

ESPECIALIDAD DE TECNOLOGÍA EN MECATRÓNICA

Trabajo final de especialidad

**“DESARROLLO DE UN ALGORITMO EN LABVIEW
QUE REALICE LA COMPARATIVA ENTRE LOS
ESPECTROS DE FRECUENCIA EN MÁQUINAS
ROTATIVAS”**

PROYECTO TERMIMAL

PRESENTA:

Ing. Cristian Hamilton Sánchez Saquín

TUTOR ACADÉMICO:

Dr. Leonardo Barriga Rodríguez

ÁREA:

Energía

TIPO DE INVESTIGACIÓN:

Desarrollo tecnológico

Santiago de Querétaro, agosto de 2017



CENTRO DE INGENIERÍA Y DESARROLLO INDUSTRIAL

CENTRO DE INFORMACIÓN Y DOCUMENTACIÓN TECNOLÓGICA

AUTORIZACIÓN
PUBLICACIÓN EN FORMATO ELECTRÓNICO DE TESIS

FECHA: 09/08/2017

EL QUE SUSCRIBE

ALUMNO (A) CRISTIAN HAMILTON SÁNCHEZ SAQUÍN.

CURP SASC941008HCSNQR01 **CVU** 747582

ORCID

CORREO ELECTRÓNICO (OPCIONAL): SAQUIN26@GMAIL.COM

EGRESADO (A) DE: TECNÓLOGO EN MECATRÓNICA-

AUTOR DE LA TESIS TÍTULO: CRISTIAN HAMILTON SÁNCHEZ SAQUÍN

POR MEDIO DEL PRESENTE DOCUMENTO AUTORIZO¹ EN FORMA GRATUITA Y PERMANENTE A QUE LA TESIS ARRIBA CITADA SEA DIVULGADA Y REPRODUCIDA PARA PUBLICARLA MEDIANTE ALMACENAMIENTO ELECTRÓNICO QUE PERMITA EL ACCESO AL PÚBLICO A LEERLA Y CONOCERLA VISUALMENTE, ASÍ COMO A COMUNICARLA PÚBLICAMENTE EN PÁGINA WEB.

LA ÚNICA CONTRAPRESTACIÓN QUE CONDICIONA LA PRESENTE AUTORIZACIÓN ES LA DEL RECONOCIMIENTO DEL NOMBRE DEL AUTOR EN LA PUBLICACIÓN QUE SE HAGA DE LA MISMA.

ATENTAMENTE

NOMBRE Y FIRMA DEL TESISISTA

¹ **LEY FEDERAL DE DERECHOS DE AUTOR**

PARA OBTENER TU **ORCID** REGÍSTRATE EN: [HTTPS://ORCID.ORG/REGISTER](https://orcid.org/register)

AV. PLAYA PIE DE LA CUESTA NO. 702 DESARROLLO SAN PABLO. C. P. 76125

SANTIAGO DE QUERÉTARO, QRO., MÉXICO MÉXICO.

TEL. 01 (442) 211 98 40

E-MAIL: JVITE@CIDESI.EDU.MX

ÍNDICE

AGRADECIMIENTOS	5
RECONOCIMIENTO	5
RESUMEN	6
ÍNDICE DE TABLAS Y FIGURAS	7
LISTA DE SÍMBOLOS, ABREVIATURAS, NOMENCLATURAS, GLOSARIO DE TÉRMINOS	9
INTRODUCCIÓN	10
PLANTEAMIENTO DEL PROBLEMA	12
JUSTIFICACIÓN	13
OBJETIVOS	13
Objetivo general	13
Objetivos específicos	13
MARCO TEÓRICO	14
Análisis de vibraciones	14
Sistema de adquisición de datos	16
Transductor	17
Tarjeta de adquisición de datos	17
LabVIEW	17
ANTECEDENTES	22
HIPÓTESIS	23
METODOLOGÍA DE LA SOLUCIÓN	23
ALCANCE	24
RESULTADOS ESPERADOS	24
DESARROLLO	25
Panel frontal	25
Diagrama de bloques	33
Arquitectura de programación	33
Productor	33
Consumidor	50
ENTREGABLES	58
FACTIBILIDAD DEL PROYECTO	58



RESULTADOS	59
CONCLUSIÓN	62
REFERENCIAS	63

AGRADECIMIENTOS

Esta Tesis la dedico agradecido en primera instancia a mis padres Rolfi y Alma Delia porque son las personas a quien quiero, respeto y admiro, por haberme dado la mejor educación y orientación durante mi vida y por no dudar en ningún momento en mi capacidad.

A mi madre y abuela Lichita por siempre apoyarme, orar por mí y sobre todo derrochar mucho amor y optimismo para salir adelante. Sin duda una mujer guerrera a la cual admiro, respeto y amo mucho y que siempre va estar presente conmigo en las buenas y en las malas.

A mis hermanos Leslie y Rolfi por su apoyo y por creer siempre en mí, para que este trabajo sirva como estímulo de superación, comprobando una vez más que podemos llegar tan lejos como las metas que nos propongamos.

RECONOCIMIENTO

Al Dr. Leonardo Barriga le doy las gracias, por haberme permitido desarrollar este proyecto bajo su tutoría, por compartir sus conocimientos, por brindarme su confianza para culminar otro paso más a lo largo de mi vida profesional.

RESUMEN

Se acondicionará un sistema de adquisición de datos a un shaker (agitador) que cumple la función de una máquina vibrando. El sistema de adquisición de datos consta de un acelerómetro triaxial 4506 Brüel and Kjaer, tarjeta de adquisición de datos Hi-Speed USB Carrier NI USB-9162 de National Instruments, un generador de señales Tektronix AFG 3151C, Shaker VR520 Vibration Research Corporation, software Excel y LabVIEW.

El sistema de adquisición de datos hará posible que obtengamos el espectro de frecuencias mediante la Transformada de Fourier, con este espectro se realizará el análisis de vibraciones y así, se determinará el tipo de falla y la severidad que se tiene en el shaker.

Para determinar la falla y severidad de la señal, se realizará la comparativa entre la señal de entrada y una norma para severidades de vibración ISO2372. Es importante recalcar que la comparativa va a ser de la vibración en general de la máquina.

ÍNDICE DE TABLAS Y FIGURAS

Ilustración 1. Gráfica de velocidad (Torres et al., 2015).....	15
Ilustración 2. Sistema de adquisición de datos (Spending, 2008).	16
Ilustración 3. Diagrama de flujo del proceso.....	23
Ilustración 4 PANEL FRONTAL DEL PROGRAMA	29
Ilustración 5 PESTAÑA DE ACELERACIÓN	30
Ilustración 6 PESTAÑA DE VELOCIDAD	31
Ilustración 7 PESTAÑA DE DESPLAZAMIENTO.....	32
Ilustración 8 DATOS GUARDADOS EN eXCEL	32
Ilustración 9 aRQUITECTURA PRODUCTOR- CONSUMIDOR.....	33
Ilustración 10 CLUSTER DE INICIO.....	34
Ilustración 11 CLUSTER DE ENVÍO DE DATOS	35
Ilustración 12 SECCIÓN LÓGICA QUE GOBIERNA EL CASE DE ENVIO DE DATOS	36
Ilustración 13 PANEL FRONTAL DE SUBVI PARA LÓGICA DE GUARDADO	37
Ilustración 14 DIAGRAMA DE BLOQUES DE SUBVI PARA LÓGICA DE GUARDADO	37
Ilustración 15 SECCIÓN DE PROGRAMA DONDE SE ENVÍA EL DATO Y SE CONBINAN ERRORES.	38
Ilustración 16 CICLO FOR PARA EL INDEXADO DE EJE X.....	39
Ilustración 17 SECCIÓN DE PROGRAMA DONDE SE USA EL SUBVI	39
Ilustración 18 PANEL FRONTAL DE SUBVI DE ACELERACIÓN	40
Ilustración 19 DIAGRAMA DE BLOQUES DE SUBVI DE ACELERACIÓN	40
Ilustración 20 PANEL FRONTAL DE SUBVI DE GUARDADO DE PROGRAMA	41
Ilustración 21 DIAGRAMA DE BLOQUES DE PROGRAMA DE GUARDADO.....	42
Ilustración 22 PANEL FRONTAL DE SUBVI QUE HACE LA CUENTA DE BOTONES	42
Ilustración 23 DIAGRAMA DE BLOQUES DE SUBVI QUE HACE LA CUENTA DE BOTONES.....	43
Ilustración 24 SECCIÓN DEL PROGRAMA DONDE SE EJECUTA EL SUBVI DE VELOCIDAD.....	43
Ilustración 25 PANEL FRONTAL DE SUBVI DE VELOCIDAD.....	44
Ilustración 26 DIAGRAMA DE BLOQUES DE SUBVI DE VELOCIDAD	44
Ilustración 27 SECCIÓN DE PROGRAMA DONDE SE EJECUTA SUBVI DE DESPLAZAMIENTO	45
Ilustración 28 PANEL FRONTAL DE SUBVI DE DESPLAZAMIENTO	46
Ilustración 29 DIAGRAMA DE BLOQUES DE SUBVI DE DESPLAZAMIENTO.....	46
Ilustración 30 PRODUCTOR.....	47
Ilustración 31 PRODUCTOR.....	47

Ilustración 32 PRODUCTOR.....	48
Ilustración 33 PRODUCTOR.....	48
Ilustración 34 PRODUCTOR.....	49
Ilustración 35 PRODUCTOR.....	49
Ilustración 36 CONSUMIDOR, DEMULTIPLEXADO DE SEÑAL	50
Ilustración 37 CASE DE NO ERROR	51
Ilustración 38 PROGRAMACIÓN DE MÁQUINA DE ESTADOS.....	52
Ilustración 39 PROGRAMACIÓN DE GUARDADO DE TIEMPO EN VELOCIDAD ...	52
Ilustración 40 PROGRAMACIÓN DE GUARDADO DE INSTANTE EN VELOCIDAD	53
Ilustración 41 PROGRAMACIÓN DE GUARDADO DE TIEMPO EN DESPLAZAMIENTO	53
Ilustración 42 PROGRAMACIÓN DE GUARDADO DE INSTANTE EN DESPLAZAMIENTO	54
Ilustración 43 PANEL FRONTAL DE SUBVI DE CONFIGURACIÓN DE EXCEL....	54
Ilustración 44 DIAGRAMA DE BLOQUES DE SUBVI DE CONFIGURACIÓN DE EXCEL	55
Ilustración 45 FIN DE CONFIGURACION DE EXCEL.....	55
Ilustración 46 CONDICIONES DE ARRANQUE	56
Ilustración 47 CONDICIÓN DE ARRANQUE, EXPLICACIÓN DEL PROGRAMA.....	57
Ilustración 48 CONDICIÓN DE ARRANQUE, LLENADO DE DATOS.....	57

LISTA DE SÍMBOLOS, ABREVIATURAS, NOMENCLATURAS, GLOSARIO DE TÉRMINOS

Vibración: Movimiento de vaivén de un cuerpo que vibra o de una parte de él.

Espectro de frecuencia: Es una medida de la distribución de amplitudes de cada frecuencia.

Análisis espectral: El análisis espectral de una señal digital tiene por objeto la descomposición de dicha señal en sus diversas componentes dentro del dominio de la frecuencia.

Transformada de Fourier: Es la herramienta matemática más adecuada para pasar del dominio del tiempo al dominio de la frecuencia, su símbolo es FFT.

Decibel: El decibel o decibelio es la unidad de medida de la intensidad (dB).

Shaker: Agitador, herramienta para simular una vibración.

Algoritmo: Serie de pasos para un fin común.

Subvi: Subrutina o subprograma de LabVIEW.

Generador de señales: Es un instrumento que proporciona señales eléctricas, típicamente, genera señales de forma cuadrada, triangular y la sinusoidal.

Máquina rotativa: Es toda aquella máquina que tiene elementos rodantes tales como: flechas, baleros, cojinetes, entre otros.

DAQ: Data acquisition, adquisición de datos.

INTRODUCCIÓN

Los algoritmos de vibraciones realizados en LabVIEW han ayudado en hacer mediciones y procesamiento de señales en maquinarias rotativas así como también en prototipos de elementos mecánicos rotatorios tanto en industrias como en laboratorios de pruebas. Conforme va pasando el tiempo se han estado mejorando los algoritmos para que de manera autónoma realicen la evaluación y el reporte del sistema medido.

Las vibraciones mecánicas pueden medirse en todos los ámbitos donde sean producidas siendo relevantes aquellas situaciones donde exista exposición humana (para fines de salud), ya sea por maquinaria (bombas, compresores, máquinas rotativas, etc.) vehículos (tierra, mar y aire), actividades (industriales, ocio, domésticas) y vibraciones estructurales en edificios (Nacional, En, En, Opci, & Digitales, n.d.). El propósito principal de la medición y análisis de las vibraciones mecánicas es su control y prevención.

Para cuantificar, medir y analizar las vibraciones se requiere de medidas experimentales (Lee & Chang, 2017). Para hacer mediciones se usan sensores de vibración los cuales pueden ser de desplazamiento, de velocidad o de aceleración, siendo el acelerómetro el más usado en la medición de vibraciones (Galindo, 2010).

La medición y análisis de vibraciones es muy utilizado en industrias que usan máquinas rotativas como una técnica de diagnóstico de fallas y evaluación de su integridad (Departamento de Ingeniería Mecánica, 2007). En el caso de los equipos rotatorios, la ventaja que presenta el análisis vibratorio es que la evaluación se realiza con la máquina funcionando, evitando con ello la pérdida de producción que genera una detención. Se sabe que cerca de 90 % de los problemas encontrados en equipos rotatorios se pueden diagnosticar con el análisis de vibraciones (Torres, Royo, & Rabanaque, 2015).

