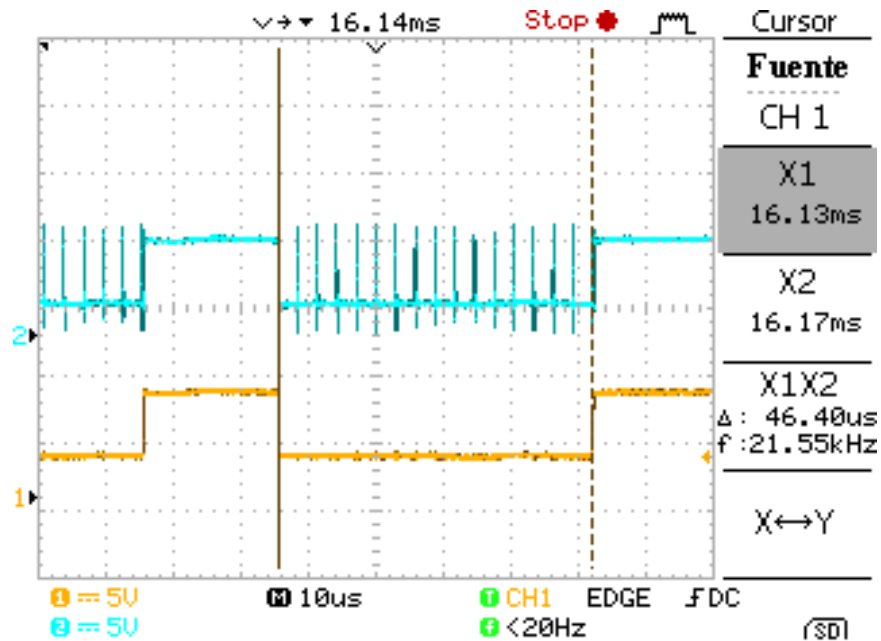
$$F_{muestreo} > 2 * F_{maxima}$$

Donde se tiene que entre mayor sea la frecuencia de muestreo, mejor será la calidad de la señal reconstruida.

La selección de esta frecuencia de muestreo está dada por ciertas características temporales del sistema, como lo son la frecuencia del microcontrolador, el tiempo de acceso a la memoria de datos y el tiempo de escritura de datos en el DAC.

Para establecer este límite, se realiza una prueba, donde mediante el osciloscopio se verifica el tiempo que tarda el sistema en obtener una muestra almacenada en memoria y convertirla en su correspondiente nivel de tensión, tal medición se presenta en la figura 40.

Figura 38. Señales de control DAC.



Fuente: Autor.

En la anterior figura, se tiene que el sistema genera una muestra en un tiempo de aproximadamente 47 [μs] que representaría una generación de frecuencia de aproximadamente 21 [Khz].

Además de la generación se ejecutan otras líneas de código para el control del sistema, entre estas se encuentra presente el funcionamiento de un conversor analógico digital, mediante el cual se interactúa con el sistema, este conversor realiza su labor en un tiempo de aproximadamente 8 [μs], por lo cual se tendría un tiempo de funcionamiento de aproximadamente 55 [μs]. Este tiempo es el mínimo en realizar un ciclo de trabajo de generación de una muestra de la señal.

Para obtener un número entero de muestras que reconstruyan la señal de 3000 Khz en vez de recrear una señal lo más periódica posible, se busca una frecuencia que se ajuste a nuestras exigencias, para lo cual se analizan las opciones presentadas en la siguiente tabla:

Tabla 5. Posibles frecuencias de muestreo.

Muestras	Periodo [μ s]	Frecuencia [Khz]
5	66.66	15
6	55.55	18
7	47.61	21

Fuente: Autor.

Según las características de nuestro sistema, la opción de 7 muestras se debe descartar, pues este tiempo es muy corto para poder generar una muestra.

La opción de 6 muestras, sería posible implementarla si no estuviera tan ajustada al tiempo máximo permitido.

Se tiene que la opción de 5 muestras con una frecuencia de 15[Khz] es la óptima, pues le da al sistema una holgura que no es inconveniente y que permite realizar modificaciones al tener más espacio para alguna línea de código requerida.

2.2.2. Calculo, almacenamiento y generación de las señales

Para la generación de señales, se debe definir la cantidad de muestras presentes en la señal de frecuencia fundamental según la tasa de muestreo. El número de muestras se halla mediante la siguiente ecuación:

$$\frac{1}{\frac{F_o}{1}} = \frac{1}{\frac{60}{\frac{Fm}{Nm}}} = 250$$

Según el número de muestras, se puede saber la resolución que tendrá el sistema en su frecuencia fundamental, esta será de:

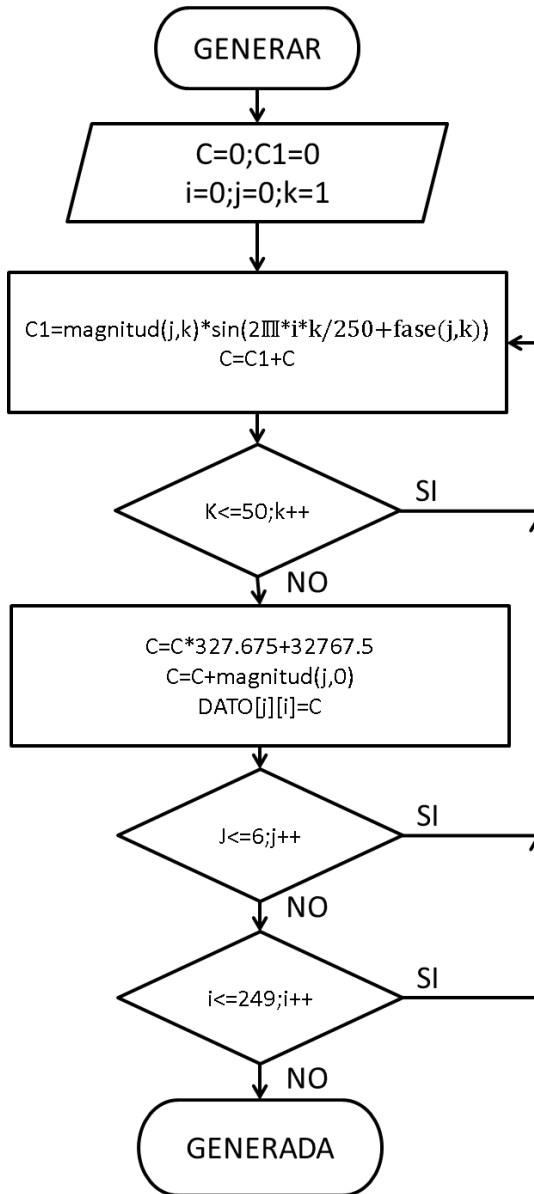
$$Resolucion^{\circ} = \frac{250}{360^{\circ}} = 0.69^{\circ}$$

Se tiene que el sistema permite la generación de señales desde el computador o directamente en el micro, el algoritmo de generación es el mismo para los dos sistemas, lo que varía entre el uno y el otro es la forma de almacenamiento y el destino inmediato.

Para la generación, se necesita de una matriz de magnitudes y de fases, de dimensiones 7×51 , donde el valor de 7 representan las 7 señales y 51, los 50 armónicos pertenecientes a cada señal, más un parámetro de nivel de tensión promedio.

El algoritmo de generación se representa mediante el diagrama de flujo representando en la figura 41.

Figura 39. Diagrama de flujo generación señal.

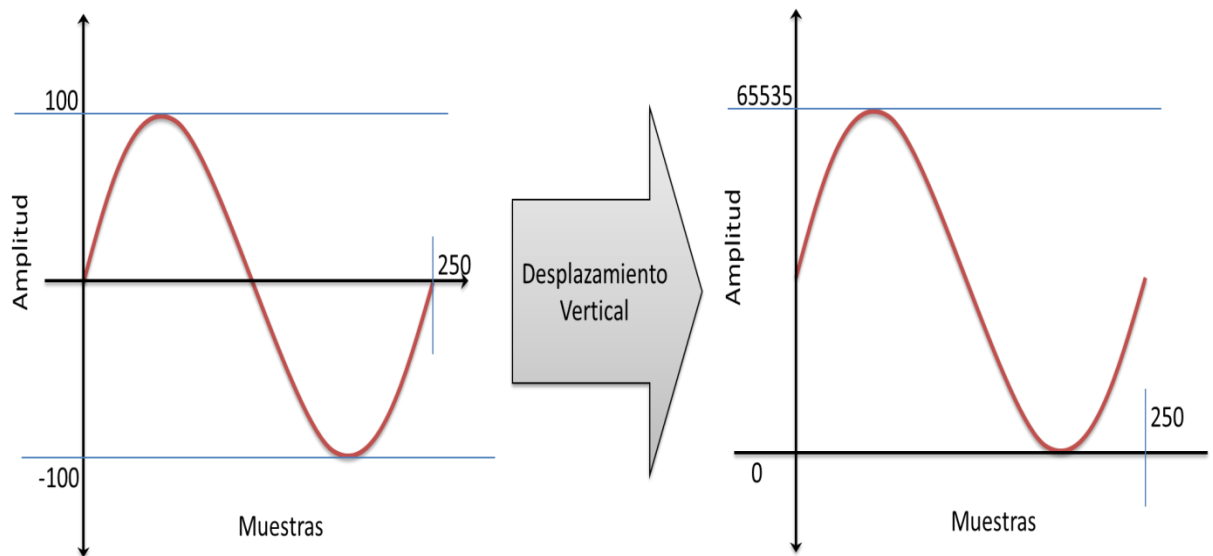


Fuente: Autor.

Los valores se calculan para una magnitud máxima de 100, teniendo como resultado un valor entre -100 y 100 continuo, donde estos límites, son los establecidos, por la tensión de referencia baja y alta del conversor digital analógico, para el cual, estos mismo se representan mediante los códigos digitales 0 y 65535 (0x0000 y 0xFFFF), por esta razón se necesita realizar una traslación

de magnitud para poder representar estos valores, y una digitalización de los valores continuos, asignando cada valor continuo que se representa mediante variables tipo coma flotante, a un a variable tipo entero de 16 bits.

Figura 40. Desplazamiento señal generada.

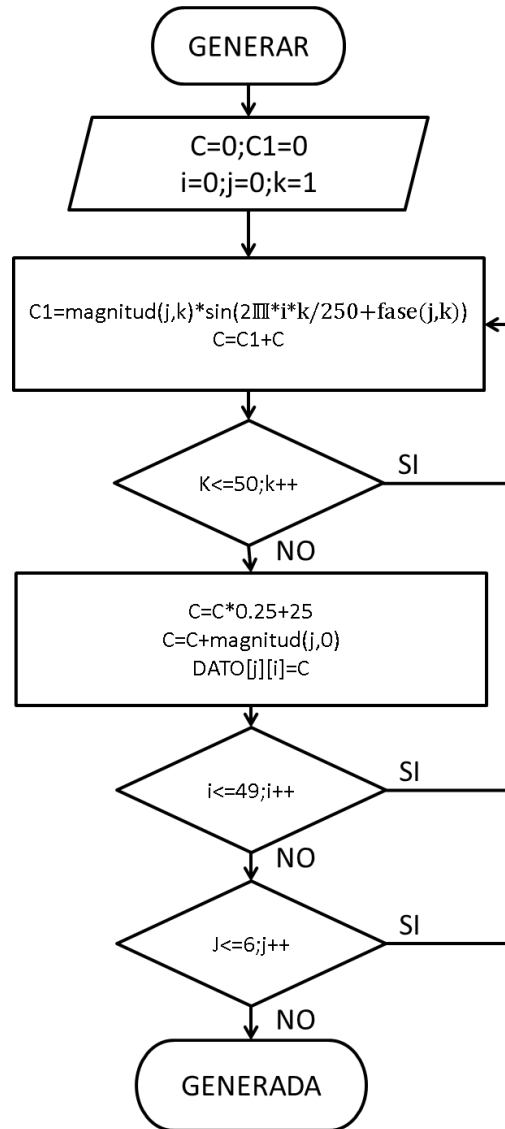


Fuente: Autor.

Una vez se tienen las señales generadas, se procede a su almacenamiento para poder ser leídas y reconstruidas.

De forma similar, se generan los valores de las señales que serán mostradas en la pantalla gráfica, la cual posee una resolución de 128X64, por lo cual se decide trabajar con 100X50 pixeles, ya que se desea mostrar una aproximación de la forma de onda. La generación de estos valores se realiza como lo indica el diagrama de la figura 43, pero en este caso, se tendrán 100 muestras en vez de las 250 presentes en la señal real además de realizar la traslación de nivel de entre -100 y 100 a 0 y 50.

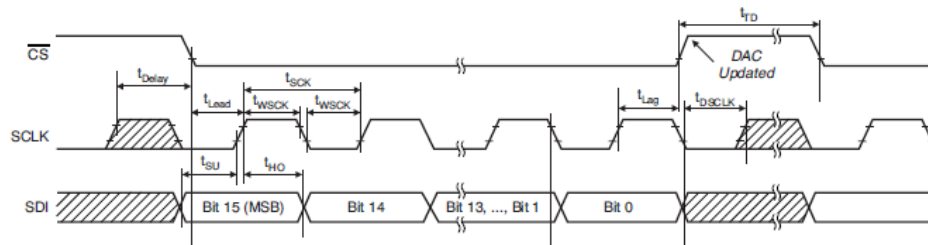
Figura 41. Diagrama de flujo cálculo de graficas de las señales.



Fuente: Autor.

Cada trama de datos en el conversor digital analógico es de la siguiente forma

Figura 42. Trama de datos DAC.



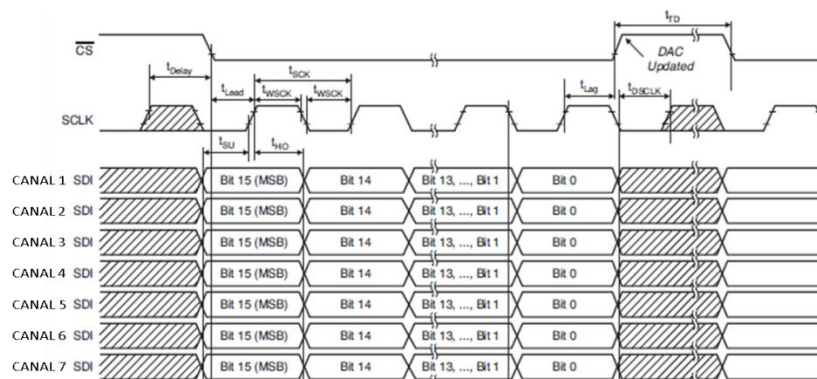
Fuente: Anexo C.

Donde cada valor que se quiera sacar, se tiene que enviar de forma serial a acompañado de dos señales de control.

Es necesario realizar un multiplexado de las 7 señales para poder visualizarlas de forma simultáneas, teniendo en cuenta, que todas deben estar perfectamente sincronizadas para no tener efectos de desfases entre las señales de salida.

Para dar solución a este problema, se decide mediante una sola trama de datos, enviar las 7 señales, tal como lo muestra la figura 45, donde a cada DAC le llegan las señales de control /CS y SCLK, junto con la señal de datos correspondiente SDn. Al utilizar este método, se optimiza la generación de las señales, pues con una sola trama de datos, se pueden generar las 7 señales.

Figura 43. Trama de datos DAC y canales.



Fuente: Anexo C.

Para la generación de la señal según la trama de datos antes mencionada, se deben acomodar los datos para poder enviarlos al dispositivo de conversión de la forma más cómoda y eficiente.

Teniendo en cuenta que cada muestra es representada por un número de 16 bits, y que son 7 señales, se decide armar un vector de bytes de 16 elementos, en el cual cada elemento contiene los bits de cierta posición de cada dato de todas las señales, este procedimiento se realiza tal como lo describe el siguiente ejemplo.

Se supone que los datos que componen una muestra son los siguientes:

Tabla 6. Ejemplo valores de una muestra.

señal	Dato	Dato binario															
		Decimal	B15	B14	B13	B12	B11	B10	B9	B8	B7	B6	B5	B4	B3	B2	B1
1	52768	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0
2	2185	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	1
3	4506	0	0	0	1	0	0	0	1	1	0	0	1	1	0	1	0
4	9020	0	0	1	0	0	0	1	1	0	0	1	1	1	1	0	0
5	17968	0	1	0	0	0	1	1	0	0	0	1	1	0	0	0	0
6	33824	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0
7	65535	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Fuente: Autor.

