



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería Mecánica Eléctrica

**DISEÑO E IMPLEMENTACIÓN DE UN ANALIZADOR DE VIBRACIONES
PARA EL MANTENIMIENTO PREDICTIVO EN SISTEMAS MECÁNICOS**

Ronald Taoffik Guerra Bailey

Asesorado por el ing. Julio Cesar Solares Peñate

Guatemala, septiembre de 2020

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**DISEÑO E IMPLEMENTACIÓN DE UN ANALIZADOR DE VIBRACIONES
PARA EL MANTENIMIENTO PREDICTIVO EN SISTEMAS MECÁNICOS**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA

POR

RONALD TAOFFIK GUERRA BAILEY

ASESORADO POR EL ING. JULIO CESAR SOLARES PEÑATE

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO ELECTRONICO

GUATEMALA, SEPTIEMBRE DE 2020

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANA	Inga. Aurelia Anabela Cordova Estrada
VOCAL I	Ing. José Francisco Gómez Rivera
VOCAL II	Ing. Mario Renato Escobedo Martínez
VOCAL III	Ing. José Milton de León Bran
VOCAL IV	Br. Christian Moisés de la Cruz Leal
VOCAL V	Br. Kevin Armando Cruz Lorente
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Pedro Antonio Aguilar Polanco
EXAMINADOR	ing. Helmut Federico Chicol Cabrera
EXAMINADOR	ing. Julio César Solares Peñate
EXAMINADOR	ing. Marvin Marino Hernández Fernández
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

DISEÑO E IMPLEMENTACIÓN DE UN ANALIZADOR DE VIBRACIONES PARA EL MANTENIMIENTO PREDICTIVO EN SISTEMAS MECÁNICOS

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería mecánica eléctrica, con fecha 15 de mayo de 2019.

Ronald Taoffik Guerra Bailey

Guatemala, 11 de noviembre de 2019

Señor
Coordinador del Área de Electrónica
Escuela de Ingeniería Mecánica Eléctrica
Facultad de Ingeniería, USAC.

Estimado Ingeniero:

Por este medio me permito dar aprobación al trabajo de Graduación titulado: **DISEÑO E IMPLEMENTACIÓN DE UN ANALIZADOR DE VIBRACIONES PARA EL MANTENIMIENTO PREDICTIVO EN SISTEMAS MECÁNICOS**, desarrollado por el estudiante **Ronald Taoffik Guerra Bailey**, ya que considero que cumple con los requisitos establecidos.

Por lo tanto, el autor de este trabajo y yo como asesor, nos hacemos responsables del contenido y conclusiones del mismo.

Sin otro en particular, aprovecho la oportunidad para saludarlo.

ID Y ENSEÑAD A TODOS


Ing. Julio César Solares Peñate
Asesor

Julio César Solares Peñate
Ingeniero Mecánico Electricista
Colegiado No. 2330



FACULTAD DE INGENIERIA

Guatemala, 15 de noviembre de 2019

Señor Director
Armando Alonso Rivera Carrillo
Escuela de Ingeniería Mecánica Eléctrica
Facultad de Ingeniería, USAC


Estimado Señor Director:

Por este medio me permito dar aprobación al Trabajo de Graduación titulado **DISEÑO E IMPLEMENTACIÓN DE UN ANALIZADOR DE VIBRACIONES PARA EL MANTENIMIENTO PREDICTIVO EN SISTEMAS MECÁNICOS**, desarrollado por el estudiante **Ronald Taoffik Guerra Bailey**, ya que considero que cumple con los requisitos establecidos.

Sin otro particular, aprovecho la oportunidad para saludarlo.

Atentamente,

ID Y ENSEÑAD A TODOS


Ing. Julio César Solares Peñate
Coordinador de Electrónica





REF. EIME 04. 2020.

El Director de la Escuela de Ingeniería Mecánica Eléctrica, después de conocer el dictamen del Asesor, con el Visto bueno del Coordinador de Área, al trabajo de Graduación del estudiante: RONALD TAOFFIK GUERRA BAILEY titulado: DISEÑO E IMPLEMENTACIÓN DE UN ANALIZADOR DE VIBRACIONES PARA EL MANTENIMIENTO PREDICTIVO EN SISTEMAS MECÁNICOS, procede a la autorización del mismo.

Ing. Armando Alonso Rivera Carrillo



GUATEMALA, 29 DE ENERO 2020.

DTG. 249.2020.

La Decana de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería Eléctrica, al Trabajo de Graduación titulado: **DISEÑO E IMPLEMENTACIÓN DE UN ANALIZADOR DE VIBRACIONES PARA EL MANTENIMIENTO PREDICTIVO EN SISTEMAS MECÁNICOS**, presentado por el estudiante universitario: **Ronald Taoffik Guerra Bailey**, y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, autoriza la impresión del mismo.

IMPRÍMASE:



Inga. Anabela Cordova Estrada
Decana



Guatemala, septiembre de 2020

AACE/asga

ACTO QUE DEDICO A:

Dios

Por ser una importante influencia en mi carrera,
entre otras cosas.

Mi madre

Aura Leticia Bailey. Su amor será siempre mi
inspiración.

Mi familia

Por ser una importante influencia en mi carrera,
entre otras cosas.

AGRADECIMIENTOS A:

**Universidad de San
Carlos de Guatemala**

Por ser quien me dio la oportunidad de llevar a cabo mi carrera, entre otras cosas.

**Mis amigos de la
facultad**

Nefi Velarde, Óscar Valverth, Ariel Chitay, Danilo Escobar.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES	VII
LISTA DE SÍMBOLOS	XI
GLOSARIO	XIII
RESUMEN	XV
OBJETIVOS.....	XVII
INTRODUCCIÓN	XIX
1. CONCEPTOS GENERALES DE MANTENIMIENTO PREDICTIVO	1
1.1. Mantenimiento predictivo.....	1
1.2. Mantenimiento predictivo en sistemas mecánicos.....	1
1.2.1. Indicadores	2
1.2.1.1. Vibraciones.....	2
1.2.2. Equipo de recolección de datos.....	3
1.3. El medidor de vibraciones	3
1.4. Analizadores de vibración portátiles	4
1.5. Características de la señal obtenida de la medición de vibraciones	4
1.5.1. Frecuencia (f).....	4
1.5.2. Desplazamiento (D)	5
1.5.3. Velocidad (V)	5
1.5.4. Aceleración (A)	5
1.6. Funciones de medición.....	5
1.6.1. Amplitud de pico (V_{Peak})	5
1.6.2. Valor cuadrático medio (V_{rms})	6

2.	CONCEPTOS GENERALES DEL ANÁLISIS DE VIBRACIONES	9
2.1.	Vibración	9
2.1.1.	Caso discreto	9
2.1.2.	Caso continuo	15
2.2.	Análisis de Fourier.....	17
2.2.1.	Series de Fourier.....	18
2.2.2.	Transformada de Fourier.....	20
3.	COMPONENTES DEL CIRCUITO.....	25
3.1.	Sensor.....	25
3.1.1.	Micrófono de estado sólido o piezoeléctrico.....	25
3.1.2.	Micrófono de bobina móvil.....	26
3.1.3.	Acelerómetro	27
3.2.	Filtros	29
3.2.1.	Paso bajo	30
3.2.2.	Filtro paso altos	30
3.2.3.	Filtros paso banda.....	31
3.2.4.	Filtros rechaza banda	31
3.3.	Amplificador operacional	32
3.3.1.	Funcionamiento.....	33
3.4.	Microcontrolador	33
3.4.1.	Microcontrolador SAM3X8E	34
3.5.	Pantalla de cristal líquido	34
3.6.	Construcción	35
3.6.1.	Sensor.....	35
3.6.2.	Filtro paso bajo 1	35
3.6.3.	Amplificador operacional	36
3.6.4.	Filtro paso bajo 2.....	36
3.6.5.	Convertor analógico digital (ADC)	36

3.6.6.	Procesamiento digital (FFT).....	36
3.6.7.	Pantalla TFT	36
4.	SENSOR	37
4.1.	Acelerómetros de fuerza balanceada	37
4.2.	Mems.....	38
4.3.	ADXL335	39
4.3.1.	Descripción de los pines.....	41
4.3.1.1.	Pin 1, 4, 9, 11, 13 y 16 NC.....	41
4.3.1.2.	Pin 2 ST.....	41
4.3.1.3.	Pines 3, 5, 6 y 7 COM.....	45
4.3.1.4.	Pines 8, 10 y 12 X, Y, Z OUT.....	45
4.3.1.5.	Pines 14 y 15 VS	45
4.3.2.	Descripción general del circuito ADXL335.....	45
4.3.3.	Teoría de operación.....	46
4.3.4.	Sensor mecánico	47
4.3.5.	Desempeño	47
4.3.6.	Desacoplo de fuente de voltaje.....	48
4.3.7.	Ajustar el ancho de banda usando capacitores a la salida	48
4.3.8.	Autoprueba	48
4.3.9.	Diseño del filtro seleccionando valores comerciales de capacitores	49
4.3.10.	Utilizando el circuito con voltajes distintos de 3 V...	50
5.	MICROCONTROLADOR.....	51
5.1.	Partes de un microcontrolador.....	51
5.1.1.	Bus de direcciones	52
5.1.2.	Bus de datos.....	53

5.1.3.	Bus de control	54
5.2.	Arduino DUE	55
5.2.1.	Componentes del Arduino Due	57
5.2.1.1.	Circuito de programación	58
5.2.1.2.	Reguladores de voltaje.....	60
6.	PANTALLA TFT	63
6.1.	ITDB02-4.3.....	66
6.1.1.	SSD1963.....	69
6.2.	Módulo de conexión ITDB02 MEGA Shield.....	70
7.	LECTURA DE EJE DEL SENSOR.....	73
7.1.	Descripción del ADC (convertor analógico a digital) del SAM3X8EA	73
7.2.	Características del ADC.....	74
7.3.	Ecuación de transferencia del digitalizador o convertor analógico digital	75
7.4.	Ecuación de transferencia del sensor ADXL335	76
7.5.	Ecuación de transferencia total	76
8.	ALGORITMO DE LA TRANSFORMADA DE FOURIER	81
8.1.	Desarrollo de la transformada de Fourier.....	81
8.2.	Datos precalculados de la transformada de Fourier.....	83
8.3.	Algoritmo no optimizado de la transformada de Fourier.....	86
8.3.1.	Parte real.....	86
8.3.2.	Parte imaginaria	87
8.3.3.	Combinación de parte imaginaria y parte real	87
8.4.	Biblioteca SplitRadixRealP.h para lenguaje c++	89
8.5.	Biblioteca UTFT.h para lenguaje c++	92

9.	DESARROLLO.....	95
9.1.	Sistema mecánico seleccionado para el experimento.....	95
9.1.1.	Motor trifásico	97
9.1.2.	Controlador de velocidad para motor trifásico	98
9.1.2.1.	Generador de señal de onda cuadrada.....	98
9.1.2.2.	Control de velocidad electrónico.....	99
9.1.3.	Sistema de mecánico generador de vibraciones armado	100
9.2.	Montaje de los módulos que componen el analizador de vibraciones	101
9.2.1.	Acoplamiento del sensor a un sistema mecánico.	104
	CONCLUSIONES	107
	RECOMENDACIONES.....	109
	BIBLIOGRAFÍA.....	111
	APÉNDICE.....	117

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Valor pico	6
2.	Valor cuadrático medio.....	7
3.	Oscilador único	9
4.	Aplicación de una fuerza a la masa entre dos resortes.....	10
5.	Sistema de 2 masas y tres resortes	10
6.	Sistema de dos masas acopladas moviéndose bajo las mismas condiciones	12
7.	Variación de velocidad y dirección diferentes en cada masa	12
8.	Comparación de los modos normales	13
9.	Múltiples modos normales del sistema.....	14
10.	Múltiples modos normales del sistema con condiciones iniciales	14
11.	Aplicación para observar los distintos modos normales.....	15
12.	Infinitos modos normales se ven como una cuerda	16
13.	Representación del número de onda	17
14.	Señal de onda cuadrada por partes	19
15.	Señal de onda cuadrada periódica y continua en el tiempo	20
16.	Relación entre el tiempo y la frecuencia.....	21
17.	Amplificador operacional	32
18.	Diagrama de bloques del sistema	35
19.	Circuito ADXL335, descripción y configuración de pines	40
20.	Procedimiento de autocomprobación para el acelerómetro	44
21.	Diagrama de bloques funcional	46
22.	Diagrama interno de un microcontrolador	51

23.	Diagrama de bus de direcciones.....	53
24.	Diagrama de bus de datos.....	54
25.	Diagrama de bus de control.....	55
26.	Placa Arduino Due.....	55
27.	Diagrama de pines del Arduino Due.....	58
28.	Circuito para el microcontrolador auxiliar.....	59
29.	Circuito regulador de voltaje de 12-7 V a 5 V.....	60
30.	Circuito regulador de voltaje de 12-7 V a 5 V.....	61
31.	Célula de cristal líquido.....	64
32.	Pixel.....	65
33.	Módulo ITDB02-4.3 vista frontal.....	66
34.	Módulo ITDB02-4.3 vista trasera.....	67
35.	Módulo ITDB02-4.3 diagrama de pines.....	68
36.	Diagrama de bloques del SSD1963.....	70
37.	ITDB02 MEGA Shield (IM120417024) vista delantera.....	70
38.	ITDB02 MEGA Shield (IM120417024) vista trasera.....	71
39.	Diagrama de bloques del ADC.....	74
40.	Motor de vibración para teléfono celular (MOT-2265).....	96
41.	Motor utilizado como generador de vibración (2834/KV1500).....	97
42.	Motor utilizado como generador de vibración (2834/KV1500).....	99
43.	Motor utilizado como generador de vibración (2834/KV1500).....	100
44.	Sistema mecánico.....	101
45.	Sensor.....	102
46.	Arduino Due.....	102
47.	Módulo acoplador de pantalla.....	103
48.	Pantalla TFT.....	103
49.	Circuito armado.....	104
50.	Arduino Due.....	104
51.	Analizador de vibraciones funcionando.....	105

TABLAS

I.	Especificaciones técnicas del Arduino Due.....	56
II.	Características eléctricas del módulo ITDB02-4.3	69

LISTA DE SÍMBOLOS

Símbolo	Significado
$\frac{d}{dt}$	Cambio en el tiempo
$\frac{\partial L}{\partial \dot{q}_s}$	Cambio del lagrangiano con respecto a la posición
$\frac{\partial L}{\partial q_s}$	Cambio del lagrangiano con respecto a la velocidad
bytes	Conjunto de 8 estados digitales
<i>n</i>	Contador de números enteros
RTD	Detector de temperatura resistivo
X	Eje cartesiano horizontal
Y	Eje cartesiano vertical
f	Frecuencia
Cos	Función coseno
Sin	Función senoidal
A	Ganancia de voltaje
Hz	Hertz, número de ciclos en cada segundo
KHz	Kilohertz millar de Hertz
pH	Medida de acidez o alcalinidad
m	Metro
uC	Microcontrolador
mm	Milésimas de metro
mm	Milímetro
nm	Nanómetro
K	Número de onda
π	Número Pi

TFT	Pantalla de cristal líquido de transistores
T	Periodo
PdM	Predictive maintenance (mantenimiento predictivo)
Σ	Sumatoria de términos
<i>t</i>	Tiempo
FFT	Transformada rápida de Fourier
ω	Velocidad angular
V	Voltaje negativo
V_{out}	Voltaje de salida
V+	Voltaje positivo

GLOSARIO

Amplitud	Mide el tamaño máximo de la señal en el eje vertical en una función periódica.
Continuo	Es concepto matemático en el cual se toman todos los valores sin discontinuidad.
Criticidad	Capacidad que tiene el hombre para hacer conscientemente afirmaciones verdaderas cayendo en cuenta de que porque las hace.
Discreto	Es un concepto matemático que denomina un conjunto de datos en los cuales se toman en ciertos números separados la misma cantidad.
Estetoscopio	Instrumento médico en forma de trompetilla que sirve para explorar los sonidos.
Infrarrojo	Radiación del espectro luminoso que tiene mayor longitud de onda y se extiende desde el extremo del rojo visible hacia frecuencias menores.
ISO	International Organization for Standardization (Organización Internacional de Normalización).

Ludeca	Es un proveedor líder de soluciones de mantenimiento predictivo y proactivo.
Mantenimiento	Conservación de una cosa en buen estado o en una situación determinada para evitar su degradación.
Masa	Todo lo que ocupa un lugar en el espacio, no viaja a la velocidad de la luz y es atraído por la gravedad.
Modos	Fenómeno de la física que consiste en tomar un único oscilador.
Monitoreo	supervisión o el control realizado a través de un monitor.
Normatividad	Conjunto de leyes o reglamentos que rigen conductas y procedimientos según los criterios y lineamientos de una institución u organización privada o estatal.
Predictivo	Capacidad de crear un criterio a futuro para el comportamiento de un proceso.
Termografía	La termografía es una técnica que permite determinar temperaturas a distancia y sin necesidad de contacto físico con el objeto a estudiar.

RESUMEN

Este proyecto consistió en el diseño y la implementación de un analizador de vibraciones para sistemas mecánicos; se eligió un sensor de aceleración analógico para medir de forma apropiada las vibraciones producidas; luego se digitalizó la señal utilizando un circuito convertidor de señales analógicas a digitales para que pudieran ser procesadas a través de un micro controlador programado en lenguaje c para obtener la transformada de Fourier casi en tiempo real para visualizar la gráfica de frecuencias en una pantalla de cristal líquido.

En este documento se presentan los códigos y diagramas de circuitos que hacen funcionar el dispositivo completamente.

Se midió las frecuencias generadas al colocar una masa en una doble hélice que se encontraba conectada a un motor trifásico y se hizo la comparación del dispositivo en funcionamiento sin ningún tipo de desbalance y se observaron las variaciones, el diagrama espectral cuando este equipo posee un desbalance en una de sus hélices, se observó una diferencia significativa que coincidía con haber colocado la masa de desbalance, por lo cual se puede dar un mantenimiento predictivo al utilizar este sensor.

OBJETIVOS

General

Proporcionar un documento que muestre como se construye un circuito capaz de digitalizar vibraciones producidas por un sistema mecánico.

Específicos

1. Demostrar qué sensor se adecúa más para la obtención de las vibraciones.
2. Construir un circuito que sea capaz de almacenar y procesar las mediciones del sensor por medio de la transformada de Fourier.
3. Presentar los datos procesados por medio de una pantalla.

INTRODUCCIÓN

Los sistemas mecánicos son conocidos por generar vibraciones los que se producen debido a las tensiones en las fajas, los engranajes, rodamientos cuando estas son accionadas desde una posición de reposo. Cuando un elemento del sistema es accionado desde la posición de reposo se espera que alcance una velocidad constante y se mantenga a esa velocidad por un periodo de tiempo establecido; después, tiene que desacelerar hasta alcanzar de nuevo la posición de reposo e iniciar el proceso nuevamente; esto es lo que genera vibraciones en el sistema mecánico.

Por ejemplo, en una banda transportadora, se espera que un elemento de producción sea colocado constantemente sobre la misma, esta tiene que desplazarlo a una velocidad constante durante un tiempo determinado; cuando esta faja empieza a fallar se pueden percibir ciertas vibraciones minúsculas lo que demuestra que la grasa de la faja ya no la lubrica lo suficiente o que uno de sus rodamientos ya está desgastado y se opone al movimiento.

En el sistema de la faja, al principio, las vibraciones serán tan pequeñas que no será posible observarlas por lo cual es necesario implementar un sensor que sea capaz de medir esas pequeñas vibraciones.

Si el sistema se encuentra funcionando correctamente será posible medir las vibraciones características a través del sensor que al presentarlas en un diagrama espectral podrán ser reconocidas como un patrón de buen funcionamiento del equipo. Con el tiempo, este patrón se estará comparando con el patrón generado por la máquina en posteriores mediciones y la diferencia entre

ambos será suficiente para determinar si la maquinaria necesita ser de nuevo engrasada, o ya sea que haya que reemplazar algunos de sus rodamientos; se puede así prevenir que la producción se detenga por completo.

1. CONCEPTOS GENERALES DE MANTENIMIENTO PREDICTIVO

1.1. Mantenimiento predictivo

El mantenimiento predictivo (PdM) es el conjunto de técnicas que están diseñadas para ayudar a determinar el estado de los equipos en servicio, con el fin de predecir en qué momento se debe realizar el mantenimiento. Este enfoque promete ahorros de coste sobre la rutina o basado en el tiempo de mantenimiento preventivo, ya que las tareas se realizan solamente cuando sea necesario.

1.2. Mantenimiento predictivo en sistemas mecánicos

El objetivo del mantenimiento basado en condición es conocer la condición de la maquinaria, de tal manera que se pueda determinar su operación de manera segura y eficiente. Las técnicas de monitoreo están dirigidas a la medición de variables físicas, que son indicadores de la condición de la máquina y mediante un análisis, efectuar la comparación con valores de referencia de acuerdo a normatividad para determinar si está en buen estado o en condiciones de deterioro.

Esta estrategia asume que hay características medibles y observables que son indicadores de la condición de la maquinaria.

El monitoreo de la condición es una herramienta poderosa para una mayor productividad y competitividad. La selección de la maquinaria para ser incluida en estos programas depende de un análisis de su criticidad, su costo,

disponibilidad, sus requerimientos de seguridad y ambientales, la confiabilidad esperada y el impacto de su falla, entre otros.

1.2.1. Indicadores

La toma de vibraciones es ampliamente utilizada en mantenimiento, con el objetivo de vigilar el comportamiento dinámico mecánico de las máquinas rotativas. La correcta utilización de esta tecnología en términos de cantidad de pruebas (tendencia) y el análisis espectral aportan información valiosa en el diagnóstico prematuro de fallas en los elementos rodantes, engranajes, bombas, compresores, ventiladores y muchas otras máquinas rotativas. Se toman mediciones de amplitud versus frecuencia de vibración, en las direcciones horizontal, vertical y axial, en cada punto.

1.2.1.1. Vibraciones

La vibración es una característica que se encuentra en prácticamente todas las máquinas industriales. Cuando la vibración aumenta más allá de los niveles normales, puede indicar el desgaste normal o puede ser que muestra el origen del problema y señalar la necesidad de una evaluación adicional para identificar las causas subyacentes, o para la acción inmediata de mantenimiento que deban tomarse. Para determinar esto, se requiere el análisis de vibración exacta.

La vibración es probablemente el mejor indicador de un problema de alineación, los niveles de vibración radial alta en el extremo de acoplamiento de una máquina o niveles de vibración axial elevada en el extremo de empuje, ambos son excelentes indicaciones de problemas de alineación.