Por lo tanto resulta ser muy atractivo el análisis y monitoreo de vibraciones, ya que no se necesita trasladar la máquina a otro sitio ni detener, apagar o desarmarla. Basta con colocar una serie de sensores en los puntos de interés, encender una computadora y correr un software que realice el análisis y nos proporcione con los datos de entrada el estado vibratorio del sistema (Galindo, 2010). Esto hace que sea muy versátil y cómodo para el cliente puesto que no perciben pérdidas monetarias.

Teniendo en cuenta lo antes mencionado, es evidente la importancia de predecir la presencia de fallas y quién se encarga de corregirlas es la persona de mantenimiento (Karnovsky & Lebed, 2016). Por ello el mantenimiento preventivo y correctivo ha hecho que el análisis de vibraciones sea una herramienta fundamental y muy usada en el ámbito industrial (Eléctricos & Vibración, 2009).

Para lograr un análisis de vibraciones se tiene que utilizar el análisis espectral (“Iso 2372 – Vibration Severity,” n.d.) y éste consiste en percibir de la adquisición de la señal un espectro de frecuencia donde la amplitud puede ser desplazamiento, velocidad o aceleración(White, 2010). Este espectro se debe obtener a partir de la Transformada de Fourier (De & Academia, 1830) y es por ello que está en el dominio de la frecuencia.

PLANTEAMIENTO DEL PROBLEMA

Se han realizado programas en LabVIEW que hacen únicamente la medición y procesamiento de señales en máquinas o elementos rotativos. En los programas se representan las gráficas obtenidas por los sensores, estas gráficas las tiene que analizar un experto en vibraciones y él es encargado de dar el reporte de severidades y posibles averías.

Se sabe que en la gran mayoría de las industrias tienen la necesidad de usar máquinas rotativas para los procesos de producción (White, 2010) tales como cortar, doblar, pintar, soldar, trasladar objetos en bandas u ocupar las bandas para conteo, verificación, envasado entre otras cosas. Debido a la importancia de las tareas que se ejecutan con ellas, esto va de la mano con tener siempre en funcionamiento y en buen estado la línea de producción ya que es una fuente de ingresos muy importante (Mosquera, Piedra, & Armas, 2001). Es por ello que saber con antelación una falla o identificar el estatus se torna una función trascendental.

Para solventar la problemática, se toma el análisis de vibración como pivote para hacer el estudio de las máquinas rotativas y conocer el estatus en el cual se encuentra (Wang, Li, Tollner, & Rains, 2012). Para llevar a cabo dicho análisis se toman herramientas matemáticas como Transformada de Fourier para llevar del dominio del tiempo (señal dada por un acelerómetro) al dominio de la frecuencia (De & Academia, 1830) para así tener un mejor panorama gráfico y con ayuda de un algoritmo de programación se interpretan los datos para que sean mostrados al usuario dando con determinación cual es el estado actual de la máquina.

JUSTIFICACIÓN

Este trabajo de investigación, es conveniente debido a la necesidad de tener siempre en buen estado y en funcionamiento la maquinaria necesaria para que un proceso de producción esté siempre activo, y de mantener un buen sistema de mantenimiento preventivo y correctivo gracias a la aplicación de los algoritmos de LabVIEW para hacer un análisis de predicción de fallas mediante vibraciones en máquinas rotativas, ya que con ello se sabrá con anticipación la existencia de averías al igual que se sabrá el estado actual de lo que se está analizando y con esto se reducen futuras pérdidas monetarias (Ocampo, 2013).

OBJETIVOS

OBJETIVO GENERAL

Desarrollar un algoritmo en LabVIEW que compare los espectros de frecuencia obtenidos de un Shaker y los obtenidos de la norma de severidad de vibraciones en máquinas rotativas ISO 2372 con el fin de identificar en qué estado se encuentra y su severidad.

OBJETIVOS ESPECÍFICOS

1. Seleccionar la estructura de programación en base a la velocidad de adquisición y procesamiento de los datos.
2. Calcular los espectros de frecuencia y gráficas de decibeles en frecuencia de la señal de entrada.
3. Comparar los espectros de frecuencia calculados de la señal de entrada y los ISO 2372.
4. Identificar qué tan severa es la falla en los espectros de frecuencia de la señal de entrada y determinar en qué clase está.
5. Programar base de datos para análisis a posteriori.
6. Redactar la Tesis.

MARCO TEÓRICO

ANÁLISIS DE VIBRACIONES

El análisis de vibraciones, la termografía, el análisis de lubricantes, entre otras son técnicas de mantenimiento predictivo que permiten hallar las causas de posibles fallos anticipándose a la avería (Ocampo, 2013). En este proyecto se utilizó la técnica de análisis de vibraciones, se explicará a detalle lo que conlleva ésta técnica.

Para empezar, se puede dar una definición y características de la vibración. La vibración es el movimiento de vaivén de una máquina o elemento de ella en cualquier dirección del espacio desde su posición de equilibrio (National Instruments, 2004). Generalmente, la causa de la vibración reside en problemas mecánicos como son: desequilibrio de elementos rotativos; desalineación en acoplamientos; engranajes desgastados o dañados; rodamientos deteriorados; fuerzas aerodinámicas o hidráulicas, y problemas eléctricos (Pozo, 2013). Estas causas como se puede suponer son fuerzas que cambian de dirección o de intensidad, estas fuerzas son debidas al movimiento rotativo de las piezas de la máquina, aunque cada uno de los problemas se detecta estudiando las características de vibración (Torres et al., 2015). Las características más importantes son: frecuencia, desplazamiento, velocidad, aceleración.

La frecuencia es una característica simple y significativa en este análisis. Se define como el número de ciclos completos en un período de tiempo (Analysis, n.d.). La unidad característica es cpm (ciclos por minuto). Existe una relación importante entre frecuencia y velocidad angular de los elementos rotativos. La correspondencia entre cpm y rpm (ciclos por minuto-revoluciones por minuto) identificará el problema y la pieza responsable de la vibración (Iso, 1800). Esta relación es debida a que las fuerzas cambian de dirección y amplitud de acuerdo a la velocidad de giro. Los diferentes problemas son detectados por las frecuencias iguales a la velocidad de giro o bien múltiplos suyos. Cada tipo de problema muestra una frecuencia de vibración distinta.

La amplitud de la vibración indica la importancia, gravedad del problema, ésta característica da una idea de la condición de la máquina (Vishwakarma, Purohit, Harshlata, & Rajput, 2017). Se podrá medir la amplitud de desplazamiento, velocidad o aceleración. La velocidad de vibración tiene en cuenta el desplazamiento y la frecuencia, es por tanto un indicador directo de la severidad de vibración (Karnovsky & Lebed, 2016). La severidad de vibración es indicada de una forma más precisa midiendo la velocidad, aceleración o desplazamiento según el intervalo de frecuencias entre la que tiene lugar, así para bajas frecuencias, por debajo de 600 cpm, se toman medidas de desplazamiento. En el intervalo entre 600 y 60.000 cpm, se mide

velocidad, y para altas frecuencia, mayores a 60.000 cpm, se toman aceleraciones (Torres et al., 2015).

La velocidad es una característica importante en la vibración, gráficamente se puede ver en la figura 1.

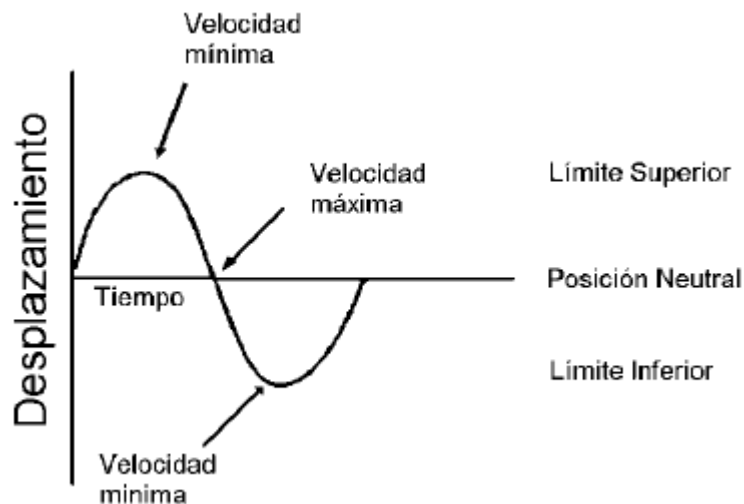


ILUSTRACIÓN 1. GRÁFICA DE VELOCIDAD (TORRES ET AL., 2015).

Se mide la velocidad de pico mayor de todo el recorrido que realiza el elemento al vibrar. La unidad es mm/s. El cambio de esta característica trae consigo un cambio de aceleración. La velocidad tiene una relación directa con la severidad de vibración, por este motivo es el parámetro que siempre se mide. Las vibraciones que tienen lugar entre 600 y 60.000 cpm se analizan teniendo en cuenta el valor de la velocidad (Torres et al., 2015).

La aceleración está relacionada con la fuerza que provoca la vibración, algunas de ellas se producen a altas frecuencias, aunque velocidad y desplazamiento sean pequeños (Espino et al., 2015).

Un punto importante a la hora de hablar de vibraciones es conocer la severidad de vibración, ella indica la gravedad que puede tener un defecto (Mosquera et al., 2001). La amplitud de la vibración expresa la gravedad del problema, pero es difícil establecer valores límites de la vibración que detecten un fallo. La finalidad del análisis de vibraciones es encontrar un aviso con suficiente tiempo para poder analizar causas y forma de resolver el problema ocasionando el paro mínimo posible en la máquina (Karnovsky & Lebed, 2016).

Una vez obtenido un histórico de datos para cada elemento de las máquinas que se estudian, el valor medio refleja la normalidad en su funcionamiento (Wang et al., 2012). Desviaciones continuas o excesivas indicarán un posible fallo que será

identificado después, teniendo en cuenta la frecuencia a la que se producen las mayores vibraciones.

Cuando no se posee histórico de datos para una máquina, puede analizarse la severidad de vibración teniendo en cuenta la norma ISO 2372 (“Iso 2372 – Vibration Severity,” n.d.).

Una vez obtenidos de una forma metódica y precisa los datos de vibraciones de una máquina donde se ha detectado un problema, es necesario identificar cual ha sido su causa y así buscar la forma y momento de reparación más eficiente, es decir, que elimine el fallo y su costo económico sea el mínimo posible.

SISTEMA DE ADQUISICIÓN DE DATOS

Es un conjunto de elementos eléctricos y computacionales capaz de obtener datos de elementos físicos. Estos sistemas son empleados para controlar, conocer o supervisar las variables de procesos físicos y está compuesto por 3 etapas:

1. Se emplean transductores para capturar la información relacionada con la vibración proveniente del sistema rotacional.
2. La señal la adquiere la tarjeta de adquisición de datos de forma analógica, para ser procesada y convertida en una señal digital que es enviada vía USB al computador.
3. La señal digital proveniente de la tarjeta capturada por el computador para su procesamiento y análisis.



ILUSTRACIÓN 2. SISTEMA DE ADQUISICIÓN DE DATOS (SPENDING, 2008).

Un transductor es un dispositivo diseñado para recibir información de una magnitud del exterior y transformarla en otra magnitud, normalmente eléctrica, la cual tenga la capacidad de poder cuantificarse y manipularse (Galindo, 2010).

Las características importantes son:

1. Rango de medida: Es el intervalo de valores que el transductor puede medir.
2. Precisión: Es el grado de exactitud que posee el instrumento.
3. Resolución: Mínima variación de la magnitud de entrada que puede apreciarse a la salida.
4. Sensibilidad: Es una cualidad que indica que tanto cambia la salida cuando la variable censada cambia.
5. Linealidad: Hace referencia al grado de coincidencia entre la curva de calibración y una línea recta.
6. Rapidez de respuesta: Puede ser un tiempo fijo o depender de cuánto varíe la magnitud a medir.

TARJETA DE ADQUISICIÓN DE DATOS

Una tarjeta de adquisición de datos es una tarjeta electrónica por medio de la cual se puede hacer un interfaz y conversión entre lo analógico y lo digital, en este caso se utiliza una bornera (terminal de conexiones) donde llega la señal analógica de los transductores para ser procesada y convertida en una señal digital que es enviada a una computadora.

LABVIEW

Es un sistema de programación gráfica para aplicaciones que involucren adquisición, control, análisis y presentación de datos (Spending, 2008) . Las ventajas que proporciona el empleo de LabVIEW se resumen en:

1. Reduce el tiempo de desarrollo de las aplicaciones ya que es intuitivo y fácil de aprender (Sumathi & Surekha, 2007).
2. Flexibilidad del sistema, permitiendo cambios y actualizaciones tanto del hardware como del software.
3. El sistema está dotado de un ambiente gráfico.
4. Tiene la posibilidad de incorporar aplicaciones escritas en otros lenguajes de programación.