El primer byte del vector, será el compuesto por los bits más significativos de todas las muestras, el siguiente, será compuesto por los bits de la posición 14 de todas las muestras, el siguiente por los bits de la posición 13 de todas las muestras y así sucesivamente hasta al bit menos significativo.

Tabla 7. Asignación de valores de una muestra a un vector de bytes.

señal	Dato Decimal	Dato binario															
		B1 5	B1 4	B1 3	B1 2	B1 1	B1 0	B 9	B 8	B 7	B 6	B 5	B 4	B 3	B 2	B 1	B 0
1	52768	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0
2	2185	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	1
3	4506	0	0	0	1	0	0	0	1	1	0	0	1	1	0	1	0
4	9020	0	0	1	0	0	0	1	1	0	0	1	1	1	1	0	0
5	17968	0	1	0	0	0	1	1	0	0	0	1	1	0	0	0	0
6	33824	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0
7	65535	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
		▼	▼	▼	▼	▼	▼	▼	▼	▼	▼	▼	▼	▼	▼	▼	▼
Vector Bytes(Hex)		61	51	48	44	43	71	58	4C	47	41	78	5C	4F	49	44	42

Fuente: Autor.

De esta forma, se representa cada muestra de las 7 señales mediante un vector de bytes, lo que hace que el manejo sea más sencillo por la arquitectura del microcontrolador empleado, y la forma como gestiona los datos la memoria externa, al escribir y leer en ella byte por byte directamente.

Una vez que se tienen los datos como vectores de bytes de 16 elementos, se procede a guardarlos en la memoria de datos externa, en la cual se almacenan desde la posición 0 ocupando cada muestra 16 bytes, de tal manera, que la primera muestra se encuentra desde la posición 0 hasta la 15, la segunda muestra desde la 16 hasta la 31, y así sucesivamente hasta ocupar 4000 posiciones de memoria.

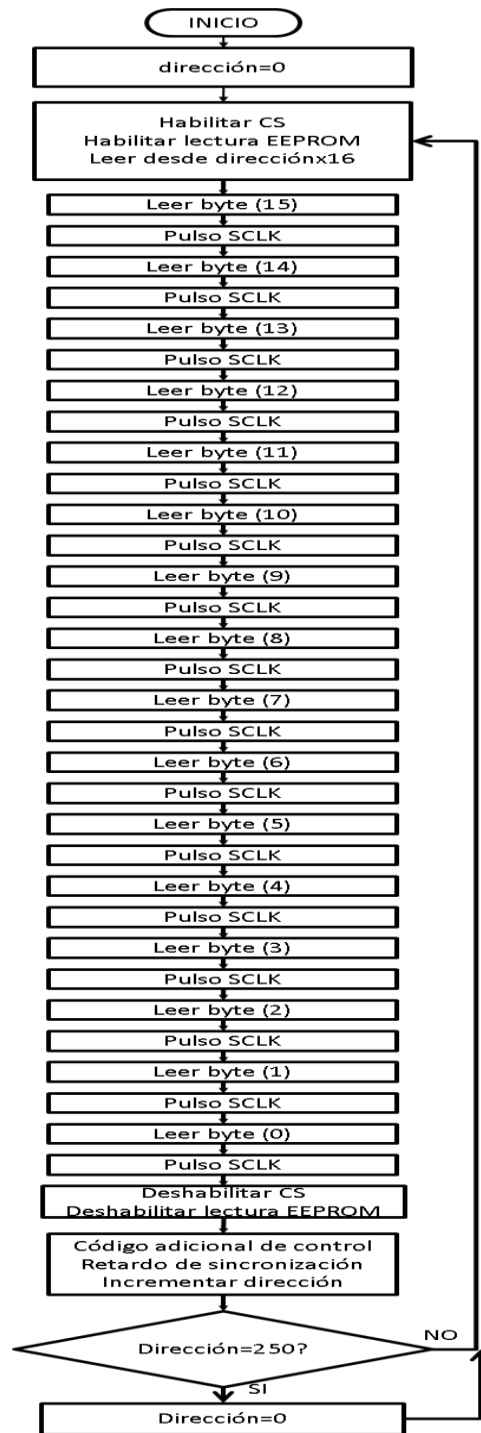
Seguido a estas posiciones de memoria se almacenan los valores correspondientes a las gráficas, que son 100 bytes por cada gráfica, para un total de 700.

De esta manera, se tiene un total de 4700 bytes de información almacenados en la memoria externa. Desde las posiciones 0 a la 4700 (0x000-0x125C).

El proceso de lectura de cada muestra, se realiza según el diagrama de flujo presentado en la figura 46. Donde se tiene que la dirección de lectura es un múltiplo de 16 por un número entre 0 y 249. Luego que se hace un recorrido completo, se tiene que se generó un periodo completo, por lo cual, se inicia de nuevo el ciclo.

Cabe tener en cuenta, que dentro de este ciclo también se tienen las líneas de control de programa, mediante las cuales se configura la adquisición o generación de señales.

Figura 44. Diagrama de flujo, lectura de dato.

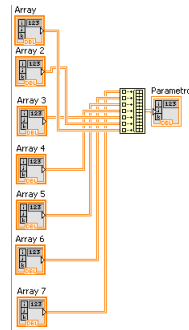


Fuente: Autor.

2.2.3. Generación de señales en LabView

Primero se crea una matriz en las que se van a ingresar los valores de las magnitudes y fases que componen las señales. Cada array¹³ de los 7 que se tienen es de orden 2x50, que representan 50 armónicos donde cada uno tiene magnitud y fase.

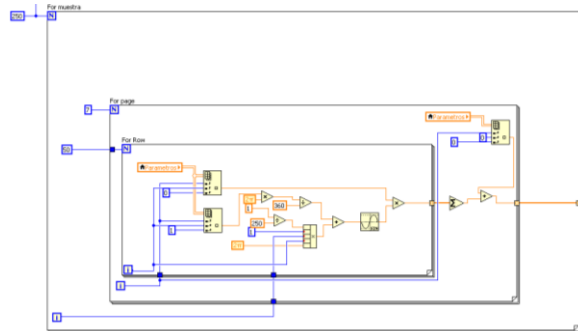
Figura 45. Matriz de parámetros.



Fuente: Autor.

Una vez se tienen los parámetros en la matriz "Parámetros", se procede a realizar el cálculo de las señales mediante el algoritmo que se presenta en la figura 39. Al implementarlo en la herramienta queda tal como lo indica la figura 46, donde en la salida se tiene un valor entre -100 y 100.

Figura 46. Diagrama generación de señales.

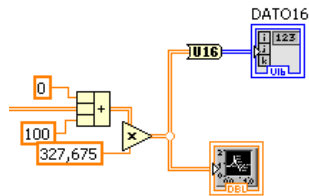


Fuente: Autor.

¹³ Vector o arreglo de vectores.

A estos valores se les tiene que realizar el corrimiento que se muestra en la figura 40, que se hace mediante el diagrama que se tiene en la figura 47, donde primero se le aplica un corrimiento vertical para que los valores queden positivos y luego se amplifica hasta tener valores entre 0 y 65535. Finalizando esta etapa se gráfica y se le asigna el valor a una variable tipo entero para discretizar las señales.

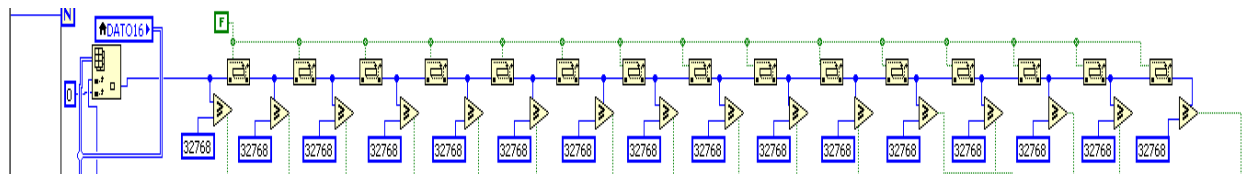
Figura 47. Desplazamiento de nivel.



Fuente: Autor.

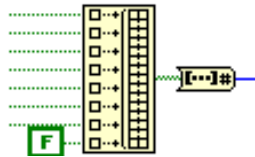
Una vez se tienen los valores de las señales discretizados y en un valor entre 0 y 65535, se procede a realizar la conversión de cada muestra de 7 valores de 16 bits a un arreglo de 16 valores de 8 bits (Byte). Para realizar esta acción, cada valor de 16 bits, se separa en bits tal como lo indica la figura 48. Luego de tener separado cada valor, se concatenan los bits n-esimos de cada valor de 16 bits, esto se realiza mediante el diagrama que se presenta en la figura 49, teniendo 16 bytes en representación de los 7 valores de 16 bits. Estos datos se van a transmitir por el puerto serie, por esto, se hace necesario realizar la conversión del valor numérico a una cadena, mediante el esquema que se presenta en la figura 50.

Figura 48. Separación valor de 16 bits en bits.



Fuente: Autor.

Figura 49. Conversión arreglo de bits a número entero.



Fuente: Autor.

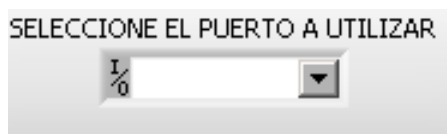
Figura 50. Conversión numero entero a cadena, para transmisión.



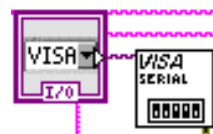
Fuente: Autor.

Una vez se tienen los datos en cadenas, se procede a realizar el envío de estos mediante la herramienta VISA la cual se configura mediante los bloques que se presentan en la figura 51 donde se selecciona el puerto a utilizar y se inicializa la comunicación. La transmisión de los datos se realiza mediante el diagrama de bloques que se tiene en la figura XX, donde se realiza el envío de 16 cadenas.

Figura 51. Puerto visa. a) Panel frontal b) Diagrama de bloques.



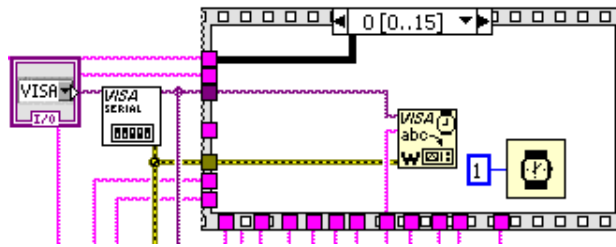
a



b

Fuente: Autor.

Figura 52. Envío de datos puerto serial.



Fuente: Autor.

2.3. Interfaz de usuario

La interfaz de usuario comprende la adquisición de datos necesarios para recrear las señales que se quieren generar. Esta se realiza de forma local o remota.

2.3.1. Ingreso de datos remota

El ingreso de datos remoto, comprende la adquisición de las señales ya creadas mediante la herramienta de software LabVIEW, este procedimiento se realiza generando un archivo de datos que contiene las señales, y luego enviando este al microcontrolador el cual cuenta con comunicación USB tipo CDC¹⁴, en la cual el dispositivo emula un puerto serial con toda la interfaz física USB. La transmisión de datos se realiza con ayuda de otra herramienta de software llamada *RealTerm*¹⁵ y de un puerto serial virtual que se puede crear mediante el software llamado *Virtual Serial Port (VSM)*¹⁶. El proceso se realiza mediante los siguientes pasos:

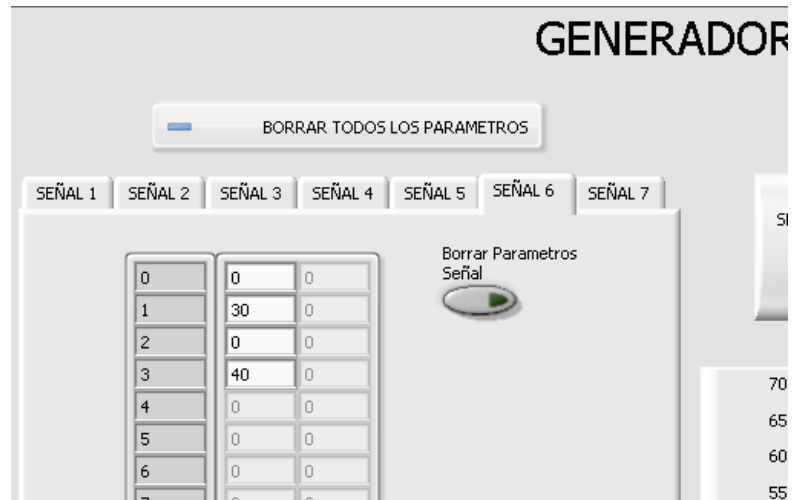
Primero en LabVIEW se configuran los parámetros de las señales a generar como se indica en la figura 53, en este caso, se está configurando el canal 6 con los armónicos 1 y 3, mediante una magnitud del 30% y 40% respectivamente.

¹⁴ *communications device class*

¹⁵ <http://realterm.sourceforge.net/>

¹⁶ <http://www.eltima.com/products/vspdxp/>

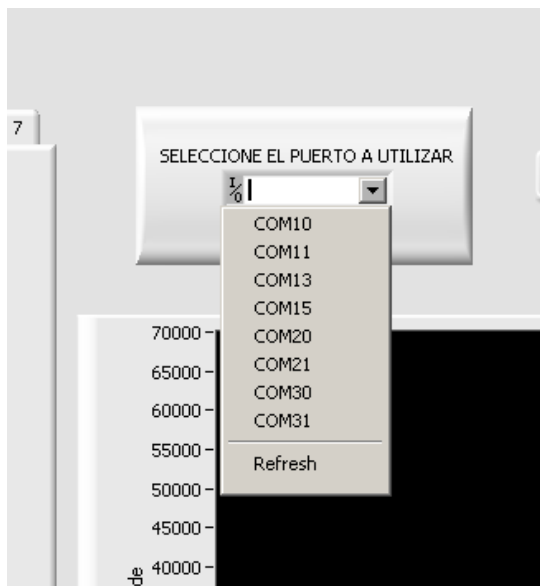
Figura 53. Ingresando parámetro interfaz remota.



Fuente: Autor.

Luego se selecciona el puerto serial por el que se van a enviar los datos tal como se indica en la figura 54, este puerto debe ser un puerto virtual por el cual se van a enviar los datos para crear el archivo que contiene la información de las señales.

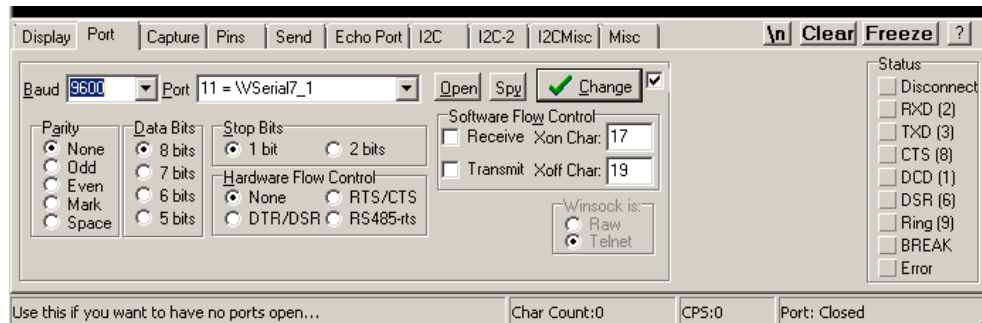
Figura 54. Selección puerto serial.



Fuente: Autor.

Seguido a esto, se selecciona en la herramienta RealTerm el otro puerto perteneciente a este puerto virtual, que en este caso es el COM11, pues se creó un puerto serial virtual que enlaza los puertos COM10<->COM11, luego de seleccionarlo, se da clic en el botón *OPEN* para habilitar la comunicación.

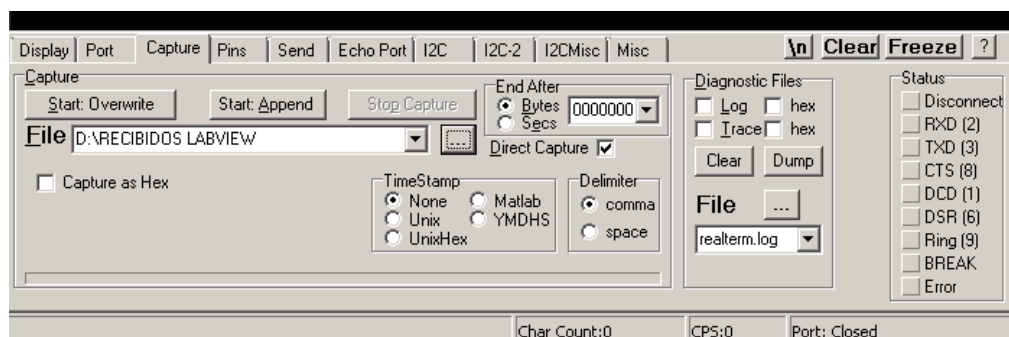
Figura 55. Selección puerto RealTerm.