Después de la detección, se puede corregir el problema con una herramienta de alineación.

1.2.2. Equipo de recolección de datos

Con un colector de datos de vibración, se puede decir, con un alto grado de certeza, por la fase del canal transversal que tiene un problema de falta de alineación, explica Alan Luedeking, vicepresidente de Ludeca y analista de vibraciones, certificado ISO, nivel I. Otras tecnologías, como la termografía de infrarrojos, puede ser útil, también. Al ver a un acoplamiento que es demasiado caliente le avisará de que usted tiene un problema, pero no necesariamente va a revelar si la causa es la falta de alineación, lubricación insuficiente, o flojedad, mientras que el análisis de vibración puede revelar qué exactamente. Su programa de confiabilidad debe incluir tanto en línea y monitoreo de vibración de mano y sistemas de tendencia para detectar estos problemas mucho antes de que se ha afectado el inventario de repuestos innecesariamente.

1.3. El medidor de vibraciones

Durante años, los técnicos se esforzaron por descifrar vibración de la máquina una señal temprana de un fallo mecánico con ayuda de una espiga, un destornillador o un estetoscopio. Aunque se usan ampliamente, estas soluciones requieren un oído entrenado y con frecuencia no identifican la causa de una condición mecánica, que deja a los equipos de mantenimiento de adivinanzas e incapaz de tomar las decisiones correctas de reparación. Muchos equipos de mantenimiento simplemente necesitan respuestas rápidas y viables. Ahí es donde medidores de vibración encajan.

Un analizador de vibraciones utilizable en práctica debe ser capaz de rastrear el análisis de los diversos componentes de vibración en toda la gama de velocidades del motor y de los miembros asociados.

1.4. Analizadores de vibración portátiles

Los departamentos de mantenimiento de hoy se les pide funcionar con menos personal y presupuestos más pequeños que nunca. En este entorno de más con menos, el personal de mantenimiento no puede permitirse el lujo de perseguir continuamente el siguiente desglose. Por lo que se necesita identificar rápidamente y con precisión fallos en desarrollo y luego llegar a la causa raíz del problema de la maquinaria, de modo que pueda repararse. Una solución tecnológica eficaz debe ser fácil de operar; reducir los requisitos de formación mientras que proporciona información rápida y efectiva a priorizar las actividades de mantenimiento.

1.5. Características de la señal obtenida de la medición de vibraciones

Vibración significa el estado de un objeto que se mueve hacia atrás, adelante repetidamente, derecha e izquierda, o arriba y abajo; generalmente se expresa mediante frecuencia, desplazamiento, velocidad y aceleración. Estos 4 elementos generalmente se denotan como F, D, V, A. Esto se ilustra simplemente como un resorte y una masa. Cuando la masa se baja de la posición inicial y se libera, la masa se mueve igual que la forma de onda de vibración que se muestra en el gráfico de la derecha.

1.5.1. Frecuencia (f)

Frecuencia significa la cantidad de veces que un objeto vibrante genera un movimiento repetitivo en 1 segundo.

Unidad: Hz

1.5.2. Desplazamiento (D)

Desplazamiento significa la amplitud (distancia) entre los picos de vibración.

Unidad: μm , mm

1.5.3. Velocidad (V)

Velocidad significa el cambio en la velocidad de desplazamiento (D) a tiempo.

Unidad: mm / s, cm / s

1.5.4. Aceleración (A)

Aceleración significa la velocidad de cambio de velocidad (V) a tiempo.

Unidad: m / s², g

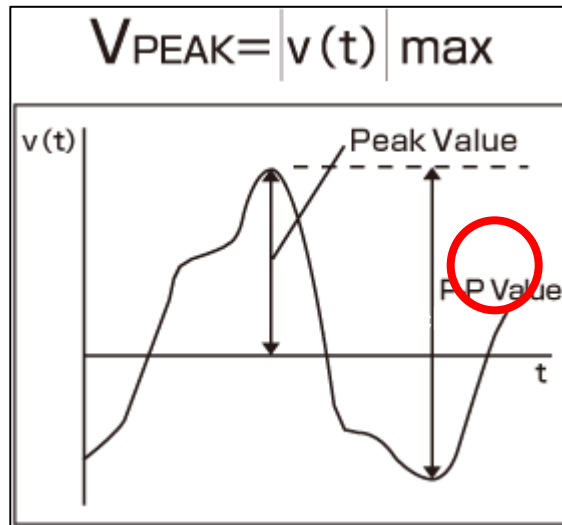
1.6. Funciones de medición

Estos son los medios para expresar la vibración en valor absoluto; se basa en las formas de onda medidas por los modos que se muestran arriba.

1.6.1. Amplitud de pico (V_{Peak})

Valor máximo en un tiempo determinado. Se utiliza para medir choques u ondas que son bastante estables.

Figura 1. Valor pico

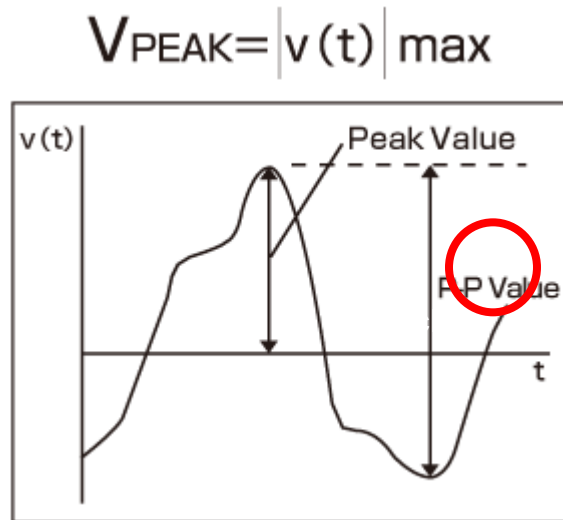


Fuente: Semantic Scholar. *Análisis de vibraciones para fines de mantenimiento predictivo*.
https://www.imv.co.jp/e/pr/vibration_measuring/chapter03/. Consulta: 2 de diciembre de 2018.

1.6.2. Valor cuadrático medio (V_{rms})

Valor cuadrático medio de los valores instantáneos en una determinada duración de tiempo. Se relaciona con el poder de la ola. El valor rms de la velocidad es uno de los factores importantes para el diagnóstico de estado de la maquinaria.

Figura 2. **Valor cuadrático medio**



Fuente: Semantic Scholar. *Análisis de vibraciones para fines de mantenimiento predictivo*.
https://www.imv.co.jp/e/pr/vibration_measuring/chapter03/. Consulta: 2 de diciembre de 2018.

2. CONCEPTOS GENERALES DEL ANÁLISIS DE VIBRACIONES

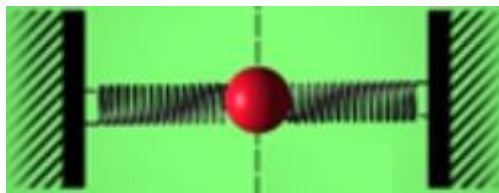
2.1. Vibración

La vibración es el movimiento de un cuerpo alrededor de un punto de reposo. Para entender mejor este concepto se estudiarán los casos discreto y continuo.

2.1.1. Caso discreto

Lo más sencillo que puede vibrar es un único oscilador, esto puede imaginarse como un centro con masa unido por medio de dos resortes que están conectados a ejes fijos en los extremos contrarios a donde está sujeta la masa.

Figura 3. **Oscilador único**



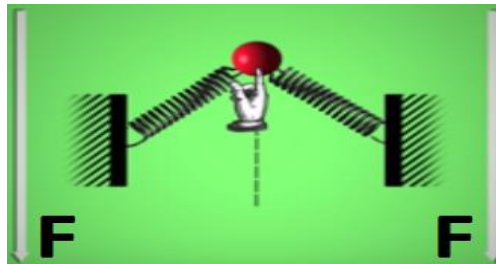
Fuente: YouTube. *La paradoja de los números.*

<https://www.youtube.com/watch?v=UdMRgTF6lxY>. Consulta: 10 de diciembre de 2018.

Al mover de la posición de equilibrio la masa, se presentará una fuerza contraria debida a la energía almacenada en los resortes de los extremos que

como resultado que la masa quiera volver a su posición original lo que genera un movimiento oscilatorio hasta que pierda por completo su energía.

Figura 4. **Aplicación de una fuerza a la masa entre dos resortes**



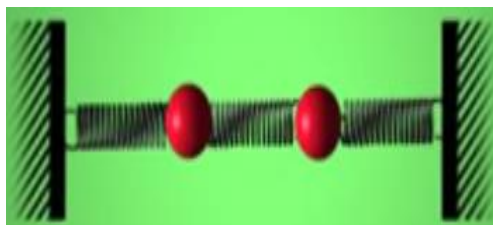
Fuente: YouTube. *La paradoja de los números*.

<https://www.youtube.com/watch?v=UdMRgTF6lxY>. Consulta: 10 de diciembre de 2018.

Por medio de la energía potencial que este sistema almacena en los resortes, se moverá hasta que el trabajo de las fuerzas no conservativas vaya reduciendo el movimiento, y alcance de nuevo su posición de equilibrio.

Ahora se repetirá el movimiento, pero se utilizarán dos masas acopladas.

Figura 5. **Sistema de 2 masas y tres resortes**



Fuente: YouTube. *La paradoja de los números*.

<https://www.youtube.com/watch?v=UdMRgTF6lxY>. Consulta: 10 de diciembre de 2018.

Este sistema posee varias formas de moverse dependiendo cómo se separen las masas y qué fuerza se les aplique al moverlas, lo cual provocará una serie de comportamientos diferentes hasta que pierdan toda la energía almacenada en los resortes y vuelvan a su posición de equilibrio.

A simple vista, no se puede predecir cómo estas condiciones iniciales afectan al movimiento final de las masas. Para eso se emplean las herramientas de la mecánica clásica: la ecuación general del momento lineal para estos sistemas (ecuación de Euler-Lagrange).

$$\frac{d}{dt} \frac{\partial L}{\partial \hat{q}_s} - \frac{\partial L}{\partial q_s} = 0$$

Donde:

L : función lagrangiana.

\hat{q}_s : coordenada generalizada: cada grado de libertad del sistema se expresa mediante una coordenada generalizada.

q_s : velocidad generalizada: derivada temporal de las coordenadas generalizadas.

t : tiempo.

Esta ecuación muestra que toda la variación del sistema masa-resorte pueden ser separadas en dos modos (los modos son las maneras como masas pueden ser variadas). A este comportamiento se le nombrará como modos normales del sistema.

En la primera forma de moverse o el primer modo normal se observa que las dos masas se mueven con la misma fuerza y el mismo desplazamiento.

Figura 6. **Sistema de dos masas acopladas moviéndose bajo las mismas condiciones**

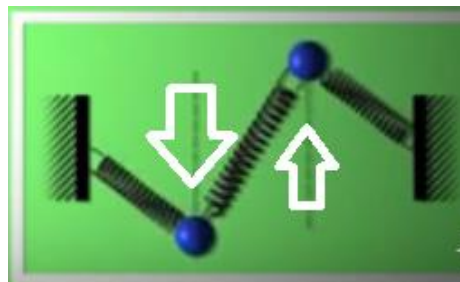


Fuente: YouTube. *La paradoja de los números.*

<https://www.youtube.com/watch?v=UdMRgTF6lxY>. Consulta: 10 de diciembre de 2018.

La otra forma de moverse, o el segundo modo, consiste en mover en distintas direcciones las dos masas y con diferentes fuerzas.

Figura 7. **Variación de velocidad y dirección diferentes en cada masa**

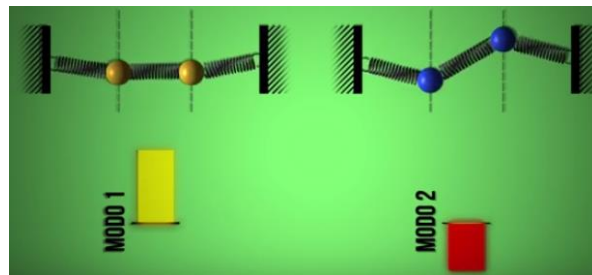


Fuente: YouTube. *La paradoja de los números.*

<https://www.youtube.com/watch?v=UdMRgTF6lxY>. Consulta: 10 de diciembre de 2018.

Con lo anterior se puede determinar que se pueden mover las masas de cualquier forma combinando los modos normales del sistema.

Figura 8. **Comparación de los modos normales**



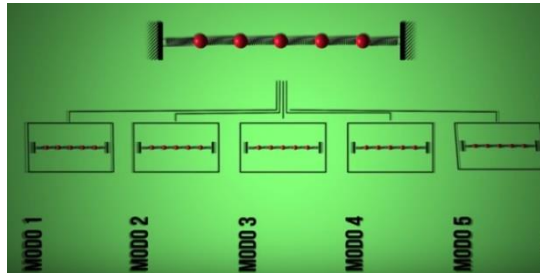
Fuente: YouTube. *La paradoja de los números.*

<https://www.youtube.com/watch?v=UdMRgTF6lxY>. Consulta: 10 de diciembre de 2018.

Las variaciones que se realicen en las condiciones iniciales del sistema, consisten en cuanto se pueden aumentar o reducir los movimientos de las masas. De modo que se obtiene el sistema resuelto para predecir el movimiento del sistema.

Observando el sistema anterior, el siguiente paso de la lógica es pensar que pasaría cuando se agreguen más masas al sistema.

Figura 9. **Múltiples modos normales del sistema**

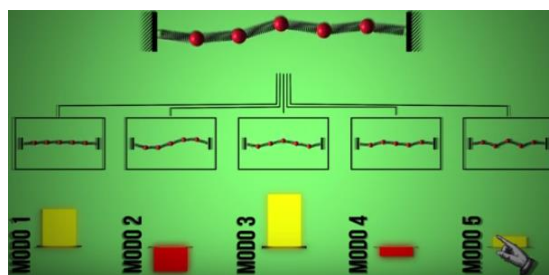


Fuente: YouTube. *La paradoja de los números.*

<https://www.youtube.com/watch?v=UdMRgTF6lxY>. Consulta: 10 de diciembre de 2018.

La idea principal se mantiene con el incremento del número de masas y resortes. Al tener muchas masas, existen muchos modos normales y su combinación, permite generar todos los movimientos posibles.

Figura 10. **Múltiples modos normales del sistema con condiciones iniciales**



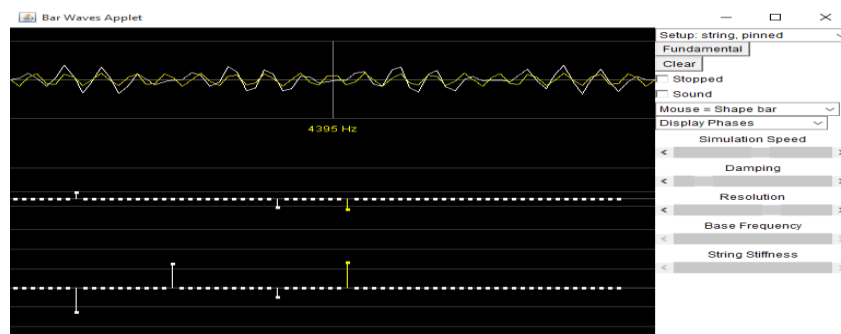
Fuente: YouTube. *La paradoja de los números.*

<https://www.youtube.com/watch?v=UdMRgTF6lxY>. Consulta: 10 de diciembre de 2018.

Los modos normales no vibran a la misma frecuencia, unos vibran a mayor frecuencia y otros a menor frecuencia.

Para observar este fenómeno con los distintos modos existe una aplicación gratuita que puede ser descargada en <http://www.falstad.com/loadedstring/>.

Figura 11. **Aplicación para observar los distintos modos normales**



Fuente: Falstad.com. *Simulación de cadena cargada*. <http://www.falstad.com/loadedstring/>.

Consulta: 11 de diciembre de 2018.

Con esta aplicación se pueden modelar varias masas y, como se observa, su gráfica de desplazamiento en función de sus condiciones ideales.

2.1.2. Caso continuo

En la naturaleza se encuentran fenómenos más complejos que para estudiarlos requieren de cálculo infinitesimal.

Se tomará el sistema simple de masas acopladas por medio de resortes, al cual se le añaden infinitas masas e infinitos resortes y, por último, las distancias entre los centros de las masas y los extremos de los resortes se hacen infinitamente pequeñas. A esto se le llama caso continuo.

En este punto dejan de existir los sistemas masa resorte puntuales por lo cual se observa un solo medio elástico susceptible a vibrar; este sistema se vería como una sola cuerda.

Figura 12. **Infinitos modos normales se ven como una cuerda**



Fuente: YouTube. *La paradoja de los números.*

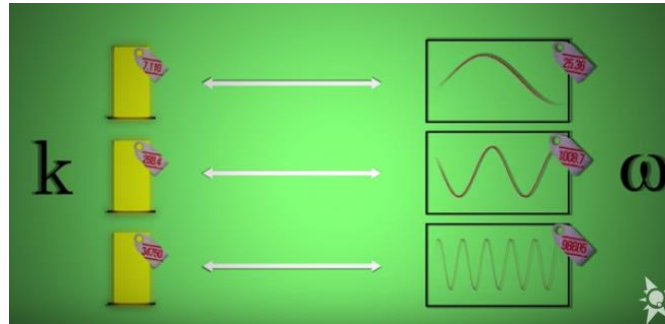
<https://www.youtube.com/watch?v=UdMRgTF6lxY>. Consulta: 11 de diciembre de 2018.

En este caso siguen existiendo modos normales, pero ahora son conocidos como ondas planas, de las cuales se observa que si se dividen dos modos normales siempre se podrán encontrar infinitos modos normales entre estos dos pareciendo solo secciones de cuerda.

Los modos normales ya no son discretos, por lo tanto, no se pueden etiquetar con números naturales, ahora se ve completamente el caso continuo en donde hay infinitos modos normales recorriendo toda la cuerda.

A esto se le llama número de onda K . Este número indica cuántas veces vibra una cuerda en una unidad de distancia determinada.

Figura 13. **Representación del número de onda**



Fuente: YouTube. *La paradoja de los números.*

<https://www.youtube.com/watch?v=UdMRgTF6lxY>. Consulta: 10 de diciembre de 2018.

El número de onda y la frecuencia de la vibración están relacionados directamente debido a que determina cuántas veces se genera una vibración en una distancia determinada; la frecuencia se traduce como cuántas veces se genera la vibración en un tiempo determinado, por lo cual se puede referir al número de onda como algo relacionado con la longitud y la frecuencia se relaciona con el tiempo.

Se observa que el movimiento de una cuerda está construido por las superposiciones de infinitas ondas planas cada una con su respectiva amplitud y frecuencia dispuestas sobre un mismo sistema.

2.2. **Análisis de Fourier**

En matemáticas, el análisis de Fourier es el estudio en donde las formas de las ecuaciones pueden ser representadas por sumatorias de simples funciones trigonométricas.

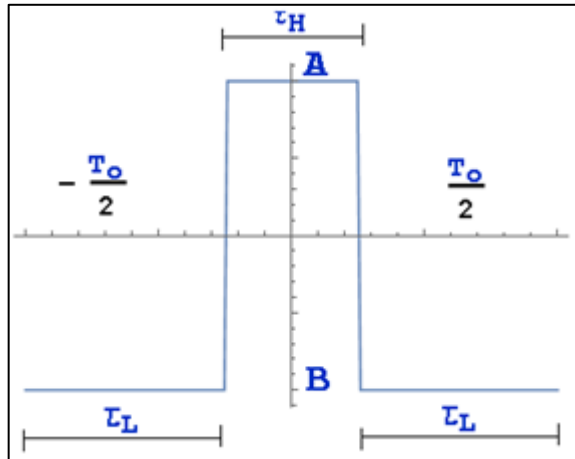
2.2.1. Series de Fourier

Si se tiene una cuerda con una forma particular de moverse y quiere saber qué modos están excitados y cuánto lo están (una cuerda también puede referirse a una onda plana como se ha visto anteriormente, por lo tanto, aplica para una señal de telecomunicaciones, la luz emitida o la señal sonora que genera un motor en funcionamiento), lo que se tiene que hacer es aplicar la transformada de Fourier. Por el contrario, si uno parte del espectro de frecuencias y quiere saber qué movimientos generaría en una cuerda, se aplica la transformada de Fourier inversa, que consiste en sumar las ondas planas pertinentes.

En el siguiente ejemplo se observa como una función cuadrada que está definida por partes puede ser representada como una serie discreta de superponían de ondas planas, cada una calibrada bajo su propia amplitud y su propia frecuencia definida a base de los términos discretos de la secuencia.

$$f(t) = \begin{cases} B - \frac{T_0}{2} < t < -\frac{\tau_H}{2} \\ A - \frac{\tau_H}{2} < t < \frac{\tau_H}{2} \\ B \frac{\tau_H}{2} < t < \frac{T_0}{2} \end{cases}$$

Figura 14. Señal de onda cuadrada por partes

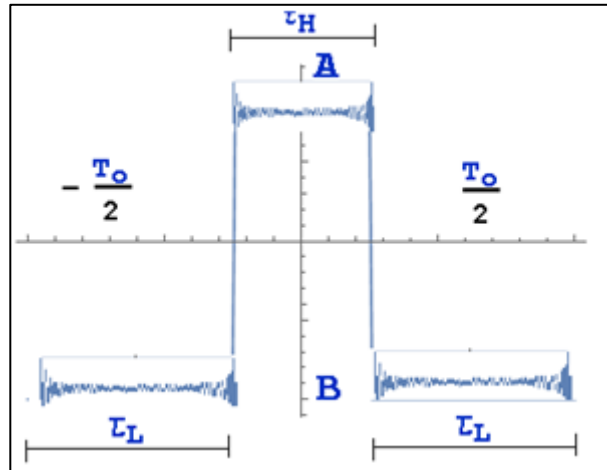


Fuente: elaboración propia, empleando wólfram matemática y ms Paint.

Se puede representar la ecuación anterior construida por partes por medio de la sumatoria de todos los modos normales utilizando la transformación a serie de Fourier que da como resultado la siguiente ecuación.

$$f(t) = \sum_{n=1}^{\infty} \frac{2(A - B) \sin\left(\frac{\pi n T_H}{T_o}\right)}{n \pi} \cos\left(\frac{2 \pi n t}{T_o}\right)$$

Figura 15. **Señal de onda cuadrada periódica y continua en el tiempo**



Fuente: elaboración propia, empleando wólffram matemática y ms Paint.