ARQUITECTURAS DE PROGRAMACIÓN EN LABVIEW

- i. Máquina de Estados
- ii. Interfaz de Usuario Basada en Eventos
- iii. Productor/Consumidor
- iv. Máquina de Estados en Cola –Productor/Consumidor

MÁQUINA DE ESTADOS

Necesito ejecutar una secuencia de eventos, pero el orden está determinado de manera interactiva. Un ejemplo muy claro es la programación de una dispensadora de refrescos, la cual se muestra su diagrama:

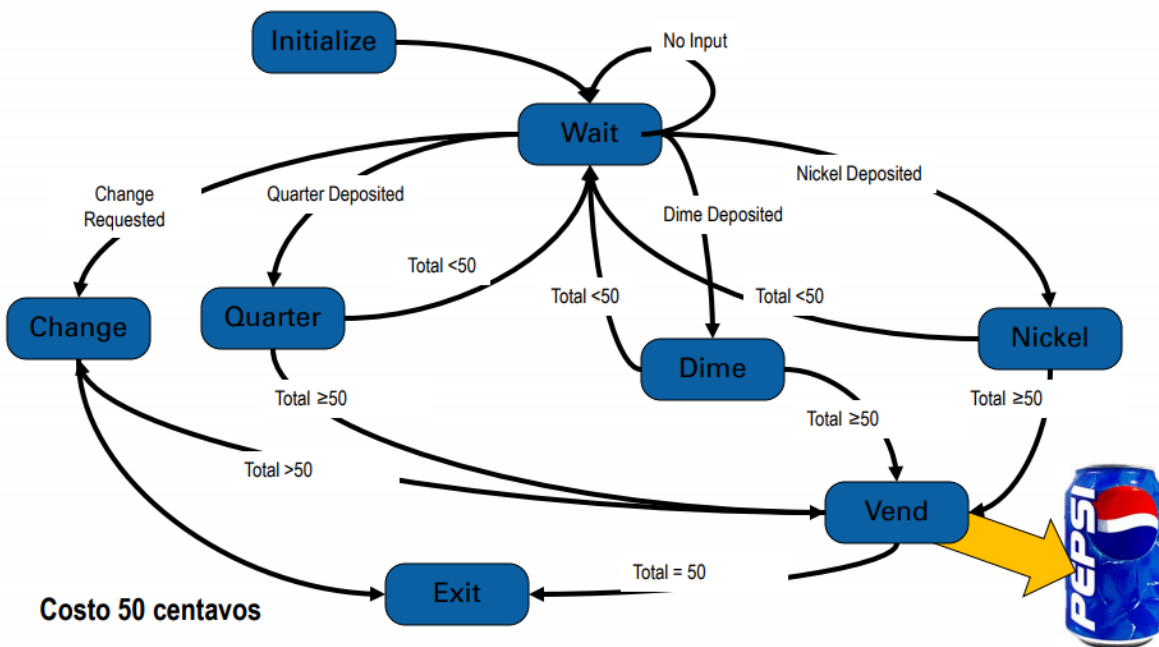


ILUSTRACIÓN 3 MÁQUINA DE ESTADOS

Los estados del programa a están previamente vistos como: Si meto una moneda para comprar un refresco tiene que valorar si da cambio o no, si da cambio tiene que hacer la resta y luego despachar, si no, solo despacha. Cada acción es un caso del Case y tienen un límite.

INTERFAZ DE USUARIOS BASADA EN EVENTOS

Estoy sondeando acciones del usuario en la interfaz, lo cual hace más lenta mi aplicación y en ocasiones no logro detectar esas acciones (Version & Number, 2014). Esta arquitectura es especial para que un operador maneje el programa, puede usarse como en modo manual o semiautomático.

Programación basada en un procedimiento

- I. Ejecuta una serie de instrucciones en secuencia
- II. Requiere estar sondeando para detectar eventos
- III. No puede determinar el orden de múltiples eventos

Programación basada en eventos

- I. Determina la ejecución al correr el VI
- II. Espera a que ocurran eventos sin consumir recursos del CPU
- III. Recuerda el orden de múltiples eventos

PRODUCTOR CONSUMIDOR

Tengo dos procesos que necesitan ejecutarse al mismo tiempo. Necesito asegurarme que uno de ellos, no haga más lento al otro. Se aplica mucho en procesos los cuales tengo una adquisición rápida de un sensor y se tiene que procesar los datos adquiridos pero en una velocidad más lenta.

Funcionamiento del patrón.

1. Ciclo maestro indica a uno o más esclavos cuando pueden ejecutarse
2. Permite ejecución asíncrona de ciclos
3. La independencia de datos rompe el flujo de datos y permite el multithreading
4. Desliga los procesos

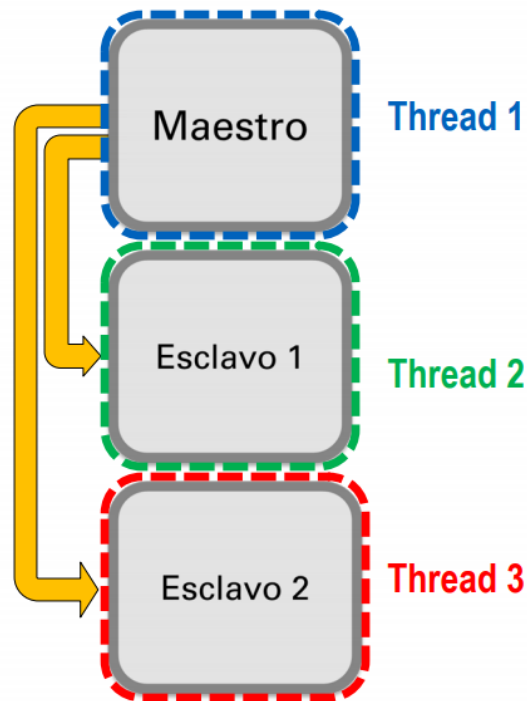


ILUSTRACIÓN 4 ESTRUCTURA DE PRODUCTOR CONSUMIDOR

Comunicación que permite hacerse entre ciclos.

1. Variables
2. Notificadores
- 3. Colas**
4. semáforos

La comunicación por colas es la que se utiliza en éste proyecto y eso es debido a que funciona como cuando se hace cola para comprar tortillas, si llegan 5 personas a comprar; las irán despachando una por una hasta que terminen las 5. Cada persona tiene que esperar su turno para que sea despachada y de esa manera todas van a comprar sus tortillas. Esta arquitectura con comunicación por colas funciona exactamente igual. Los datos se irán anidando en una cola que los enviará al consumidor 1 por 1 hasta que se manden todos y se cerciora que ningún dato se pierda.

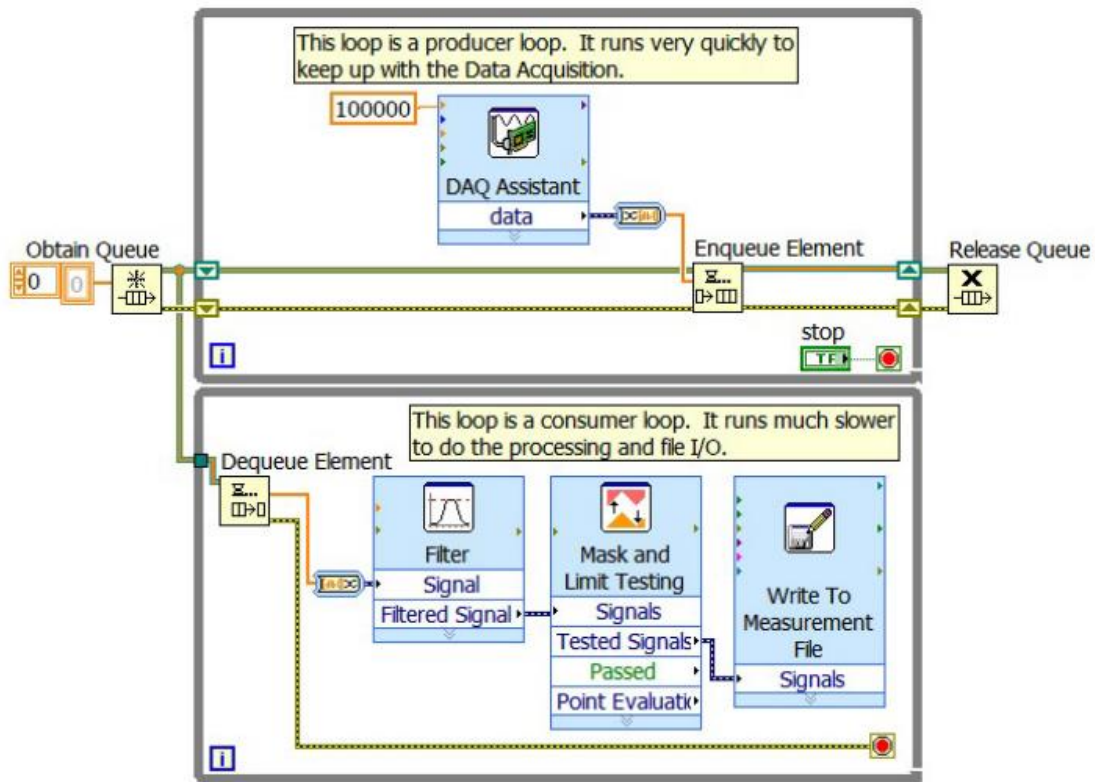


ILUSTRACIÓN 5 DIAGRAMA DE BLOQUES DE ARQUITECTURA PRODUCTOR CONSUMIDOR

MÁQUINA DE ESTADOS EN COLA –PRODUCTOR/CONSUMIDO

Esta estructura es exactamente igual que la anterior pero ahora el consumidor tiene una máquina de estados que la gobierna el productor. De esta manera se sabe que se tienen un cierto número de casos a ejecutar.

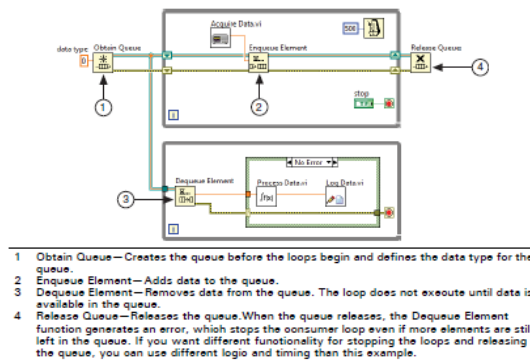


ILUSTRACIÓN 6 MÁQUINA DE ESTADOS EN COLA CON PRODUCTOR CONSUMIDOR

ANTECEDENTES

- Estupiñán P, 2006. Se describe el diseño e implementación de un analizador de vibraciones basado en un Sistema de Adquisición de Datos (*SAD*), con capacidad para monitorear y analizar vibraciones en uno y dos canales simultáneamente. El analizador permite calcular espectros, formas de onda, analizar órbitas, coherencia, así como también almacenar los datos adquiridos para su posterior análisis.
- Galindo, 2010. Tiene como objetivo principal la construcción y validación de un sensor de vibraciones construido a partir de un acelerómetro mecánico-electrónico, el cual se armó en un circuito impreso. Las señales obtenidas el sensor se capturaron con Labview, para posteriormente exportarlos a un archivo ASCII. Los datos obtenidos se analizaron con MATLAB.
- Weilin Wang, Changying Li, Ernest W. Tollner y Glen C. Rains, 2011. Desarrollo de un software para la adquisición de un espectro de frecuencia de una imagen en constante movimiento usando LabVIEW. Se recolectan los espectros de frecuencia y tiempo real de 3 tipos de imágenes usando 2 tipos de arquitecturas de programación tales como máquinas de estados y productor consumidor.
- Piero Espino, J. Rendón, J. Valdez, Y. Davizón, J. Nuñez y C. Hinojosa, 2015. Diseño, implementación y uso de interfaces gráficas desarrolladas en LabVIEW para el aprendizaje de análisis y control de sistemas de un grado de libertad sujeto a una excitación libre. Este programa es desarrollado para la educación de mecánica de vibraciones.
- Jared R. Ocampo, 2017. Se crea un sistema de bajo costo que permite monitorear vibraciones y diagnosticar disturbios que podrían causar problemas en máquinas rotatorias usando el software de LabVIEW y un microcontrolador pic.

HIPÓTESIS

Si se conocen los espectros de frecuencia de una máquina rotativa entonces es posible realizar la comparativa con lo establecido en la norma ISO 2372, mediante el desarrollo de un algoritmo implementado en LabVIEW que permitirá conocer la severidad de las mismas.

METODOLOGÍA DE LA SOLUCIÓN

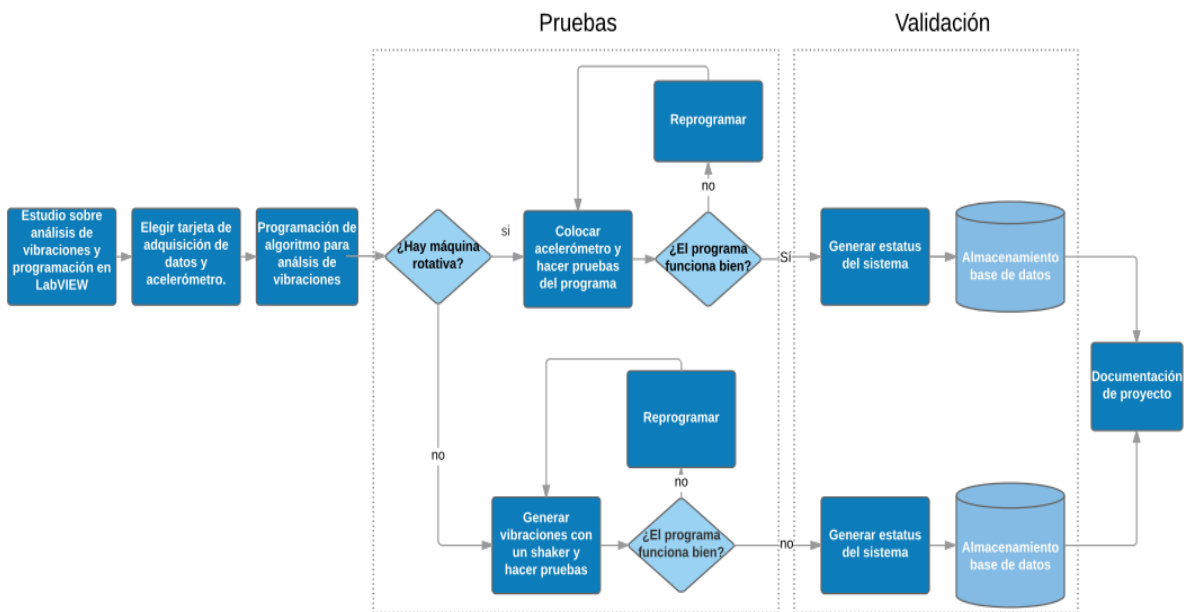


ILUSTRACIÓN 7. DIAGRAMA DE FLUJO DEL PROCESO.