Fuente: Autor.

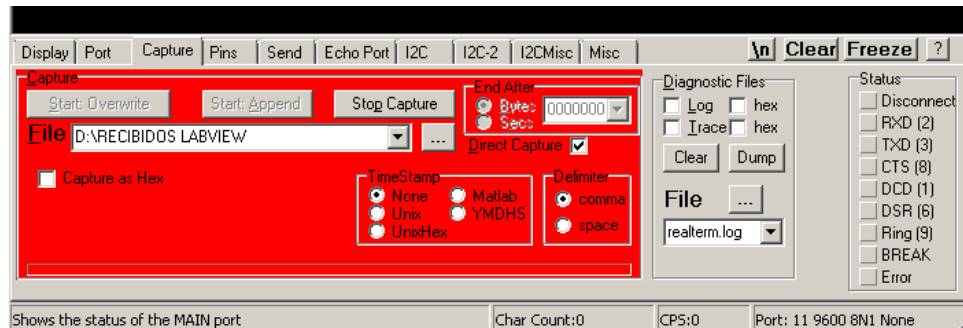
Luego, se procede a capturar los datos que se van a enviar desde LabView, para esto, se va a la pestaña *CAPTURE*, donde se selecciona un archivo de destino, seguido a esto se da clic en el botón *Start Overwrite* para empezar a escribir en el archivo creado tal como se indica en las figuras 56 y 57.

Figura 56. Seleccionar archivo destino de datos recibidos.



Fuente: Autor.

Figura 57. Recepción de datos habilitada.



Fuente: Autor.

Luego de habilitar la recepción de datos, se procede a enviarlos desde labview haciendo clic en el botón enviar datos, una vez que se envían los datos perfectamente, se grafica en la interfaz la señal generada.

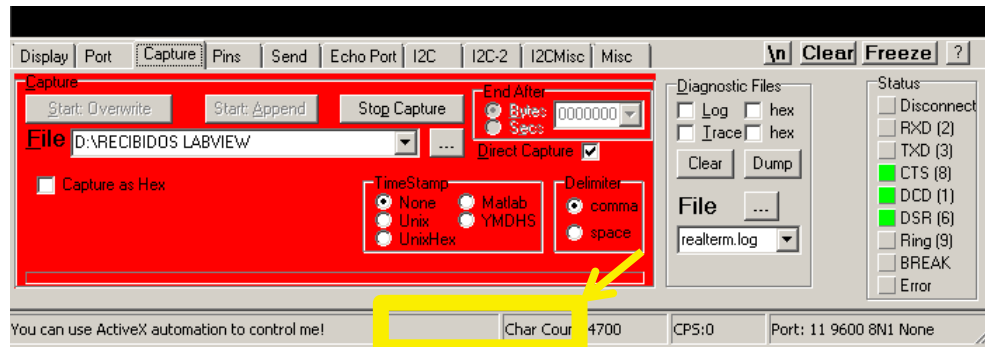
Figura 58. Datos enviados.



Fuente: Autor.

Una vez que los datos son enviados, el indicador *Char Count* muestra la cantidad de los datos recibidos como se muestra en la figura 59, una vez terminado el proceso, se da clic en *Stop Capture* para finalizar la recepción de los datos.

Figura 59. Recepción de datos.

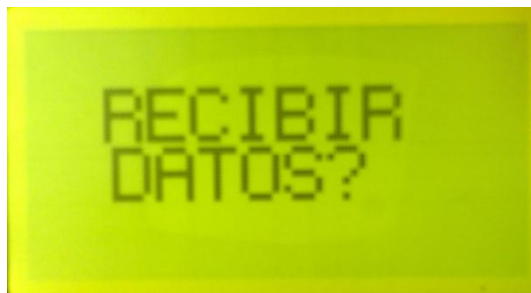


Fuente: Autor.

Ya teniendo el archivo que contiene la información de las señales, se procede a realizar el envío de este al microcontrolador, el cual a medida que va recibiendo los datos, lo va almacenando en la memoria EEPROM de a paquetes de 64 bytes.

Para esto se entra al modo recepción de datos mediante el botón **USB**, donde en primera medida, el sistema pregunta si se desea realizar la conexión con el PC, tal como se muestra en la figura 60, para aceptar la recepción, se oprime el botón **OK**, en caso rechazar el procedimiento, se utiliza cualquier otra tecla.

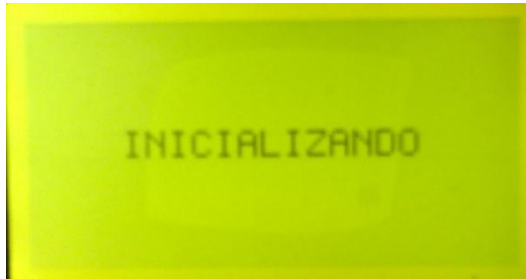
Figura 60. Habilitar comunicación USB.



Fuente: Autor.

Seguido a esto, el sistema queda inicializando la comunicación hasta que se establece el enlace tal como lo muestra la figura 61, en caso de no establecerse la comunicación al cabo de 20 [s], el sistema se reiniciara.

Figura 61. Sistema esperando enlace con el computador.

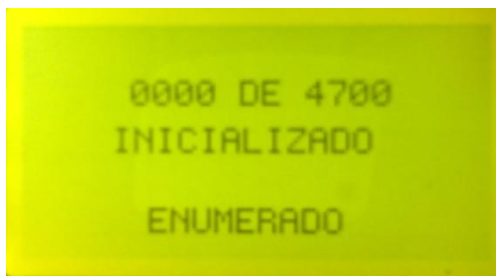


Fuente: Autor.

En este paso se deben instalar los drivers pertenecientes al sistema, que se pueden encontrar en¹⁷.

Al establecer la conexión, el sistema presenta lo que se muestra en la figura 62, indicando que la comunicación ya se encuentra establecida y esta esperando, además de la cantidad de datos que se van a recibir.

Figura 62. Sistema enlazado con el computador.

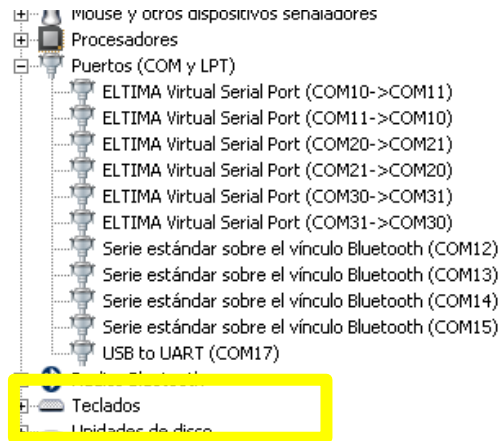


Fuente: Autor.

Una vez se tiene que la comunicación se encuentra establecida, se necesita saber cuál es el puerto asignado por el computador al sistema. Para esto, se va al administrado de dispositivos del computador y se busca entre los puertos COM y LPT, y se busca el dispositivo llamado “*USB to UART*”, el COM perteneciente a este es el que se necesita para él envío de datos.

¹⁷ <https://www.dropbox.com/sh/3jqaq5ji6elqm26/TQIf12YhBE>

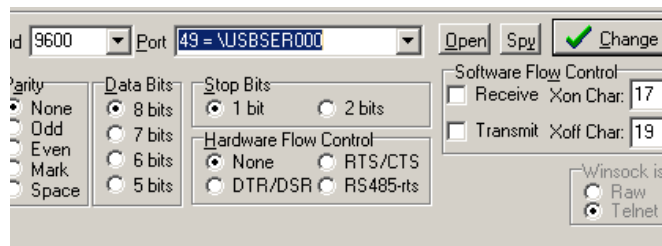
Figura 63. Asignación puerto COM al sistema.



Fuente: Autor.

Conociendo el puerto asignado, se vuelve a la aplicación *SerialTerm*, se selecciona el puerto asignado y se hace clic en el botón *Open* para establecer el enlace.

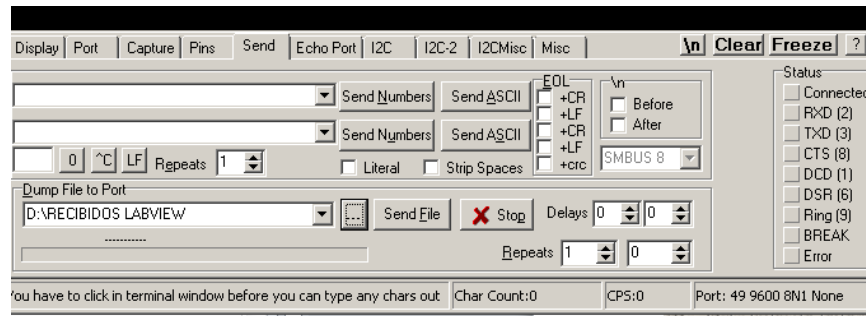
Figura 64. Selección puerto del sistema.



Fuente: Autor.

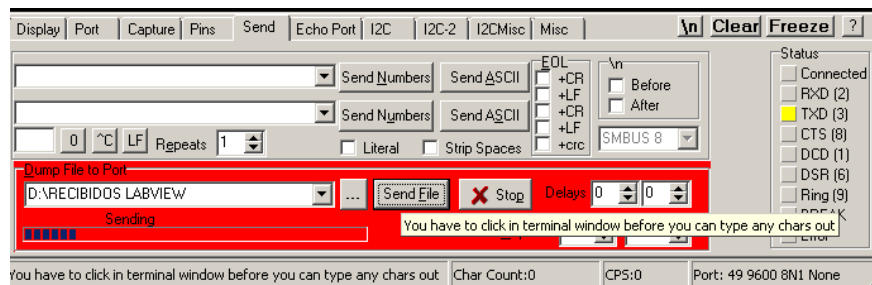
Luego se va a la pestaña *Send*, se selecciona el archivo a enviar y se realiza el envío de los datos, haciendo clic en el botón *Send File*.

Figura 65. Selección archivo de datos.



Fuente: Autor.

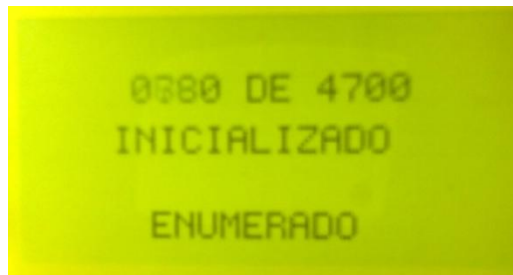
Figura 66. Envío de datos PC-Sistema.



Fuente: Autor.

A medida que se van recibiendo los datos, el sistema presenta la cantidad de datos recibidos en la pantalla, tal como se muestra en la figura 67. Luego de recibir la totalidad de los datos muestra en pantalla lo que se indica en la figura 68.

Figura 67. Sistema recibiendo datos.



Fuente: Autor.

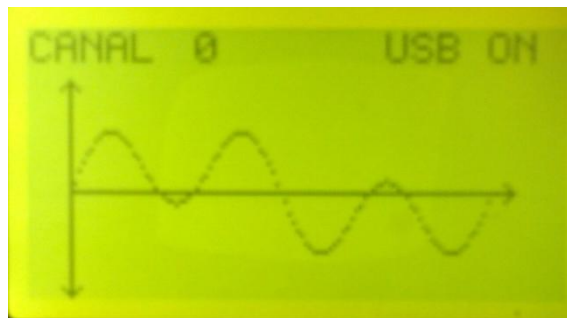
Figura 68. Sistema indicando que la transferencia ha sido completa.



Fuente: Autor.

Luego de esto, el sistema vuelve a su estado normal de generación y visualización en pantalla de las señales, donde se tiene como resultado el presentado en la figura 69, que tiene la misma forma de la señal que se creó y que se tiene en la figura 58.

Figura 69. Visualización señal generada.



Fuente: Autor.

El indicador de "USB ON" en la parte derecha superior, indica que el sistema ya se encuentra enlazado con el computador, y que solo basta con entrar al modo de recepción de datos y recibir archivos desde la herramienta ya mencionada.

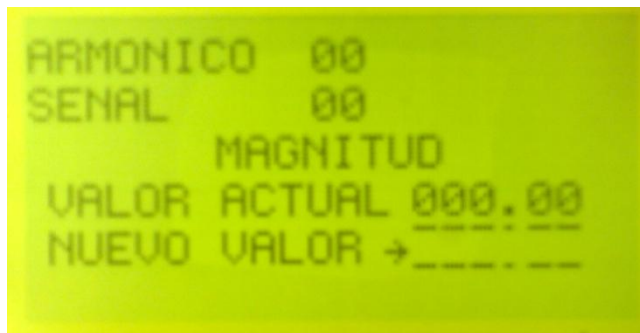
2.3.2. Ingreso de datos local

El ingreso de datos local, comprende la creación de una matriz de datos que contiene los parámetros de las señales a generar, los cuales son las magnitudes y fases de los 50 armónicos que presentan las 7 señales.

Estos parámetros se presentan mediante una matriz de dimensión 7x51x2, donde la primera dimensión representa las 7 señales, la segunda el nivel de tensión promedio más los 50 armónicos, y la tercera dimensión si el parámetro indicado es magnitud con un "0" o si es fase con un "1".

Una vez que se ingresa al modo de adquisición de parámetros, se realiza la navegación por esta matriz mediante el teclado para poder visualizar y modificar los parámetros que se quieran, una vez que se entra a este modo, se muestra en pantalla lo indicado en la figura 70.

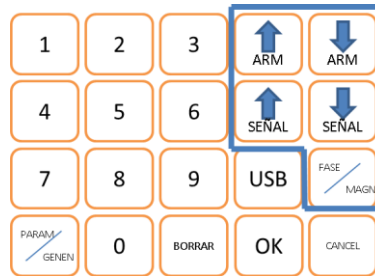
Figura 70. Modo adquisición de parámetros.



Fuente: Autor.

Para llegar al parámetro requerido, se realizan los desplazamientos necesarios mediante los botones indicados en la figura 71, mostrando a su vez en "VALOR ACTUAL" el valor correspondiente que presenta dicho parámetro.

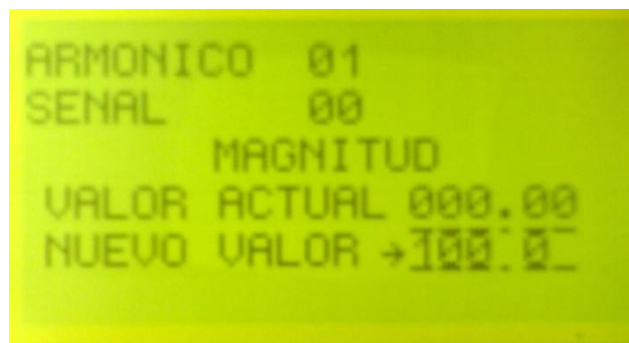
Figura 71. Teclas de desplazamiento.



Fuente: Autor.

Para modificar algún parámetro, solo se necesita ubicarse sobre él y escribir el número que se quiera. Si se desea visualizar el valor, solo se devuelve a la posición y se leerá el valor automáticamente.

Figura 72. Ingreso de dato parámetro.



Fuente: Autor.

Figura 73. Visualización parámetro escrito.



Fuente: Autor.

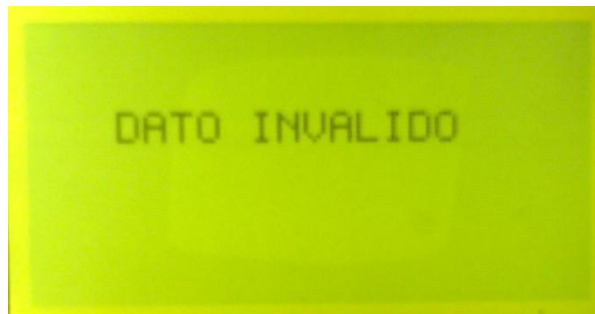
Los valores de magnitudes se encuentran restringidos a valores menores o iguales a 100, por esto, si se ingresa un valor que no cumpla aparecerá un mensaje de “DATO INVALIDO”, como se muestra en las figuras 74 y 75.

Figura 74. Ingreso de valor 111,11.





Fuente: Autor.