La gráfica anterior es una representación continua en el tiempo de la ecuación de onda cuadrada hecha por partes anteriormente; esto significa que es posible representar cualquier forma de señal periódica en el tiempo como una suma de modos normales con una amplitud diferente y en una frecuencia que es un múltiplo de la frecuencia fundamental de la señal.

2.2.2. Transformada de Fourier

La transformada de Fourier descompone una ecuación en función del tiempo en una ecuación en función de la frecuencia, de manera similar a cómo un tono musical de cuerda se puede expresar como la amplitud (o volumen) de sus notas constitutivas. Esto lo hace por medio de graficar solamente la amplitud de cada uno de los términos en el eje y , y colocando su respectiva frecuencia en el eje x ; por ejemplo:

De la ecuación de la superposición de modos normales se toman solo las amplitudes.

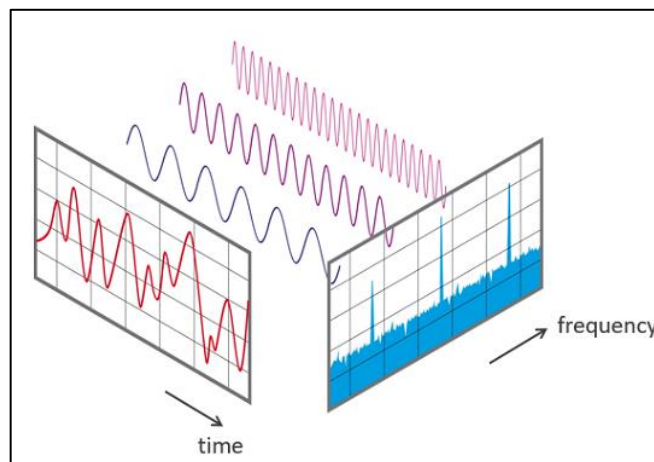
$$Amplitud(n) = \frac{2(A - B) \sin\left(\frac{\pi n T_H}{T_o}\right)}{n \pi}$$

Siendo la respectiva frecuencia angular para cada n-ésimo modo normal.

$$\omega(n) = \frac{2 \pi n}{T_o}$$

Y su respectiva fase. En el caso del pulso de onda cuadrada centrado en el origen la fase es constante y es cero; ya que, al ser una función par, su componente impar es cero para la cual se hace una gráfica diferente en donde ahora el eje x representa esas respectivas fases.

Figura 16. **Relación entre el tiempo y la frecuencia**



Fuente: NTI. *Soluciones de medición para una amplia gama de aplicaciones*. <https://www.nti-audio.com/>. Consulta: 11 de diciembre de 2018.

En este documento se observará la construcción y calibración de un equipo electrónico para la obtención y el procesamiento de la señal de vibración para obtener la transformada de Fourier de la señal de un equipo mecánico, el cual será puesto a prueba por medio de un desbalance en un entorno controlado, para crear una base de datos de información que produce cada una de las ondas planas con sus armónicos; se puede así predecir sus fallas en funcionamiento continuo monitoreándola en tiempo real para poder predecir que pieza de la máquina es necesario ajustar para aumentar el tiempo de vida y la calidad de la producción que esta proporciona.

Para la obtención de la señal, su procesamiento y la presentación de la información obtenida se procederá a construir un equipo electrónico que como mínimo posea los siguientes elementos:

- Transductor: capaz de convertir la señal de vibración en una señal eléctrica análoga a la vibración.
- Filtro de señal eléctrica: se encargará eliminar la inducción de otras señales que no tengan relación con la señal de la vibración.
- Amplificador de señal: aumentará la amplitud de la señal eléctrica convirtiéndola al rango del voltaje correcto para ser leída por el circuito de muestreo dentro del microcontrolador.
- Un segundo filtro de señal a la salida del amplificador anterior para eliminar el posible ruido inducido por la conversión de energía que se produce en el amplificador.

- Un circuito encargado de tomar el voltaje y que se integra en el microcontrolador que tiene la función de hacer una conversión de señal analógica a digital; puede así ser almacenada en memoria.
- Un circuito de memoria con suficiente espacio para guardar la cantidad de muestras necesarias para el cálculo de la transformada de Fourier.
- Un procesador lo suficientemente rápido para hacer los cálculos lo suficientemente.
- Una memoria para almacenar la información de los resultados de los cálculos.
- Una pantalla donde sea posible ver el espectro de frecuencias obtenido de las transformaciones anteriores.

3. COMPONENTES DEL CIRCUITO

3.1. Sensor

Un sensor es un aparato que tiene la capacidad de producir señales eléctricas a través de cambios físicos o químicos; a estas magnitudes se les denomina variables de instrumentación: temperatura, presión, humedad, aceleración, velocidad, entre otros.

El tipo de sensor que se utilizará será alguno que pueda detectar el movimiento. Por las mismas construcciones de cada sensor este es notablemente alguna especie de filtro de frecuencia lo que disminuye su efectividad a la hora de medir el movimiento. Ya que los sistemas que se tienen como objetivo son motores conectados a poleas y engranajes no se necesita lograr medir altas frecuencias. Por lo tanto, se podrán utilizar micrófonos de estado sólido, acelerómetros o bobinas móviles conectados a un imán.

3.1.1. Micrófono de estado sólido o piezoeléctrico

Las ondas sonoras hacen vibrar el diafragma y, su movimiento, hace que se mueva el material contenido en su interior (cuarzo, sal de Rochelle, carbón, entre otros) Las fricciones entre las partículas del material generan sobre su superficie una tensión eléctrica.

La señal eléctrica de salida es análoga (amplitud y frecuencia) a la onda sonora que la generó.

La respuesta en frecuencia de los micros electrostáticos es muy irregular, por lo que su uso en ámbitos de audio profesional está desaconsejado.

Sin embargo, los componentes piezoeléctricos son utilizados para la construcción de micrófonos especializados para instrumentos musicales.

Micrófonos piezoeléctricos son:

- El micrófono de cristal
- El micrófono de cerámica
- Los componentes piezoeléctricos en guitarras eléctricas

Para medir los sistemas mecánicos vibrantes es posible que se pierdan algunas frecuencias ya que estas son inmediatamente interrumpidas convirtiendo estos en filtros paso bajo con una frecuencia de corte cerca de los 20 KHz, ya que el movimiento se tiene que medir indirectamente a base del sonido que produce la máquina.

3.1.2. Micrófono de bobina móvil

El micrófono de bobina móvil consta de un diafragma rígido suspendido frente a un imán permanente potente, que cuenta con una hendidura en la que va acoplada una bobina móvil solidaria. Cuando las ondas sonoras excitan el diafragma (de 20-30 mm de diámetro), la bobina sólida se mueve a su vez (hacia delante y hacia atrás) dentro de la ranura del imán. Así la bobina vibra de forma proporcional al sonido registrado y al oscilar dentro del campo magnético del imán permanente se genera una corriente eléctrica proporcional a este sonido.

Hay dos tipos de micrófono:

- Los que tienen suficiente hilo fino enrollado a la bobina como para entregar un nivel suficiente de corriente a la salida.
- Otros con menos espiras que requieren una preamplificación. En este caso, el circuito preamplificador es un transistor JFET que está alojado en el propio cuerpo del micrófono.

Con este dispositivo pueden medirse directamente las magnitudes de las vibraciones ya que al colocarlo directamente en el sistema mecánico el sonido ya no importará y será por transferencia de energía mecánica que la bobina se mueva produciendo una mejor señal y un ancho de banda en función de que tan rápido puede la bobina móvil regresar a su posición de equilibrio.

3.1.3. Acelerómetro

Se denomina acelerómetro a cualquier instrumento destinado a medir aceleraciones. Esto no es necesariamente la misma que la aceleración de coordenadas (cambio de la velocidad del dispositivo en el espacio), sino que es el tipo de aceleración asociada con el fenómeno de peso experimentado por una masa de prueba que se encuentra en el marco de referencia del dispositivo.

Un ejemplo en el que este tipo de aceleraciones son diferentes es cuando un acelerómetro medirá un valor sentado en el suelo, ya que las masas tienen un peso y no se muestra un cambio en la velocidad. Sin embargo, un acelerómetro en caída gravitacional libre hacia el centro de la Tierra medirá un valor de cero; ya que, a pesar de que su velocidad es cada vez mayor, está en un marco de referencia en el que no tiene peso.

Hay dos tipos de acelerómetros los cuales son:

- Acelerómetro mecánico:

Es el acelerómetro más simple. Se construye uniendo una masa a un dinamómetro cuyo eje está en la misma dirección que la aceleración que se desea medir.

- Acelerómetro de estado sólido o piezoeléctrico:

El acelerómetro es uno de los transductores más versátiles, siendo el más común el piezoeléctrico por compresión. Este consiste que cuando se somete a presión un segmento cristalino piezoeléctrico, se produce una carga eléctrica proporcional a la fuerza aplicada. Los elementos piezoeléctricos están hechos normalmente de circonato de cuarzo que crean diferencias de potencial al generar algún tipo de deformación.

Los elementos piezoeléctricos se encuentran comprimidos por una masa, sujeta al otro lado por un muelle y todo el conjunto dentro de una caja metálica. Cuando el dispositivo está bajo efectos de aceleración, el disco piezoeléctrico se ve sometido a una fuerza variable, proporcional a la aceleración de la masa. Debido al efecto piezoeléctrico se desarrolla un potencial variable que será proporcional a la aceleración. Este voltaje variable se puede registrar sobre un osciloscopio o voltímetro.

Este dispositivo junto con los circuitos eléctricos asociados se puede usar para la medida de velocidad y desplazamiento además de la determinación de formas de onda y frecuencia. La ventaja principal de estos dispositivos radica en

que pueden ser muy pequeños por lo cual no ejercen influencia sobre el sistema que están midiendo.

El intervalo de frecuencia típica es de 2 Hz a 10 KHz. Su uso es común en mantenimiento predictivo, donde se emplea para detectar defectos en máquinas rotativas y alternativas; se detecta, por ejemplo, el mal estado de un rodamiento o cojinete en una etapa temprana antes de que se llegue a la avería. De acuerdo con la ley fundamental de la dinámica o segunda ley de Newton.

3.2. Filtros

Un filtro electrónico es un elemento que disminuye los efectos de una determinada frecuencia o gama de frecuencias de una señal eléctrica que pasa a través de este, estas frecuencias se rechazan debido a que interfieren con la forma de la señal que hace difícil distinguir sus propiedades.

Las frecuencias se discriminan debido a que son a veces de un origen diferente al de la señal que se está analizando, con esto se logra una muestra clara de la cual solo se obtienen los datos pertinentes que concuerdan con el tipo de fenómeno que se está estudiando.

Hay varios tipos de filtros los cuales son:

- Filtro paso bajo
- Filtro paso altos
- Filtro paso banda
- Filtro rechaza banda

3.2.1. Paso bajo

Los filtros paso bajos tienen la capacidad de discriminar las altas frecuencias.

Ya que solo interesan las frecuencias en el rango de 10 kHz lo cual representa un motor a 600 Krpm's o menor, se utilizarán solo filtros paso bajo para evitar ruidos en la señal que no provengan del sistema estudiado.

Un filtro paso bajo corresponde a un filtro electrónico caracterizado por permitir el paso de las frecuencias más bajas y atenuar las frecuencias más altas. El filtro necesita para su funcionamiento de dos terminales en la entrada y dos en la salida contenidos en una forma teórica como un dispositivo de una caja negra también, denominada cuadripolo o bipuerto; así todas las frecuencias se pueden presentar a la entrada, pero a la salida solo estarán presentes las que permita pasar el filtro.

Este dispositivo evitará que las señales de alta frecuencia en (las cuales es donde normalmente se presenta el ruido eléctrico) intervengan con las mediciones del sensor.

3.2.2. Filtro paso altos

Su objetivo es atenuar las bajas frecuencias y dejar pasar las altas.

Este sistema se encarga de disminuir los efectos de las longitudes de onda muy largas que pertenecen a otros fenómenos físicos como vibraciones generadas por sismo y la señal de corriente alterna que está presente en todo lugar donde se utilice debido a su gran amplitud.

3.2.3. Filtros paso banda

Estos dispositivos se encargan de evitar frecuencias bajas y frecuencias altas al mismo tiempo; deja que solo el rango de frecuencias de interés sea percibido por el medidor.

Su función radica en la forma en la cual es capaz de presentar una respuesta a una cantidad de frecuencias contenidas en un rango, el cual es acotado por las bajas frecuencias del lado izquierdo y las altas frecuencias del lado derecho; deja una forma rectangular llamada banda bajo condiciones ideales.

Este tipo de filtros es utilizado en casi todo tipo de señales, pero las más conocidas son las señales de audio como en un afinador de guitarra, que para detectar que se está produciendo la frecuencia necesaria para que la cuerda esté afinada necesita un circuito que elimine el efecto de todas las demás frecuencias cercanas.

3.2.4. Filtros rechaza banda

Estos filtros se encargan de discriminar un rango de frecuencias y dejar pasar todas las demás, son lo contrario a un filtro paso banda.

Su principal objetivo es atenuar los efectos de un rango de frecuencias específica para que la señal obtenida de la medición del fenómeno no se altere bajo estos efectos.

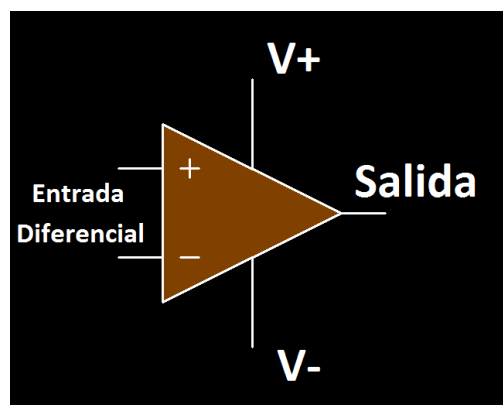
Estos circuitos se utilizan en casos en donde una frecuencia que se presenta constantemente es reconocida como algo que no es parte del fenómeno estudiado o simplemente es parte de este, pero no posee interés alguno.

3.3. Amplificador operacional

Son circuitos que por medio de un arreglo de transistores se transfieren y se convierten en pequeñas variaciones de voltaje en su entrada en grandes variaciones en su salida; en otras palabras, estos aportan una alteración a la señal es un factor de escala que aumenta el tamaño de la señal sin alterar su forma en el tiempo.

Un amplificador operacional (op-amp o opamp) es un circuito electrónico de alta ganancia de tensión. El amplificador tiene una entrada diferencial y, por lo general, una salida de un solo extremo. En esta configuración, un amplificador produce una salida potencial que es típicamente cientos de veces más grande que la diferencia de potencial entre sus terminales de entrada.

Figura 17. Amplificador operacional



Fuente: elaboración propia, empleando Circuit wizard v1.5.

3.3.1. Funcionamiento

Las entradas diferenciales del amplificador constan de una entrada no inversora (+) con la tensión V_+ y una entrada inversora (-) con el voltaje V_- ; idealmente, el op-amp amplifica solo la diferencia de voltaje entre los dos, que es llamada la tensión de entrada diferencial. La tensión de salida del amplificador operacional V viene dada por la ecuación.

$$V_{out} = A (V_+ - V_-)$$

Donde A es la de bucle abierto ganancia del amplificador (el término bucle abierto se refiere a la ausencia de un bucle de retroalimentación desde la salida a la entrada).

3.4. Microcontrolador

Un microcontrolador (abreviado uC, 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 y periféricos de entrada y salida.

Este componente se utilizará para realizar la conversión de la señal analógica proveniente del acelerómetro la cual será procesada de forma que se pueda obtener la transformada rápida de Fourier (FFT).

3.4.1. Microcontrolador SAM3X8E

Es un microcontrolador de la familia SAM3X/A de la empresa ATMEL. Este es un miembro de una familia de microcontroladores flash basados en microprocesador ARM CORTEX M3 RISC de 32 bits de alto rendimiento. Funciona a una velocidad máxima de 84 MHz y cuenta con hasta 512 Kbytes de Flash y hasta 100 Kbytes de SRAM. El conjunto incluye un periférico.

Alta velocidad USB Host y el puerto de dispositivos de bajo consumo integrado con una conexión Ethernet, una MCI alta velocidad para SDIO / SD / MMC, una interfaz de bus externo con el regulador NAND Flash (NFC), 5 UARTs, 2 TWIS, 4 SPI, así como un PWM, tres temporizadores de propósito general de 32 bits 3 canales, una baja potencia de RTC, 256-bit registros de propósito general de copia de seguridad, un ADC de 12 bits y un DAC de 12 bits.

3.5. Pantalla de cristal líquido

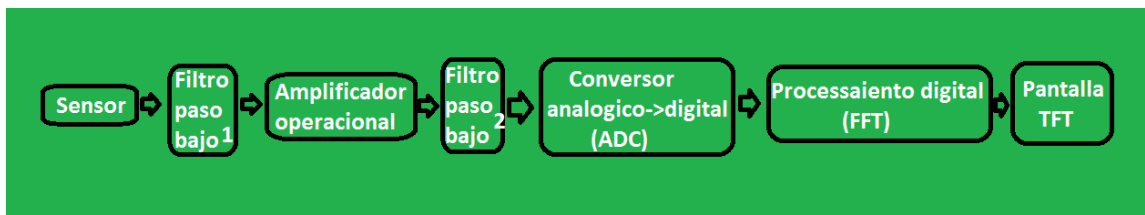
TFT-LCD (*Thin Film Transistor-Liquid Crystal Display*), pantalla de cristal líquido de transistores de película fina) es una variante de pantalla de cristal líquido (LCD) que usa tecnología de transistor de capa delgada (TFT) para mejorar su calidad de imagen. Las LCD de TFT son un tipo de LCD de matriz activa, aunque esto es generalmente sinónimo de LCD. Son usados en tabletas, computadoras, celulares, televisores, visualizadores de pantalla plana y proyectores. En informática, los monitores de TFT han desplazado la tecnología de CRT, y están comúnmente disponibles en tamaños de 12 a 30 pulgadas. En 2006 entraron en el mercado de las televisiones.

Se presentarán las gráficas de las amplitudes con sus respectivas frecuencias luego de ser procesadas por el microcontrolador.

3.6. Construcción

En el siguiente diagrama de bloques se detalla el conjunto de circuitos y su ubicación con respecto a la toma de señal y la presentación en la pantalla TFT.

Figura 18. Diagrama de bloques del sistema



Fuente: elaboración propia, empleando MS WORD 2016.

3.6.1. Sensor

Este es el dispositivo encargado de proporcionar la señal en tiempo real.

3.6.2. Filtro paso bajo 1

El primer filtro se encargará de discriminar las frecuencias causadas por el ruido eléctrico para los bajos voltajes que produce el sensor.

3.6.3. Amplificador operacional

Este se encargará de proporcionar una amplitud adecuada para ser muestreada por el sensor ADC.

3.6.4. Filtro paso bajo 2

Este será el encargado de eliminar el ruido causado por el circuito amplificador.

3.6.5. Conversor analógico digital (ADC)

El ADC se encarga de muestrear la señal producida por el sensor y convertirla en distinguibles niveles de voltaje los cuales son convertidos a números binarios con los que es posible tratarlos digitalmente.

3.6.6. Procesamiento digital (FFT)

Es un trozo de código para un lenguaje de programación basado en matemática discreta el cual es capaz de obtener la transformada rápida de Fourier de la señal que el sensor proporciona; puede así proporcionar una matriz de muestras que contienen la información de las amplitudes de las funciones armónicas en función de su frecuencia.

3.6.7. Pantalla TFT

Esta es una matriz de puntos de color rojo, verde y azul los cuales en su sincronización serán capaces mostrar la gráfica de las distintas amplitudes y sus frecuencias.

4. SENSOR

4.1. Acelerómetros de fuerza balanceada

Es un dispositivo que normalmente integra tres componentes está hecho para estudiar señales sísmicas de muy baja frecuencia, es utilizado para analizar la sismicidad de la tierra, como para la localización de epicentros de sismos.

Este tipo de acelerómetro se compone de una bobina bajo una placa metálica conductora que al polarizarse con voltaje genera un campo magnético que atrae la placa conductora a un punto de equilibrio; esta placa conductora se sostiene como un resorte por su propia forma por medio de unos pines, está formada de dos partes las cuales a modo de antena capturan el campo magnético que genera la bobina, el cual después es medido por medio de un circuito amplificador y un conjunto de filtros electrónicos; cuando ocurre un evento sísmico esta placa se mueve de su punto de equilibrio lo cual, generando una variación de campo magnético.

Este tipo de sensores es mayormente utilizado en vulcanología para detectar eventos sísmicos de bajas frecuencias que se generan cuando el magma dentro del volcán rompe la roca.

Desventajas:

- Son muy costosos.
- Son difíciles de conseguir.

- Son demasiado grandes y su peso podría alterar el comportamiento de la maquinaria.
- El movimiento al transportarlos puede altear su respuesta con respecto a su función de transferencia.

Ventajas:

- Tienen una alta sensibilidad a frecuencias muy bajas.
- Presentan bajos niveles de ruido en relación con la señal.
- Poseen una documentación muy exacta ya que incluyen la función de transferencia del sensor y la señal.

4.2. Mems

Los componentes fabricados con elementos en el orden de los micrómetros y que son eléctricos y mecánicos como transductores de gran precisión son circuitos basados en tecnología en el orden de los micrómetros creados como transductores para transformar distintas magnitudes físicas en señales eléctricas.

Estos versátiles dispositivos se componen de mecanismos diminutos en el orden de nanómetros dentro de los chips de silicón, estos pueden contener dentro de sí elementos extremadamente simples, como una masa en desplazamiento a elementos muy complejos como dispositivos con electroimanes.

Hoy en día, los MEMS son utilizados por su pequeño tamaño en dispositivos como tabletas o celulares; estos dispositivos son utilizados para vigilar el comportamiento del dispositivo para convertirse en una interfaz con la que el usuario puede interactuar; por ejemplo cuando se gira el celular se observa que la pantalla gira de manera de posicionarse en la forma más cómoda con respecto

al usuario, este es un claro ejemplo del uso del acelerómetro triarías interno que indica la dirección de la aceleración gravitacional, por lo tanto, cuando una de las tres coordenadas cambia de una a otra; esto indicará la posición que posee el celular.

Desventajas:

- No pueden medir frecuencias muy bajas
- No está disponible su función de transferencia

Ventajas

- Su pequeño tamaño no ejerce peso sobre la maquinaria y, por lo tanto, no altera a la vibración original.
- No pierde sus propiedades al ser transportado.
- Es de muy bajo costo.
- Es fácil de conseguir.