ALCANCE

Desarrollar un algoritmo en LabVIEW que sea capaz de mostrar la clase y severidad de una máquina rotativa, gráficas en decibeles y frecuencias de: aceleración, velocidad y posición. Así como también tenga la envergadura de guardar los datos necesarios para que el usuario pueda hacer análisis en cualquier momento.

RESULTADOS ESPERADOS

1. Estructura de programación en la cual pueda ejecutar dos actividades con diferentes velocidades de ejecución.
2. Cálculo de velocidad y desplazamiento así como también las gráficas en frecuencia y decibeles.
3. Comparativa de espectros de frecuencia.
4. Interfaz que indique el estatus del sistema.
5. Base de datos con información guardada por el usuario.
6. Redacción de la Tesis.

DESARROLLO

Este programa realiza un análisis de vibraciones mediante acelerómetro y una tarjeta de adquisición de datos. El programa nos proporciona el estatus del sistema vibratorio mediante la comparación de espectros de frecuencia del Shaker y la tabla de severidades de la norma ISO2372.

Dicho estatus del sistema se muestra en un indicador tipo String en la ventana PRINCIPAL del Panel Frontal, mismo que nos dirá la clase de la máquina (son 4 clases) y la severidad en la que se encuentra.

Es importante mencionar que la comparativa no lo hace en la señal de aceleración, más bien lo hace con la señal de velocidad – en el marco teórico se explica por qué- es por ello que la ventana a la cual se debe hacer observaciones es en a de Velocidad. Se tienen 4 ventanas para hacer el análisis en “tiempo real” las cuales son:

- PRINCIPAL
- FFT Velocidad
- FFT Aceleración
- FFT Desplazamiento

El programa de manera autónoma nos genera el estatus, pero nos muestra las diversas gráficas para ir viendo el comportamiento de ciertas máquinas, como por ejemplo el comportamiento del arranque, de un paro o de la excitación del sistema con una fuerza externa. Para ello se aplicó dos opciones de guardado; la primera es para guardar datos de manera instantánea, esto quiere decir que al instante de ser presionado el botón se guardarán los datos vistos en la gráfica; la segunda es para un guardado por tiempo, esto se aplica para ver el comportamiento del sistema por un periodo cualquiera de tiempo. Esto ayuda a que los datos guardados puedan ser analizados en cualquier momento o con otro software.

MATERIALES

- ✓ Acelerómetro triaxial 4506 Brüel and Kjaer.



ILUSTRACIÓN 8 ACELEROMETRO TRIAXIAL 4506 BRÜEL AND KJAER.

- ✓ Tarjeta de adquisición de datos Hi-Speed USB Carrier NI USB-9162 de National Instruments.

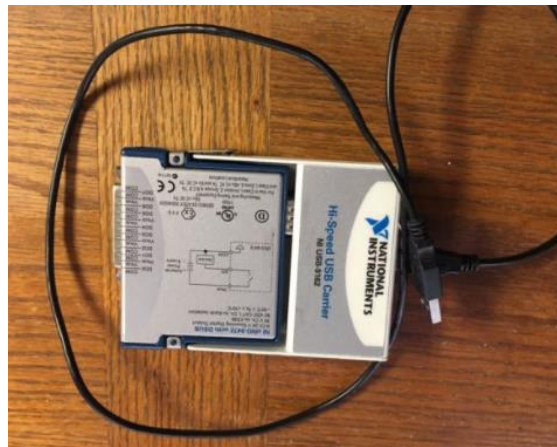


ILUSTRACIÓN 9 TARJETA DE ADQUISICIÓN DE DATOS HI-SPEED USB CARRIER NI USB-9162 DE NATIONAL INSTRUMENTS.

- ✓ Shaker VR520 Vibration Research Corporation.



ILUSTRACIÓN 10 SHAKER VR520 VIBRATION RESEARCH CORPORATION.

- ✓ Generador de señales Tektronix AFG 3151C.



ILUSTRACIÓN 11 GENERADOR DE SEÑALES TEKTRONIX AFG 3151C.

- ✓ Software LabVIEW.
- ✓ Software Excel.

En seguida se muestran las imágenes de como se montó el acelerómetro en el Shaker para posteriormente enviar la señal a la tarjeta de adquisición de datos y ser leído por la computadora. Al Shaker se le manda una señal desde el generador de señales y cumple la función de imitar la señal que le está entrando. Por lo tanto será nuestra máquina vibratoria.

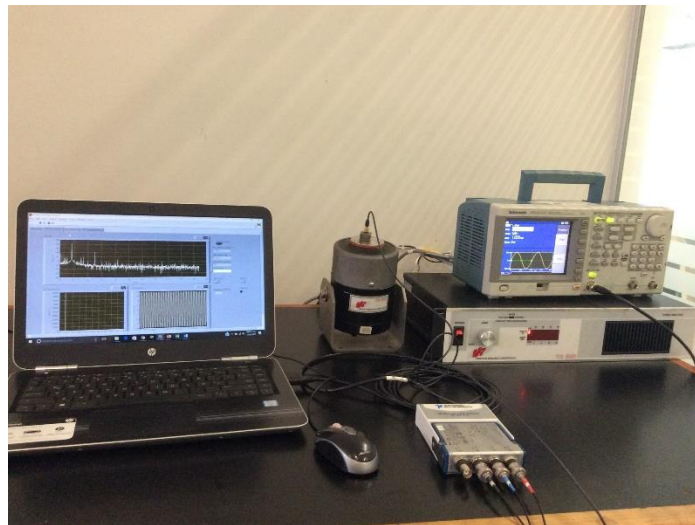


ILUSTRACIÓN 12 MATERIALES USADOS.

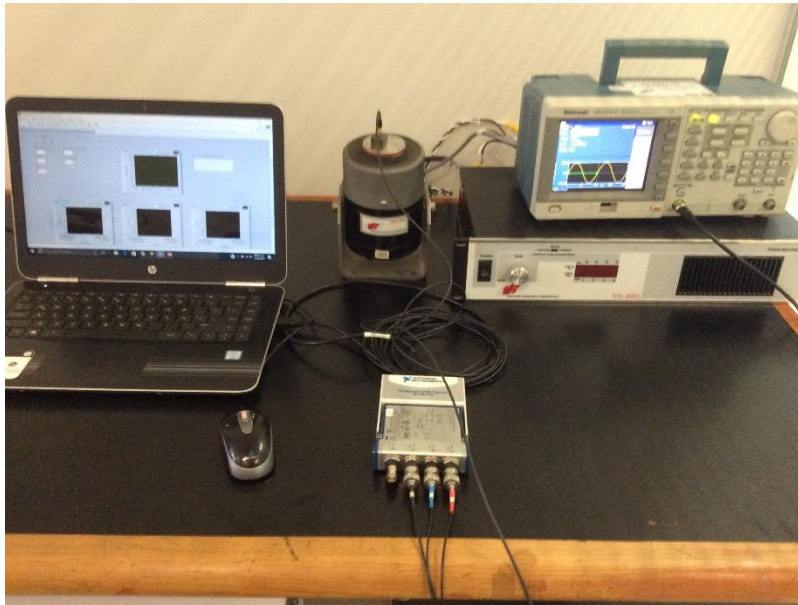


ILUSTRACIÓN 13 CONEXIONES

Teniendo una noción de que es lo que hace el programa y cuáles son los materiales que se están usando y cómo se aplican, toca explicar el diagrama de bloques y el panel frontal.

PANEL FRONTAL

El programa tiene 5 datos de entrada los cuales son:

1. Número de muestras: Datos para el acelerómetro.
2. Zoom: Acercamiento de gráficas.
3. Hp: Para clasificar la máquina.
4. Frecuencia de muestreo: Dato para el acelerómetro.
5. Cpm máquina: Ciclos por minuto de la máquina.

Son los datos necesarios para que el programa funcione. Con estos datos se van a mostrar las gráficas que generan los 3 canales que tiene el acelerómetro.

En la ilustración siguiente se tiene 4 gráficas, una por cada eje y otra donde junta los 3. Tiene un String donde nos da el estatus del sistema (Clase de máquina y estado de severidad). Cuenta con 3 pestañas: Aceleración, velocidad y desplazamiento.

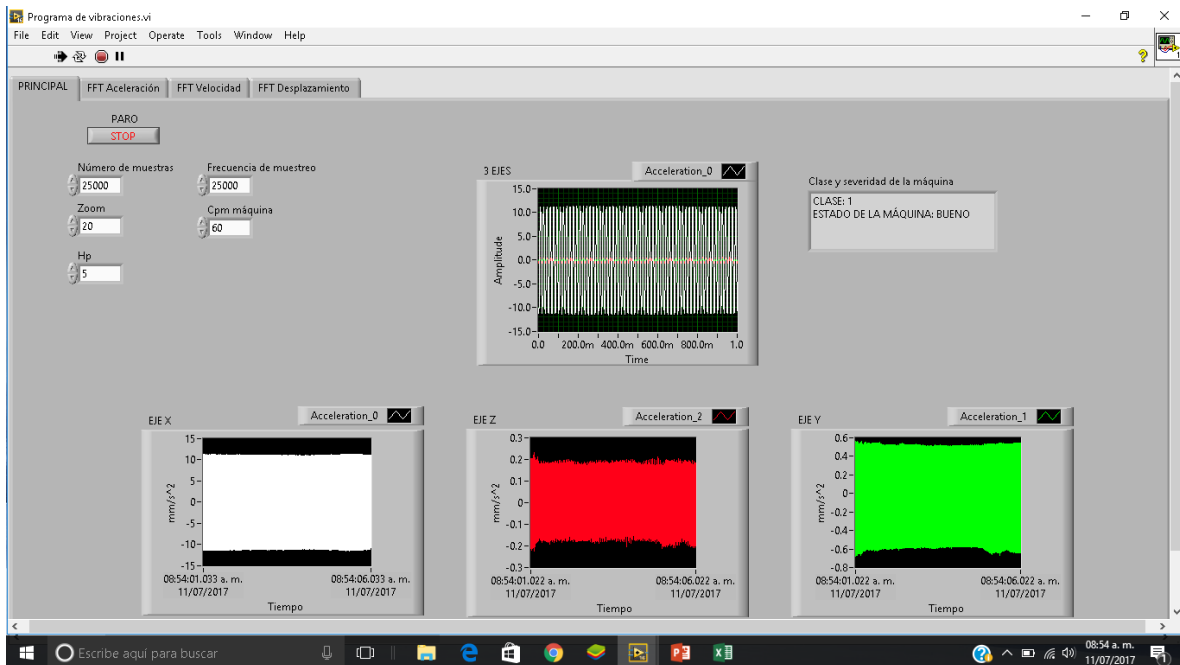


ILUSTRACIÓN 14 PANEL FRONTAL DEL PROGRAMA

En la siguiente ilustración nos pasamos a la pestaña de aceleración, la cual nos muestra la gráfica; FFT dB, ésta gráfica nos dice las frecuencias del sistema pero en decibeles; FFT Aceleración que nos muestra la señal de aceleración pero en frecuencia; Aceleración es la gráfica que proviene tal cual del acelerómetro. Tiene 5 indicadores:

1. Valor máximo de aceleración.
2. Valor máximo en dB.
3. Frecuencia máxima en aceleración.

4. Muestras guardadas.
5. Led que indica cuando está grabando.

Por otro lado cuenta con 4 controles:

1. Congelar gráfica: Detiene la gráfica para que sea analizada, es como si le tomasen una foto.
2. Escala en dB: Cumple la función de set point, es la escala de los decibeles.
3. Instante: Graba de manera instantánea los datos.
4. Tiempo: Graba los datos el tiempo que el usuario necesite.

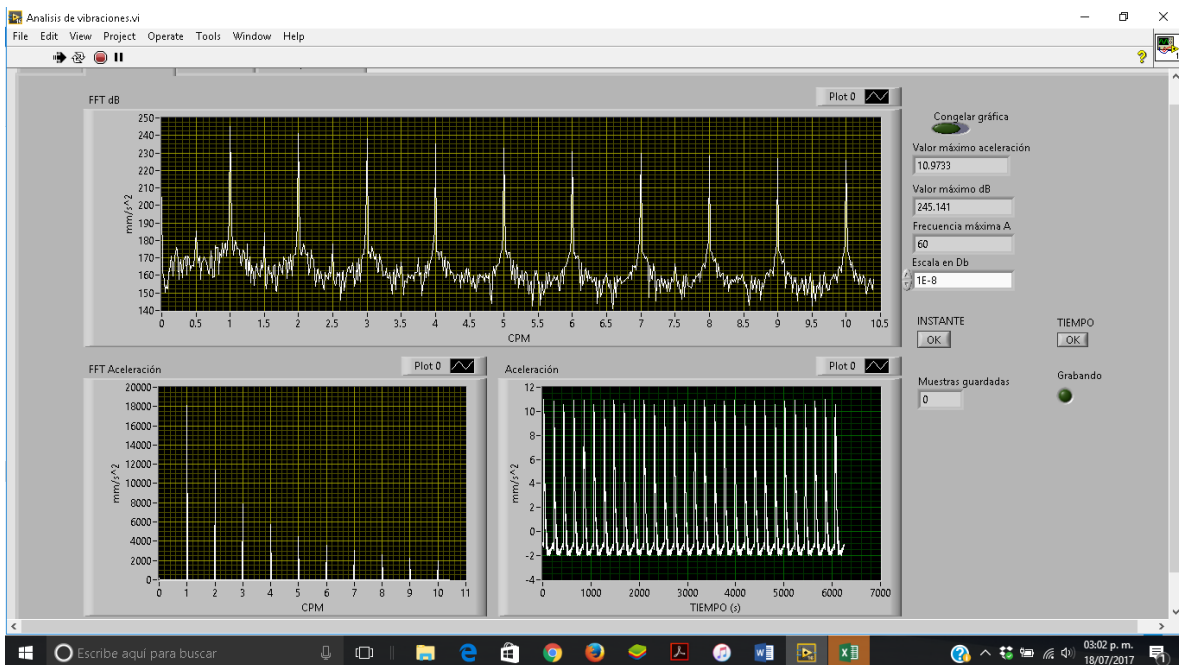


ILUSTRACIÓN 15 PESTAÑA DE ACCELERACIÓN

En la siguiente ilustración nos pasamos a la pestaña de aceleración, la cual nos muestra la gráfica; FFT dB, ésta gráfica nos dice las frecuencias del sistema pero en decibeles; FFT Velocidad que nos muestra la señal de Velocidad pero en frecuencia; Velocidad es la gráfica que proviene tal cual del acelerómetro. Tiene 5 indicadores:

6. Valor máximo de velocidad.
7. Valor máximo en dB.
8. Frecuencia máxima en velocidad.
9. Muestras guardadas.
10. Led que indica cuando está grabando.