Figura 75. Mensaje al ingresar valor no valido.



Fuente: Autor.

En caso de equivocarse al ingresar un dato, se debe pulsar el botón  para borrar los números que se hallan escrito hasta el momento.

Si se desea salir del modo parámetros, se debe pulsar el botón ,

en caso de querer generar las señales se pulsa el botón .

Una vez que se procede con la generación de señales, aparece en pantalla el indicador de las muestras calculadas a medida que se realiza el proceso, luego que se finaliza, se calculan los valores de las gráficas, donde en pantalla de muestra el progreso de la generación.

3. PRUEBAS

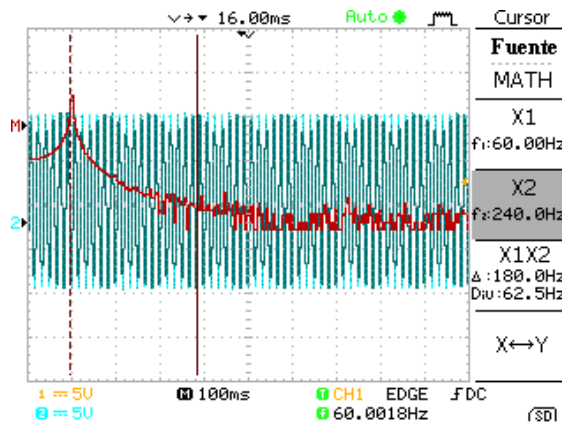
Dentro de este capítulo, se realizarán algunas pruebas con el fin de verificar el buen funcionamiento del dispositivo

3.1. Distorsión Armónica

Para la medición de la distorsión armónica, se analiza el espectro de algunas señales generadas con el fin de buscar armónicos indeseados que contribuyan en la mala calidad de la misma, para eso, se realizaron pruebas sobre señales generadas de 60 [Hz], 600[Hz], 1200[Hz], 1800[Hz], 2400[Hz] y 3000[Hz], tomando datos del espectro perteneciente a dicha señal, sobre el cual se calculan las componentes indeseadas.

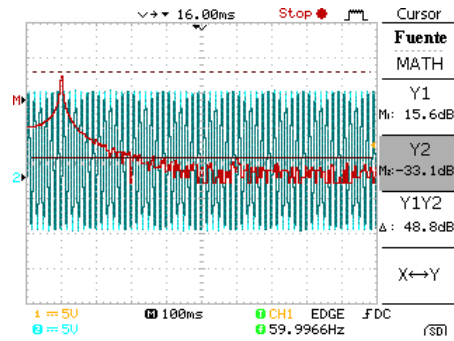
En las siguientes figuras, se presentan unas mediciones tomadas mediante un osciloscopio en función analizador de espectro, mediante las cuales, se realiza una cuantificación de los componentes de armónicos mediante los cuales se estimara la distorsión armónica presente en cada señal.

Figura 76. Espectro señal 60[Hz], cursor frecuencia.



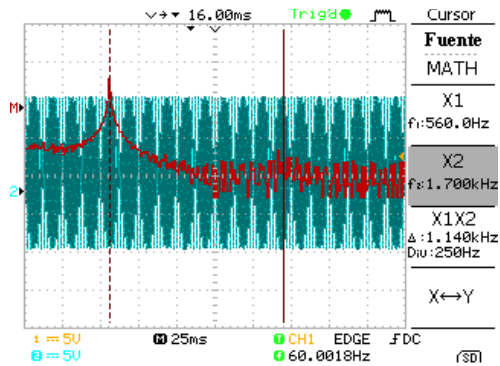
Fuente: Autor.

Figura 77. Espectro señal 60[Hz], cursor Amplitud.



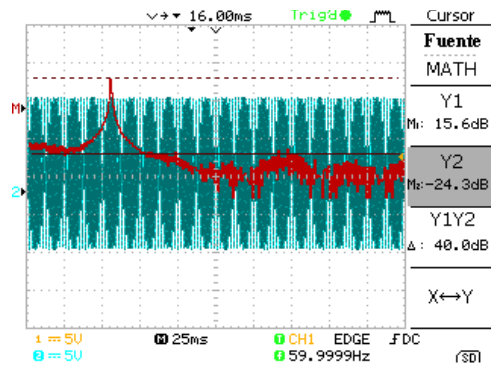
Fuente: Autor.

Figura 78. Espectro señal 600[Hz], cursor frecuencia.



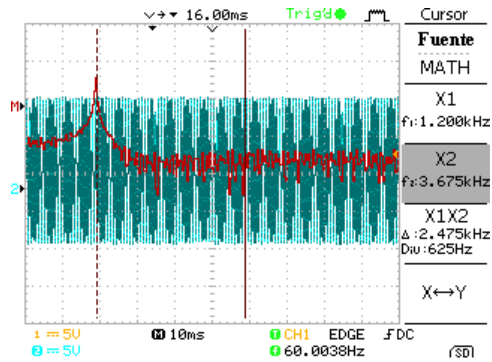
Fuente: Autor.

Figura 79. Espectro señal 600[Hz], cursor Amplitud.



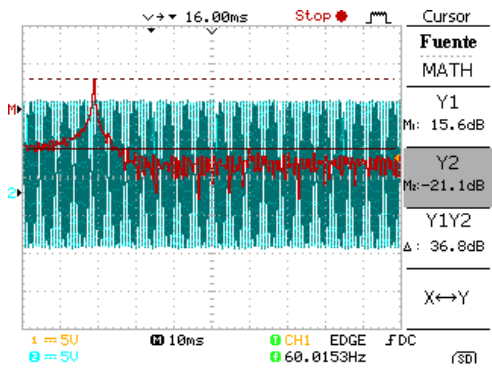
Fuente: Autor.

Figura 80. Espectro señal 1200[Hz], cursor frecuencia.



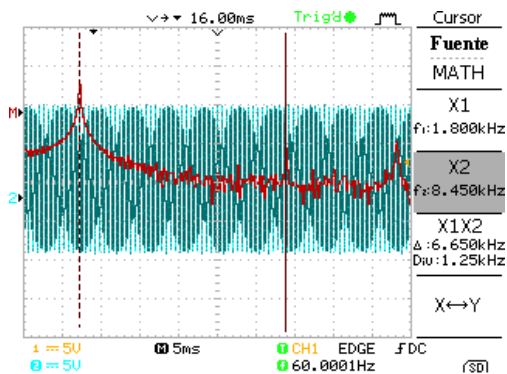
Fuente: Autor.

Figura 81. Espectro señal 1200[Hz], cursor amplitud.



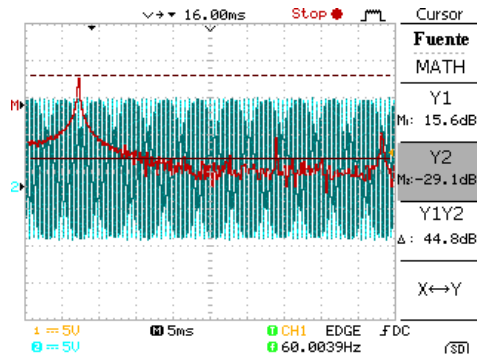
Fuente: Autor.

Figura 82. Espectro señal 1800[Hz], cursor frecuencia.



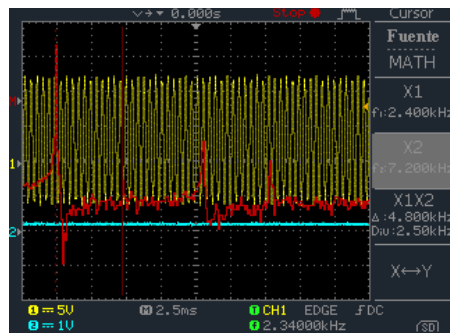
Fuente: Autor.

Figura 83. Espectro señal 1800[Hz], cursor Amplitud.



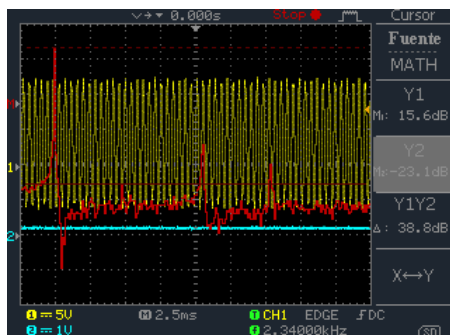
Fuente: Autor.

Figura 84. Espectro señal 2400[Hz], cursor frecuencia.



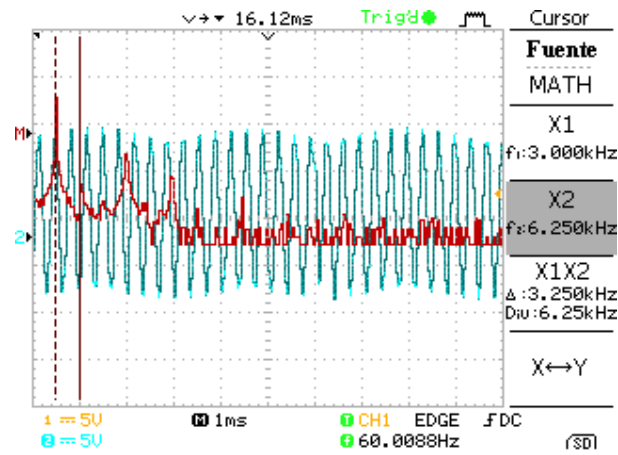
Fuente: Autor.

Figura 85. Espectro señal 2400[Hz], cursor amplitud.



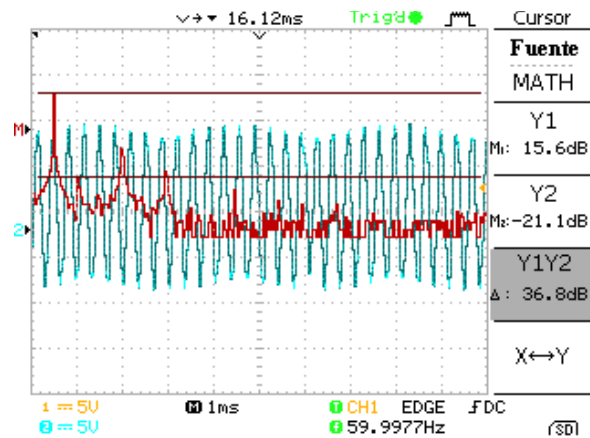
Fuente: Autor.

Figura 86. Espectro señal 3000[Hz], cursor frecuencia.



Fuente: Autor.

Figura 87. Espectro señal 3000[Hz], cursor Amplitud.



Fuente: Autor.

De las anteriores figuras, se extrae la información presentada en la siguiente tabla,

Tabla 8. Magnitudes de armónicos diferentes frecuencias.

Frecuencia [Hz]	Magnitudes Armónicos[dB]					
	1	2	3	4	5	6
60	15.6	0	0	0	0	0
600	15.6	0	0	-24.3	0	0
1200	15.6	0	0	-21.1	0	0
1800	15.6	0	0	-29.1	0	0
2400	15.6	0	-22	-26	0	0
3000	15.6	-21.1	0	-9.1	0	0

Fuente: Autor.

Para el cálculo de la distorsión armónica, se utiliza la siguiente expresión

$$THD\% = \frac{\sqrt{V_2^2 + V_3^2 + \dots + V_n^2}}{V_1} * 100$$

Donde las magnitudes deben estar en unidades de [V].

Tabla 9. Distorsión armónica señales generadas.

Frecuencia	THD (%)
60	0
600	1.01
1200	1.46
1800	0.58
2400	1.55
3000	6

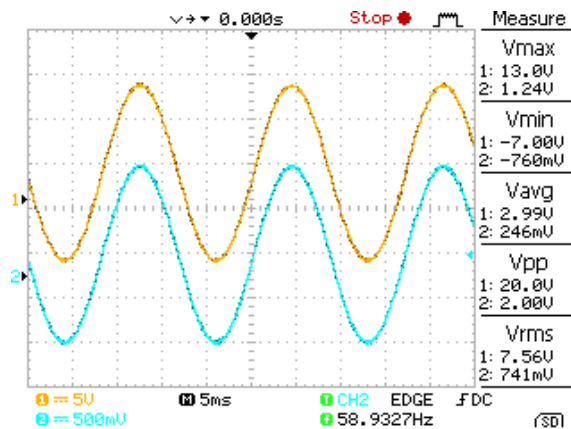
Fuente: Autor.

3.2. Verificación amplitudes generadas

Se generaron señales correspondientes a determinados valores de voltaje pico, para una frecuencia de 60[Hz], la comprobación de dichos valores, va estar especificada por los la tensión pico a pico.

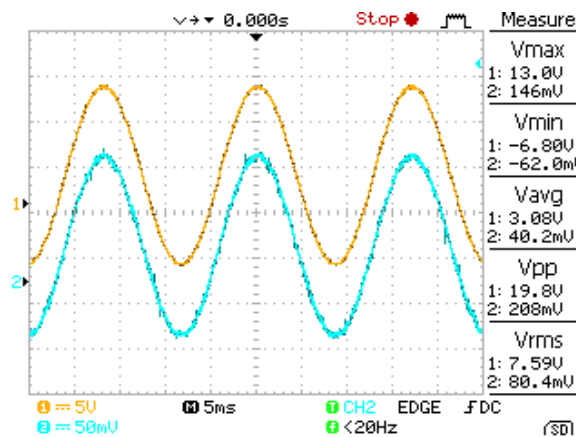
Las mediciones se presentan en las siguientes figuras, donde se generan señales con amplitudes de 10[V], 1[V], 0.1 [V], 0.05 [V], 0.01 [V] y 0.005[V].

Figura 88. Señales de 10[V] y 1[V] de tensión pico 60[Hz].



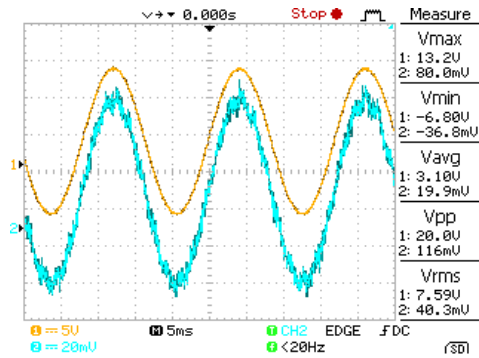
Fuente: Autor.

Figura 89. Señales de 10[V] y 0,1[V] de tensión pico 60[Hz].



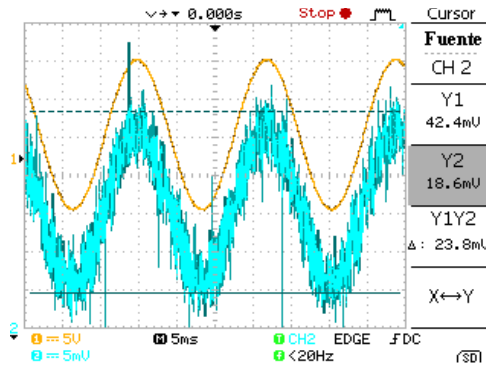
Fuente: Autor.

Figura 90. Señales de 10[V] y 0,05[V] de tensión pico 60[Hz].



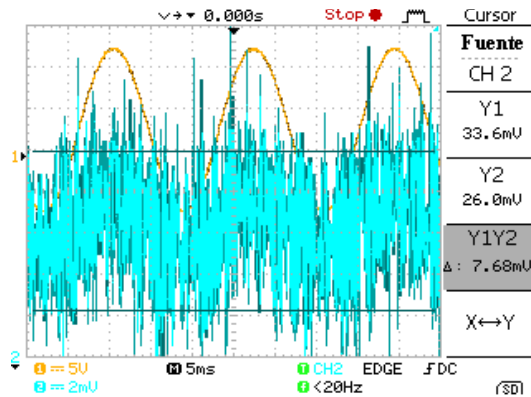
Fuente: Autor.

Figura 91. Señales de 10[V] y 0,01[V] de tensión pico 60[Hz].



Fuente: Autor.

Figura 92. Señales de 10[V] y 0,005[V] de tensión pico 60[Hz].

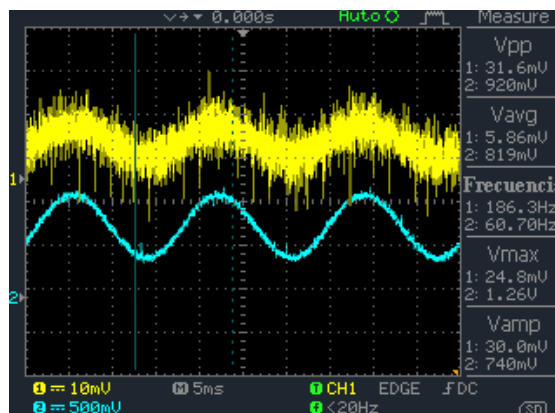


Fuente: Autor.