Por lo tanto, para razones de este documento se seleccionará un sensor de este tipo para hacer las mediciones de forma adecuada, ya que la maquinaria normalmente no funciona en bajas frecuencias no es necesario tener un sensor que sea capaz de medir ese tipo de eventos.

4.3. ADXL335

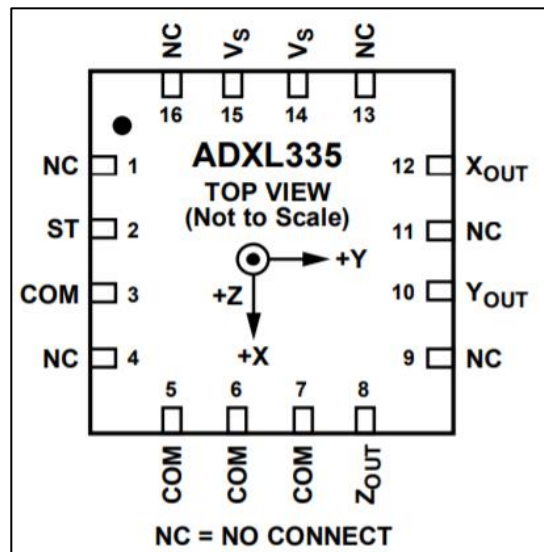
Este es el código de dispositivo del sensor seleccionado debido a que fue el que mostró las mejores características que se presentan a continuación.

Este es un sensor analógico que posee tres ejes independientes nombrados como x, y, z, de los cuales solo se utilizará uno para medir la vibración en un solo

sentido. El sensor se alimenta con una fuente de corriente directa y puede funcionar entre 1,8 y 3,6 voltios lo cual permitirá usar la misma fuente para el circuito de la pantalla y el microcontrolador, y tiene un consumo de corriente de 350 microamperios.

Tiene una excelente estabilidad frente a los cambios de temperatura lo cual es muy útil ya que la maquinaria siempre genera calor, posee un ancho de banda de 0,5 Hz a 1 600 Hz para los ejes x, y, con un ancho de banda para el eje z de 0,5 Hz a 500 Hz, lo cual es suficiente para medir la mayoría de maquinaria ya que traducido a RPM (revoluciones por minuto) los anchos de banda quedarían como 30 a 96 000 ,00 RPM para los ejes x, y, teniendo 30 a 30 000,00 RPM para el eje z.

Figura 19. **Circuito ADXL335, descripción y configuración de pines**



Fuente: elaboración propia, empleando MS WORD 2016.

4.3.1. Descripción de los pines

Estos son cada uno de los pines con su respectiva descripción de funcionamiento para el circuito integrado ADXL335.

4.3.1.1. Pin 1, 4, 9, 11, 13 y 16 NC

NC es una representación de *no connected* traducido como no conectado.

4.3.1.2. Pin 2 ST

Self-test, este pin sirve para para comprobar que el sensor esté funcionando correctamente. Cuando la función de autoprueba esta activada es posible aplicar una fuerza electrostática de prueba al elemento mecánico dentro del circuito que hace que la parte móvil se aleje de la posición de equilibrio, que emula una aceleración de entrada definida. La salida de este dispositivo en el modo de autocomprobación será comparada con la salida del sensor en modo normal (es importante saber que el modo normal es cuando el modo de autocomprobación está desactivado).

Para comprender mejor el modo de autocomprobación se debe tomar en cuenta lo siguiente.

- Se debe realizar una tabla comparativa de los dos modos del acelerómetro los cuales son con la autocomprobación habilitada y deshabilitada.
- Al intercambiar los modos en el acelerómetro se requiere una cantidad de tiempo para que se estabilice.

- La autocomprobación se puede realizar a diferentes rangos de muestreo de escala completa.
- Para reducir el error en la toma de muestras, se pueden tomar de 5 a 10 muestras en intervalos de más altas frecuencias y luego promediarlos.

Cuando el acelerómetro está en modo normal significa que en cada uno de las componentes se registra una posición inicial.

$$A_x = X_0$$

$$A_y = Y_0$$

$$A_z = Z_0$$

Donde:

A_x, A_y, A_z son, respectivamente, las aceleraciones características en una posición inicial X_0, Y_0, Z_0 .

En la misma posición estacionaria pero ahora en modo de autocomprobación se obtiene que el acelerómetro se ha desplazado de su posición original.

$$A_x' = X_0 + \Delta_x$$

$$A_y' = Y_0 + \Delta_y$$

$$A_z' = Z_0 + \Delta_z$$

Donde:

$\Delta_x, \Delta_y, \Delta_z$ representan el cambio de aceleración con respecto al punto de equilibrio.

Por lo tanto, se puede obtener el resultado de la autocomprobación observando la diferencia entre los datos obtenidos en los dos modos.

$$\Delta_x = |\Delta'_x - \Delta_x|$$

$$\Delta_y = |\Delta'_y - \Delta_y|$$

$$\Delta_z = |\Delta'_z - \Delta_z|$$

Si: $\min_x < \Delta_x < \max_x$, $\min_y < \Delta_y < \max_y$ y $\min_z < \Delta_z < \max_z$ significa que la autocomprobación demuestra que el sensor está funcionando de forma adecuada.

Donde: $\min_{x,y,z}$ y $\max_{x,y,z}$ están relacionados con el rango de escala completa en la unidad del bit menos significativo.

Por lo general, hay dos modos de autocomprobación: self-test0 y self-test1. La diferencia entre self-test0 y self-test1 es que la masa de prueba se mueve en direcciones opuestas con la misma magnitud. Eso significa que los resultados de la autocomprobación 0 deben estar cerca de los resultados de la autocomprobación1 con el signo opuesto. Por lo tanto, solo necesita elegir el modo self-test0 o self-test1 para la autopueba en cualquier posición estacionaria.

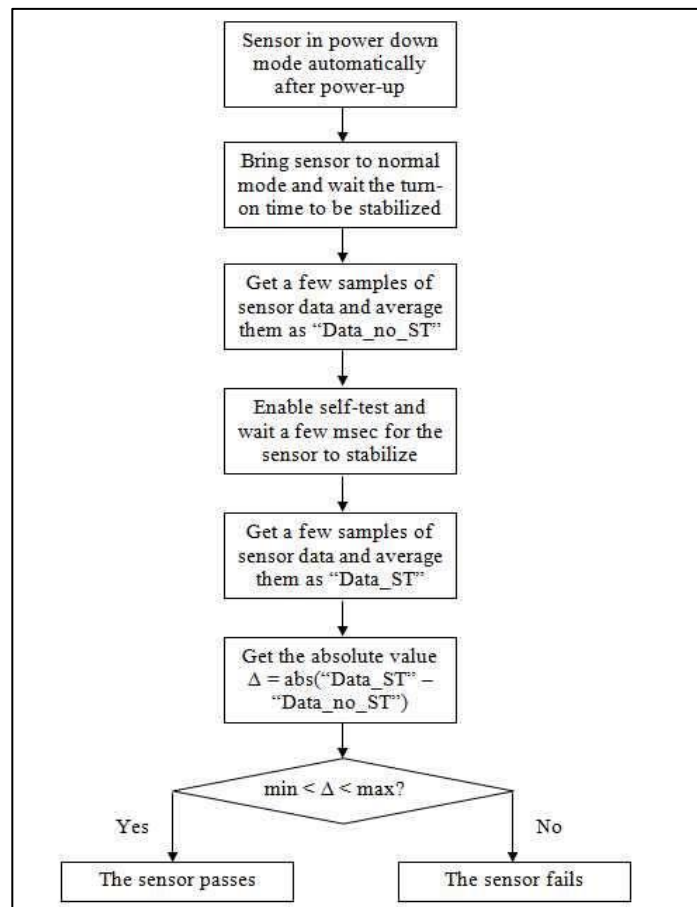
Los datos crudos (sin procesar) del acelerómetro en modo normal (autocomprobación deshabilitada) es igual a la mitad de la suma de los datos brutos en autocomprobación0 y los datos brutos en autocomprobación1, porque

los valores Δ en modo autocomprobación0 y autocomprobación1 se deben cancelar.

Es decir:

$$A_{X,Y,Z}(MODO\ NORMAL) = \frac{A_{X,Y,Z}'(autoprueba0) + A_{X,Y,Z}'(autoprueba1)}{2}$$

Figura 20. Procedimiento de autocomprobación para el acelerómetro



Fuente: MEMS, EDN. *La función de autocomprobación incorporada en los sensores inerciales.*

<https://m.eet.com/media/1167688/fig1.jpg>. Consulta: 11 de diciembre de 2018.

4.3.1.3. Pines 3, 5, 6 y 7 COM

Se refiere a la palabra común y representa que va conectado a 0 voltios con una fuente de corriente directa.

4.3.1.4. Pines 8, 10 y 12 X, Y, Z OUT

Estos son los pines que corresponden a cada una de las salidas analógicas del circuito ya que este posee una componente para cada eje.

4.3.1.5. Pines 14 y 15 VS

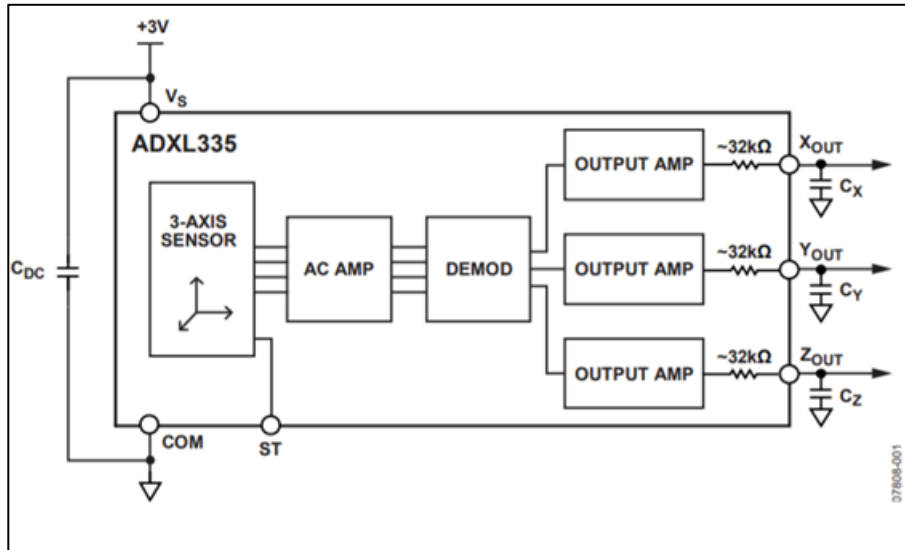
En esto va conectada la fuente de voltaje que produce entre 1,8 y 3,6 voltios de corriente directa.

4.3.2. Descripción general del circuito ADXL335

El ADXL335 es un pequeño chip de bajo consumo de potencia, utiliza solo 350 μA y una fuente de corriente directa que puede ser de 1,8 V a 3,6 V; este dispositivo tiene un rango de escala de ± 3 G (siendo G la fuerza de gravedad), el cual puede ser utilizado para medir la aceleración gravitacional; también, posee la habilidad de utilizarse como inclinómetro basado en la inclinación gravitacional. Su uso está optimizado para medición de movimiento, choque o vibración.

El usuario puede seleccionar el ancho de banda del acelerómetro usando capacitores en la salida de cada eje. El ancho de banda puede ser seleccionado en un rango de 0,5 Hz a 1 600 Hz para los ejes X y Y, para el eje Z el rango es de 0,5 Hz a 550 Hz.

Figura 21. Diagrama de bloques funcional



Fuente: MEMS, EDN. *La función de autocomprobación incorporada en los sensores inerciales.*
<https://m.eet.com/media/1167688/fig1.jpg>. Consulta: 11 de diciembre de 2018.

4.3.3. Teoría de operación

El chip contiene una superficie de poli silicona (superficie-micro mecanizada), la señal está acondicionada como un sistema, está hecho de una arquitectura de lazo abierto para la medición de la aceleración. Las señales de salida son voltajes analógicos proporcionales a la aceleración. El acelerómetro es capaz de medir la aceleración estática de la gravedad en su aplicación de inclinómetro con un resultado dinámico de aceleración resultante del movimiento, el choque o la vibración.

El sensor está construido de una superficie de poli silicona con una estructura micro mecanizada construida dentro. La poli silicona está suspendida en una estructura sobre la superficie de la oblea y provee una resistencia a las

fuerzas de aceleración. La estructura de deflexión significa que está usando un arreglo de capacitores en forma diferencial que consiste en placas independientes y placas unidas a la masa. Las placas independientes están giradas 180 grados fuera de fase con las ondas cuadradas. La deflexión en la variación de la aceleración en el movimiento de la masa y el desbalance diferencial de los capacitores resulta en una salida de amplitud de voltaje proporcional a la aceleración. La técnica de demodulación sensible a la fase es usada para determinar la magnitud y dirección de la aceleración.

El demodulador de salida es amplificado y llevado fuera del chip por medio de una resistencia de 32 K Ω . El usuario puede modificar el ancho de banda del dispositivo añadiendo un capacitor para filtrar las impurezas en la resolución de la medición ayudando a prevenir el solapamiento en la señal.

4.3.4. Sensor mecánico

El ADXL335 usa una simple estructura para muestrear los ejes; como resultado, los tres ejes son muestreados de forma perpendicular teniendo un pequeño cruce entre ejes, la desalineación mecánica del sensor es la fuente del cruce entre los ejes del sensor. Esta alineación puede ser calibrada usando un nivelador al posicionar el sensor.

4.3.5. Desempeño

No es necesario utilizar un circuito de compensación de temperatura adicional, ya que las técnicas aseguran un alto rendimiento. Como resultado no hay error de cubanización o comportamiento no mono tónico, haciendo que la histéresis de la temperatura sea muy baja (típicamente menos de 3 mg por encima de -25°C a $+70^{\circ}\text{C}$ de temperatura).

4.3.6. Desacoplo de fuente de voltaje

Para las aplicaciones se utiliza un capacitor de 0,1 0,1 μF colocado cerca del circuito ADXL335 en los pines de la fuente de voltaje para desacoplar el acelerómetro del ruido de la fuente de voltaje adecuadamente. Además, en las aplicaciones en donde el ruido está presente en un reloj de frecuencia de 50 KHz, este puede causar errores en la medición de aceleración por lo que se recomienda evitar.

4.3.7. Ajustar el ancho de banda usando capacitores a la salida

El ADXL335 tiene la capacidad de limitar el ancho de banda de la medición usando capacitores en los pines de salida de cada señal. Los capacitores tienen que ser añadidos en estos pines para implementar un filtro paso bajo para evitar el solapamiento de señales y reducir así el ruido. La ecuación que rige este ancho de banda para 3 dB es:

$$F_{-3 \text{ dB}} = \frac{1}{2\pi \cdot 32 \text{ K}\Omega \cdot C_{X, Y, Z}}$$

La tolerancia de la resistencia interna varía entre $\pm 15 \%$ del valor nominal de 32 k Ω ; se recomienda un valor mínimo de capacitancia de 0,0047 μF para todos los ejes.

4.3.8. Autoprueba

El pin ST controla la autoprueba como se había mencionado antes. Cuando el pin es puesto en el voltaje positivo la fuerza electrostática es insertada al tubo del acelerómetro. El resultado del movimiento del tubo permite probar si el

acelerómetro está funcionando. Típicamente, el cambio en la salida es de -0,8 g (correspondientes a -325 mV) en el eje X, +1,08 g (+325 mV) en el eje Y y 1,83 g (550 mV) para el eje z. El pin ST puede estar desconectado o conectado a la línea común en su uso normal.

Se recomienda nunca exponer el pin ST a voltajes mayores que $V_s + 0,3$ V.

4.3.9. Diseño del filtro seleccionando valores comerciales de capacitores

El ancho de banda seleccionable en el acelerómetro determina la resolución de la medición. El filtrado también se utiliza para disminuir el piso del ruido para implementarlo en la resolución de la medición. La resolución depende del ancho de banda del filtro análogo en los pines de salida.

La salida del ADXL335 tiene un ancho de banda mayor a los 500 Hz típicamente. Se puede filtrar la señal en este punto para limitar el error de solapamiento de señal. El ancho de banda análogo no puede ser menor de la mitad del muestreo análogo a digital.

El ruido en el ADXL335 tiene la característica de ser el típico ruido blanco gaussiano, el cual hace una contribución igual en todas las frecuencias y se describe en términos de $\mu\text{g}/\sqrt{\text{Hz}}$. Se puede limitar el ancho de banda a frecuencias más bajas para maximizar la resolución en la aplicación y el rango dinámico del acelerómetro.

En el modo de un solo polo, el ruido típico para el circuito es determinado por:

$$\text{rms ruido} = \text{densidad del ruido} \times (\sqrt{\text{BW} \times 1,6})$$

Esta obtiene los valores pico del ruido. El ruido pico a pico puede ser estimado usando métodos estadísticos solamente.

4.3.10. Utilizando el circuito con voltajes distintos de 3 V

El ADXL335 ha sido probado específicamente a 3 V, además, este puede ser utilizado con un rango de voltajes entre 1,8 V y 3,6 V. Hay que notar la variación de algunos parámetros cuando se varía la fuente de voltaje.

La salida del circuito es proporcional, además la sensibilidad de la salida varía con respecto a la fuente de voltaje. A 3,6 V la sensibilidad de la salida es 360 mV/g y a 2 V la sensibilidad de la salida es 192 mV/g.

La salida de 0 g es normalmente la mitad de la fuente de voltaje.

El ruido de la salida es absoluto en voltios; además, la densidad de ruido espectral disminuye cuando la fuente de voltaje aumenta. Esto es porque el factor de escala aumenta mientras el ruido en el voltaje permanece constante.

5. MICROCONTROLADOR

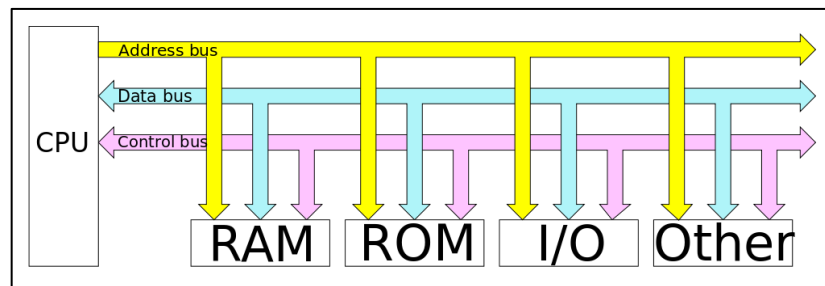
El microcontrolador es un circuito integrado que se asemeja a una computadora completa y puede ser programado para llevar una o varias tareas específicas en función de su capacidad de procesamiento y memoria.

El microcontrolador es un circuito integrado que contiene los mismos componentes que una computadora, solo que, para desempeñar una tarea específica, por lo tanto, poseen una memoria ROM (memoria solo de lectura), una memoria RAM (memoria de acceso aleatorio), un procesador y puertos de entrada y salida los cuales son utilizados para la comunicación con otros dispositivos como pantallas, sensores, circuitos de control como relevadores y otros.

5.1. Partes de un microcontrolador

A continuación, se describen las partes de un microprocesador.

Figura 22. Diagrama interno de un microcontrolador



Fuente: elaboración propia.

5.1.1. Bus de direcciones

Es un conjunto de pines que definen el tamaño de la memoria que el procesador puede utilizar (aunque aumentando los circuitos puede ser expandida); esta memoria se dividirá en bloques principales uno llamado memoria ROM, RAM, puertos de entrada y salida también otros que a veces no se utilizan por no ser necesarios para la tarea.

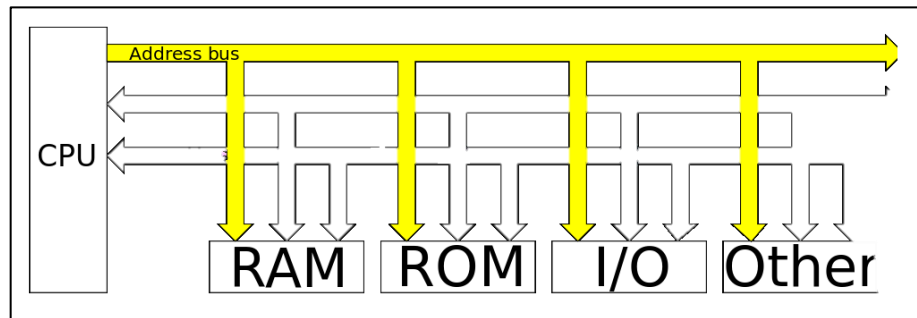
Los pines del bus de direcciones definen la cantidad de lugares a los que se puede acceder; por ejemplo, si se tiene una cantidad de 4 pines las combinaciones posibles entre estos (refiriéndose a tener algunos apagados y otros encendidos) es de 16 posibilidades contando la posibilidad de que todos los pines estén en cero voltios.

Cada una de estas posiciones o direcciones es el equivalente a tener una gaveta de un mueble, por lo tanto, se tendría un mueble con 16 gavetas en las cuales las primeras cuatro se reservaran para datos que no pueden ser borrados (memoria ROM) o se conocen como instrucciones principales; las otras 4 gavetas pueden ser para almacenar los datos que las instrucciones almacenadas produzcan y los cuales son variables en el tiempo (memoria RAM); otras cuatro gavetas pueden ser utilizadas para poner los resultados finales de las operaciones anteriores (datos de entrada y salida) y las últimas cuatro gavetas pueden ser utilizadas para datos de rendimiento de los mismos procesos realizados.

Normalmente, en los microcontroladores se encuentran con una cantidad de RAM bastante grande comparada con la memoria ROM debido a que la cantidad de instrucciones no es tan grande como los datos que generan las operaciones realizadas y tienen una cantidad muy pequeña de direcciones de

puertos de entrada y salida ya que normalmente no son necesarios tantos pines para llevar tareas de control de mecanismos físicos.

Figura 23. **Diagrama de bus de direcciones**



Fuente: elaboración propia.

5.1.2. **Bus de datos**

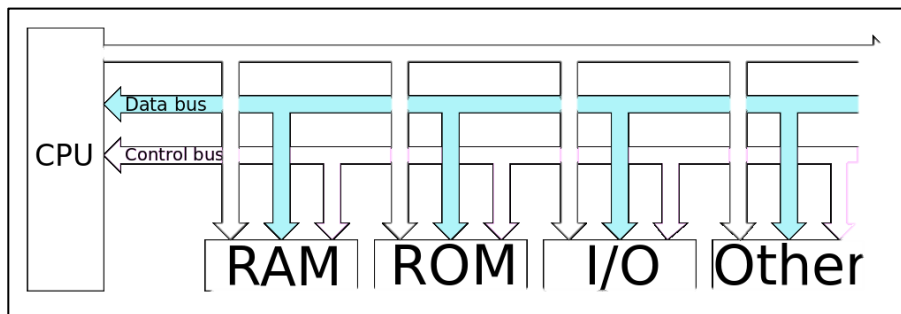
El bus de datos es un conjunto de pines dentro del chip que lleva la información de los resultados, a diferencia de los pines de direcciones que solo indican el lugar en donde estos datos van a ser almacenados.