Por otro lado cuenta con 4 controles:

5. Congelar gráfica: Detiene la gráfica para que sea analizada, es como si le tomasen una foto.
6. Escala en dB: Cumple la función de set point, es la escala de los decibeles.
7. Instante: Graba de manera instantánea los datos.
8. Tiempo: Graba los datos el tiempo que el usuario necesite.

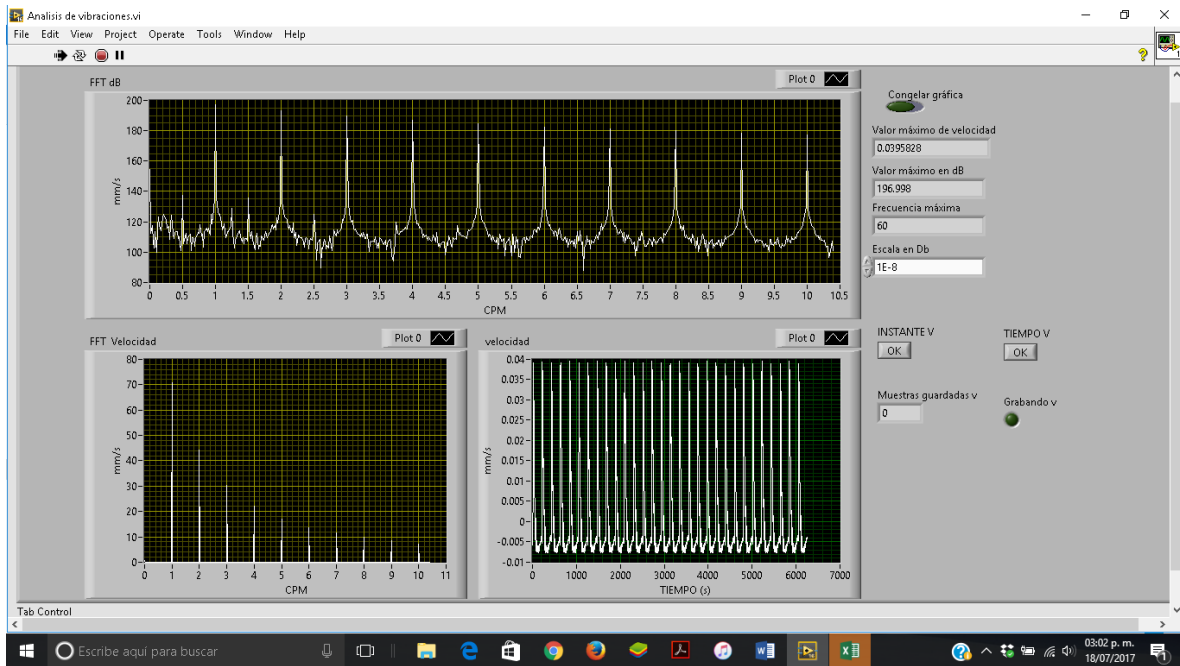


ILUSTRACIÓN 16 PESTAÑA DE VELOCIDAD

En la siguiente ilustración nos pasamos a la pestaña de Desplazamiento, la cual nos muestra la gráfica; FFT dB, ésta gráfica nos dice las frecuencias del sistema pero en decibeles; FFT Desplazamiento que nos muestra la señal de Desplazamiento pero en frecuencia; Desplazamiento es la gráfica que proviene tal cual del acelerómetro. Tiene 5 indicadores:

11. Valor máximo de velocidad.
12. Valor máximo en dB.
13. Frecuencia máxima en velocidad.
14. Muestras guardadas.
15. Led que indica cuando está grabando.

Por otro lado cuenta con 4 controles:

9. Congelar gráfica: Detiene la gráfica para que sea analizada, es como si le tomasen una foto.
10. Escala en dB: Cumple la función de set point, es la escala de los decibeles.
11. Instante: Graba de manera instantánea los datos.
12. Tiempo: Graba los datos el tiempo que el usuario necesite.



ILUSTRACIÓN 17 PESTAÑA DE DESPLAZAMIENTO

Al final se muestra el Excel con la base de datos guardada.

	eje X	eje Y	eje X	eje Y	eje X	eje Y	eje X	eje Y	eje X	eje Y	eje X	eje Y	eje X	eje Y
1	0	0.006501	0	0.006664	0	0.006926	0	0.007123	0	0.007551	0	0.007819	0	0.007702
2	1	0.000128	1	0.000064	1	0.000056	1	0.000136	1	0.000018	1	0.00015	1	0.00005
3	2	0.000089	2	0.000038	2	0.000021	2	0.000071	2	0.000033	2	0.000034	2	0.00001
4	3	0.000021	3	0.000044	3	0.000014	3	0.000016	3	0.000017	3	0.000017	3	0.00001
5	4	0.000009	4	0.000049	4	0.000001	4	0.000017	4	0.000012	4	0.000004	4	0.00001
6	5	0.000039	5	0.000046	5	0.000005	5	0.000014	5	0.000021	5	0.000047	5	0.00001
7	6	0.000059	6	0.000017	6	0.000028	6	0.000006	6	0.000026	6	0.000007	6	0.00005
8	7	0.000067	7	0.000002	7	0.000039	7	0.000036	7	0.000024	7	0.000003	7	0.00001
9	8	0.000093	8	0.000164	8	0.000153	8	0.000138	8	0.00014	8	0.000139	8	0.00015
10	9	0.000074	9	0.000046	9	0.000014	9	0.000025	9	0.000051	9	0.000065	9	0.00002
11	10	0.000095	10	0.000066	10	0.000037	10	0.000044	10	0.000026	10	0.000009	10	0.00006
12	11	0.000153	11	0.000108	11	0.000199	11	0.000139	11	0.000182	11	0.000166	11	0.00016
13	12	0.000116	12	0.000115	12	0.000268	12	0.000129	12	0.000226	12	0.00019	12	0.00016
14	13	0.0001	13	0.000037	13	0.000164	13	0.000153	13	0.000051	13	0.000161	13	0.0001
15	14	0.000141	14	0.000093	14	0.000081	14	0.000087	14	0.000092	14	0.000104	14	0.00005
16	15	0.000018	15	0.000061	15	0.000069	15	0.000062	15	0.000066	15	0.000044	15	0.00004
17	16	0.000143	16	0.000013	16	0.000044	16	0.000058	16	0.000007	16	0.000056	16	0.00016
18	17	0.000006	17	0.000008	17	0.000016	17	0.000037	17	0.000026	17	0.000035	17	0.00004
19	18	0.000031	18	0.000002	18	0.000032	18	0.000099	18	0.000046	18	0.000018	18	0.00006
20	19	0.000092	19	0.000167	19	0.000139	19	0.000169	19	0.000086	19	0.000123	19	0.0001
21	20	0.000274	20	0.00017	20	0.000211	20	0.000264	20	0.000111	20	0.000272	20	0.00016
22	21	0.000271	21	0.000009	21	0.000211	21	0.000177	21	0.000134	21	0.000208	21	0.00011

ILUSTRACIÓN 18 DATOS GUARDADOS EN EXCEL

DIAGRAMA DE BLOQUES

ARQUITECTURA DE PROGRAMACIÓN

Se escogió una arquitectura de programación capaz de adquirir datos del acelerómetro y procesarlos pero a velocidades diferentes puesto que se necesita alta velocidad para adquirir y poca velocidad para procesar. Se previó que los datos adquiridos (a una velocidad alta) no se hayan perdido en el camino, que exista una fila de datos para que se envíen de uno en uno y de esa manera no se pierdan.

Para no perder ningún dato y usar ciclos en paralelo, se escogió la estructura de productor consumidor con máquina de estados.

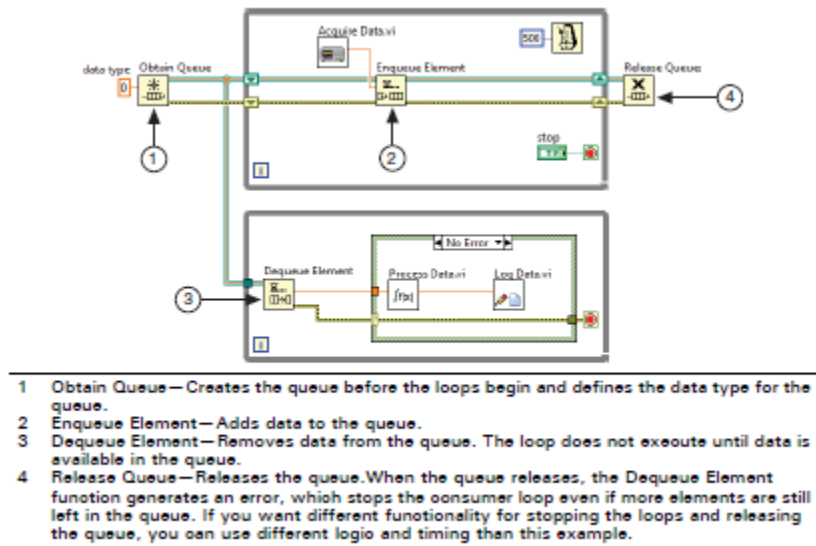


ILUSTRACIÓN 19 ARQUITECTURA PRODUCTOR- CONSUMIDOR

La arquitectura se divide en dos Loops, en seguida se van a explicar la función de cada uno basándose en la ilustración anterior, se tomarán los mismos números.

PRODUCTOR.

Se observa en “Data type” que es el tipo de dato de entrada, puede ser de cualquier tipo: Booleano, Cluster, String, Numérico, Array o cualquier otro. Este dato es el que gobierna el envío ya que se tienen que enviar exactamente el mismo dato y se tiene que manipular el mismo dato, es por ello que primero se escoja los datos que se van a enviar para posteriormente se manipulen igual.

El dato que se escogió es un Cluster, mismo que contiene todos los datos que se van a estar enviando para el consumidor (parte 3 del programa “dequeue Element”), en la siguiente ilustración se muestra el Cluster guía para la transmisión y manipulación de datos.

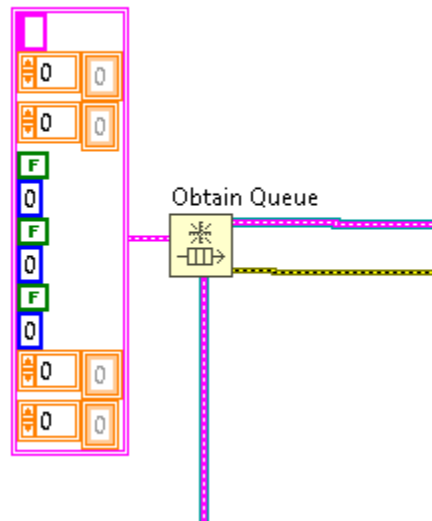


ILUSTRACIÓN 20 CLUSTER DE INICIO

Se observa que se tienen 4 tipos de datos:

1. String
2. Array doblé
3. Booleano
4. Numeric I32

Son 11 datos en total de los cuales hay 4 diferentes. Esta es la plantilla a usar en todo el programa, este Cluster es la guía, por lo tanto siempre que se manipule se debe de tomar en cuenta que lleva 11 datos y están en el orden de la imagen.

En seguida se muestra el Cluster que entra al Enqueue Element del productor:

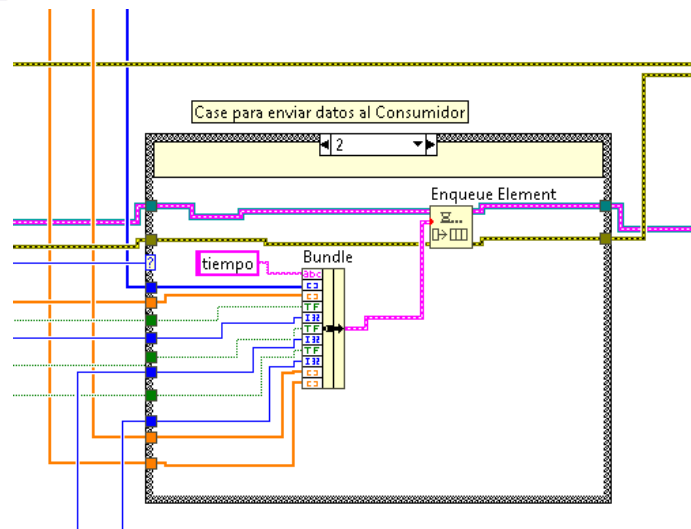


ILUSTRACIÓN 21 CLUSTER DE ENVÍO DE DATOS

Se observa que dentro de un “Case” está el envío de los datos para el consumidor. La función “Bundle” Concatena todos los datos de entrada y los convierte en un Cluster. Si se observa con detención, el orden en que se están concatenando es el mismo del Cluster guía. De manera forzosa tienen que ser iguales, si uno de los elementos no se mete en el mismo orden y exactamente del mismo tipo, nos marcará un error en el cable de salida.