Como se tiene en las imágenes antes mencionadas, los valores configurados para dichas señales son generados tal como se espera, aunque para las tensiones de 10 [mV] y 5[mV] no se perciba muy bien.

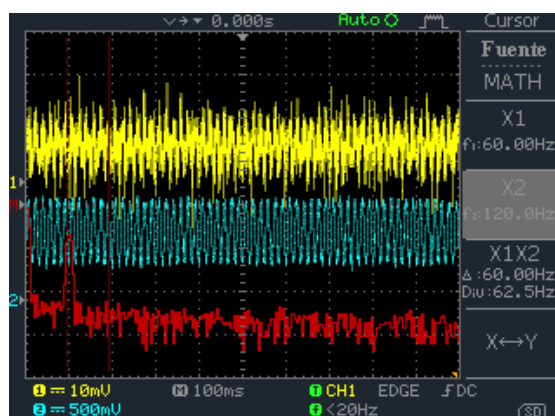
Se generaron señales para valores de 10 [mV] y 1[mV], las cuales para poder ser visualizadas en el osciloscopio, fue necesario de una etapa de amplificación mediante un amplificador de instrumentación INA214 el cual posee una ganancia fija de 100 [V/V], en las siguientes figuras, se presentan las señales de salida del sistema (señal) y estas amplificadas (azul).

Figura 93. Señal con Amplitud de 10 [mV].



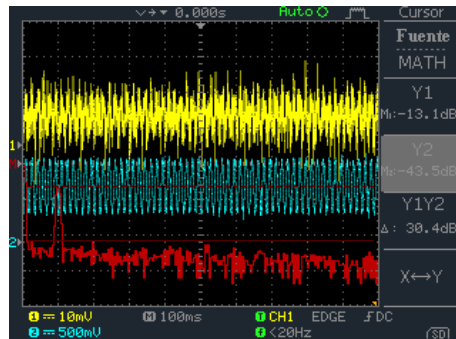
Fuente: Autor.

Figura 94. Espectro señal de salida 10 [mV], cursor frecuencia.



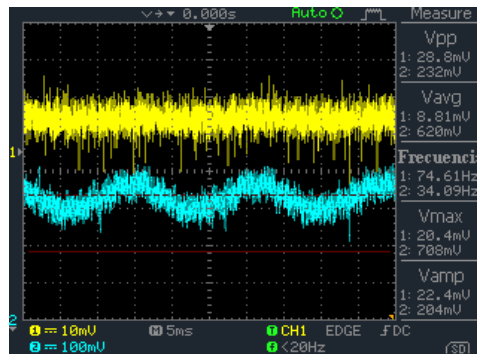
Fuente: Autor.

Figura 95. Espectro señal de salida 10 [mV], cursor Amplitud.



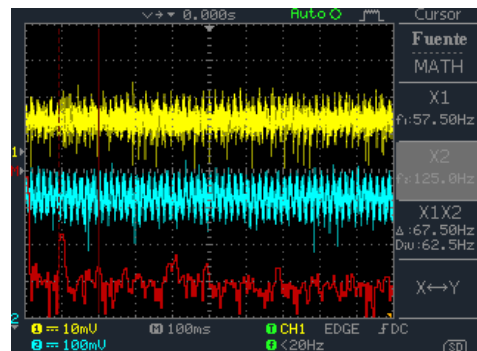
Fuente: Autor.

Figura 96. Señal con Amplitud de 1 [mV].



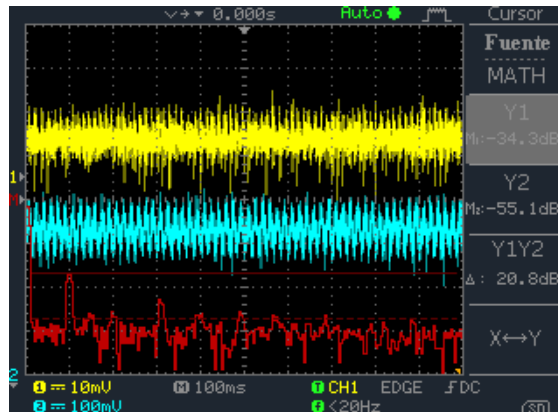
Fuente: Autor.

Figura 97. Espectro señal de salida 1 [mV], cursor frecuencia.



Fuente: Autor.

Figura 98. Espectro señal de salida 1 [mV], cursor Amplitud.



Fuente: Autor.

Tabla 10. Magnitudes de armónicos diferentes frecuencias.

Amplitud [mV]	Magnitudes Armónicos[dB]					
	1	2	3	4	5	6
10	-13.1	-43.5	0	0	0	0
1	-34.3	-55.1	0	-48.0	0	0

Fuente: Autor.

Tabla 11. Distorsión armónica señales generadas.

Amplitud [mV]	THD (%)
10	3
1	22

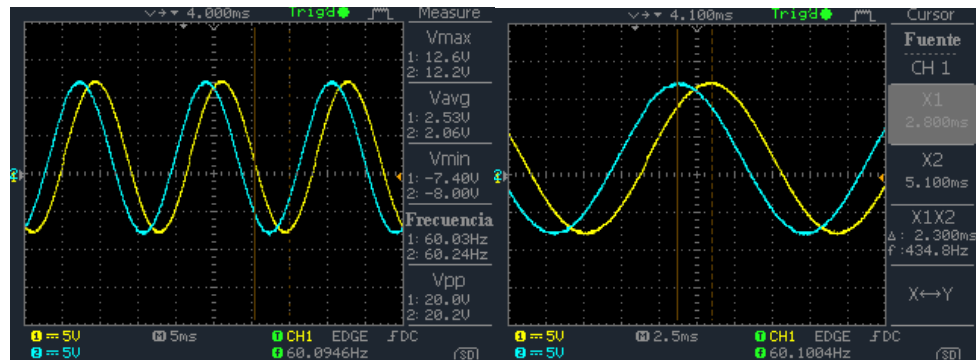
Fuente: Autor.

Se tiene que hasta una señal de 10 [mV] el sistema presenta una distorsión armónica de 3%.

3.3. Verificación fases generadas

Para esta prueba, se generaron señales de una frecuencia de 60 [Hz], con fases de 45[°], 90[°], 180[°] y 270[°], cuyos resultados se aprecias en las siguientes figuras.

Figura 99. Señal de 60[Hz], desfase 45[°].



Fuente: Autor.

Figura 100. Señal de 60[Hz], desfase de 90[°].

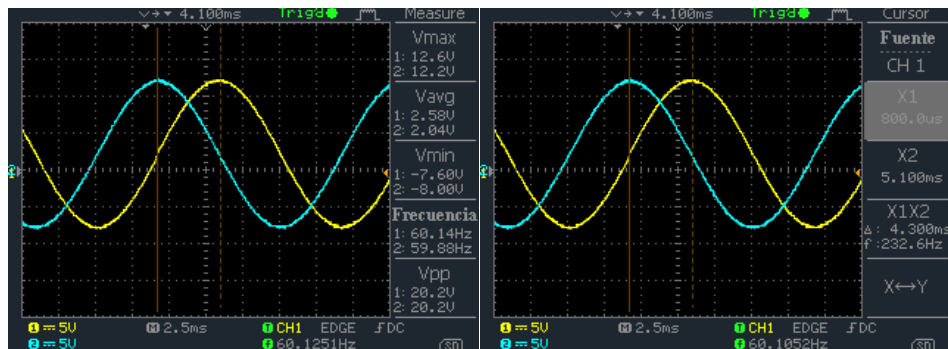
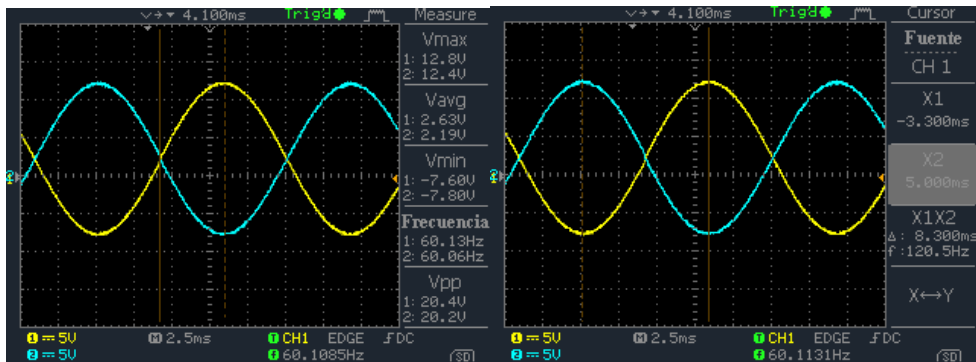
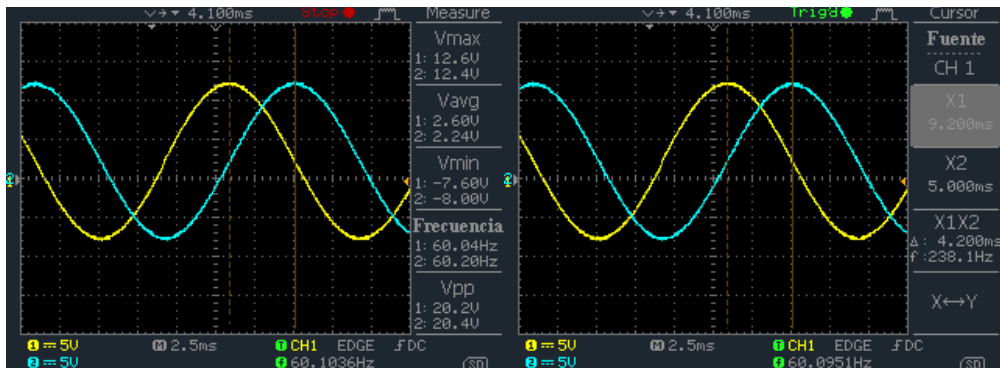


Figura 101. Señal de 60[Hz], desfase de 180[°].



Fuente: Autor.

Figura 102. Señal de 60[Hz], desfase de 270[°].



Fuente: Autor.

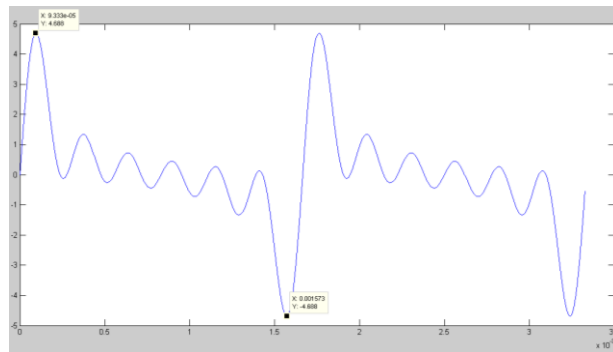
Donde se tiene mediante verificación visual, que los resultados son los esperados, pues los desfases esperados con los que se aprecian en las mediciones realizadas.

3.4. Verificación Señales Teóricas contra Experimentales

Para la siguiente prueba, se generan señales teóricas en la herramienta de software MATLAB¹⁸ con las mismas componentes de una señal generada real, para compararlas y determinar si la forma adquirida es la esperada.

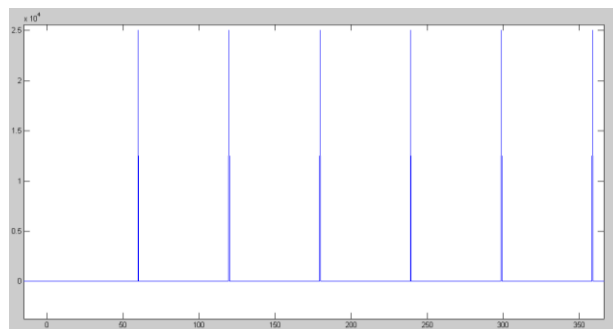
Como primera prueba, se genera una señal formada por los armónicos 1,2,3,4,5 y 6, con magnitudes de 1[V], en las siguientes figuras se presentan los resultados experimental y teóricos, sobre los cuales se hará una comparación visual y de alguno parámetros.

Figura 103. Señal teórica 1, 2, 3, 4,5 armónicos temporal.



Fuente: Autor.

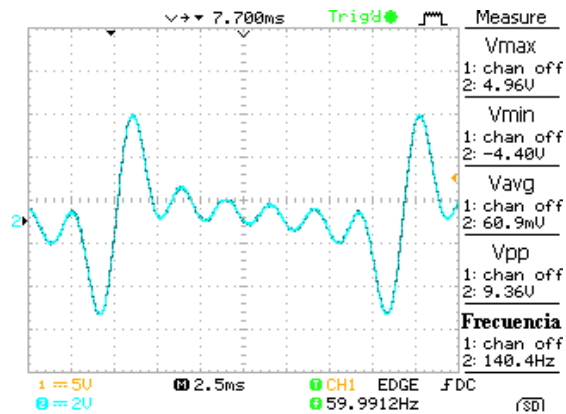
Figura 104. Señal teórica 1, 2, 3, 4,5 armónicos frecuencia.



Fuente: Autor.

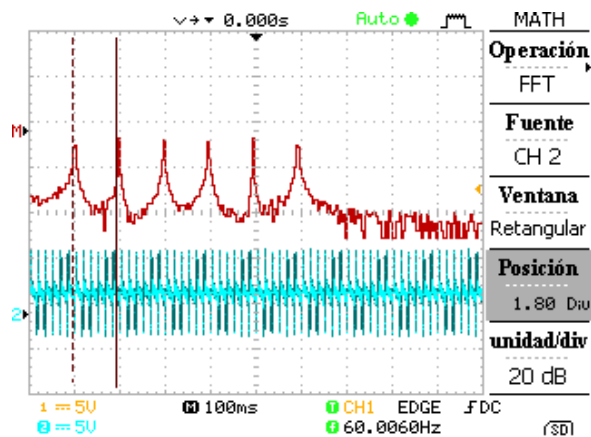
¹⁸ <http://www.mathworks.com/products/matlab/>

Figura 105. Señal experimental 1, 2, 3, 4,5 armónicos temporal.



Fuente: Autor.

Figura 106. Señal experimental 1, 2, 3, 4,5 armónicos frecuencia.



Fuente: Autor.

Se tiene que la señal teórica posee una tensión pico a pico de 9.36 [V], al igual que la señal experimental, además se tiene una similitud entre las señales en el espacio temporal y poseer las mismas componentes frecuenciales.

Como segunda prueba, se decide generar una señal cuadrada de 60[Hz], la cual se representan en series de Fourier mediante la siguiente expresión

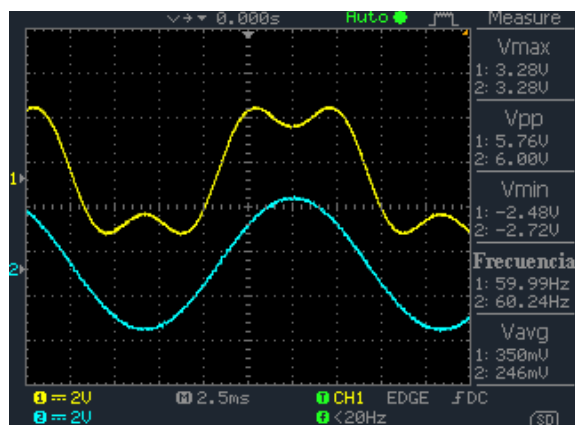
$$Vm * \left(\frac{\sin(x)}{1} + \frac{\sin(3x)}{3} + \frac{\sin(5x)}{5} + \dots + \frac{\sin(nx)}{n} \right); n = \text{impares.}$$

Para Valor de v_m de 3 [V], teniendo en cuenta las siete primeras componentes, la secuencia quedaría

$$3 \cdot \sin(x) + \sin(3x) + 0,6 \sin(5x) + 4,2 \cdot \sin(7x) + 3,3 \sin(9x) + 2,7 \sin(11x) + 2,3 \sin(12x)$$

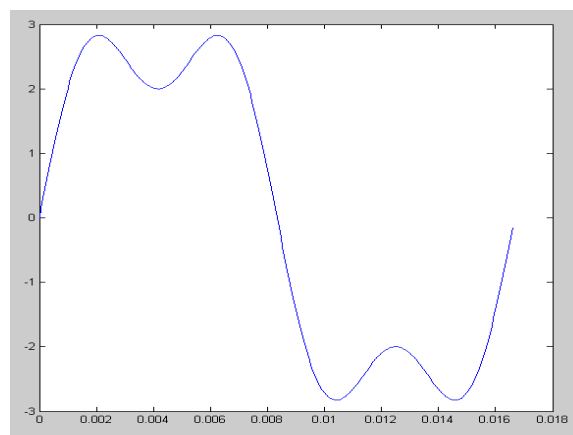
En las siguientes imágenes, se muestra como varia la señal que se requiere a medida la suma de cada vez más armónicos, mediante la comparación de la medición de la señal real contra la señal teórica simulada.