Con respecto a lo anteriormente planteado, el bus de direcciones indica en que número de gaveta se almacenará y el bus de datos es lo que se introducirá en esa gaveta.

El bus de datos tiene una cantidad de pines limitada y normalmente es un determinante para saber qué tipo de procesador es con el cual se está trabajando. Cuando se habla de un procesador de 8 bits se refiere a que este posee una cantidad de 8 pines en su bus de datos por lo cual es capaz de ofrecer hasta 256 combinaciones entre ellos. Al momento que se redacta este

documento ya es común hablar de microcontroladores de 32 bits los cuales incluso son baratos, aunque difíciles de conseguir ya que normalmente no se necesita tanta capacidad para la mayoría de tareas.

Figura 24. **Diagrama de bus de datos**

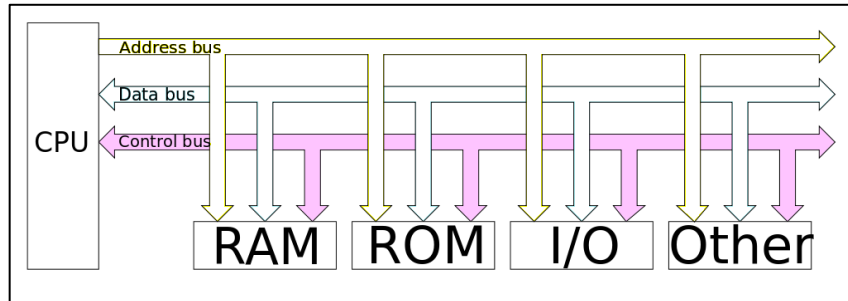


Fuente: elaboración propia.

5.1.3. **Bus de control**

El bus de control es como un subdireccionamiento para evitar que los datos se traslapen en los distintos dispositivos; este conjunto de pines está directamente relacionado con las instrucciones programadas en la memoria interna moviendo los datos resultantes de un lugar a otro por medio del bus de direcciones.

Figura 25. Diagrama de bus de control



Fuente: elaboración propia.

5.2. Arduino DUE

Es la primera placa de desarrollo basada en un microcontrolador de 32 bits. Con 54 pines de entrada y salida, 12 pines de entrada analógica, esta es una tarjeta de desarrollo para tareas avanzadas en procesamiento de señales.

Figura 26. Placa Arduino Due



Fuente: Arduino Due. *Placa de microcontrolador*. <https://store.arduino.cc/due>. Consulta: 11 de diciembre de 2018.

El Arduino Due es una placa de microcontrolador basada en un Atmel SAM3X8E con un procesador ARM Cortex-M3. Esta es la primera placa de desarrollo basada en un procesador ARM de 32 bits que contiene 4 módulos UART (puertos seriales embebidos en el dispositivo) a 84 MHz de velocidad de reloj, una conexión capacitada para USB OTG 2 conversores analógicos, 2 conectores de poder, un conector de poder, un conector SPI y un JTAG, también con un botón de reinicio y uno de borrado.

Tabla I. **Especificaciones técnicas del Arduino Due**

Microcontrolador	AT91SAM3X8E
Voltaje de operación	3,3 V
Voltaje de entrada	7-12 V
Límites de voltaje	6-16 V
Pines digitales	54 (12 salidas PWM)
Pines de entrada analógica	12
Pines de salida analógica	2 (DAC)
Corriente de salida total en todos los pines	130 mA
Corrientes DC para 3,3 V	800 mA
Corrientes DC Para 5 V	800 mA
Memoria flash	512 KB
SRAM	96 KB
Velocidad de reloj	84 MHz
Longitud	101,52 mm
Ancho	53,3 mm
Peso	36 g

Fuente: elaboración propia.

Las ventajas que posee este microcontrolado para calcular la transformada de Fourier en tiempo real para un sensor que está muestreando a una frecuencia máxima de 100 Hz es indispensable para que pueda producirse un procesamiento lo suficientemente rápido debido a que a la vez que está muestreando, procesando datos calculando la transformada de Fourier y actualizando los datos en la pantalla TFT, es necesario tener una alta frecuencia

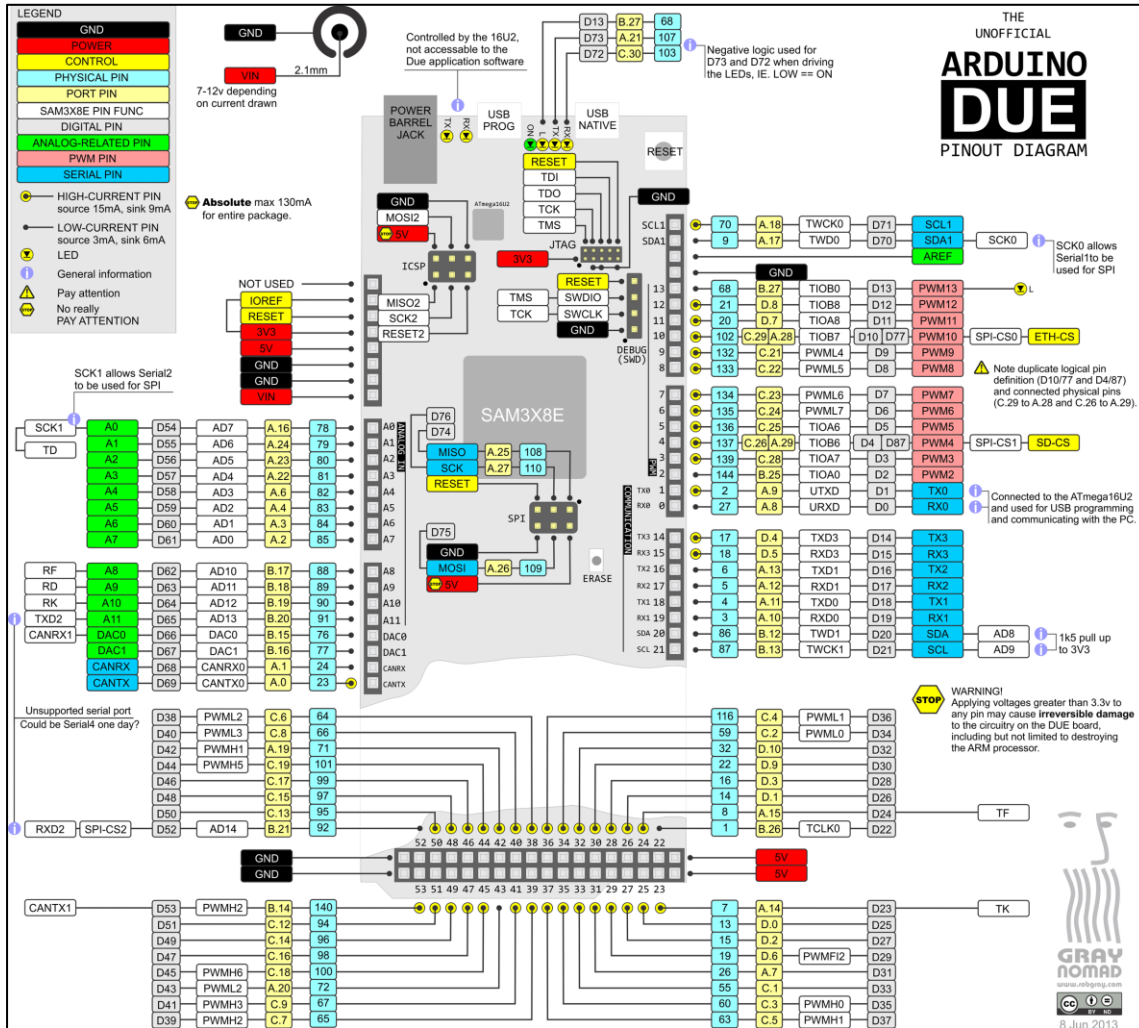
de reloj, una gran tolerancia al ruido de la fuente debido a que en la placa se ha incluido un regulador MPM3610 el cual es un convertor de corriente directa a corriente directa sincrónico el cual es utilizado en aplicaciones de tratamiento de imágenes (de acuerdo a la hoja de datos).

5.2.1. Componentes del Arduino Due

En esta sección se describirá cada uno de los dispositivos que componen la placa de desarrollo Arduino Due entre los cuales están un microcontrolador que se utiliza para grabar el programa dentro el microcontrolador de 32 bits, un conjunto de reguladores de voltaje, el conjunto de conectores utilizados para la fácil conexión con otros dispositivos. En el siguiente diagrama pueden observarse varios colores los cuales representan cada uno de los tipos de conexión que posee la placa: pines de voltaje, de control, pines físicos, analógicos, digitales, de comunicación, puertos enteros y pines de funciones; también, son descritas la función de los botones integrados en la placa y las especificaciones de voltaje que posee el conector de entrada de voltaje para no necesitar usar los puertos USB.

La descripción de esta placa se hace muy intuitiva debido a la relación que hay entre cada función de los pines el lugar en donde fueron colocados respectivamente.

Figura 27. Diagrama de pines del Arduino Due



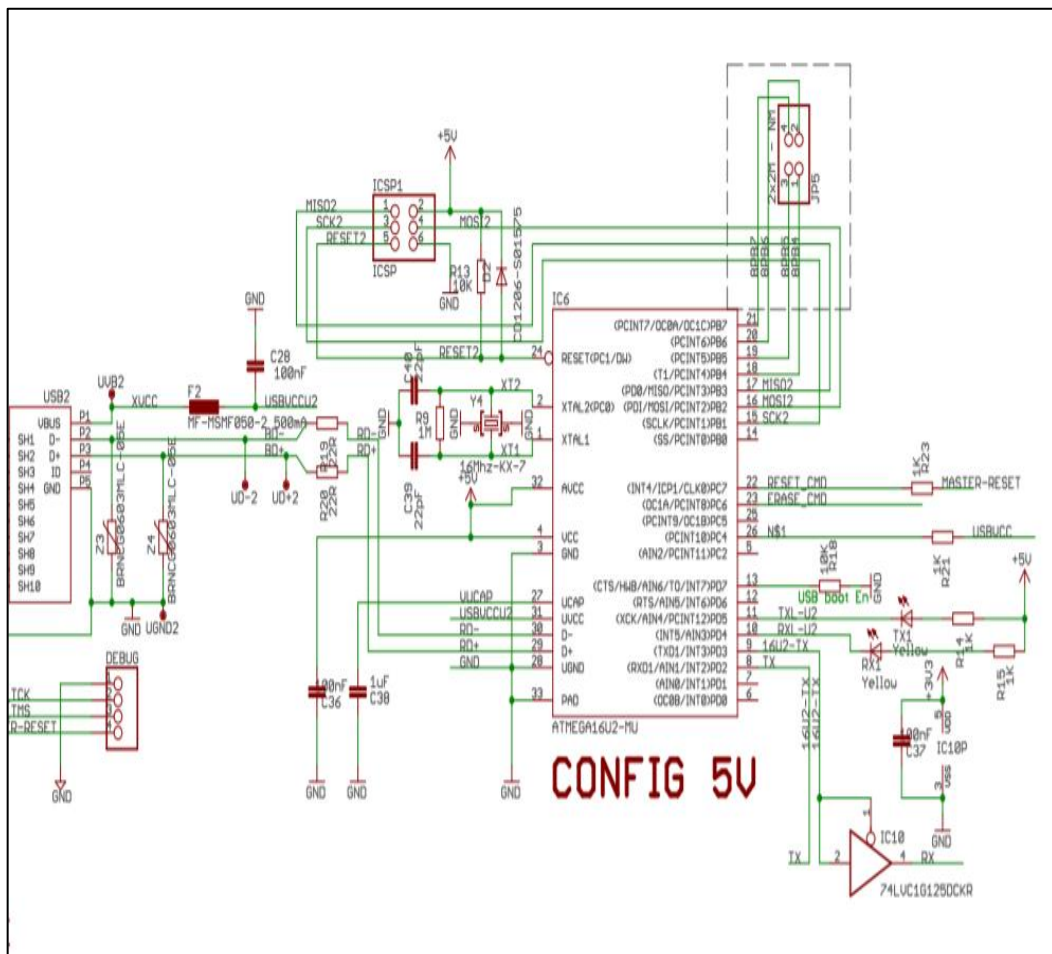
Fuente: IcircuiT. *Arduino Boards-Pin mapping*. <https://icircuit.net/arduino-boards-pin-mapping/141>. Consulta: 11 de diciembre de 2018.

5.2.1.1. Circuito de programación

Para programar el microcontrolador la placa de desarrollo implementa un microcontrolador auxiliar ATMEGA16U2-MU el cual ya tiene un programa

instalado que se encarga de convertir el puerto USB de computadora en un puerto en una interfaz FFPI (*Fast Flash Programming Interface*, interfase de programación rápida para memoria flash) que tiene como finalidad trasladar las instrucciones al SAM3X8E.

Figura 28. **Circuito para el microcontrolador auxiliar**



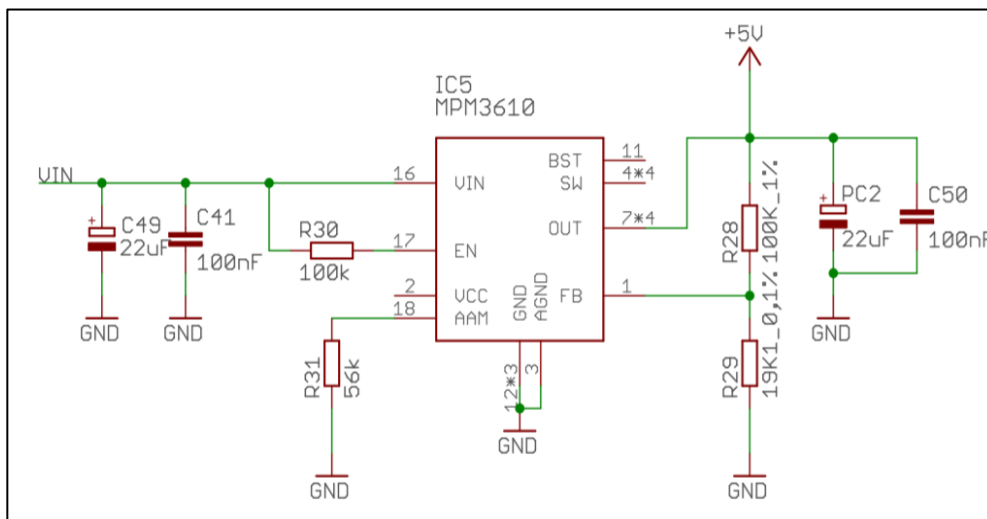
Fuente: IcircuiT. *Arduino Boards-Pin mapping*. <https://icircuit.net/arduino-boards-pin-mapping/141>. Consulta: 11 de diciembre de 2018.

5.2.1.2. Reguladores de voltaje

Los reguladores de voltaje se encargan de estabilizar y proporcionar la amplitud de voltaje correcto en corriente directa, para disminuir al mínimo los errores en el proceso de los ciclos de reloj para que el programa se mantenga constante y estable; también, funcionan como una protección para el circuito, evitando que voltajes que superen el nominal de este circuito se presenten y dañen por completo el microcontrolador.

En el Arduino Due se encuentran dos reguladores: el primero es utilizado para la conexión de voltaje externo que regula la entrada de voltaje de entre 12 y 7 voltios a 5 voltios; luego, está otro que regula el voltaje desde 5 V a 3,3 el cual es el utilizado para polarizar el SAM3X8E.

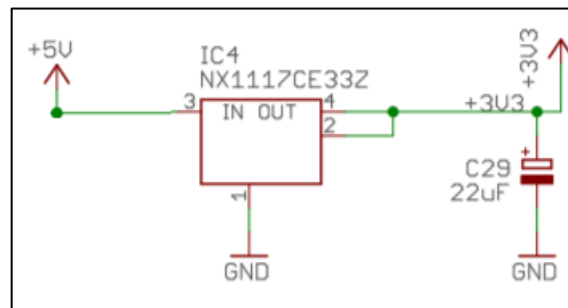
Figura 29. Circuito regulador de voltaje de 12-7 V a 5 V



Fuente: IcircuiT. *Arduino Boards-Pin mapping*. <https://icircuit.net/arduino-boards-pin-mapping/141>. Consulta: 11 de diciembre de 2018.

El diagrama del regulador con salida de 3,3 V es menos complejo debido a que dependen de un componente que ya no representa una conversión de mayores voltajes de entrada.

Figura 30. **Circuito regulador de voltaje de 12-7 V a 5 V**



Fuente: IcircuiT. *Arduino Boards-Pin mapping*. <https://icircuit.net/arduino-boards-pin-mapping/141>. Consulta: 11 de diciembre de 2018.

6. PANTALLA TFT

TFT significa *thin-film transistor* o transistor de película fina, es un tipo de transistor de efecto de campo que se va construyendo de finas películas.

Las pantallas TFT tienen una luz de fondo que es producida a través de uno o varios leds que se encuentran en uno de los lados de la pantalla o detrás de esta. Luego tiene un panel de vidrio que está compuesto por varias capas en donde la primera se compone de dos filtros polarizadores que son paneles de vidrio con líneas horizontales o verticales y estos van juntos y orientados ortogonalmente entre sí y se ubica en la parte superior; su trabajo es polarizar la luz no polarizada en consecuencia crea luz polarizada en un plano perpendicular al panel lo cual significa que redirige la luz en dirección a los ojos de quien esté viendo el panel.

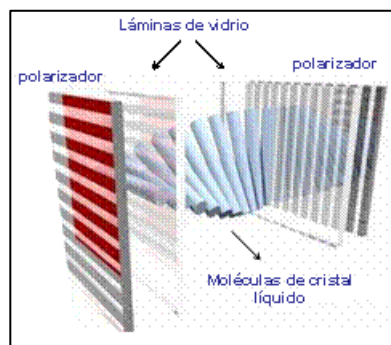
La luz polarizada debe pasar al otro lado girando noventa grados a través del panel; esta tarea se realiza a través de un componente llamado cristal líquido.

Las pantallas utilizan los cristales líquidos neumáticos torcidos, las moléculas en este cristal están torcidas casi noventa grados entre estas, las capas de alineación se colocan en la parte superior e inferior del cristal líquido neumático y se utilizan para girar las moléculas exactamente noventa grados para evitar que la luz atravesase, para producir el negro.

El panel de cristal líquido está recubierto por dos electrodos transparentes que, al aplicarles una diferencia de potencial, se encargan de girar las moléculas neumáticas dentro del cristal, que dejan pasar la luz a través del panel polarizado.

Estos electrodos pueden ajustar el plano de polarización por lo cual puede controlarse la intensidad de la luz que los atraviesa, por lo tanto, se logra un control de la intensidad del color por medio de este método. Las moléculas de cristal líquido se retuercen en proporción a la tensión eléctrica aplicada.

Figura 31. **Célula de cristal líquido**



Fuente: Vkent. *Pantalla cristal líquido*.

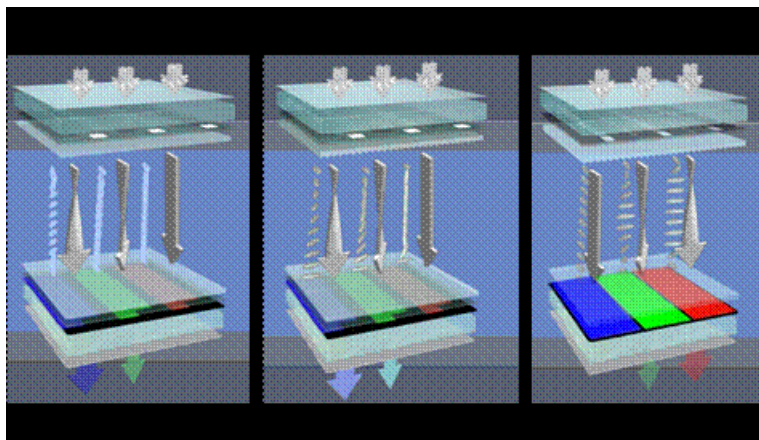
<https://sites.google.com/site/elrincondelacienciavkent/home/numeros-antiores/numero-47/pantalla-cristal-liquido>. Consulta: 11 de diciembre de 2018.

Por último, un filtro de color que hace que los otros colores se reflejen y dejen pasar solo un color como el rojo, verde o azul, se coloca sobre cada uno de estos paneles.

El conjunto de un panel con un solo filtro de color es llamado subpixel y contiene todos los componentes anteriormente mencionados.

Por otro lado, cuando se tienen tres subpixeles, uno de color rojo, otro verde y un azul se le llama pixel propiamente.

Figura 32. **Pixel**



Fuente: Vkent. *Pantalla cristal líquido.*

<https://sites.google.com/site/elrincondelacienciavkent/home/numeros-anteriores/numero-47/pantalla-cristal-liquido>. Consulta: 11 de diciembre de 2018.

Cada píxel de una pantalla se compone de estos tres filtros de color con sus respectivos componentes, y una pantalla completa es un conjunto de estos píxeles distribuidos en forma matricial en toda la pantalla.

Las pantallas TFT, en estos tiempos, son sistemas completos que poseen un circuito para la sincronización de la trama vertical y la trama horizontal que poseen la capacidad de presentar diferentes colores retroiluminados lo cual permite verlas en condiciones de oscuridad, pero si hay demasiada luz hace que sea difícil distinguir las imágenes presentadas.

Para el circuito se eligió un tipo de pantalla que fuese compatible con el módulo Arduino Due por lo cual se utiliza un circuito integrado capaz de reproducir imágenes en un panel TFT.

En el momento en que estas pantallas fueron tomando auge en el mundo de la tecnología se concibieron herramientas para facilitar el uso una de estas herramientas; ya concebidas como módulos totalmente transparentes son los creados por la empresa ITEAD Studio los cuales presentan un módulo completo para la implementación de pantallas TFT de varios tamaños.

Debido a las capacidades que se encontraron, en necesario implementar una pantalla de cuatro pulgadas o más de diagonal ya que al presentar la transformada de Fourier en una más pequeña es difícil encontrar una medición clara de la frecuencia solo con verla.

6.1. ITDB02-4.3

Este es un módulo de pantalla TFT que incluye un cristal líquido con una diagonal de cuatro puntos tres pulgadas de diámetro en una relación de que por cada 16 unidades en la línea horizontal se recorren 9 en la vertical.