En seguida se va a enlistar qué es cada cable que entra al Bundle:

1. String: Esta variable es muy importante puesto que nos dirá a qué estado de la máquina de estados del consumidor se tendrá que ir para ejecutar la acción. En resumen es el que escoge que ejecutará el consumidor.
2. Numeric: Sale de un “For” que nos proporciona la cantidad de datos que irán en el eje X de las gráficas que se van a guardar en Excel. Se calcula con el número de datos dividido entre 2 y comienza en cero.
3. Array 1D: Es el arreglo de los datos en decibeles para las gráficas de aceleración.
4. Booleano: Envía True o False si se presiona el botón de guardado por tiempo en aceleración.
5. Numeric I32: Es un contador del botón Instante de aceleración, cuenta cuantas veces se ha guardado la muestra y se reinicia cuando el programa se detiene.
6. Booleano: Envía True o False si se presiona el botón de guardado por tiempo en velocidad.
7. Numeric I32: Es un contador del botón Instante de velocidad, cuenta cuantas veces se ha guardado la muestra y se reinicia cuando el programa se detiene.
8. Booleano: Envía True o False si se presiona el botón de guardado por tiempo en desplazamiento.

9. Numeric I32: Es un contador del botón Instante de desplazamiento, cuenta cuantas veces se ha guardado la muestra y se reinicia cuando el programa se detiene.
10. Array 1D: Es el arreglo de los datos en decibeles para las gráficas de velocidad.
11. Array 1D: Es el arreglo de los datos en decibeles para las gráficas de desplazamiento.

Se observa que es la misma cantidad y orden de los datos del Cluster guía.

Para manipular el Case que manda los datos, es un Numeric I32 que proviene de la lógica de todos los botones de guardado. Se genera un arreglo booleano para que se convierta en un Numeric. La condición será de 6 bits, como solo se pueden encender uno a la vez, se tiene la combinación en numérico de: 0,1,2,4,8,16,32. Eso son los números de cada caso del Case. En seguida se muestra la programación de este segmento:

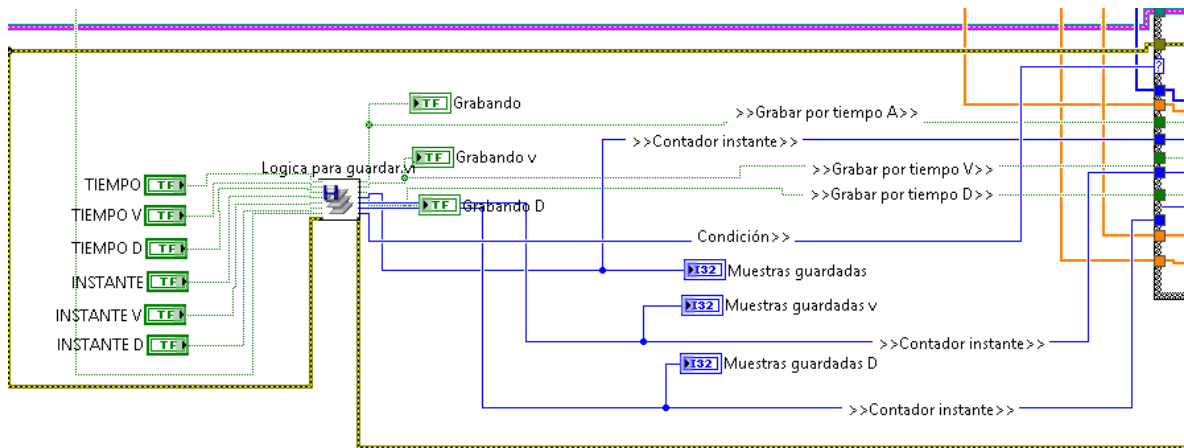


ILUSTRACIÓN 22 SECCIÓN LÓGICA QUE GOBIERNA EL CASE DE ENVIO DE DATOS

Se observa un Subvi llamado: Lógica para guardar. Este subvi se encarga de hacer la lógica booleana y la conversión de datos a numérico y las salidas para los leds. Tal como se muestra en la siguiente ilustración.

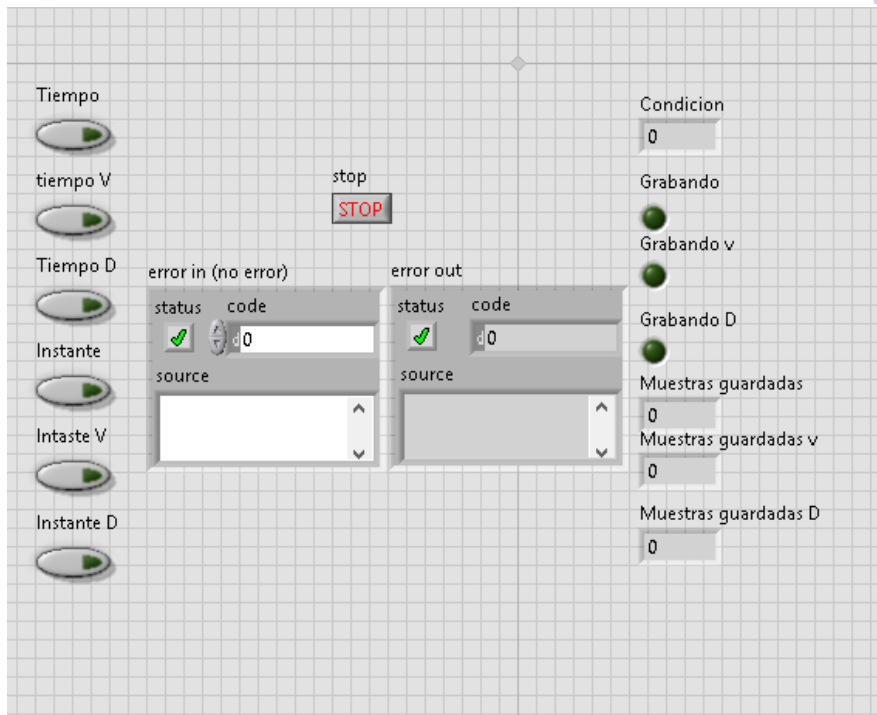


ILUSTRACIÓN 23 PANEL FRONTAL DE SUBVI PARA LÓGICA DE GUARDADO

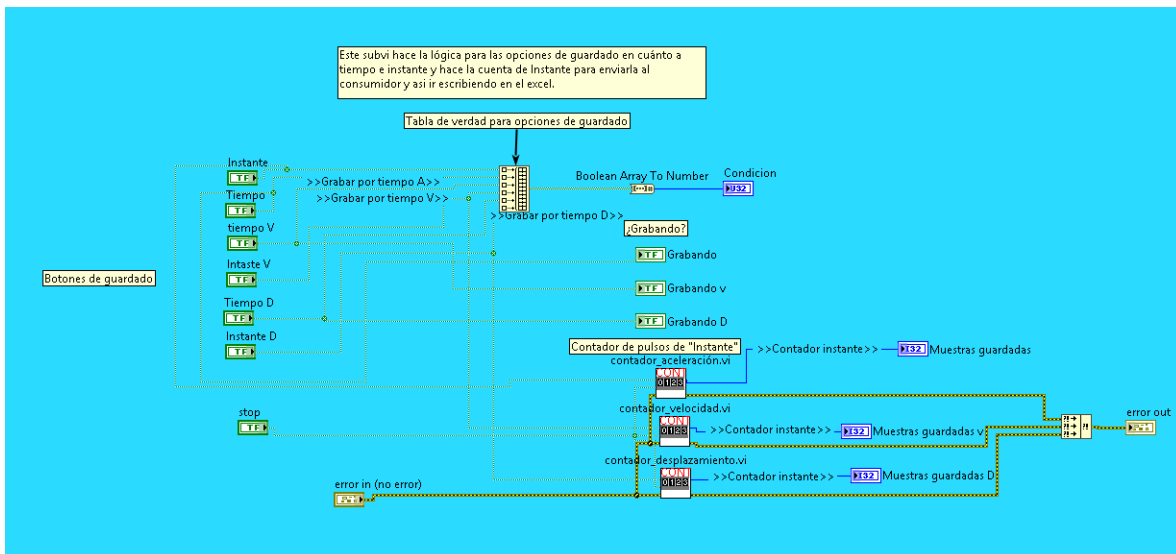


ILUSTRACIÓN 24 DIAGRAMA DE BLOQUES DE SUBVI PARA LÓGICA DE GUARDADO

Ahora toca enviar los datos, se muestra como en la salida del Case se manda el cable del Cluster y los errores que se están juntando en la función “Merge Error”.

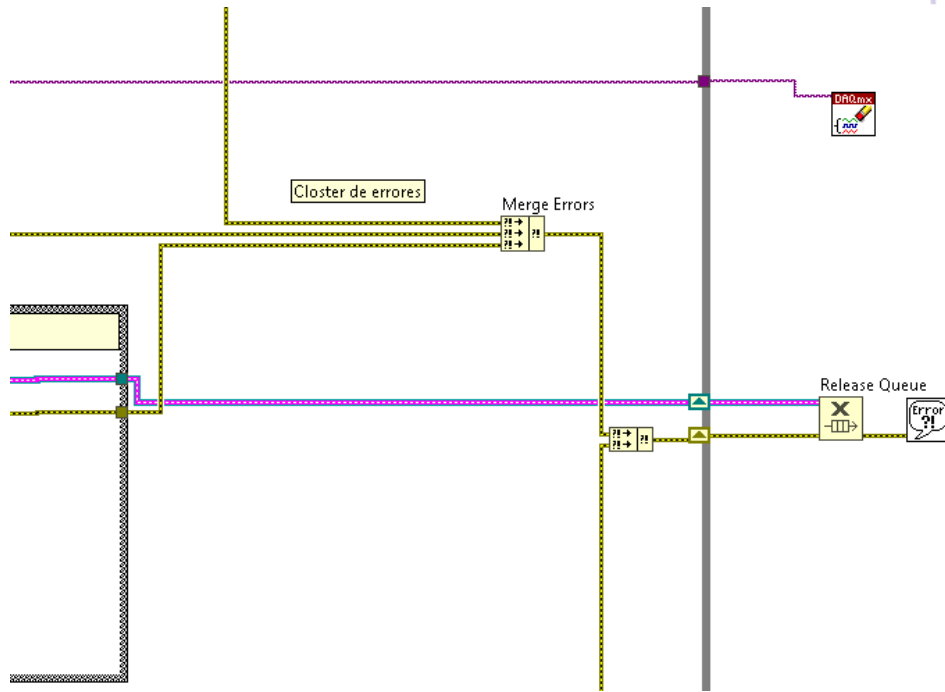


ILUSTRACIÓN 25 SECCIÓN DE PROGRAMA DONDE SE ENVÍA EL DATO Y SE CONBINAN ERRORES.

Ahora se va a mostrar cómo se calculan u obtiene los 11 datos del Cluster que se envía del productor al consumidor:

1. String: Se genera en cada caso con un String constante.
2. Numeric: Sale del ciclo For de la ilustración, mismo que en el número de iteraciones sale de un Subvi y es el número de muestras dividido en 2.

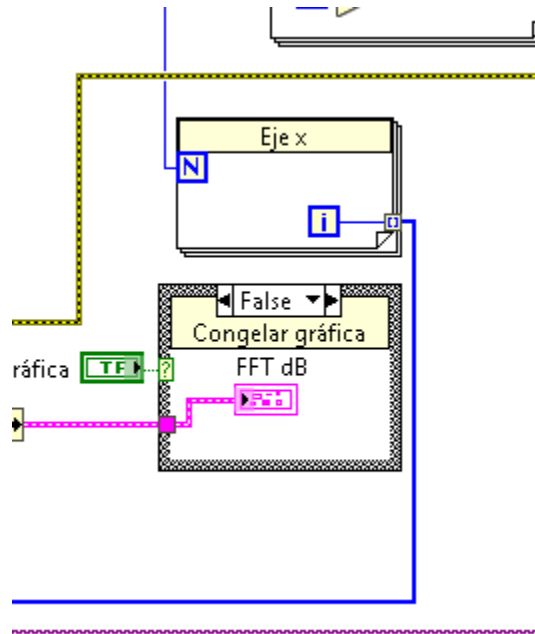


ILUSTRACIÓN 26 CICLO FOR PARA EL INDEXADO DE EJE X

3. Arreglo 1D: Arreglo de datos en decibeles que viene del Subvi de aceleración.

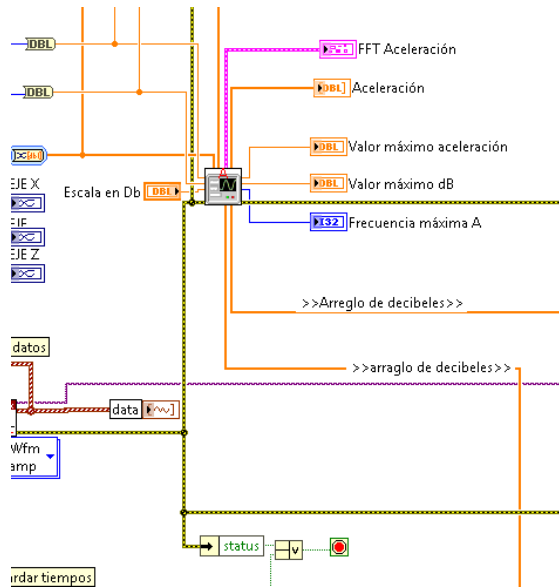


ILUSTRACIÓN 27 SECCIÓN DE PROGRAMA DONDE SE USA EL SUBVI

En seguida se muestra la programación del subvi.