Figura 107. Señal Real componentes 1 y 2.



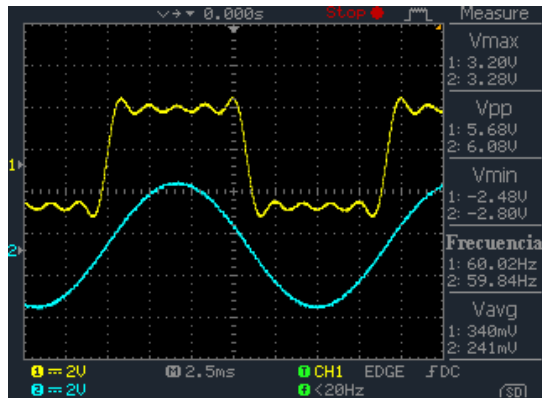
Fuente: Autor.

Figura 108. Señal Simulada componentes 1 y 2.



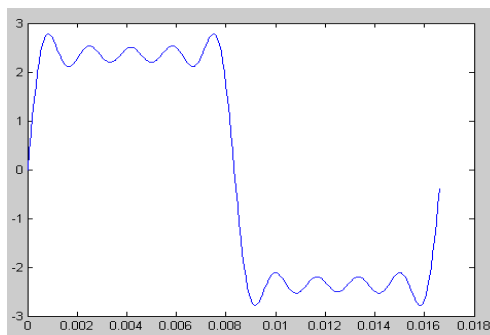
Fuente: Autor.

Figura 109. Señal Real componentes 1, 2,3 ,4 y 5.



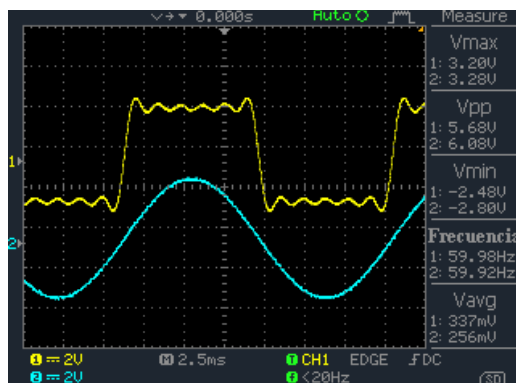
Fuente: Autor.

Figura 110. Señal Simulada componentes 1, 2,3 ,4 y 5.



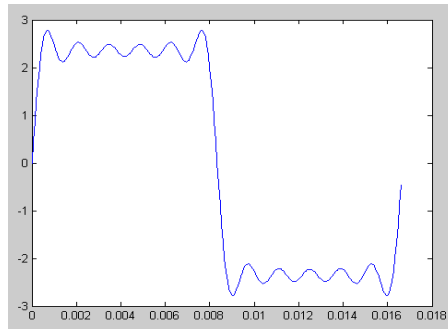
Fuente: Autor.

Figura 111. Señal Real componentes 1, 2, 3, 4, 5 y 6.



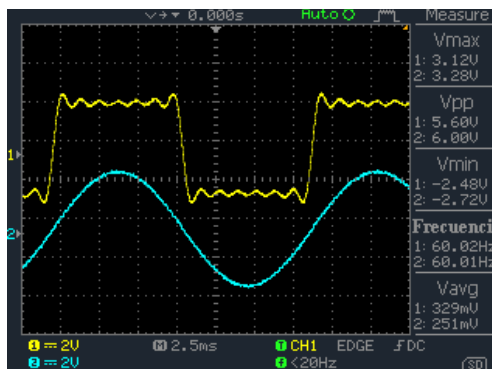
Fuente: Autor.

Figura 112. Señal Simulada componentes 1, 2, 3, 4, 5 y 6.



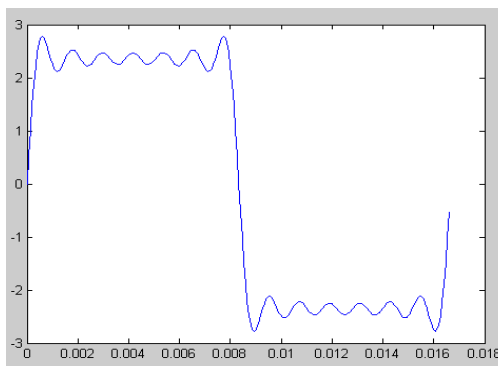
Fuente: Autor.

Figura 113. Señal Real componentes 1, 2, 3, 4, 5, 6 y 7.



Fuente: Autor.

Figura 114. Señal Simulada componentes 1, 2, 3, 4, 5, 6 y 7.



Fuente: Autor.

Según las pruebas acabadas de realizar, se tiene que las señales generadas son idénticas a las simuladas.

4. CONCLUSIONES

- Se construyó un sistema en base al microcontrolador PIC18F4550 capaz de generar 7 señales, donde a cada señal se le pueden configurar 50 armónicos con sus respectivas magnitudes en porcentaje del valor máximo y fases en grados más un nivel de DC.
- Se obtuvo una tasa de muestreo con un periodo de 66 [µS].
- Se obtuvieron señales con una baja distorsión armónica, teniendo como máximo, un 6% en el peor caso para una frecuencia de 3 [Khz].
- Se diseñó un sistema que puede generar una señal con una tensión máxima de 10 [Vp] y 10[Vdc], con una resistencia de salida de 470 [Ω] y una corriente máxima de 21[mA].
- Se implementó una interfaz física local mediante una pantalla de cristal líquido y teclado, cuyo manejo facilita la administración de los parámetros correspondientes a las señales.
- Se utilizó el periférico USB del microcontrolador seleccionado para establecer la comunicación, entre el mismo y el computador, eliminando la necesidad de componentes adicionales para establecer la comunicación.
- Se diseñó mediante la herramienta de software labview un programa que genera las señales deseadas mediante una interfaz gráfica amigable con el usuario que permite visualizar las formas de ondas en una sola gráfica.
- Las señales se generaron con respecto al punto de cruce por cero para poder sincronizar todas salidas.
- Mediante las interfaces graficas se puede comparar mediante inspección visual las salidas requeridas sobre las obtenidas.
- Se utilizó una herramienta de software para realizar la transferencia de información entre el computador y el dispositivo, creando un archivo con las señales generadas desde labview y luego enviando este al microcontrolador.

5. RECOMENDACIONES

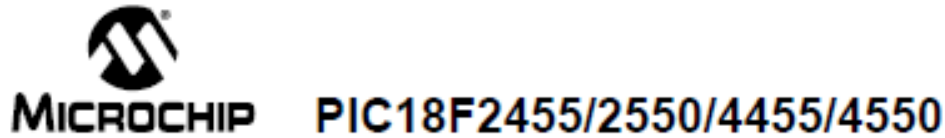
- Utilizar un microcontrolador con más prestaciones en cuanto a herramientas de cálculo, para poder realizar dicha tarea de una forma más rápida.
- Utilizar una pantalla LCD de más resolución para poder visualizar una aproximación más cercana a las señales reales generadas.
- Al utilizar el componente DAC8871, se hace necesaria tener energizado la parte digital y analógica a la vez, pues de no ser así, se producen consumos de potencia mayores.
- Mediante la implementación de memorias con comunicación paralela, se pueden obtener mejores tiempos de acceso a los datos almacenados en estas.

REFERENCIAS

- [1] NASSER kehtarnavaz – MANSOUR keramat. DSP System Design Using The TMS320C6000. Prentice Hall, 2001.
- [2] Alan v. Oppenheim y Alan S. Willsky. Señales y sistemas. (Ed. Prentice Hall), (1994)
- [3] PARGUAIN, Joselito Building a Stable DAC External Reference Circuit SLAA172 marzo 2003.
- [4] ESTEVEN RODRIGUEZ freddy ferney - PEÑA RODRIGUEZ elkin omar. *Diseño e implementación de un generador de formas de ondas de tensión y corriente para sistemas trifásico*. Trabajo de grado. UIS, 2009.
- [5] ACOSTA ROMERO Jorge Alberto - CORZO SUAREZ Edgar Alfonso. *Medición de energía eléctrica usando circuitos integrados de propósito específico*. Trabajo de grado. UIS, 2009.
- [6] CALA RODRIGUEZ Jairo Augusto. Diseño, simulación y caracterización de un bipotenciostato de precisión. Trabajo de grado. UIS, 2012.

ANEXOS

ANEXO a. Hoja de datos PIC18F4550



28/40/44-Pin High-Performance, Enhanced Flash USB Microcontrollers with nanoWatt Technology

Universal Serial Bus Features:

- USB V2.0 Compliant
- Low Speed (1.5 Mb/s) and Full Speed (12 Mb/s)
- Supports Control, Interrupt, Isochronous and Bulk Transfers
- Supports up to 32 endpoints (16 bidirectional)
- 1-Kbyte dual access RAM for USB
- On-chip USB transceiver with on-chip voltage regulator
- Interface for off-chip USB transceiver
- Streaming Parallel Port (SPP) for USB streaming transfers (40/44-pin devices only)

Power-Managed Modes:

- Run: CPU on, peripherals on
- Idle: CPU off, peripherals on
- Sleep: CPU off, peripherals off
- Idle mode currents down to 5.8 μ A typical
- Sleep mode currents down to 0.1 μ A typical
- Timer1 oscillator: 1.1 μ A typical, 32 kHz, 2V
- Watchdog Timer: 2.1 μ A typical
- Two-Speed Oscillator Start-up

Flexible Oscillator Structure:

- Four Crystal modes including High Precision PLL for USB
- Two External Clock modes, up to 48 MHz
- Internal oscillator block:
 - 8 user-selectable frequencies, from 31 kHz to 8 MHz
 - User-tunable to compensate for frequency drift
- Secondary oscillator using Timer1 (32 kHz)
- Dual oscillator options allow microcontroller and USB module to run at different clock speeds
- Fail-Safe Clock Monitor
 - Allows for safe shutdown if any clock stops

Peripheral Highlights:

- High-current sink/source 25 mA/25 mA
- Three external interrupts
- Four Timer modules (Timer0 to Timer3)
- Up to 2 Capture/Compare/PWM (CCP) modules:
 - Capture is 16-bit, max. resolution 6.25 ns (T_{CH}/16)
 - Compare is 16-bit, max. resolution 100 ns (T_{CV})
 - PWM output: PWM resolution is 1 to 10-bit
- Enhanced Capture/Compare/PWM (ECCP) module:
 - Multiple output modes
 - Selectable polarity
 - Programmable dead time
 - Auto-Shutdown and Auto-Restart
- Enhanced USART module:
 - LIN bus support
- Master Synchronous Serial Port (MSSP) module supporting 3-wire SPI™ (all 4 modes) and I²C™ Master and Slave modes
- 10-bit, up to 13-channels Analog-to-Digital Converter module (A/D) with programmable acquisition time
- Dual analog comparators with input multiplexing

Special Microcontroller Features:

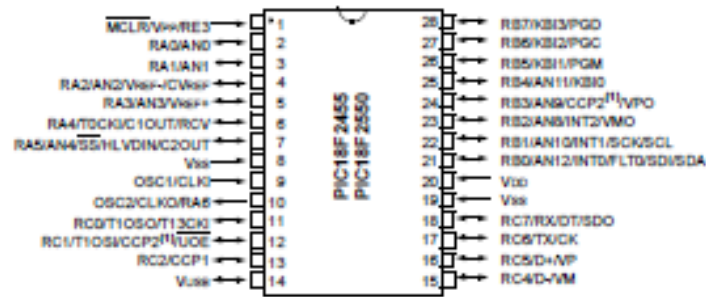
- C compiler optimized architecture with optional extended instruction set
- 100,000 erase/write cycle Enhanced Flash program memory typical
- 1,000,000 erase/write cycle Data EEPROM memory typical
- Flash/Data EEPROM Retention: > 40 years
- Self-programmable under software control
- Priority levels for interrupts
- 8 x 8 Single-Cycle Hardware Multiplier
- Extended Watchdog Timer (WDT):
 - Programmable period from 41 ms to 131s
- Programmable Code Protection
- Single-Supply 5V In-Circuit Serial Programming™ (ICSP™) via two pins
- In-Circuit Debug (ICD) via two pins
- Optional dedicated ICD/ICSP port (44-pin devices only)
- Wide operating voltage range (2.0V to 5.5V)

Device	Program Memory		Data Memory		I/O	10-bit A/D (ch)	CCP/ECCP (PWM)	SPP	MSSP		I ² C	Comparators	Timers 8/16-bit
	Flash (bytes)	# Single-Word Instructions	SRAM (bytes)	EEPROM (bytes)					SPI™	Master PC™			
PIC18F2455	24K	12288	2048	256	24	10	2/0	No	Y	Y	1	2	1/3
PIC18F2550	32K	16384	2048	256	24	10	2/0	No	Y	Y	1	2	1/3
PIC18F4455	24K	12288	2048	256	35	13	1/1	Yes	Y	Y	1	2	1/3
PIC18F4550	32K	16384	2048	256	35	13	1/1	Yes	Y	Y	1	2	1/3

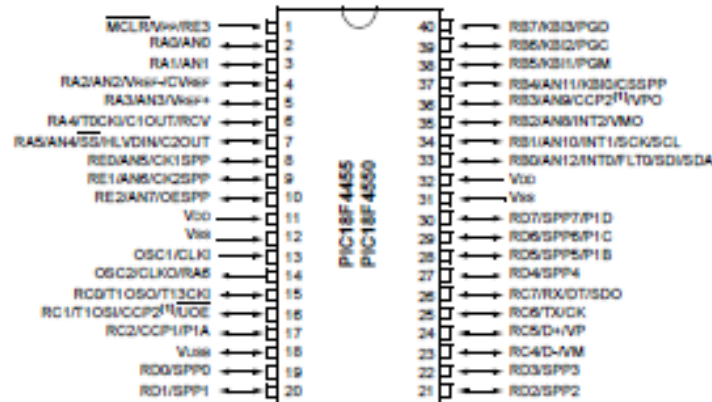
PIC18F2455/2550/4455/4550

Pin Diagrams

28-Pin PDIP, SOIC



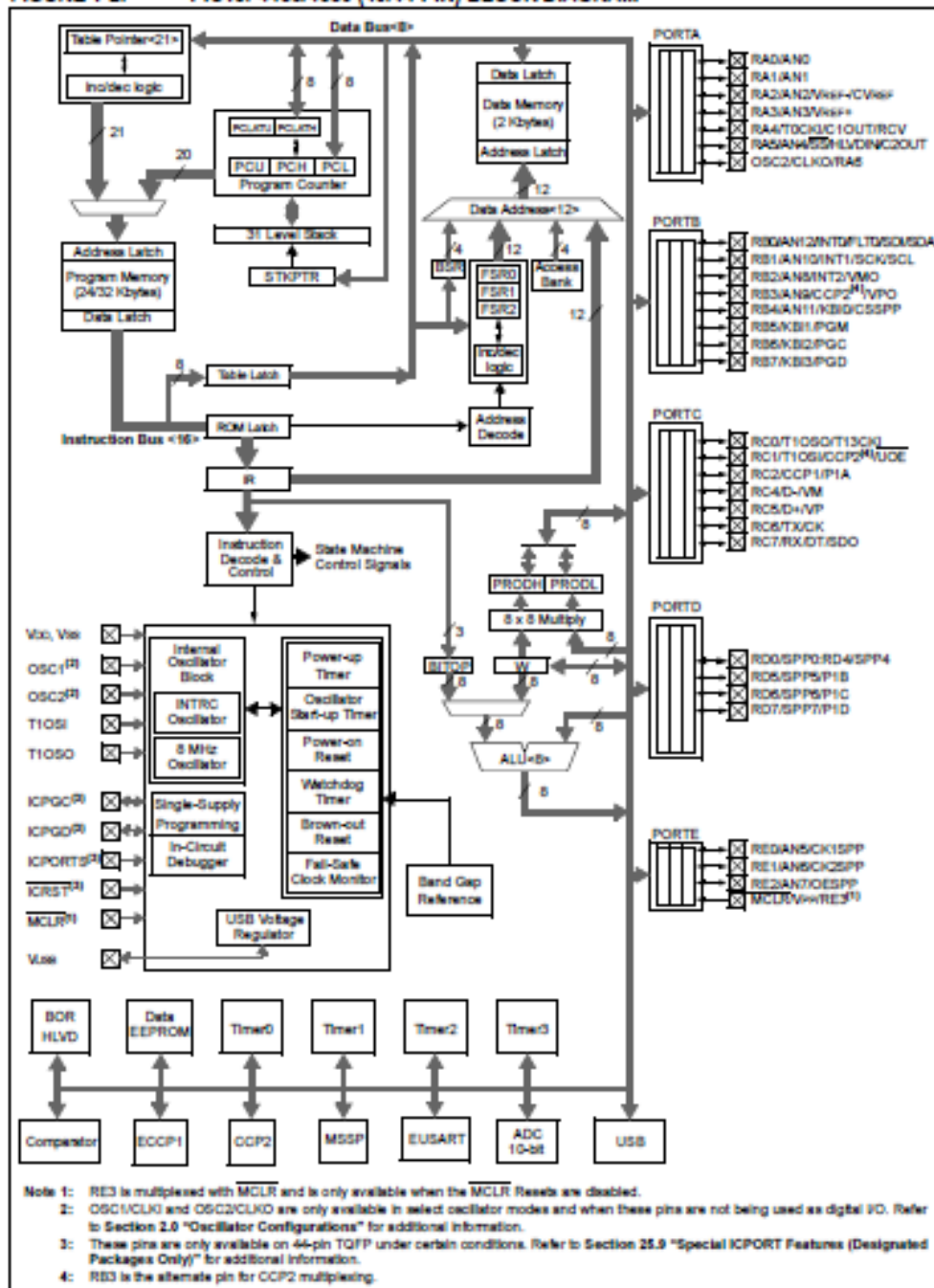
40-Pin PDIP



Note 1: RB3 is the alternate pin for CCP2 multiplexing.