Figura 33. **Módulo ITDB02-4.3 vista frontal**



Fuente: Amazon. *Yongse TFT LCD de 4,3 pulgadas de pantalla táctil del módulo Mega2560 DUE Disponible para Arduino*. <https://www.amazon.es/Yongse-pulgadas-pantalla-Mega2560-Disponible/dp/B01F7TTV9O>. Consulta: 11 de diciembre de 2018.

Figura 34. **Módulo ITDB02-4.3 vista trasera**



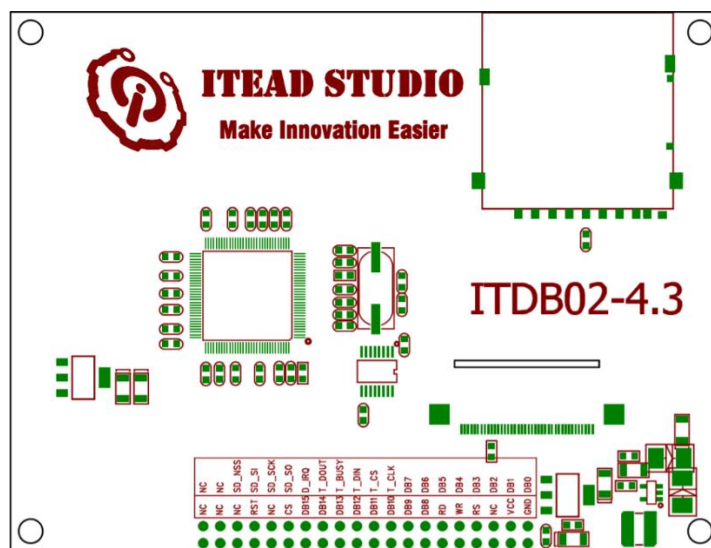
Fuente: Amazon. *Yongse TFT LCD de 4,3 pulgadas de pantalla táctil del módulo Mega2560 DUE Disponible para Arduino.* <https://www.amazon.es/Yongse-pulgadas-pantalla-Mega2560-Disponible/dp/B01F7TTV9O>. Consulta: 11 de diciembre de 2018.

El ITDB02-4.3 es un módulo de pantalla TFT LCD con la capacidad de reproducir sesenta y cinco mil colores de 272 píxeles de altura y 480 píxeles de longitud de resolución, desarrollado especialmente para placas de Arduino UNO o Mega2560 R3. El controlador de esta pantalla es el circuito integrado SSD196 el cual soporta hasta 16 bits en su interface de datos con un control de interface basado en cuatro cables; además, este módulo contiene una pantalla táctil y un puerto para conectar una tarjeta de memoria.

El módulo ITDB02-4.3 está soportado en la biblioteca de programación llamada UTFT Library en la cual se encuentran las herramientas necesarias para la codificación de la información para ser desplegada de forma intuitiva.

Para el uso de esta pantalla es necesario valerse de un módulo de conexión auxiliar debido al intercambio que hay en la disposición de los pines de las placas de desarrollo Arduino por lo cual se recomienda utilizar la tarjeta ITDB02 MEGA Shield (IM120417024) la cual se utiliza para acoplar la pantalla al Arduino DUE.

Figura 35. **Módulo ITDB02-4.3 diagrama de pines**



Fuente: Itead Studio. *Hoja de datos del módulo ITDB02-4.3.* <https://www.itead.cc/>. Consulta: 11 de diciembre de 2018.

Tabla II. **Características eléctricas del módulo ITDB02-4.3**

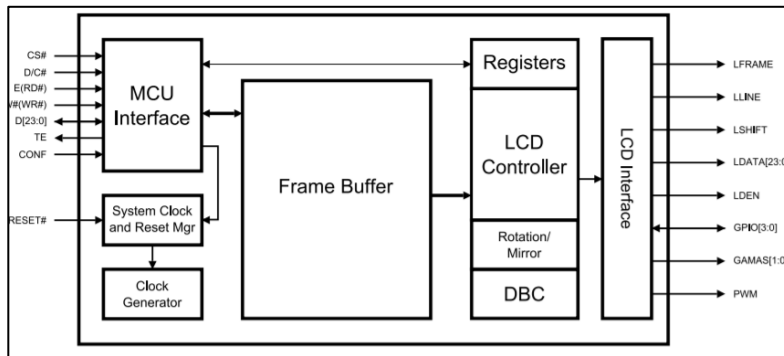
Especificaciones		mínima	típica	máxima	unidad
Voltaje		4,5	5	5,5	VDC
Voltaje de salida VH	Voltaje objetivo = 3,3 V	3	3,3	3,6	V
	Voltaje objetivo = 5 V	4,5	5	5,5	V
salida de voltaje VL:		-0,3	0	0,5	V
Consumo de corriente			20	40	mA

Fuente: elaboración propia.

6.1.1. **SSD1963**

El SDS1963 es un circuito integrado que se utiliza como un controlador digital que tiene una capacidad de 1 215 k bytes de memoria que soporta pantallas hasta de una cantidad de pixeles de 864 horizontalmente por 480 verticalmente y 24 bits de profundidad de color. Este equipa una interface de microcontrolador con diferentes anchos de canales para recibir comandos de datos de gráficos. Este funciona como una interface de soporte sin una ram en común con el controlador, previendo una profundidad de color hasta de 24 bits.

Figura 36. Diagrama de bloques del SSD1963

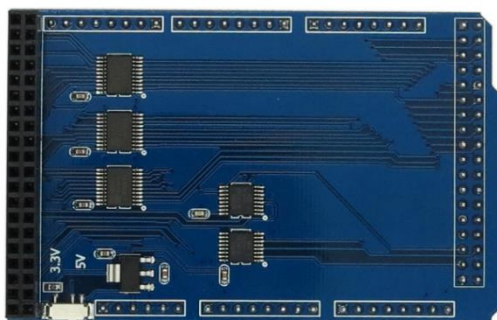


Fuente: Solomon Systech. *Hoja de datos del circuito SSD1963*. <http://www.solomon-systech.com/>. Consulta: 11 de diciembre de 2018.

6.2. Módulo de conexión ITDB02 MEGA Shield

A continuación, se presenta el módulo de conexión “ITDB02 MEGA Shield”.

Figura 37. ITDB02 MEGA Shield (IM120417024) vista delantera



Fuente: ITEAD. *ITDB02 Arduino mega shield starter kit for 3.3v-5v mainboard compatible with Arduino mega pins*. <https://www.itead.cc/itdb02-arduino-mega-shield.html>. Consulta: 11 de diciembre de 2018.

Figura 38. **ITDB02 MEGA Shield (IM120417024) vista trasera.**



Fuente: ITEAD. *ITDB02 Arduino mega shield starter kit for 3.3v-5v mainboard compatible with Arduino mega pins*. <https://www.itead.cc/itdb02-arduino-mega-shield.html>. Consulta: 11 de diciembre de 2018.

El módulo opera con 3,3 voltios lo cual no lo hace compatible con los pines de Arduino Mega; entre las características principales puede observarse que es compatible con los pies del lector de tarjetas incluidos en el módulo ITDB02-4.3 y también el panel táctil.

Está formada por 3 circuitos integrados de tipo regulador de cocho puertos conformados por el dispositivo 74HC7541, dos reguladores sexagesimales de tres estados y una interface de 40 pines para pantallas LCD; por último, contiene un regulador para entrada de 5 voltios y una salida de 3,3 voltios ya que este es el voltaje de operación de los pines en el Arduino Due.

El conjunto de reguladores se encarga de convertir el voltaje de las señales del Arduino Due a señales del módulo de pantalla, y redireccionar el mapa de pines para que coincidan de forma mecánica con las señales de control entre ambos módulos.

7. LECTURA DE EJE DEL SENSOR

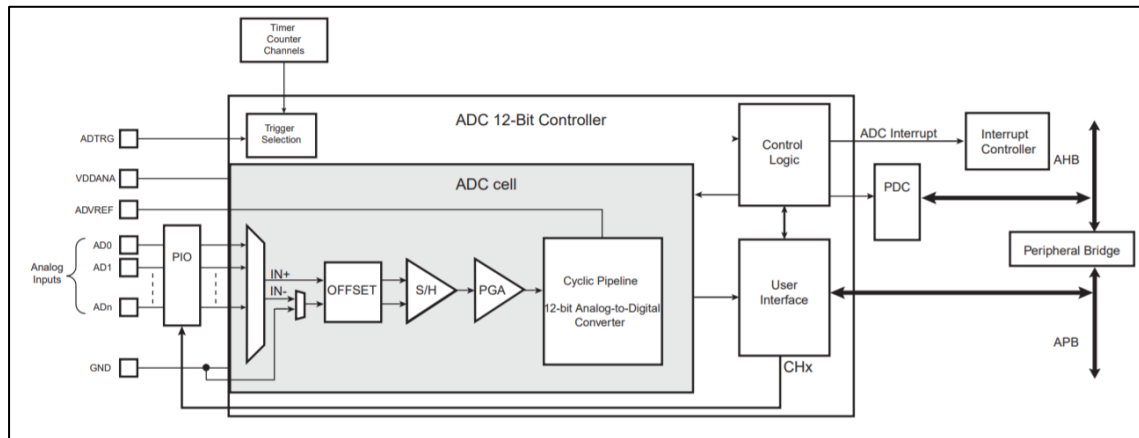
Como el sensor utilizado es analógico en sus salidas se utiliza el conversor analógico dentro del microcontrolador; para esta actividad es necesario configurar adecuadamente llamando a los registros necesarios para activar el dispositivo; se utiliza uno de los pines del microcontrolador como entrada analógica.

7.1. Descripción del ADC (conversor analógico a digital) del SAM3X8EA

El módulo ADC está basado en un convertidor de 12 bits administrado por un módulo controlador de ADC. Este también integra un multiplexor de 16 entradas a una salida (un multiplexor es un circuito que conmuta las entradas a la misma salida por medio de unos pines que seleccionan cada una de las entradas en diferentes tiempos).

La conversión puede extenderse de 0v a ADVREF (voltaje de referencia colocado en este pin) el cual permite una conversión lineal de 12 bits, eso significa que el voltaje de entrada será convertido a un conjunto de números entre 0 y 2^n siendo n la cantidad de bits que el conversor posea.

Figura 39. Diagrama de bloques del ADC.



Fuente: Microchip. Hoja de datos del micro controlador SAM3X8EA.

<https://www.microchip.com/wwwproducts/en/ATsam3x8e>. Consulta: 11 de diciembre de 2018.

Un circuito automático comparador permite la detección de valores a través de un umbral en un rango determinado.

7.2. Características del ADC

- Resolución de 12 bits.
- Tasa de conversión de 1 MHz.
- Operación de fuente de alimentación de amplio rango.
- Voltaje de entrada diferencial o de entrada simple seleccionable.
- Ganancia programable para el rango máximo de entrada de escala completa 0 – VDD.
- Multiplexor integrado que ofrece hasta 16 entradas analógicas independientes.
- Activación y desactivación individual de cada canal.
- Disparador de hardware o software.

- Pin de disparo externo.
- Salidas de contador de temporizador.
- Línea de eventos PWM.
- Unidad de entrada de falla PWM.
- Soporte PDC.
- Posibilidad de configuración de tiempos ADC.
- Dos modos de suspensión y secuenciador de conversión.
- Activación automática al activar y volver al modo de suspensión después de conversiones.
- Posibilidad de secuencia de canal personalizada.
- Modo de espera para respuesta rápida del tiempo de activación.
- Capacidad de apagado.
- Comparación automática de ventanas de valores convertidos.
- Registros de protección contra escritura.

7.3. Ecuación de transferencia del digitalizador o conversor analógico digital

Por lo cual, asumiendo que el voltaje de ADVREF es el interno del microcontrolador el cual equivale a 3,3 v, se transforma en 0 para 0 V y 2^{12} para 3,3 V ($2^{12} = 4\ 096$); esta conversión se hace de forma lineal por lo cual puede inferirse una ecuación basada en el modelo de variación proporcional. La siguiente ecuación muestra convertir el valor del ADC a voltios lo cual es la primera ecuación de transferencia que se observa siendo propia del convertidor analógico digital.

$$\text{Voltios} = \frac{\text{Valor de ADC}}{4\ 096} * 3,3\ V$$

7.4. Ecuación de transferencia del sensor ADXL335

Para convertir los valores de voltaje a medidas de aceleración se utiliza la función de transferencia del sensor la cual es sugerida por la hoja de datos.

Como se conoce que la sensibilidad del sensor varía proporcionalmente de forma lineal con el voltaje de polarización, si se utiliza un voltaje de fuente de 3,6 V, la sensibilidad de la salida es 360 mV/g (siendo g la aceleración gravitacional); y si se utiliza un voltaje de fuente de = 2 V, la sensibilidad de la salida es 195 mV/g, por lo cual se puede inferir que se mantiene lineal con un error de 5 mV/g. Con estos parámetros sugeridos por la hoja de datos puede ser calculada la sensibilidad del sensor lo cual servirá para obtener su apropiada función de transferencia por medio de una regla de conversión de factor unitario.

$$3,6 V = \frac{360 mV}{g}$$

Entonces:

$$3,3 V \frac{\frac{360 mV}{g}}{3,6 V} = \frac{330 mV}{g}$$

7.5. Ecuación de transferencia total

La ecuación de transferencia total se refiere a convertir los datos del conversor analógico, los cuales son números, a dimensionales directamente en datos asociados a una magnitud física; en este caso, aceleración a razón de la fuerza gravitacional.

Combinando los datos anteriores se puede obtener la función de transferencia total:

Sabiendo que la sensibilidad del sensor es:

$$\frac{330 \text{ mV}}{g}$$

Es necesario invertir esta función para poder calcular cuántos g se tienen por unidad de voltio.

$$\frac{g}{330 \text{ mV}} = \frac{g}{0,33 \text{ V}} = \frac{3,03 \text{ g}}{V}$$

Sabiendo que el sensor tiene un rango de medición de – 3 g a 3 g, siendo el cero una medición de voltaje intermedia la cual se mantendrá constante a menos que exista una variación de aceleración en un eje en cualquiera de las dos direcciones, se espera que cuando el sensor esté en una posición de reposo la medición de voltaje sea la mitad del valor nominal de voltaje de polarización del sensor.

$$\text{Voltios} = \frac{\text{Valor de ADC}}{4\ 096} \cdot 3,3 \text{ V}$$

Haciendo una multiplicación de las dos funciones de transferencia se obtiene la función total del sistema en conjunto.

$$\frac{\text{Valor de ADC}}{4\ 096} \cdot 3,3 \text{ V} \cdot \frac{3,03 \text{ g}}{V}$$

Como el sensor solo es capaz de medir hasta + 3 g, se descartarán los últimos decimales que superan este valor (0,03 g por voltio) lo cual acota los valores de ADC no dejando que estos superen cierto rango ya que representarían una saturación del sensor lo cual impediría calcular de forma correcta la transformada de Fourier.

Por ahora se utilizará la mitad del valor de la sensibilidad para la polarización del sensor para marcar el cero del sensor analógico que deja la ecuación con un corrimiento negativo.

$$\text{Valor de aceleración} = \frac{\text{Valor de ADC}}{4\,096} \cdot 3,3\,V \frac{3,03\,g}{V} - 4,9995\,g$$

Ahora se calcularán los rangos máximos y mínimos de ADC permitidos; fuera de estos rangos el sensor se presentará saturado por lo cual no podrán ser tomados en cuenta para generar el algoritmo.

$$\pm 3\,g = \frac{\text{Valor de ADC}}{4\,096} \cdot 3,3\,V \frac{3,03\,g}{V} - 4,9995\,g$$

Simplemente sustituyendo para cada uno de estos valores se pueden obtener los valores de ADC descartables.

$$\text{Valor de ADC} = 4\,096 \frac{\pm 3\,g + 4,9995\,g}{3,3\,V \frac{3,03\,g}{V}}$$

$$\text{Valor de ADC MÁXIMO} = 4\,096 \frac{3\,g + 4,9995\,g}{3,3\,V \frac{3,03\,g}{V}} = 3\,276$$

$$\text{valor de ADC MÍNIMO} = 4\,096 \frac{-3\text{ g} + 4,9995\text{ g}}{3,3\text{ V} \frac{3,03\text{ g}}{\text{V}}} = 819$$

Por tanto, entre los valores de 819 y 3 276 estarán los datos correctos; por arriba o por debajo de estos valores, el sensor estará saturado.

8. ALGORITMO DE LA TRANSFORMADA DE FOURIER

A continuación, se hará una descripción de cómo se construye el código para evaluar las amplitudes en función de la frecuencia, las cuales son necesarias para obtener un espectro de frecuencias característico de las vibraciones producidas por un sistema mecánico.

8.1. Desarrollo de la transformada de Fourier

Ecuación de la DFT (transformada de Fourier discreta):

$$x[k] = \sum_{n=0}^{N-1} X[n] e^{-j2\pi(k\frac{n}{N})}$$

Donde:

$x[k]$: representa la frecuencia discreta k-ésima posición

$X[n]$: representa la muestra en el n-ésimo momento

N: total de muestras tomadas

N: representa el tiempo discreto en el que fue tomada la muestra

K: representa la frecuencia discreta

Puede expandirse por medio de la identidad de Euler en la siguiente ecuación:

$$x[k] = \sum_{n=0}^{N-1} X[n] \cos \left[\frac{2\pi k}{N} n \right] + j \sum_{n=0}^{N-1} X[n] \sin \left[\frac{2\pi k}{N} n \right]$$

La siguiente ecuación muestra cuánto vale cada intervalo de frecuencia.

$$f_i = \frac{f_s}{N} = \frac{k}{N}$$

f_i : intervalo de frecuencia

f_s : frecuencia de muestreo

La siguiente ecuación muestra cuánto vale cada uno de los intervalos de frecuencia en función de las muestras:

$$k(n) = \frac{f_s}{N} n$$

Por lo tanto, $0 < k(n) < \frac{f_s}{N} n - \frac{f_s}{N}$ ya que se toma en cuenta la posición cero entonces hay que restarle un intervalo al final.

$$0 < k(n) < \frac{f_s}{N} (n - 1)$$

Por lo tanto, k representa el intervalo de la frecuencia de muestreo. Simplificaciones para la creación del algoritmo.

$$x[k] = \sum_{n=0}^{N-1} X[n] \cos \left(\frac{2\pi k}{N} n \right) + j \sum_{n=0}^{N-1} X[n] \sin \left(\frac{2\pi k}{N} n \right)$$

Parte real:

$$Re[x[k]] = \sum_{n=0}^{N-1} X[n] \cos\left(\frac{2\pi k}{N} n\right)$$

Parte imaginaria:

$$Im[x[k]] = \sum_{n=0}^{N-1} X[n] \sin\left(\frac{2\pi k}{N} n\right)$$

Módulo (lo cual representa la amplitud de la señal en cada una de sus frecuencias).

$$|x[k]| = \sqrt{Re[x[k]]^2 + Im[x[k]]^2}$$

Sustituyendo:

$$|x[k]| = \sqrt{\sum_{n=0}^{N-1} X[n] \cos\left(\frac{2\pi k}{N} n\right)^2 + \sum_{n=0}^{N-1} X[n] \sin\left(\frac{2\pi k}{N} n\right)^2}$$

8.2. Datos precalculados de la transformada de Fourier

Se nota que hay datos repetitivos, por lo tanto, pueden ser precalculados para que el algoritmo solo tenga que realizar la multiplicación; y no calcular senos ni cosenos, lo cual, consume tiempo de procesamiento.

Por lo tanto, se nombran esos datos periódicos y se designan como:

Para la parte real:

$$a(n) = \cos \left[\frac{2\pi k}{N} n \right]$$

Y, para la parte imaginaria:

$$b(n) = \sin \left[\frac{2\pi k}{N} n \right]$$

Se puede notar que:

$$\cos \left[\frac{2\pi k}{N} n \right] \quad y \quad \sin \left[\frac{2\pi k}{N} n \right]$$

Son los datos calculados del coseno y el seno en la posición n con una frecuencia angular constante de $\omega = \frac{2\pi k}{N}$. Para convertirlos en funciones calculables es necesario conocer el periodo de muestreo.

$$t_s = \frac{1}{k}$$

t_s : periodo de muestreo

Esto se utilizará para multiplicarlo por cada una de las muestras convirtiéndolo así en función de seno y coseno calculables dependientes de n .

$$a(n) = \cos \left(\frac{2\pi k}{N} t_s n \right)$$

$$b(n) = \sin\left(\frac{2\pi k}{N} t_s n\right)$$

Sustituyendo t_s :

$$a(n) = \cos\left(\frac{2\pi k}{N} \frac{1}{k} n\right)$$

$$b(n) = \sin\left(\frac{2\pi k}{N} \frac{1}{k} n\right)$$

Anulando factores:

$$a(n) = \cos\left(\frac{2\pi \cancel{k}}{N} \frac{1}{\cancel{k}} n\right)$$

$$b(n) = \sin\left(\frac{2\pi \cancel{k}}{N} \frac{1}{\cancel{k}} n\right)$$

Finalmente, se obtiene la simplificación para generar los valores precalculados.

$$a(n) = \cos\left(\frac{2\pi}{N} n\right)$$

$$b(n) = \sin\left(\frac{2\pi}{N} n\right)$$

Lo cual demuestra que se pueden calcular estos valores sin tomar en cuenta la frecuencia de muestreo.

$$|x[k]| = \sqrt{\sum_{n=0}^{N-1} X[n]a(n)^2 + \sum_{n=0}^{N-1} X[n]b(n)^2}$$

8.3. Algoritmo no optimizado de la transformada de Fourier

El algoritmo:

$$x[k] = \sum_{n=0}^{N-1} X[n]a(n) + j \sum_{n=0}^{N-1} X[n]b(n)$$

$$Re[x[k]] = \sum_{n=0}^{N-1} X[n]a(n)$$

$$Im[k] = \sum_{n=0}^{N-1} X[n]b(n)$$

8.3.1. Parte real

Algoritmo para el cálculo de la parte real.

$$Re[x[k]] = \sum_{n=0}^{N-1} X[n]a(n)$$

Si:

```
For(int k = 0; k=N-1 ;k++)
{
```

```

    For(int n =0; n=N-1; n++)
    {
        ReX[K]+= X[n]a(n);
    }
}

```

8.3.2. Parte imaginaria

Algoritmo para el cálculo de la parte imaginaria.

$$Im[x[k]] = \sum_{n=0}^{N-1} X[n]b(n)$$

Si:

```

For(int k = 0; k=N-1 ;k++)
{
    For(int n =0; n=N-1; n++)
    {
        ImX[K]+= X[n]b(n);
    }
}

```

8.3.3. Combinación de parte imaginaria y parte real

para optimizar los cálculos de la parte real e imaginaria se puede optimizar realizando los dos cálculos en los mismos ciclos y posicionando los datos como pares o impares.

```

For(int k = 0; k=N-1 ;k++)
{
  For(int n =0; n=N-1; n++)
  {
    ReX[2K]+= X[n]a(n); //cálculo de pares
    ImX[2K + 1]+= X[n]b(n);//calculo de impares
  }
}

```

Por último, es necesario calcular el módulo para conocer cada amplitud en cada intervalo de frecuencia.

$$|x[k]| = \sqrt{\sum_{n=0}^{N-1} X[n]a(n)^2 + \sum_{n=0}^{N-1} X[n]b(n)^2}$$

Por el algoritmo anterior:

$$|x[k]| = \sqrt{ReX[2K]^2 + ImX[2K + 1]^2}$$

El algoritmo quedaría de la siguiente forma:

```

For(int k = 0; k=N-1 ;k++)
{
  Abs[k] = sqrt(ReX[2K]^2 + ImX[2K + 1]^2);
}

```

Por último, se puede optimizar el algoritmo una vez más utilizando los ciclos anteriores.

```

For(int k = 0; k=N-1 ;k++)
{
  For(int n =0; n=N-1; n++)
  {
    ReX[2K]+= X[n]a(n); //cálculo de pares
    ImX[2K + 1]+= X[n]b(n);//calculo de impares
  }
  Abs[k] = sqrt(ReX[2K]2 + ImX[2K + 1]2);
}

```

8.4. Biblioteca SplitRadixRealP.h para lenguaje c++

Esta biblioteca es la herramienta que ayudará a calcular la transformada de Fourier dentro del microcontrolador. Especialmente un archivo “.h”, conocido como archivo encabezado es el encargado de situar la estructura de las variables dentro del código; por lo cual dentro de este archivo no se recomienda escribir ciclos iterativos; es utilizado mayormente para la creación de las variables (asignación de tipo y de nombre) y su inicialización (referente a colocarle un valor inicial).