PIC18F2455/2550/4455/4550

FIGURE 1-2: PIC18F4455/4550 (40/44-PIN) BLOCK DIAGRAM



ANEXO b. Hoja de datos DCP020505



DCP02 Series

SBVS011K—MARCH 2006—REVISED FEBRUARY 2008

**Miniature, 2W, Isolated
UNREGULATED DC/DC CONVERTERS**

FEATURES

- Up To 89% Efficiency
- Thermal Protection
- Device-to-Device Synchronization
- SO-28 Power Density of 106W/in³ (6.5W/cm³)
- EN55022 Class B EMC Performance
- UL1950 Recognized Component
- JEDEC 14-Pin and SO-28 Packages

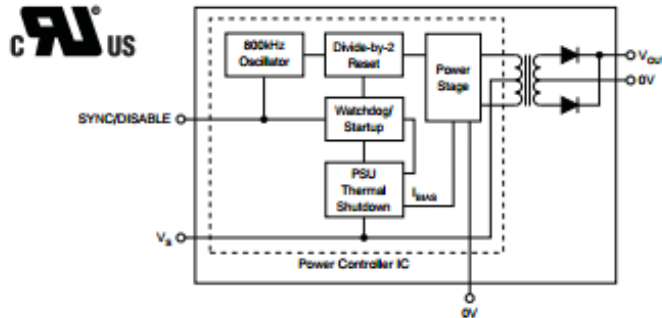
APPLICATIONS


- Point-of-Use Power Conversion
- Ground Loop Elimination
- Data Acquisition
- Industrial Control and Instrumentation
- Test Equipment

DESCRIPTION

The DCP02 series is a family of 2W, isolated, unregulated DC/DC converters. Requiring a minimum of external components and including on-chip device protection, the DCP02 series provides extra features such as output disable and synchronization of switching frequencies.

The use of a highly integrated package design results in highly reliable products with power densities of 79W/in³ (4.8W/cm³) for DIP-14, and 106W/in³ (6.5W/cm³) for SO-28. This combination of features and small size makes the DCP02 suitable for a wide range of applications.



 Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet. All trademarks are the property of their respective owners.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of the Texas Instruments standard warranty. Deviations may occur due to

Copyright © 2006–2008, Texas Instruments Incorporated

ANEXO c. Hoja de datos DAC8871



16-Bit, Single-Channel, $\pm 18V$ Output (Unbuffered), Ultra-Low Power, Serial Interface DIGITAL-TO-ANALOG CONVERTER

FEATURES

- 16-Bit Resolution
- Output: $\pm 18V$ for $\pm 18V$ Reference Input
- $\pm 18V$ Supply Operation
- Very Low Power
- High Accuracy INL: 1LSB
- Low Noise: $10nV/\sqrt{Hz}$
- Fast Settling: $1\mu s$ to 1LSB
- Fast SPI™ Interface: Up To 50MHz
- 16-Pin TSSOP Package
- Selectable Reset to Zero or Midscale

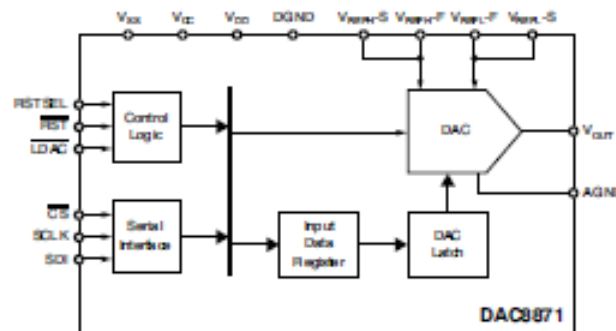
APPLICATIONS

- Portable Equipment
- Automatic Test Equipment
- Industrial Process Control
- Data Acquisition Systems
- Optical Networking

DESCRIPTION

The DAC8871 is a 16-bit, single-channel, serial input, voltage output digital-to-analog converter (DAC). The output range is determined by the reference voltage, V_{REFH} and V_{REFL} . By properly selecting the reference, the output can be unipolar or bipolar, and up to $\pm 18V$. The DAC8871 provides excellent linearity (1LSB INL), low noise, and fast settling ($1\mu s$ to 1LSB of full scale output) over the specified temperature range of $-40^{\circ}C$ to $+105^{\circ}C$. The output is unbuffered, which reduces the power consumption and the error introduced by the buffer. This device features a standard high-speed clock (up to 50MHz), and a 3V or 5V SPI serial interface to communicate with the DSP or microprocessors. For optimum performance, a set of Kelvin connections to external reference are provided.

The DAC8871 is available in a TSSOP-16 package.



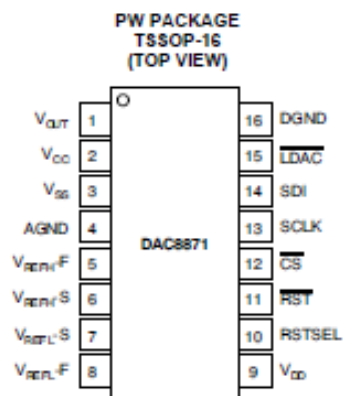
Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

TI DSP is a trademark of Texas Instruments.
 SPI, QSPI are trademarks of Motorola, Inc.
 Microwire is a trademark of National Semiconductor.
 All other trademarks are the property of their respective owners.

PRODUCTION DATA information is current as of publication date.
 Products conform to specifications per the terms of the Texas
 Instruments standard warranty. Production processing does not
 necessarily include testing of all parameters.

Copyright © 2007–2008, Texas Instruments Incorporated

PIN CONFIGURATION (NOT TO SCALE)



TERMINAL FUNCTIONS

TERMINAL NO.	NAME	DESCRIPTION
1	V_{OUT}	Analog output of the DAC
2	V_{CC}	Positive analog power supply: +15V
3	V_{SS}	Negative analog power supply: -15V
4	AGND	Analog ground
5	V_{REFH-F}	V_{REFH} reference input (Force). Connect to external V_{REFH} .
6	V_{REFH-S}	V_{REFH} reference input (Sense). Connect to external V_{REFH} .
7	V_{REFL-S}	V_{REFL} reference input (Sense). Connect to external V_{REFL} .
8	V_{REFL-F}	V_{REFL} reference input (Force). Connect to external V_{REFL} .
9	V_{DD}	Digital power. +5V for 5V interface logic; +3V for 3V logic.
10	RSTSEL	Power-On-Reset select. Determines V_{OUT} after power-on reset. If tied to V_{DD} , the DAC latch is set to mid-scale after power-on, and V_{OUT} is $(V_{REFH} - V_{REFL})/2$. If tied to DGND, the DAC latch is cleared ('0'), and V_{OUT} is V_{REFL} .
11	\overline{RST}	Reset (active low)
12	\overline{CS}	Chip select input (active low). Data are not clocked into SDI unless \overline{CS} is low.
13	SCLK	Serial clock input
14	SDI	Serial data input. Data are latched into input register on the rising edge of SCLK.
15	\overline{LDAC}	Load DAC control input (active low). When \overline{LDAC} is low, the DAC latch is simultaneously updated with the content of the input register.
16	DGND	Digital ground

TYPICAL CHARACTERISTICS (continued)

At $T_A = +25^\circ\text{C}$, $V_{DD} = +5\text{V}$, $V_{CC} = +15\text{V}$, $V_{SS} = -15\text{V}$, $V_{REFH} = +10\text{V}$, and $V_{REFL} = -10\text{V}$, unless otherwise noted.

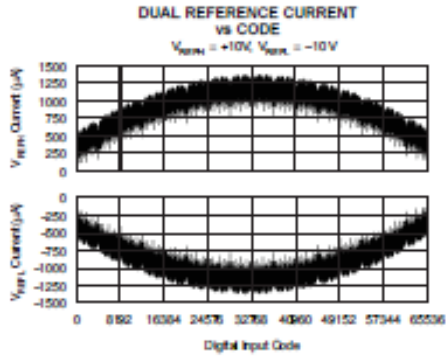


Figure 27.

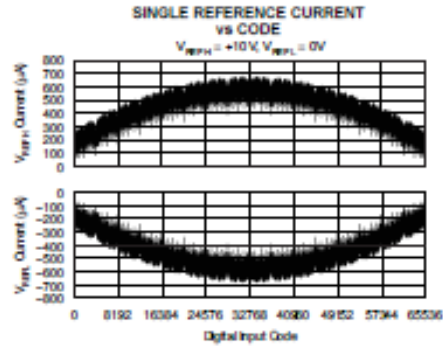


Figure 28.

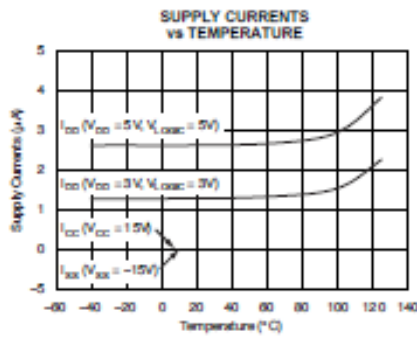


Figure 29.

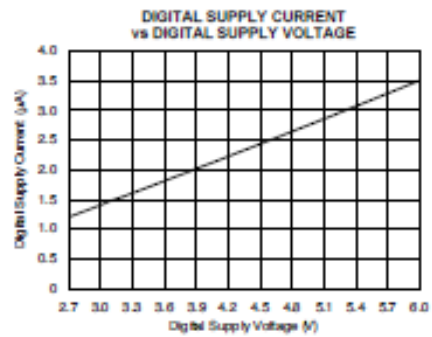


Figure 30.

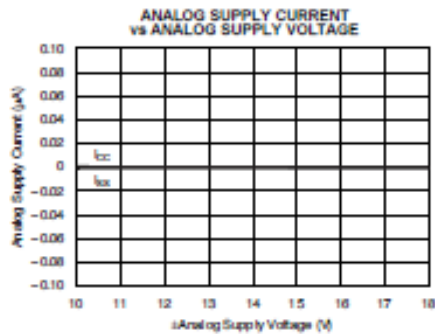


Figure 31.

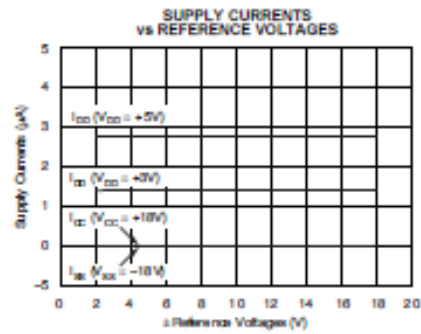


Figure 32.

THEORY OF OPERATION

GENERAL DESCRIPTION

The DAC8871 is a 16-bit, single-channel, serial-input, voltage-output DAC. It operates from a dual power supply ranging from $\pm 13.5\text{V}$ to $\pm 19.8\text{V}$, and typically consumes $10\mu\text{A}$. The output range is from V_{REFL} to V_{REFH} . Data are written to this device in a 16-bit word format, via an SPI serial interface. To ensure a known power-up state, the DAC8871 is designed with a power-on reset function. After power on, the state of the RSTSEL pin sets the value of the input register and DAC latch, which sets the output state of the V_{OUT} pin. Refer to the [Power-On Reset and Hardware Reset](#) section for more details.

Kelvin sense connections for the reference and analog ground are also included.

DIGITAL-TO-ANALOG SECTIONS

The DAC architecture consists of two matched DAC sections and is segmented. A simplified circuit diagram is shown in [Figure 38](#). The four MSBs of the 16-bit data word are decoded to drive 15 switches, E1 to E15. Each of these switches connects one of 15 matched resistors to either V_{REFH} or V_{REFL} . The remaining 12 bits of the data word drive switches S0 to S11 of a 12-bit voltage mode R-2R ladder network.

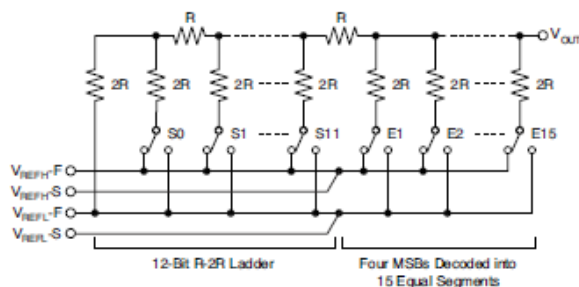


Figure 38. DAC Architecture

OUTPUT RANGE

The output of the DAC is:

$$V_{\text{OUT}} = \frac{V_{\text{REFH}} - V_{\text{REFL}}}{65536} \times \text{Code} + V_{\text{REFL}} \quad (1)$$

Where Code is the decimal data word loaded to the DAC latch.

For example, if V_{REFH} is +10V, and V_{REFL} is -10V, the range of V_{OUT} is from -10V (Code = 0000h) to +10V (Code = FFFFh).

The range of V_{REFL} is from -18V to $(V_{\text{REFH}} - 1.25\text{V})$, and the range of V_{REFH} is 0V to +18V. The output from the DAC8871 can be unipolar (from 0V to +18V) or bipolar by setting the proper V_{REFL} and V_{REFH} values.

APPLICATION INFORMATION

REFERENCE INPUT

The DAC full-scale output voltage is determined by the reference voltage, as shown in the [Output Range](#) section.

Reference input V_{REFH} can be any voltage from 0V to +18V. Reference input V_{REFL} can be any voltage from -18V to $(V_{REFH} - 1.25V)$. The current into the V_{REFH} input and out of V_{REFL} depends on the DAC output voltages. Refer to [Figure 27](#) and [Figure 28](#) for details. The reference input appears as a varying load to the reference. If the reference can sink or source the required current, a reference buffer is not required. The DAC8871 features a reference drive (force) and sense connection that minimizes the internal errors caused by the changing reference current and the circuit impedances. [Figure 40](#) shows a typical reference configuration.

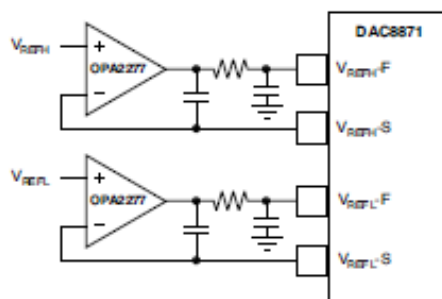


Figure 40. Buffered Reference Connection

POWER-SUPPLY BYPASSING

For accurate, high-resolution performance, bypassing the supply pins with a 10 μ F tantalum capacitor in parallel with a 0.1 μ F ceramic capacitor is recommended.

POWER-SUPPLY SEQUENCING

The analog supplies (V_{CC} and V_{DD}) must power up before the digital supply (V_{DD}). All three supplies must power up before the reference voltages (V_{REFH} and V_{REFL}) are applied. Additionally, because the DAC input shift register is not reset during a power-on reset (or a hardware reset through the \overline{RST} pin), the \overline{CS} pin must not be unintentionally asserted during power-up of the device. It is recommended that the \overline{CS} pin be connected to V_{DD} through a pull-up resistor to avoid improper power-up.

Likewise, the state of the \overline{LDAC} pin must not be accidentally changed during power-up. It is recommended that the \overline{LDAC} pin be connected to V_{DD} through a pull-up resistor, unless it is permanently tied to ground.

To ensure that the ESD protection circuitry of this device is not activated, all other digital pins must be kept at ground potential until V_{DD} is applied.



REF5010, REF5020
REF5025, REF5030
REF5040, REF5045, REF5050

www.ti.com

SBOS410E – JUNE 2007 – REVISED JUNE 2010

Low-Noise, Very Low Drift, Precision VOLTAGE REFERENCE

Check for Samples: REF5010, REF5020, REF5025, REF5030, REF5040, REF5045, REF5050

FEATURES

- **LOW TEMPERATURE DRIFT:**
 - High-Grade: 3ppm/°C (max)
 - Standard-Grade: 8ppm/°C (max)
- **HIGH ACCURACY:**
 - High-Grade: 0.05% (max)
 - Standard-Grade: 0.1% (max)
- **LOW NOISE:** 3 μ V_{pp}/V
- **EXCELLENT LONG-TERM STABILITY:**
 - 5ppm/1000hr (typ) after 1000 hours
- **HIGH OUTPUT CURRENT:** \pm 10mA
- **TEMPERATURE RANGE:** –40°C to +125°C

APPLICATIONS

- 16-BIT DATA ACQUISITION SYSTEMS
- ATE EQUIPMENT
- INDUSTRIAL PROCESS CONTROL
- MEDICAL INSTRUMENTATION
- OPTICAL CONTROL SYSTEMS
- PRECISION INSTRUMENTATION

DESCRIPTION

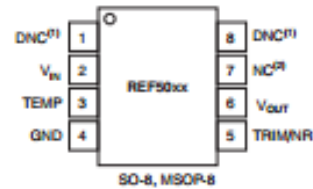
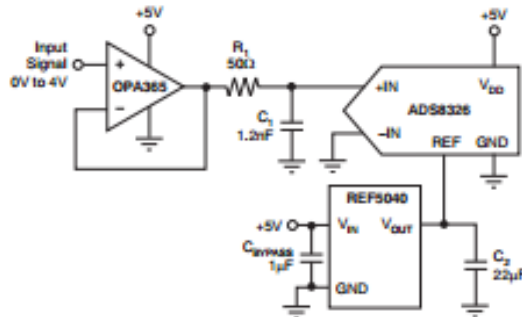
The REF50xx is a family of low-noise, low-drift, very high precision voltage references. These references are capable of both sinking and sourcing, and are very robust with regard to line and load changes.

Excellent temperature drift (3ppm/°C) and high accuracy (0.05%) are achieved using proprietary design techniques. These features, combined with very low noise, make the REF50xx family ideal for use in high-precision data acquisition systems.

Each reference voltage is available in both standard- and high-grade versions. They are offered in MSOP-8 and SO-8 packages, and are specified from –40°C to +125°C.

REF50xx Family

MODEL	OUTPUT VOLTAGE
REF5020	2.048V
REF5025	2.5V
REF5030	3.0V
REF5040	4.096V
REF5045	4.5V
REF5050	5.0V
REF5010	10.0V



NOTES: (1) DNC = Do not connect.
(2) NC = No internal connection.

APPLICATION CIRCUITS

NEGATIVE REFERENCE VOLTAGE

For applications requiring a negative and positive reference voltage, the REF50xx and OPA735 can be used to provide a dual-supply reference from a 5V supply. Figure 33 shows the REF5025 used to provide a 2.5V supply reference voltage. The low drift performance of the REF50xx complements the low offset voltage and zero drift of the OPA735 to provide an accurate solution for split-supply applications. Care must be taken to match the temperature coefficients of R_1 and R_2 .

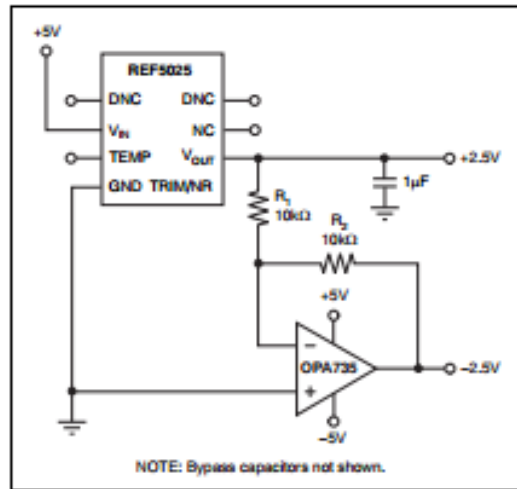


Figure 33. The REF5025 and OPA735 Create Positive and Negative Reference Voltages

DATA ACQUISITION

Data acquisition systems often require stable voltage references to maintain accuracy. The REF50xx family features low noise, very low drift, and high initial accuracy for high-performance data converters. Figure 34 shows the REF5040 in a basic data acquisition system.

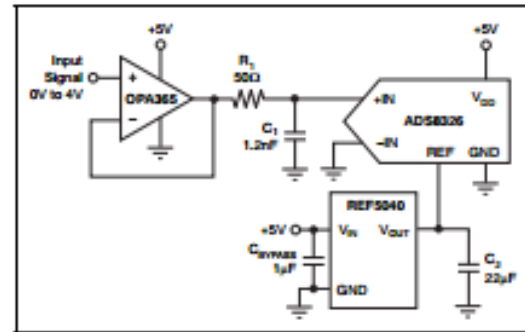
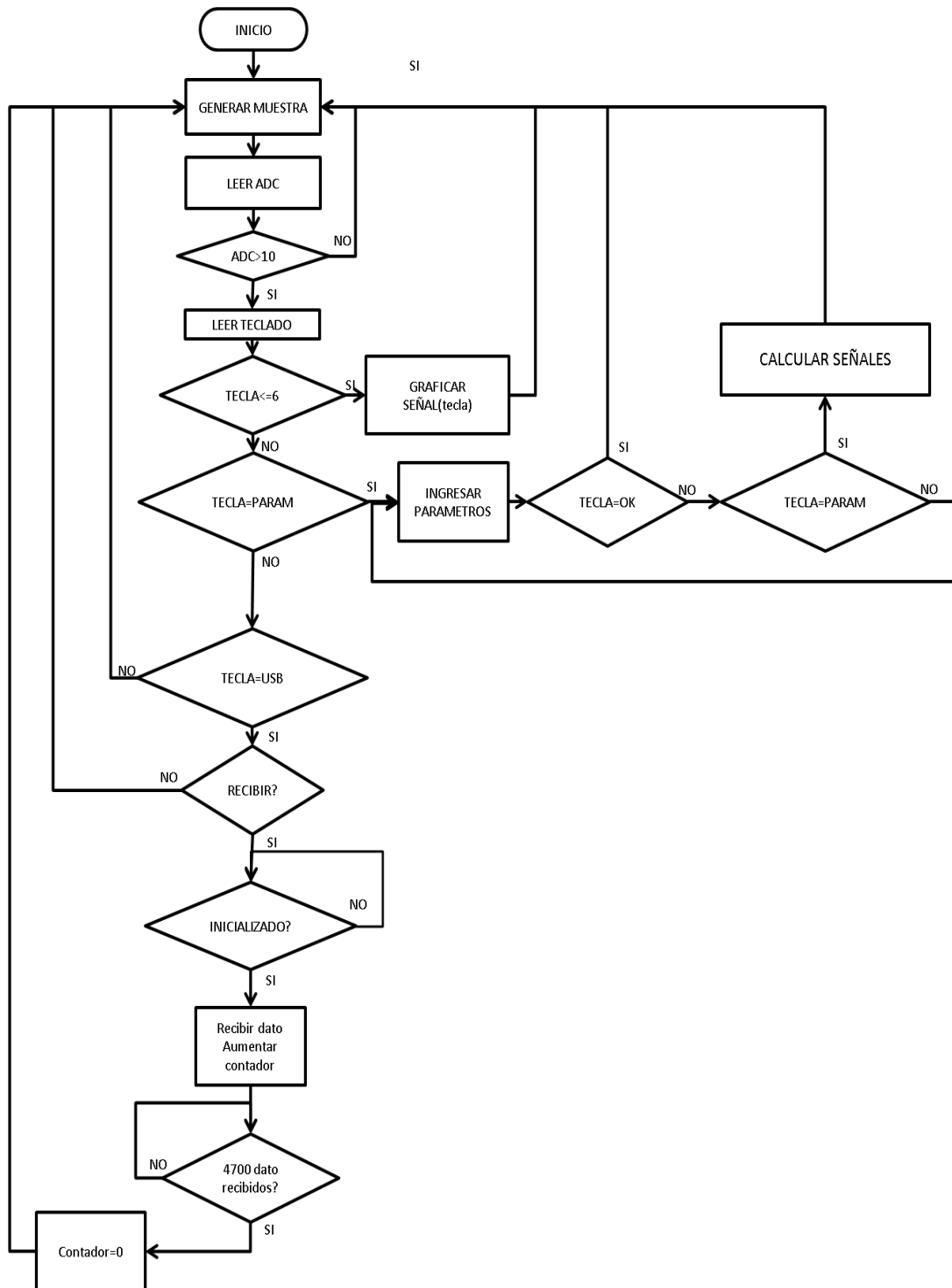
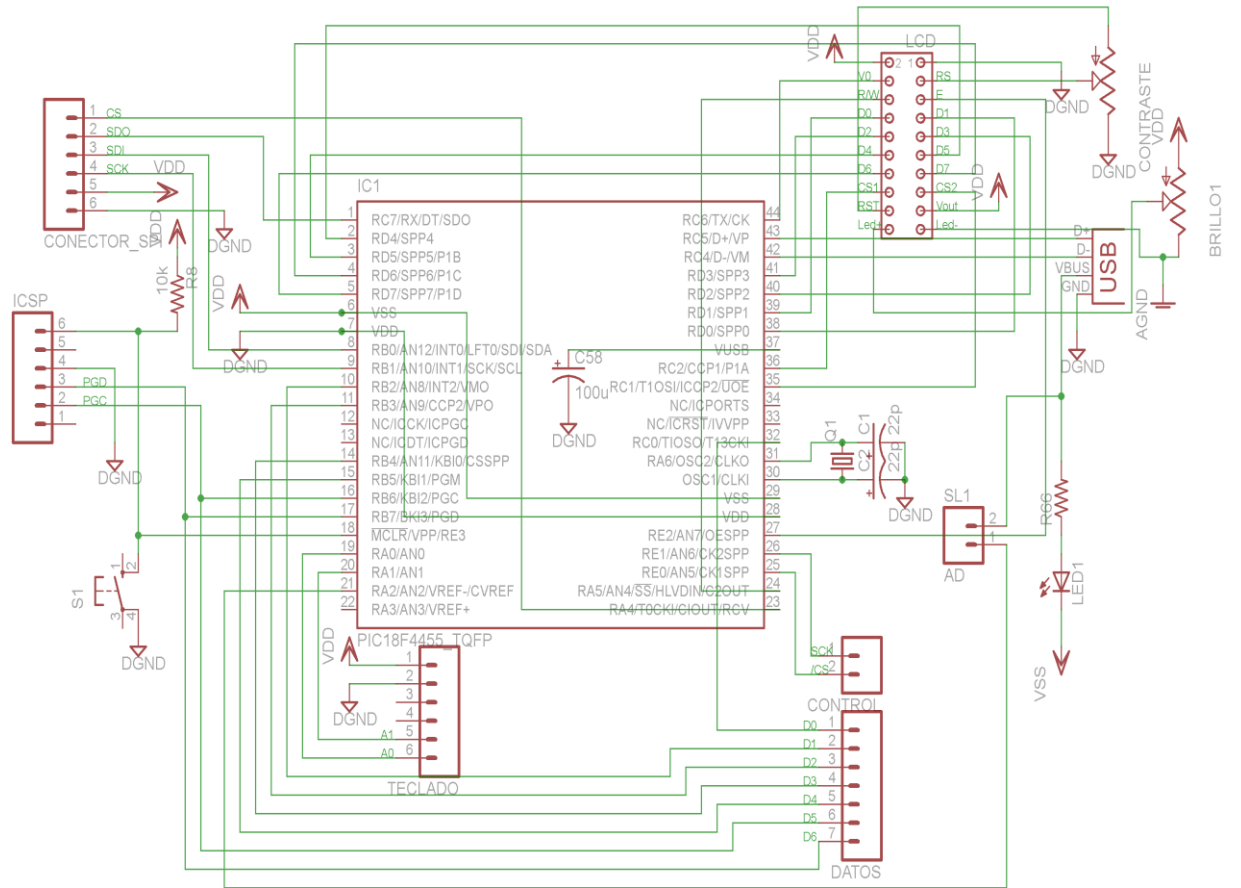


Figure 34. Basic Data Acquisition System

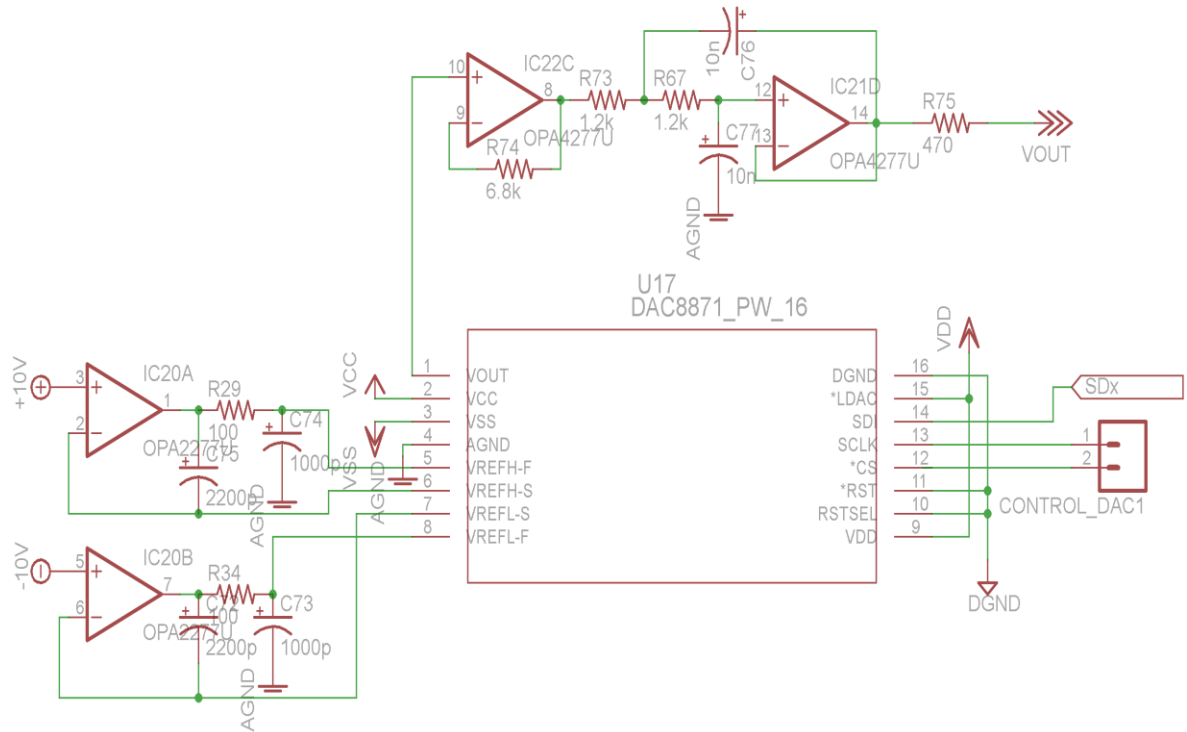
ANEXO e. Diagrama de funcionamiento del generador de señales trifásico.



ANEXO f. Esquema circuito de control generador de señales trifásico.



ANEXO g. Esquema circuito canal conversor digital analógico completo.



ANEXO h. Foto Proyecto