Dentro del archivo SplitRadixRealP.h se observa el siguiente código al abrirlo en cualquier procesador de texto.

```

#ifndef _SPLITRADIXREALP_H_
#define _SPLITRADIXREALP_H_

```

Estas líneas se encargan de definir un nombre para la biblioteca; la primera sentencia pregunta si no hay alguna biblioteca con ese nombre, y la segunda

línea define ese nombre si anteriormente no se ha encontrado una definida con el nombre.

Las siguientes son otro conjunto de bibliotecas auxiliares que ayudarán a la operación de esta, por lo cual son llamadas de la siguiente forma:

```
#include <stdio.h>
```

Esta biblioteca no es necesaria dentro del microcontrolador, se encarga de direccionar las salidas y las entradas desde una computadora, sirve para poder probar la biblioteca haciendo un llamado a las funciones y operando los datos de entrada para proporcionar una salida.

```
#include <stdlib.h>
```

Esta biblioteca contiene ciertas funciones para la gestión de los datos como una división de enteros, y otra operación de valor absoluto, es una biblioteca estándar del lenguaje que sirve para gestionar los datos como asignación de variables.

```
#include <math.h>
```

Esta biblioteca incluye algoritmos para las operaciones matemáticas como senos, cosenos, potencias, raíz cuadrada, entre otros.

```
#include <inttypes.h>
```

Contiene funciones para poder presentar los resultados de forma ordenada.

```
#define RAD_CONV 180,0 / 3,14159265
```


Define una variable para la conversión entre radianes y grados.

```
#define FFT_SIZE      2048
```

Esta es la cantidad de datos precalculados.

```
#define MIRROR      FFT_SIZE / 2
```

Representa la mitad de los datos precalculados.

```
#define LOG2_FFT      11 /* log2 FFT_SIZE */
```

Tamaño de la variable de logaritmo en base 2 (`/* */` lo que se escribe dentro de estos símbolos son comentarios y no cuentan como parte del código).

```
#define NWAVE          2048 /* full length of Sinewave[] */
```

Tamaño de la función senoidal en forma discreta, se refiere al número de muestras que componen la función.

```
#define NQUAT          NWAVE / 4
```

Esta es una variable que define sobre un cuarto de la señal senoidal.

Lo demás que se encuentra dentro de la biblioteca son los valores precalculados y el uso de la función antes vista, adaptada para los cálculos necesarios.

8.5. Biblioteca UTFT.h para lenguaje c++

Esta herramienta se utiliza para la comunicación con la pantalla TFT ya que contiene compatibilidad con los módulos utilizados y posee un gran conjunto de funciones las cuales aceleran el proceso de programación dentro del microcontrolador.

Esta herramienta tiene un soporte multiplataforma para placas LCD TFT en color *Copyright (C) 2015 Rinky-Dink Electronics, Henning Karlsen. Todos los derechos reservados.*

Esta biblioteca es una actualización de ITDB02_Graph, ITDB02_Graph16 y las bibliotecas RGB_GLCD para Arduino y chipKit (otra placa de desarrollo parecida a Arduino). Como el número de los módulos y controladores de pantalla compatibles comenzaron a aumentar, era hora de que se construyera una biblioteca única y universal, ya que será mucho más fácil de mantener en el futuro.

La funcionalidad básica de esta biblioteca se basó originalmente en el código de demostración proporcionado por ITEAD studio (para los módulos ITDB02) y NKC Electronics (para el módulo / escudo RGB GLCD).

Esta biblioteca admite una serie de gráficos de 8 bits, 16 bits en serie.

Para obtener una lista completa de los módulos y controladores de pantalla probados, véase el documento `UTFT_Supported_display_modules _ & _ controllers.pdf`.

Cuando se utilizan módulos de visualización de 8 bits y 16 bits, hay algunos requisitos que debe cumplir. Estos requisitos se pueden encontrar en el documento UTFT_Requirements.pdf. No hay requisitos especiales cuando se utilizan pantallas en serie. Se puede encontrar la última versión de la biblioteca en <http://www.RinkyDinkElectronics.com/>. Esta biblioteca es software libre; se puede redistribuirlo y / o modificarlo bajo los términos de la licencia CC BY-NC-SA 3.0.

9. DESARROLLO

De la siguiente forma deben llevarse a cabo los experimentos necesarios para poner en funcionamiento el dispositivo para que los datos obtenidos de este tengan validez.

Para un ejemplo práctico y simple se utilizará un motor trifásico con un módulo de control de velocidad para observar el cambio de revoluciones por minuto en la gráfica en tiempo real.

9.1. Sistema mecánico seleccionado para el experimento

El sistema está compuesto por un motor trifásico con una hélice de dos aspas conectado a un control de velocidad.

El fin de las aspas es que en una de ellas puede colocarse una masa, que hará perder el punto de equilibrio en el sistema provocando una vibración extra. Este tipo de dispositivo se encuentra en los teléfonos inteligentes dotándolos de la capacidad de proveer vibraciones a modo de alarma para evitar emitir ondas sonoras.

Figura 40. **Motor de vibración para teléfono celular (MOT-2265)**



Fuente: Inteligencia artificial. *Vibradores industriales*.

<http://inteligenciaartificialyrobotica.com/esp/item/340/mot-2265-motor-vibrador>. Consulta: 11 de diciembre de 2018.

Este pequeño motor de corriente continua produce vibraciones al girar un eje excéntrico en más de 10 000 RPM cuando se alimenta a 3 V. Los motores de este tipo se encuentran comúnmente en los teléfonos celulares y otros dispositivos que utilizan la vibración para una respuesta táctil, su pequeño tamaño (11,6 × 4,6 × 4,8 mm) y peso ligero (0,8 g) hacen que sea fácil de integrar en sistemas con restricciones de espacio reducidos. El motor tiene 1,5" y está encerrado en un manguito de caucho extraíble que le da superficies planas para el montaje y evita que castañetearan contra lo que se monta. Está destinado para el funcionamiento alrededor de 3 V (2,4 V a 3,5 V recomendado), y la polaridad no es importante (el motor puede funcionar CW o CCW) citando a la página web inteligencia artificial y robótica del vendedor del artículo.

9.1.1. Motor trifásico

El motor utilizado para el experimento es un dispositivo especializado para usarse en aviones a control remoto, es un motor trifásico pequeño con la capacidad de levantar hasta 1 libra si se le ponen aspas suficientemente largas.

En su etiqueta posee la siguiente descripción 2834/KV1500. Esto se lee como, kv o constante de velocidad, que comprende la cantidad de revoluciones por minuto por cada voltio. Si se quisiera saber de forma teórica a cuántas revoluciones está girando el motor, se tiene que multiplicar los kv por la cantidad de voltios que proporcione la fuente de voltaje directa al motor, tomando en cuenta las pérdidas en el circuito de control de velocidad. Hay que tomar en cuenta que este dato es tomado sin ninguna carga en el motor, eso significa que debe girar solo sin ningún otro dispositivo como la hélice.

Figura 41. **Motor utilizado como generador de vibración (2834/KV1500)**



Fuente: Inteligencia artificial. *Vibradores industriales.*

<http://inteligenciaartificialyrobotica.com/esp/item/340/mot-2265-motor-vibrador>. Consulta: 11 de diciembre de 2018.

Para lograr la correcta medición de las revoluciones por minuto que genera el motor se debe hacer por medio de una medición indirecta.

Para obtener la cantidad real de revoluciones por minuto se colocó un sensor de luz mientras se le apuntaba una luz directamente, y con un trozo de papel que se sujetaba en la parte del motor, se hacía que este trozo de papel cubriera la luz cada vez que pasara a través de este; luego, se observaba la señal producida por medio de un contador de frecuencia lo cual mostraba cuántas veces el sensor quedaba cubierto en un segundo; haciendo la multiplicación de estas veces por 60 segundos se obtuvieron las revoluciones por minuto reales que producía el motor; este dato será utilizado para observar en el medidor que las revoluciones por minuto coinciden con lo visto en la pantalla.

El número 2 834 denomina el tamaño del dispositivo 3,4 cm de alto y 2,8 cm de ancho.

9.1.2. Controlador de velocidad para motor trifásico

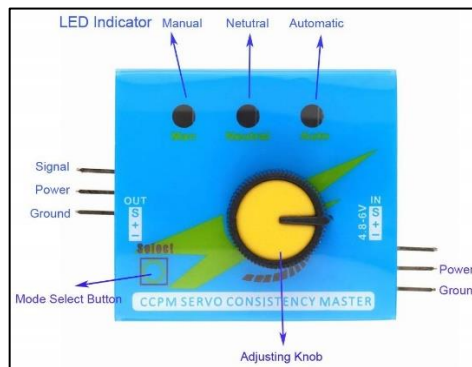
El control de velocidad se basa en dos circuitos totalmente distintos: el primero es un generador de señal de onda cuadrada y el segundo transforma esta onda cuadrada en 3 señales que tienen un retardo entre estas.

9.1.2.1. Generador de señal de onda cuadrada

Este es un dispositivo compuesto con una perilla y un circuito que se apaga y se enciende 50 veces por segundo.

La perilla hace variar el tiempo de encendido sin afectar que se sigan dando las 50 veces de encendido y apagado del dispositivo; cuando el tiempo de encendido aumenta el de apagado disminuye.

Figura 42. **Motor utilizado como generador de vibración (2834/KV1500)**



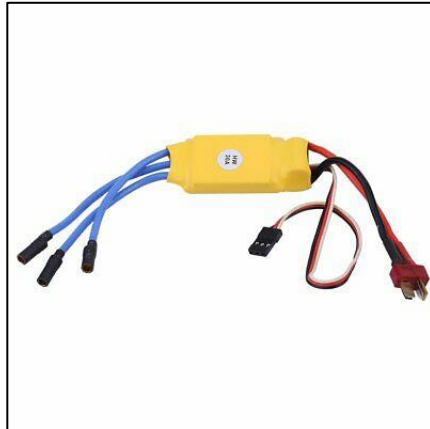
Fuente: Amazon. *Lower priced items to consider.* <https://www.amazon.com/RCmall-Digital-Consistency-Controller-Checker/dp/B01BY5LOZE>. Consulta: 11 de diciembre de 2018.

9.1.2.2. Control de velocidad electrónico

Conocido por sus siglas en inglés ESC (*Electronic Speed Controller*), es un circuito encargado de proporcionar las 3 señales para el motor de tres fases toma como señal de entrada un generador de onda cuadrada a 50 cambios de voltaje por segundo y convertirla en la forma necesaria para hacer girar el motor en función del ancho de pulso o el tiempo que los ciclos de la señal se ponen en un voltaje positivo de 5 voltios.

Este dispositivo requiere un voltaje más alto en la entrada que se encarga de alimentar solo el motor y también el generador de onda cuadrada con 5 voltios.

Figura 43. **Motor utilizado como generador de vibración (2834/KV1500)**

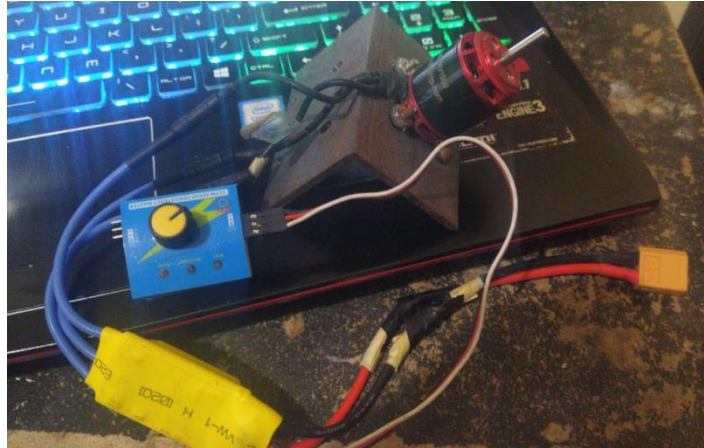


Fuente: Ebay. *HW30A Controlador de Velocidad Brushless ESC para DJI EMAX FPV DRONE RC Quadcopter 4 un.* <https://www.ebay.com/itm/30A-Brushless-Speed-Controller-ESC-for-DJI-EMAX-FPV-Drone-RC-Quadcopter-4pcs/143425713010?hash=item2164d6b372:g:zZsAAOSw0Uddt6DK>. Consulta: 11 de diciembre de 2018.

9.1.3. Sistema de mecánico generador de vibraciones armado

Al tener todos los componentes es necesario interconectarlos entre ellos para observar los cambios que puedan verse en la señal por medio de girar la perilla del generador de señal de onda cuadrada.

Figura 44. **Sistema mecánico**



Fuente: elaboración propia.

9.2. Montaje de los módulos que componen el analizador de vibraciones

El analizador de vibraciones está compuesto por cuatro módulos principales:

- Sensor
- Arduino Due
- Módulo acoplador de pantalla TFT
- Pantalla TFT

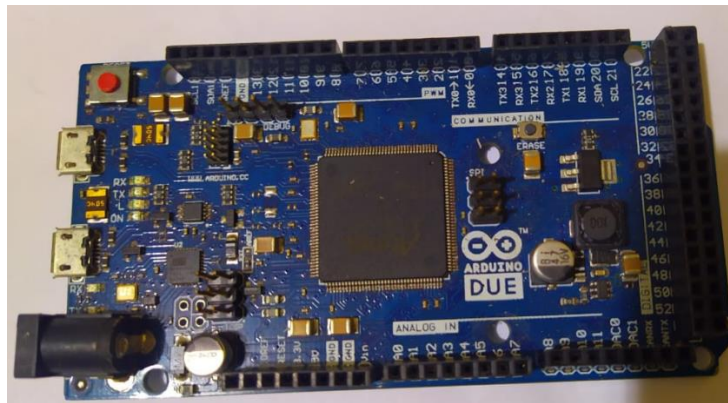
Los cuales se deberán instalar de la siguiente forma.

Figura 45. **Sensor**



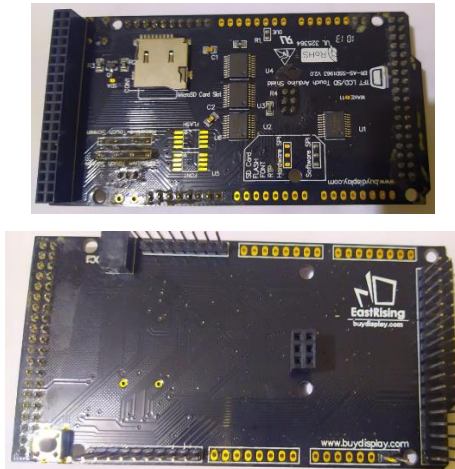
Fuente: elaboración propia.

Figura 46. **Arduino Due**



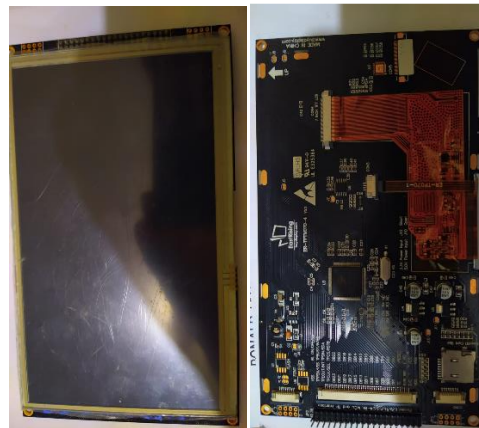
Fuente: elaboración propia.

Figura 47. **Módulo acoplador de pantalla**



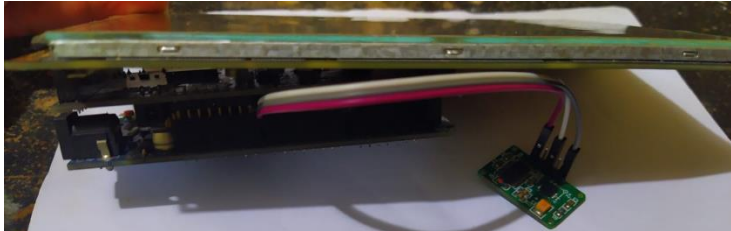
Fuente: elaboración propia.

Figura 48. **Pantalla TFT**



Fuente: elaboración propia.

Figura 49. **Circuito armado**



Fuente: elaboración propia.

9.2.1. **Acoplamiento del sensor a un sistema mecánico**

El dispositivo que mide las vibraciones, se coloca en el sistema sin utilizar el resto del módulo debido a que tiene que aportar la menor cantidad de masa al sistema estudiado.

Una vez conectado el dispositivo a una fuente de voltaje se debe presionar el botón RESET dentro del módulo Due.

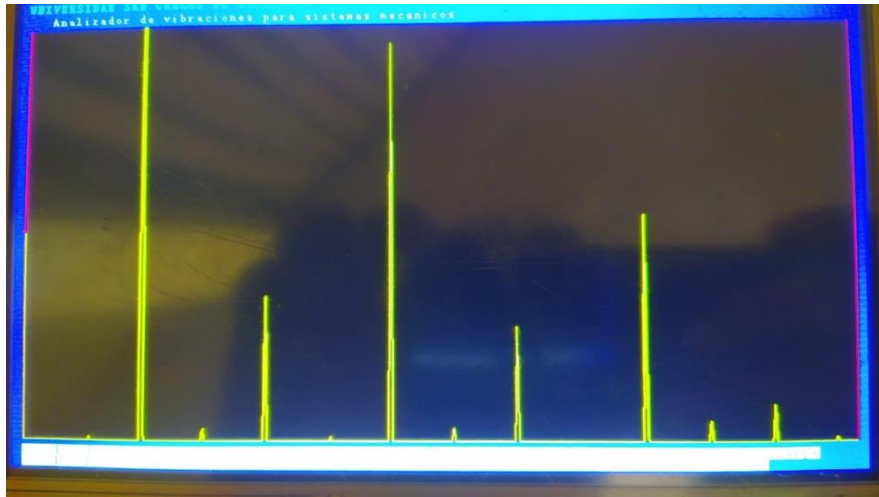
Figura 50. **Arduino Due**



Fuente: elaboración propia.

Para que el sistema inicie y empiece a mostrar las mediciones de ruido del ambiente en escala de revoluciones por minuto.

Figura 51. **Analizador de vibraciones funcionando**



Fuente: elaboración propia.

CONCLUSIONES

Para los sistemas mecánicos es posible proporcionar un estudio de vibraciones utilizando equipos de bajo costo concerniente a como se desarrolló el circuito en este documento.

1. Se determinó que el mejor sensor para la medición de la señal de vibración es el acelerómetro debido a su fácil calibración y gran precisión.
2. Se construyó un circuito capaz de almacenar y procesar las mediciones del sensor por medio de la transformada de Fourier utilizando un microcontrolador de bajo costo y alta velocidad.
3. Por medio de presentar los datos en una pantalla, que a su vez dota al medidor de portabilidad y es amigable con el usuario, puede mostrarse una huella característica de las vibraciones para su posterior comparación.

RECOMENDACIONES

1. Observar que en el código se elijan pines libres para colocar los voltajes de polarización del sensor para evitar el uso de una fuente externa.
2. Colocar el sensor por medio de un cable de unos 50 cm de largo y que este sea afectado por la tensión de los cables lo menos posible eligiendo el eje de medición perpendicular a la dirección de los cables.
3. El Arduino DUE es un circuito muy sensible por lo cual se recomienda no superar 3,3 voltios en sus pines ya que esto podría dañar el dispositivo de forma irreparable.

BIBLIOGRAFÍA

1. Amazon. *Yongse TFT LCD de 4,3 pulgadas de pantalla táctil del módulo Mega2560 DUE Disponible para Arduino*. [en línea]. <<https://www.amazon.es/Yongse-pulgadas-pantalla-Mega2560-Disponible/dp/B01F7TTV9O>>. [Consulta: 19 de octubre de 2018].
2. ANALOG DEVICE. *Hoja de datos del 3-Axis accelerometer module - ADXL330*. [en línea]. <https://www.sparkfun.com/datasheets/Components/ADXL330_0.pdf>. [Consulta: 11 de octubre de 2018].
3. _____. *Small, low power, 3-Axis ± 3 g accelerometer ADXL335*. [en línea]. <<https://www.analog.com/media/en/technical-documentation/data-sheets/ADXL335.pdf>>. [Consulta: 11 de octubre de 2018].
4. ARDUINO FORUM. *UTFT Supported display modules & controllers*. [en línea]. <<https://forum.arduino.cc/index.php?action=dlattach;topic=629773.0;attach=321372>>. [Consulta: 9 de diciembre de 2018].
5. ARDUINO. *Arduino Due*. [en línea]. <https://content.arduino.cc/assets/DUE_V02g_sch.pdf>. [Consulta: 8 de diciembre de 2018].
6. _____. *Arduino Due*. [en línea]. <<https://store.arduino.cc/due>>. [Consulta: 8 de diciembre de 2018].

7. _____. *Diagrama del circuito esquemático de la placa Arduino DUE*. [en línea]. <<https://www.youtube.com/watch?v=UdMRgTF6IxY>>. [Consulta: 18 de octubre de 2018].
8. ATMEL. *8-bit Microcontroller with 8/16/32K Bytes of ISP Flash and USB Controller*. [en línea]. <<https://cdn.sos.sk/productdata/18/15/eb99303a/atmega8u2-mur.pdf>>. [Consulta: 8 de diciembre de 2018].
9. _____. *SAM3X / SAM3A Series*. [en línea]. <https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-11057-32-bit-Cortex-M3-Microcontroller-SAM3X-SAM3A_Datasheet.pdf>. [Consulta: 8 de diciembre de 2018].
10. BHAGWANDAS PANNALAL, Lathi y DING, Zhi *Modern digital and analog communication systems*. 4a ed. New York, United States: Prentice Hall, 2010. 1004 p.
11. BENGOCHEA LOVO, Álvaro. *5º Curso: tratamiento digital de señal*. [en línea]. <https://www.academia.edu/14908832/5o_Curso-Tratamiento_Digital_de_Se%C3%B1al>. [Consulta: 11 de octubre de 2018].
12. BOXTEC. *4.3" TFT LCD Screen Module (ITDB02-4.3)*. [en línea]. <<https://shop.boxtec.ch/tft-lcd-screen-module-itdb02-p-40969.html?language=en>>. [Consulta: 8 de diciembre de 2018].

13. CRESPO CEPEDA, José Luis. *Cómo entender cualquier vibración*. [en línea]. <<https://www.youtube.com/watch?v=UdMRgTF6lxY>>. [Consulta: 9 de diciembre de 2018].
14. EDN NETWORKS. *Home, analog design center, how to article the embedded self-test feature in MEMS inertial sensors*. [en línea]. <<https://www.edn.com/design/analog/4390843/The-embedded-self-test-feature-in-MEMS-inertial-sensors>>. [Consulta: 8 de diciembre de 2018].
15. ELECTRONIKA I TY. *ITDB02 Arduino MEGA Shield v2.1*. [en línea]. <https://elty.pl/en_US/p/ITDB02-Arduino-MEGA-Shield-v2.1/279>. [Consulta: 11 de diciembre de 2018].
16. ELRINCONDELACIENCIAVKENT. *Pantallas de cristal líquido (LCD)*. [en línea]. <<https://sites.google.com/site/elrincondelacienciavkent/home/numeros-anteriores/numero-47/pantalla-cristal-liquido>>. [Consulta: 8 de diciembre de 2018].
17. Escuela Universitaria de Ingeniería Técnica Industrial Bilbao. *La transformada rápida de Fourier*. [en línea]. <<http://www.ehu.eus/Procesadodesenales/tema7/ty3.html>>. [Consulta: 11 de diciembre de 2018].
18. HERNÁNDEZ SAMPIERI, Roberto; FERNÁNDEZ COLLADO, Carlos y LUCIO, Pilar Baptista. *Metodología de la investigación*. Atlacomulco, México: McGraw-Hill, 1999. 497 p.

19. HSU, Hwei P. *Análisis de Fourier*. United States: Prentice Hall, 1999. 318 p.
20. IMV CORPORATION. *Vibration technical guide*. [en línea]. <<http://www.mantetronic.com/es/servicios/vista/category/mantenimiento-predictivo-mecanico>>. [Consulta: 11 de diciembre de 2018].
21. Inteligencia artificial y robótica. *MOT-2265 motor vibrador*. [en línea]. <<http://inteligenciaartificialyrobotica.com/esp/item/340/mot-2265-motor-vibrador>>. [Consulta: 8 de diciembre de 2018].
22. Itead Blog. *ITDB02 Arduino MEGA Shield*. [en línea]. <https://www.itead.cc/wiki/ITDB02_Arduino_MEGA_Shield>. [Consulta: 8 de diciembre de 2018].
23. _____. *Lcd 16bit mega shield 2.0.sch*. [en línea]. <ftp://imall.iteadstudio.com/IM120417024_ITDB02_Arduino_MEGA_Shield/SCH_IM120417024_ITDB02ArduinoMEGAShield.pdf>. [Consulta: 8 de diciembre de 2018].
24. Itead Studio. *Hoja de datos de la pantalla TFT de 4.3"*. [en línea]. <ftp://imall.iteadstudio.com/IM120419007_ITDB02_4.3/DS_IM120419007_ITDB02_4.3.pdf>. [Consulta: 9 de diciembre de 2018].
25. _____. *ITDB02 4.3" TFT LCD display module shield for Arduino*. [en línea]. <<https://www.itead.cc/itdb02-4-3.html>>. [Consulta: 9 de diciembre de 2018].

26. _____. *ITDB02 Arduino mega shield starter kit for 3.3V-5V mainboard compatible with Arduino mega pins*. [en línea]. <<https://www.itead.cc/itdb02-4-3.html>>. [Consulta: 9 de diciembre de 2018].
27. MANTETRONIC. *Mantenimiento predictivo mecanico – Mantetronic S.A.S*. [en línea]. <<http://www.mantetronic.com/es/servicios/vista/category/mantenimiento-predictivo-mecanico>>. [Consulta: 11 de diciembre de 2018].
28. MICROCHIP. *PIC16F882/883/884/886/887 Data Sheet*. [en línea]. <<http://ww1.microchip.com/downloads/en/devicedoc/41291d.pdf>>. [Consulta: 11 de diciembre de 2018].
29. MOTORTEC. *Lo que hay que saber para elegir los motores para un cuadracóptero*. [en línea]. <<https://www.prometec.net/elegir-motores-cuadrapotero/>>. [Consulta: 11 de diciembre de 2018].
30. MPS the future of analog IC technology. *MPM3610*. [en línea]. <https://www.monolithicpower.com/en/documentview/productdocument/index/version/2/document_type/Datasheet/lang/en/sku/MPM3610GQV/document_id/2090/>. [Consulta: 19 de octubre de 2018].
31. NEXPERIA. *74HC365; 74HCT365*. [en línea]. <https://assets.nexperia.com/documents/data-sheet/74HC_HCT365.pdf>. [Consulta: 19 de octubre de 2018].

32. Painless360. *RC basics - understanding electronic speed controllers (ESC)*. [en línea]. <<https://www.youtube.com/watch?v=OZNxbxL7cdc/>>. [Consulta: 19 de octubre de 2018].
33. Rassoul Amirfattahi, R. A. *Calculation of computational complexity for Radix-2p fast fourier transform algorithms for medical signals*. [en línea]. <<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3967424/>>. [Consulta: 19 de octubre de 2018].
34. SCHWENKE, Thomas. *¿Cómo funciona una pantalla LCD?* [en línea]. <<https://www.youtube.com/watch?v=zL2TX-xknjI>>. [Consulta: 19 de octubre de 2018].
35. Solomon. *Systech semiconductor technical data. 1215KB Embedded Display SRAM LCD Display Controller*. [en línea]. <<https://cdn.displaytech-us.com/sites/default/files/driver-ic-data-sheet/Solomon-Systech-SSD1963.pdf>>. [Consulta: 19 de octubre de 2018].

APÉNDICE

Apéndice 1. Código dentro del microcontrolador

```
#include <SplitRadixRealP.h>
#define SMP_RATE 700L //700
#define CLK_MAIN 84000000UL
#define TMR_CNTR CLK_MAIN / ( 3*SMP_RATE)//3
//#define FFT_SIZE 2048
#define MIRROR FFT_SIZE / 2
#define INP_BUFF FFT_SIZE

volatile uint16_t sptr = 0 ;
volatile int16_t flag = 0 ;
uint16_t inp[2][INP_BUFF] = { 0}; // DMA BUFFER
int f_r[FFT_SIZE] = { 0};
int out1[MIRROR] = { 0}; // Magnitudes
int out2[MIRROR] = { 0}; // Magnitudes
const int dc_offset = 1023; //2047;
unsigned long time_start;
unsigned int time_hamng, time_revbn, time_radix, time_gainr, time_sqrtl,
time_sqrtl2;
SplitRadixRealP radix;
// TFT -----
#include <UTFT.h> // UTFT rutina
#include <UTouch.h> // UTouch rutina

extern uint8_t SmallFont[]; // fondo pequeño
extern uint8_t BigFont[]; // fondo grande
extern uint8_t SevenSegNumFont[]; // Variable
auxiliarSevenSegNumFont

UTFT myGLCD(SSD1963_800480,38,39,40,41);

int mode = 2;
int mx;
//pantalla
#define xz 800
```

Continuación del apéndice 1.

```
#define yz 480

void setup(){
  delay(10);
  Serial.begin (115200) ;
  adc_setup ();
  tmr_setup ();
  //pinMode( A0, INPUT);           // entrada
  // TFT setup -----
  myGLCD.InitLCD(LANDSCAPE);      // inicializamos pantalla con
  imagenes acostadas
  // -----
  pinMode(8, OUTPUT); //Luz de fondo es necesario encenderla o la
  pantaria no proyectara imagen
  digitalWrite(8, HIGH); //encender luz de fondo
  // -----

//voltages del sensor
  pinMode(18, OUTPUT);
  pinMode(19, OUTPUT);
  digitalWrite(18, LOW); //GND
  digitalWrite(19, HIGH); // V+

  myGLCD.lcdOn();
  delay(10);
  myGLCD.setContrast(64);
  //demo();
  myGLCD.setFont(SmallFont);

  myGLCD.clrScr();

  myGLCD.setColor(0,0,255); // color azul
  myGLCD.fillRect(0, 0, xz, yz); // rectangulo azul fondo de pantalla

  myGLCD.setColor(255,0,0); // color Rojo
  myGLCD.fillRect(10, 30, xz-10, yz-30); // tamaño del rectangulo limitador de
  la pantalla

  myGLCD.setColor(0, 255, 0); // color verde
  myGLCD.setBackColor(0,0,255); // color de fondo de las letras azul
```

Continuación del apéndice 1.

```
myGLCD.print(" UNIVERSIDAD SAN CARLOS DE GUATEMALA ", LEFT,
1); // titulo
```

```
myGLCD.setColor(255, 255, 0); // color amarillo
myGLCD.setBackgroundColor(0,0,255);// color de fondo de las letras azul
myGLCD.print("  Analizador de vibraciones para sistemas mecanicos ",
LEFT, 15); // titulo
```

```
myGLCD.setColor(0,0,0); // -- color fondo del osciloscopio
myGLCD.setBackgroundColor(255,255,255);          // moji background color
// --color de letras
myGLCD.setFont(SmallFont);
myGLCD.print(" | | | | | | | | | | | | | | | | | | | | |
21krpm", 10, 457);
myGLCD.print("  1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17
18 19 20 21 ", 10, 467);
```

```
}
//-----
inline int mult_shft12( int a, int b)
{
  return ((a * b) >> 12);//LE AUTA LOS 12 BITS MENOS SIGNIFICATIVOS
Y DEVUELVE SOLO LOS MAS ALTOS
}
//-----
```

```
void loop(){

// while (myTouch.dataAvailable() == true) { // interupt Touch
// if ( mode == 1 ){ mx = 2;}
// if ( mode == 2 ){ mx = 1;}
  mode = 0;
  //}
  if ( flag ){
// FFT -----
  uint16_t indx_a = flag -1;
  uint16_t indx_b = 0;
  time_start = micros(); // tiempo en microsegundos en el cual el arduino
empezo a funcionar

  for ( uint16_t i = 0, k = (NWAVE / FFT_SIZE); i < FFT_SIZE; i++ )
  {
```

Continuación del apéndice 1.

```
uint16_t windw = Hamming[i * k];
f_r[i] = mult_shft12((inp[indx_a][indx_b++] - dc_offset), windw);
}

time_hamng = micros() - time_start;
time_start = micros();
radix.rev_bin( f_r, FFT_SIZE);
time_revbn = micros() - time_start;
time_start = micros();
radix.fft_split_radix_real( f_r, LOG2_FFT);
time_radix = micros() - time_start;
time_start = micros();
radix.gain_Reset( f_r, LOG2_FFT -1);
time_gainr = micros() - time_start;
time_start = micros();
radix.get_Magnit1( f_r, out1);
time_sqrtl = micros() - time_start;
time_start = micros();
radix.get_Magnit2( f_r, out2);
time_sqrtl2 = micros() - time_start;
// -----
switch(mode){

case 0:
    prnt_out4( f_r);          // tipo de linea
    break;

case 1:
    prnt_out5( f_r, MIRROR);  // tipo de linea
    break;
case 2:
    prnt_out6( f_r, MIRROR);  // tipo de linea
    break;
}
flag = 0;
}
}

void prnt_out4( int *yd)
{          // fft data,count
```

Continuación del apéndice 1.

```
int xr=10;
int yr=451;                // cambio linea original int xr=18; int yr=250;

int x1;
int x2;
int y1;
int y2;

yd[0]=0; yd[1]=0;

for ( uint32_t x = 2; x < 780; x++){          // 1 - 450

    yd[x] = abs( yd[x] );                    //

    if ( yd[x] > 270 )
    {
        yd[x] = 270;
    } // limitador de altura

    x2 = x + xr;                            // x2
    x1 = x2 - 1;                            // x1

    y1 = yr - 4*yd[x-1];                    // y1
    y2 = yr - 4*yd[x];                      // y2

    if ( y1 < 30 )
    {
        y1 = 30;
    } // LIMITADOR BAJO

    if ( y2 < 30 )
    {
        y2 = 30; yd[x] = 270;
    } // ALTO

    myGLCD.setColor(0,0,0);                // pantalla en negro
    myGLCD.drawLine(x2, yr, x2, 30);       // dibujar rectangulo
    myGLCD.setColor(255,255,0);            // azul
    myGLCD.drawLine(x1, y1, x2, y2);       //
}
}
```

Continuación del apéndice 1.

```
//-----  
  
void prnt_out5( int *yd, int ct){          // fft conteo de datos  
    int xr=10;  
    int yr=451;                          // cambio linea original int xr=18; int yr=250;  
    int x1;  
    int x2;  
    int y1;  
    int y2;  
    yd[0]=0;  
    yd[1]=0;  
  
    for ( uint32_t x = 2; x < 780; x++)  
    {          // 1 -  
        yd[x] = abs( yd[x] );              //  
        if ( yd[x] > 270 )  
        {  
            yd[x] = 270;  
        }          //limitador de pantalla //cambio estaba comentada if ( yd[x] >  
270 ){ yd[x] = 270;}  
  
        x2 = x + xr;                       // x2 : 480/(1024/3)  
        x1 = x2 - 1;                       // x1  
        y1 = yr - 4*yd[x-1];              // y1  
        y2 = yr - 4*yd[x];                // y2  
  
        if ( y1 < 30 )  
        {  
            y1 = 30;  
        }          // Limitter  
  
        if ( y2 < 30 )  
        {  
            y2 = 30;  
            yd[x] = 270;  
        } // disp,data Limitter  
        myGLCD.setColor(0,0,0);          // pantalla negra  
        myGLCD.drawLine(x2, yr, x2, 30); // NEGRO  
        myGLCD.setColor(255,255,0);      // LINEA AMARILLA  
        myGLCD.drawLine(x1, y1, x2, y2); //  
    }  
}
```


Continuación del apéndice 1.

```
}  
  
//-----  
  
void prnt_out6( int *yd, int ct){          // fft data,count  
  
    int x0=10;// aqui empiezan a generarse las barras en x  
  
    int w = 10;                // w : ancho de las barras   ---x0=20  
  
    int x1;//Eta es la pocion de la barra anterior  
    int x2;//esta es la pocion del final de la barra  
  
    int y1=450; //altura maxima de la pantalla de medicion  
    int y2;      // y1   --cambio 300  
  
    int c1=250; int c2=190; int c3=130; int c4=70; //  
    yd[0]=0;  
    yd[1]=0;                // first clear  
  
    for ( int x = 2; x < 780; x = x + w )//este es con respecto al ancho de la  
    pantalla  
    {        // 1 - 780  
        y2 = 0;                // y2 reset  
        for ( int k = 0; k < w; k++ ){        // 0 to 9  
            y2 = y2 + abs(yd[ x + k]);        // x + 0 to x + 9  
        }                //  
  
        y2 = y1 - 4*(y2 / w );                // y2 *4  
        x1 = x + x0;                // x1  
        x2 = x1 + w - 2;                // x2  
        if ( y2 < 30 ){ y2 = 30;}        // limitador --cambio 30 30  
  
        myGLCD.setColor(0,0,0);        // pantalla negra  
        myGLCD.fillRect(x1, y2, x2+2, 30);        // pantalla negra  
  
        if ( y2 < c4 ){                // < 70  
            myGLCD.setColor(0,255,255);  
            myGLCD.fillRect(x1, c4, x2, y2);  
            myGLCD.setColor(0,255,0);  
            myGLCD.fillRect(x1, c3, x2, c4);
```

Continuación del apéndice 1.

```
    myGLCD.setColor(0,0,255);
    myGLCD.fillRect(x1, c2, x2, c3);
    myGLCD.setColor(255,0,255);
    myGLCD.fillRect(x1, c1, x2, c2);
    myGLCD.setColor(255,0,0);
    myGLCD.fillRect(x1, y1, x2, c1);
}
if ( y2 > c4 && y2 < c3 ){           // y2 = 70 - 130
    myGLCD.setColor(0,255,0);
    myGLCD.fillRect(x1, c3, x2, y2);
    myGLCD.setColor(0,0,255);
    myGLCD.fillRect(x1, c2, x2, c3);
    myGLCD.setColor(255,0,255);
    myGLCD.fillRect(x1, c1, x2, c2);
    myGLCD.setColor(255,0,0);
    myGLCD.fillRect(x1, y1, x2, c1);
}
if ( y2 > c3 && y2 < c2 ){           // y2 = 130 - 190
    myGLCD.setColor(0,0,255);
    myGLCD.fillRect(x1, c2, x2, y2);
    myGLCD.setColor(255,0,255);
    myGLCD.fillRect(x1, c1, x2, c2);
    myGLCD.setColor(255,0,0);
    myGLCD.fillRect(x1, y1, x2, c1);
}
if ( y2 > c2 && y2 < c1 ){           // y2 = 190 - 250
    myGLCD.setColor(255,0,255);
    myGLCD.fillRect(x1, c1, x2, y2);
    myGLCD.setColor(255,0,0);
    myGLCD.fillRect(x1, y1, x2, c1);
}
if ( y2 > c1 ){                       // > 240
    //myGLCD.setColor(255,255,0);
    //myGLCD.fillRect(x1, y1, x2, y2);
}
}
}
//-----

void pio_TIOA0 (){
    PIOB->PIO_PDR = PIO_PB25B_TIOA0 ; // desactivar PIO control
```

Continuación del apéndice 1.

```

    PIOB->PIO_IDR = PIO_PB25B_TIOA0 ; // desactivar PIO interrupts
    PIOB->PIO_ABSR |= PIO_PB25B_TIOA0 ; // cambiar al periferico B
}
//-----
void tmr_setup (){
    pmc_enable_periph_clk(TC_INTERFACE_ID + 0 *3 + 0); // reloj TC0 canal
    0
    TcChannel * t = &(TC0->TC_CHANNEL)[0] ; // puntero para TC0
    registros para el canal 0
    t->TC_CCR = TC_CCR_CLKDIS ; // Desactivar los registros
    mientras se configuran
    t->TC_IDR = 0xFFFFFFFF ; // desactivar las interrupciones
    t->TC_SR ; // read int status reg to clear pending
    t->TC_CMR = TC_CMR_TCCLKS_TIMER_CLOCK1 | // use TCLK1
    (PRESCALADOR DIVIDIDO 2, = 42MHz)
    TC_CMR_WAVE | // MODO DE FORMA DE ONDA
    TC_CMR_WAVSEL_UP_RC | // CONTEO HACIA
    ARRIBA PWM USANDO RC COMO LIMITE
    TC_CMR_EEVT_XC0 | // ESTABLECER EVENTOS
    EXTERNOS DESDE XC0 (ESTE CONFIGURA TIOB COMO SALIDA)
    TC_CMR_ACPA_CLEAR | TC_CMR_ACPC_CLEAR |
    TC_CMR_BCPB_CLEAR | TC_CMR_BCPC_CLEAR ;
    t->TC_RC = TMR_CNTR; // el contador se reinicia en RC, para
    colocar periodos de 42MHz de reloj
    t->TC_RA = TMR_CNTR /2; // otra parte de onda cuadrada
    t->TC_CMR = (t->TC_CMR & 0xFFFF0FFF) | TC_CMR_ACPA_CLEAR |
    TC_CMR_ACPC_SET ; //limpiar los comparadores
    t->TC_CCR = TC_CCR_CLKEN | TC_CCR_SWTRG ; // reactivar los
    controladores de reloj.
}
//-----
void adc_setup (){
    pmc_enable_periph_clk(ID_ADC);
    adc_init(ADC, SystemCoreClock, ADC_FREQ_MAX,
    ADC_STARTUP_FAST);
    NVIC_EnableIRQ (ADC_IRQn); // activar conversor analogico a
    digital
    adc_disable_all_channel(ADC);
    adc_enable_interrupt(ADC, ADC_IER_RXBUFF);
}

```

Continuación del apéndice 1.

```

ADC->ADC_RPR = (uint32_t) inp[0];    // DMA buffer
ADC->ADC_RCR = INP_BUFF;
ADC->ADC_RNPR = (uint32_t) inp[1];    // PROXIMO DMA buffer
ADC->ADC_RNCR = INP_BUFF;
ADC->ADC_PTCR = 1;
adc_set_bias_current(ADC, 0x01);
// adc_enable_tag(ADC);
adc_enable_channel(ADC, ADC_CHANNEL_7); // EL PIN DE ENTRADA
DE SEÑAL ES EL ANALOGICO 0 AN0
adc_configure_trigger(ADC, ADC_TRIG_TIO_CH_0, 0);
adc_start(ADC);
}
//-----
void ADC_Handler (void){
    if((adc_get_status(ADC) & ADC_ISR_RXBUFF) ==
ADC_ISR_RXBUFF){
        flag = ++sptr;
        sptr &= 0x01;
        ADC->ADC_RNPR = (uint32_t) inp[sptr];
        ADC->ADC_RNCR = INP_BUFF;
    }
}
}

```

Fuente: elaboración propia.