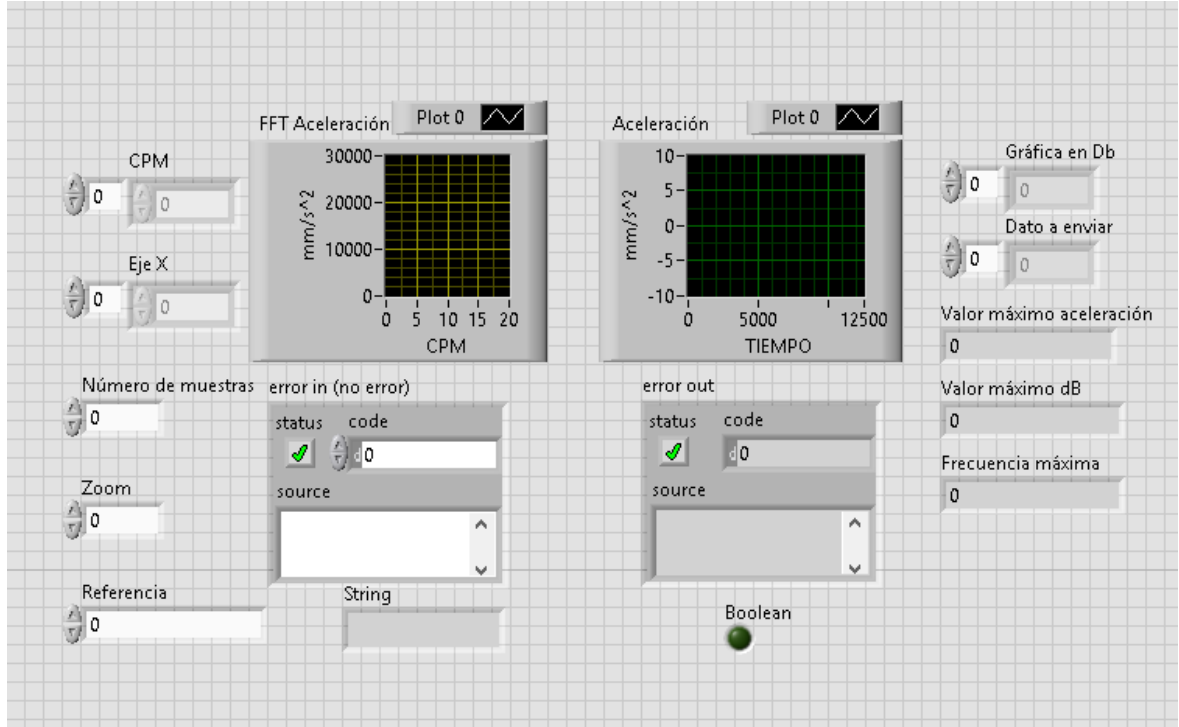


ILUSTRACIÓN 28 PANEL FRONTAL DE SUBVI DE ACCELERACIÓN

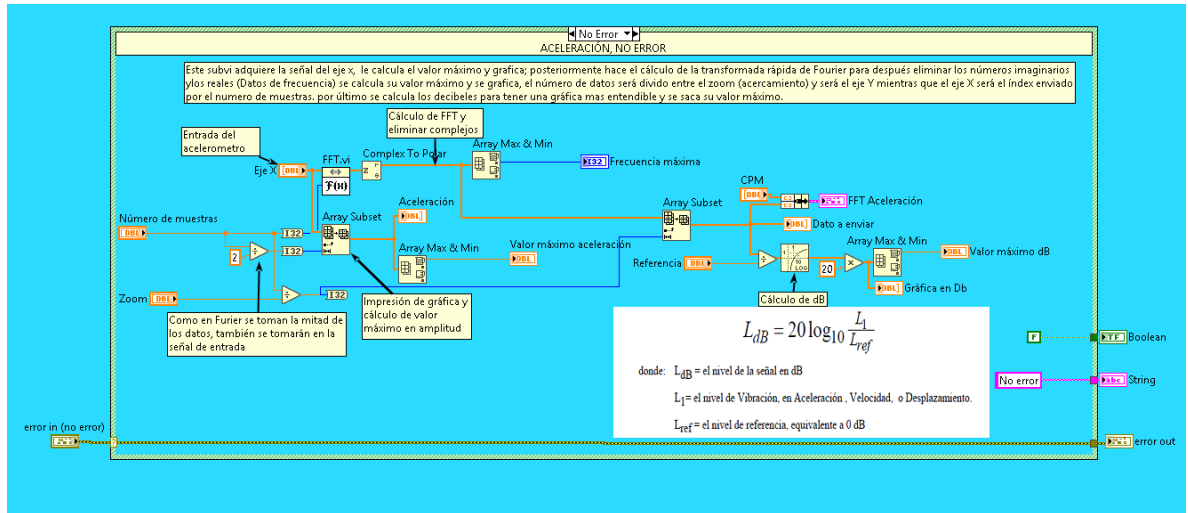


ILUSTRACIÓN 29 DIAGRAMA DE BLOQUES DE SUBVI DE ACCELERACIÓN

Este Subvi ejecuta la conversión de los datos del acelerómetro los cuales están en el dominio del tiempo y los pasa al dominio de la frecuencia gracias a la Transformada rápida de Fourier para posteriormente hacer las gráficas en frecuencia, dar las frecuencias altas y bajas así como también hacer el cálculo de decibeles y mostrar las gráficas y valores correspondientes.

Es importante recalcar que el Subvi cuenta con la estructura estándar para la ejecución, la cual es un Case de error, mientras no exista error que ejecute el programa y cuando exista error que se detenga y que mande el estatus que dicho Subvi está fallando junto con el código del error.

4. Booleano: Grabar por tiempo viene del Subvi de la ilustración 14.
5. Numeric I32: Es un contador del botón Instante de aceleración, cuenta cuantas veces se ha guardado la muestra y se reinicia cuando el programa se detiene. Se ilustra el Subvi para la ejecución de todos los puntos de guardado que van del punto 5 al 9.

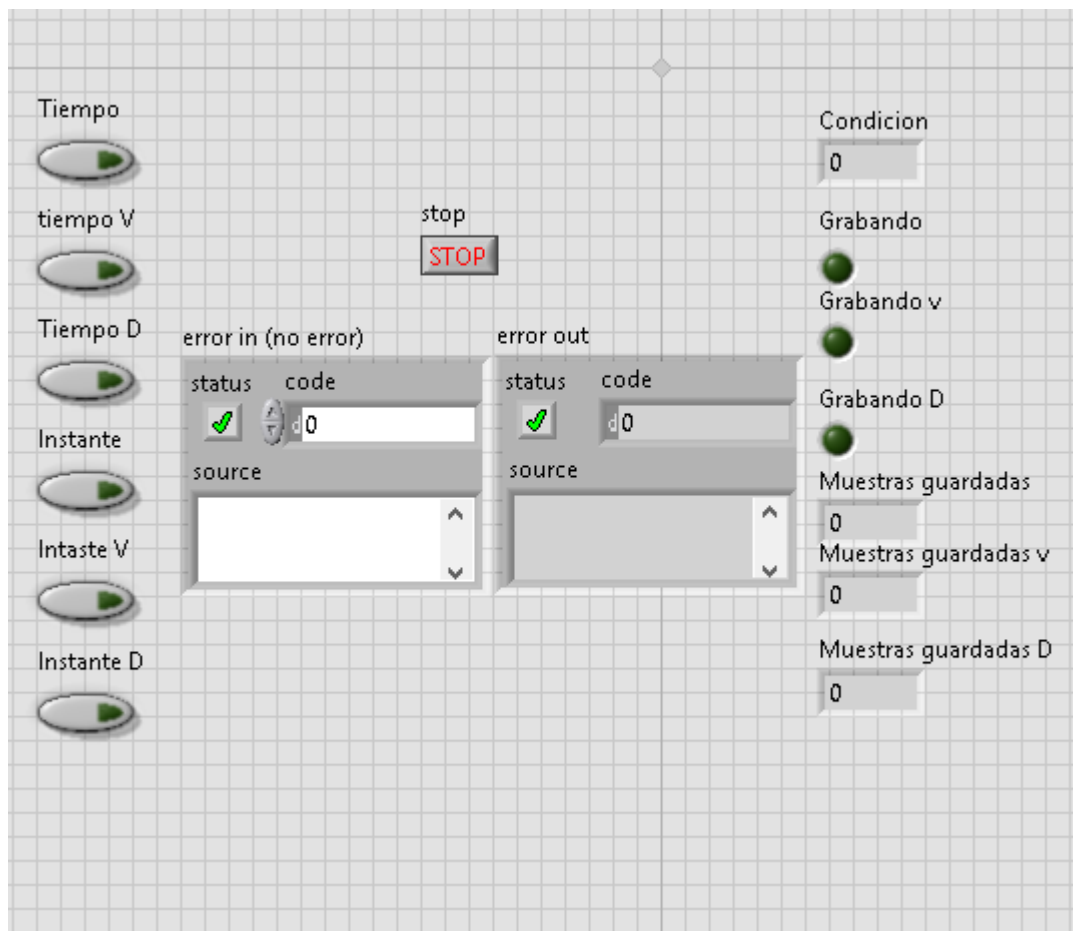


ILUSTRACIÓN 30 PANEL FRONTAL DE SUBVI DE GUARDADO DE PROGRAMA

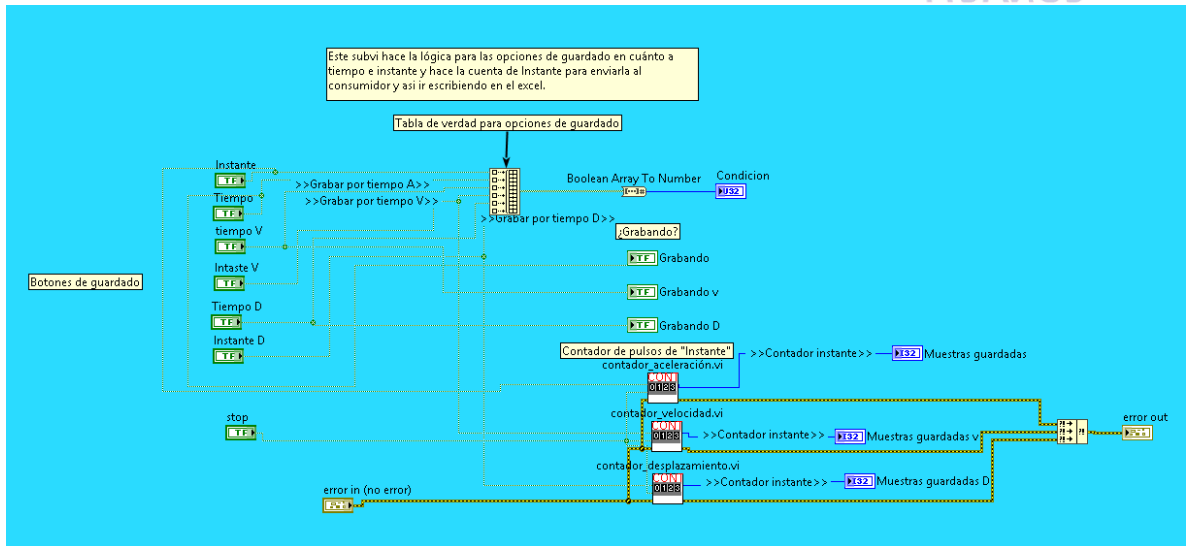


ILUSTRACIÓN 31 DIAGRAMA DE BLOQUES DE PROGRAMA DE GUARDADO

Dentro de este Subvi hay otro Subvi que ejecuta la cuenta cuando presionan guardar:

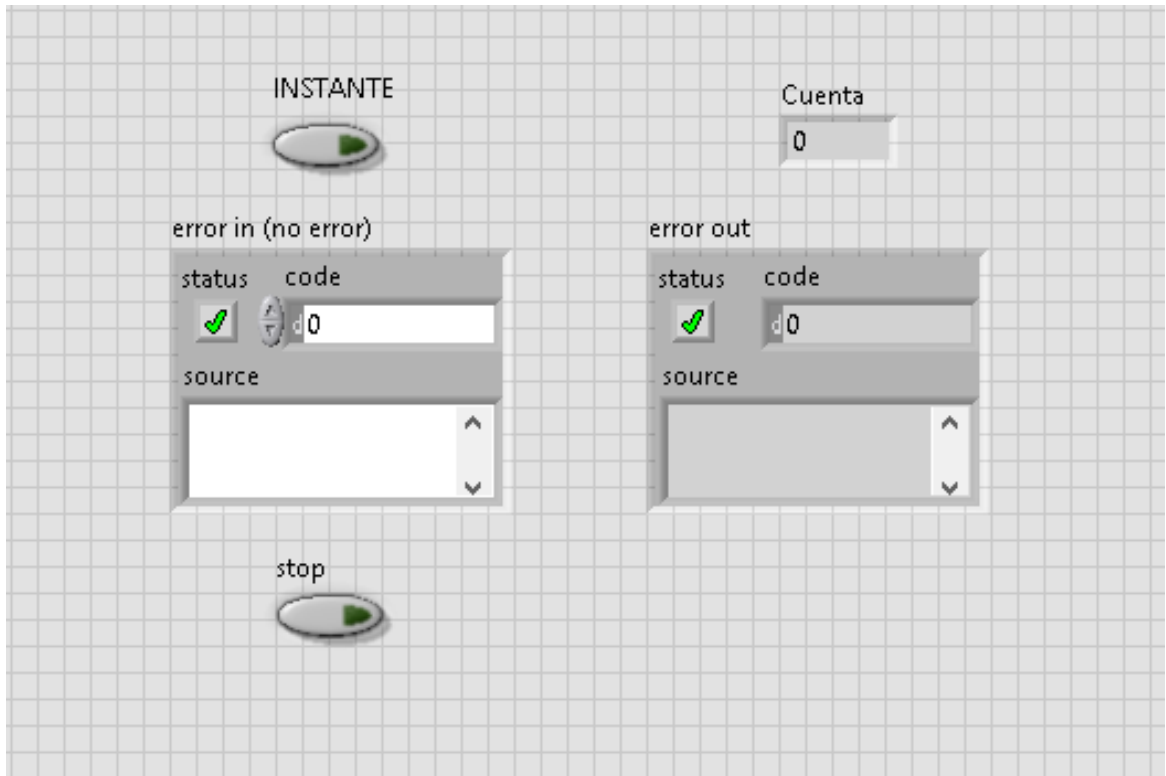


ILUSTRACIÓN 32 PANEL FRONTAL DE SUBVI QUE HACE LA CUENTA DE BOTONES

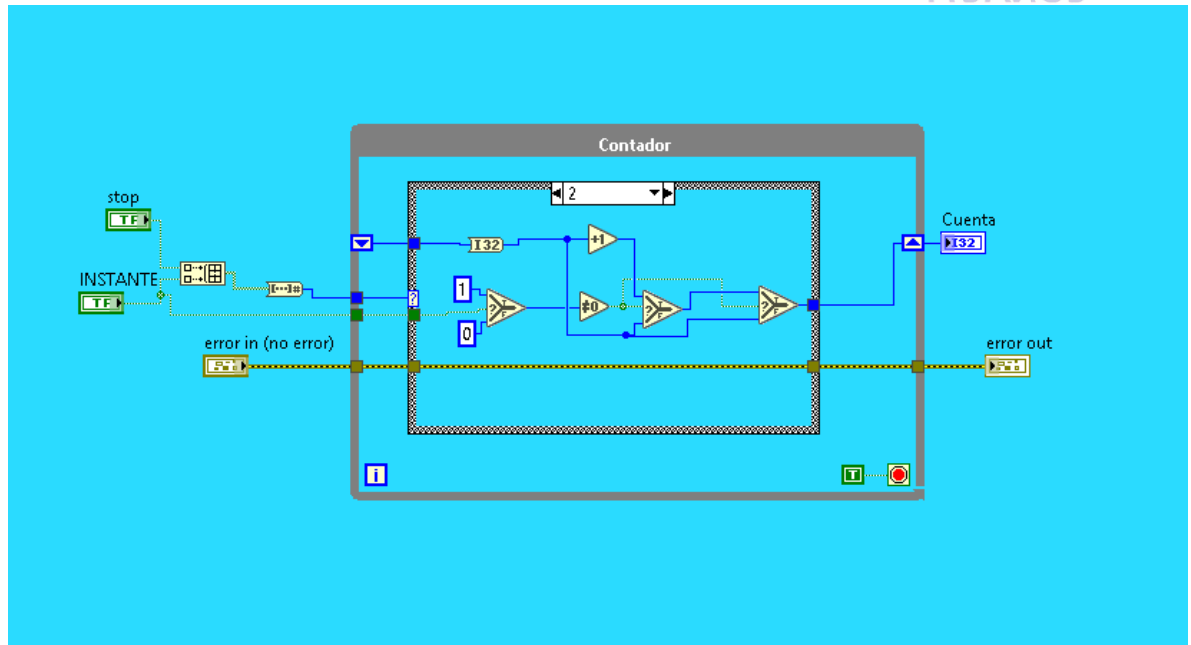


ILUSTRACIÓN 33 DIAGRAMA DE BLOQUES DE SUBVI QUE HACE LA CUENTA DE BOTONES

Teniendo el diagrama a bloques se observa que es una Variable Global Funcional, misma que se usa para hacer cuentas durante cualquier parte del programa. Este código cuenta las veces que se presiona un botón y se reinicia cuando se termina el programa.

6. Array 1D: Es el arreglo de los datos en decibeles para las gráficas de velocidad. Proviene de esta parte del código

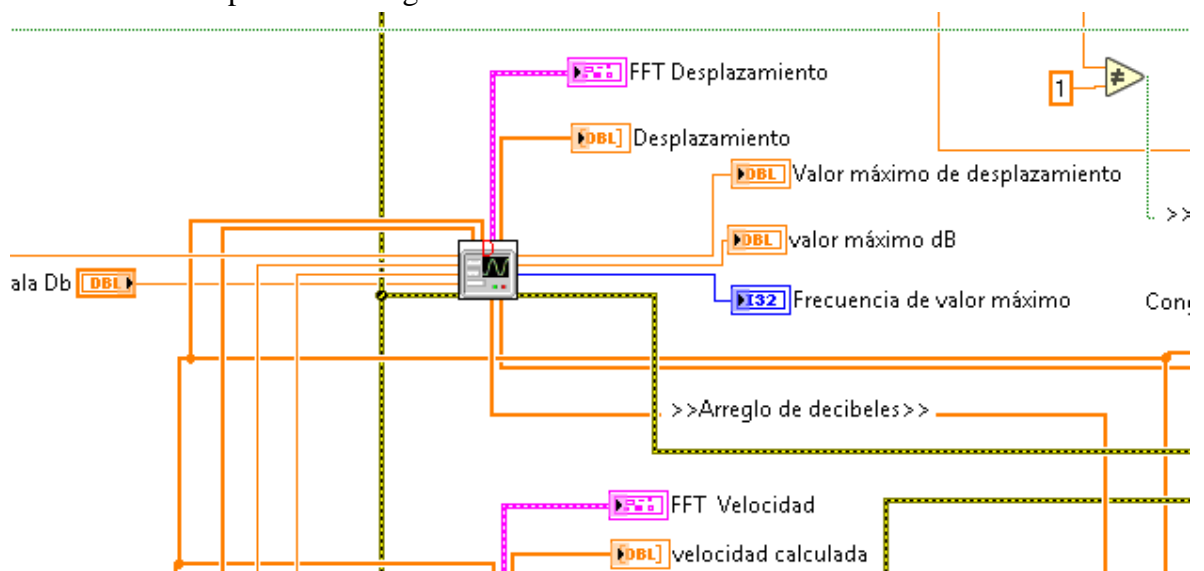


ILUSTRACIÓN 34 SECCIÓN DEL PROGRAMA DONDE SE EJECUTA EL SUBVI DE VELOCIDAD

Y sale del Subvi llamado Velocidad:

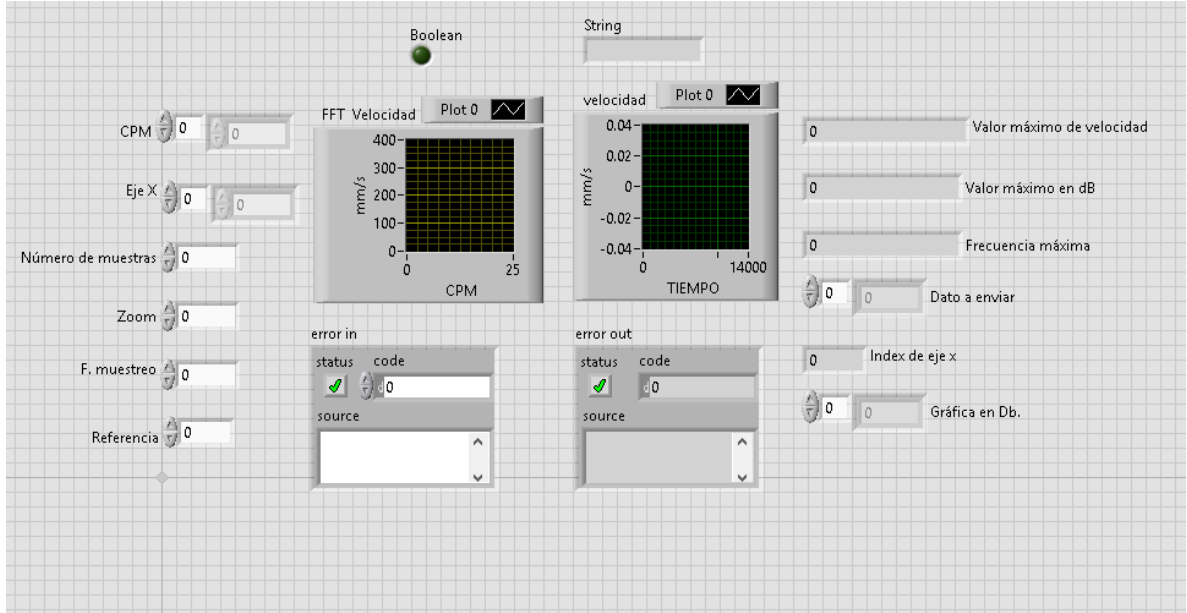


ILUSTRACIÓN 35 PANEL FRONTAL DE SUBVI DE VELOCIDAD

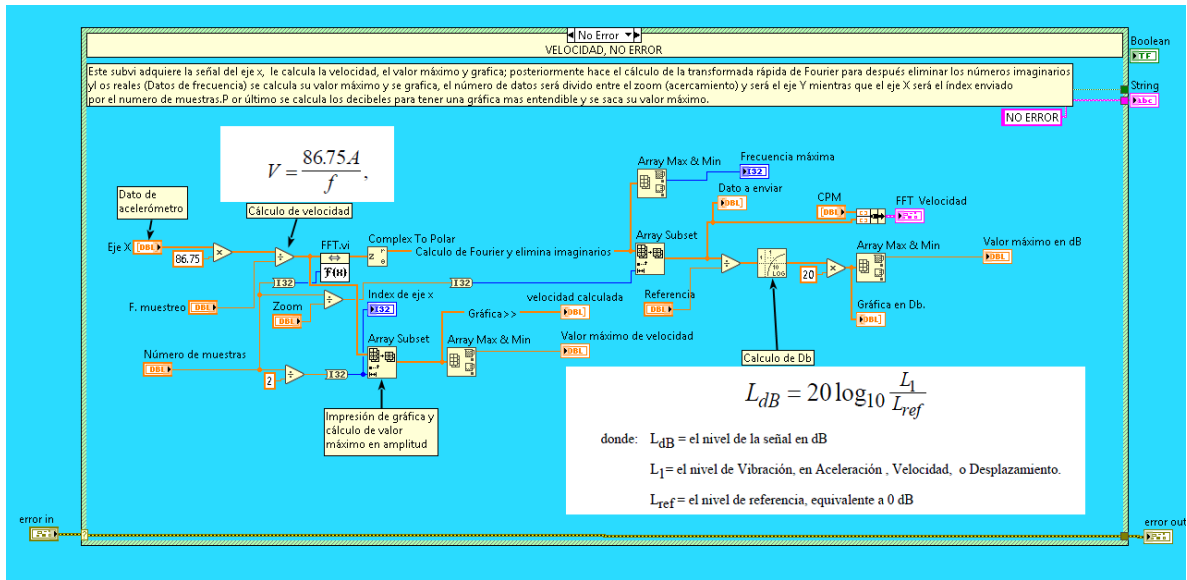


ILUSTRACIÓN 36 DIAGRAMA DE BLOQUES DE SUBVI DE VELOCIDAD

Este Subvi ejecuta la conversión de los datos del acelerómetro los cuales están en el dominio del tiempo y los pasa al dominio de la frecuencia gracias a la Transformada

rápida de Fourier para posteriormente hacer las gráficas en frecuencia, dar las frecuencias altas y bajas así como también hacer el cálculo de decibeles y mostrar las gráficas y valores de velocidades.

Es importante recalcar que el Subvi cuenta con la estructura estándar para la ejecución, la cual es un Case de error, mientras no exista error que ejecute el programa y cuando exista error que se detenga y que mande el estatus que dicho Subvi está fallando junto con el código del error.

7. Array 1D: Es el arreglo de los datos en decibeles para las gráficas de desplazamiento. Proviene de esta sección de código

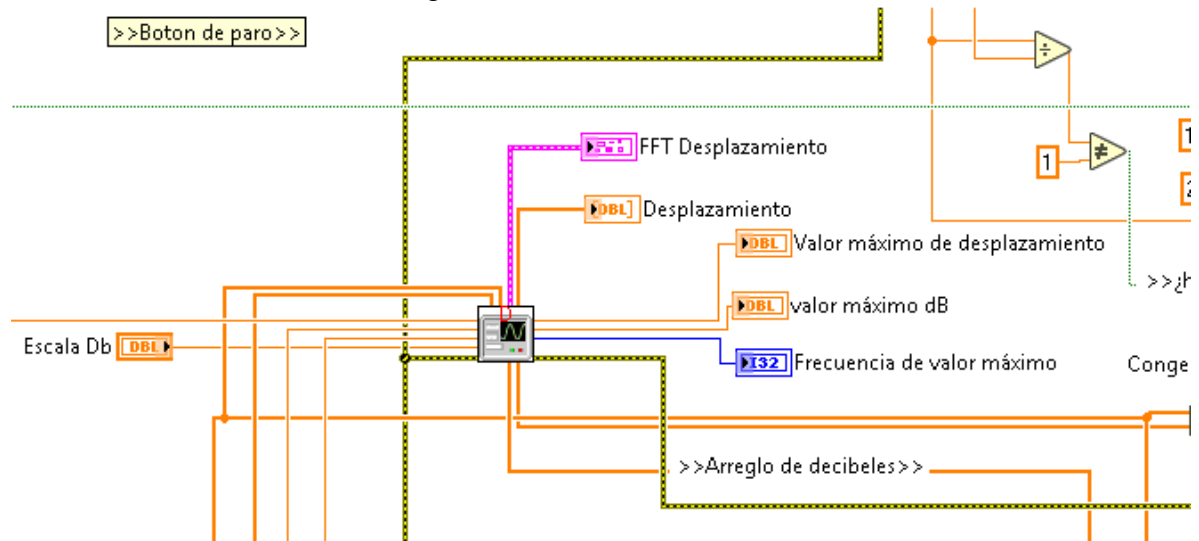


ILUSTRACIÓN 37 SECCIÓN DE PROGRAMA DONDE SE EJECUTA SUBVI DE DESPLAZAMIENTO

Y se calcula en el Subvi de Desplazamiento:

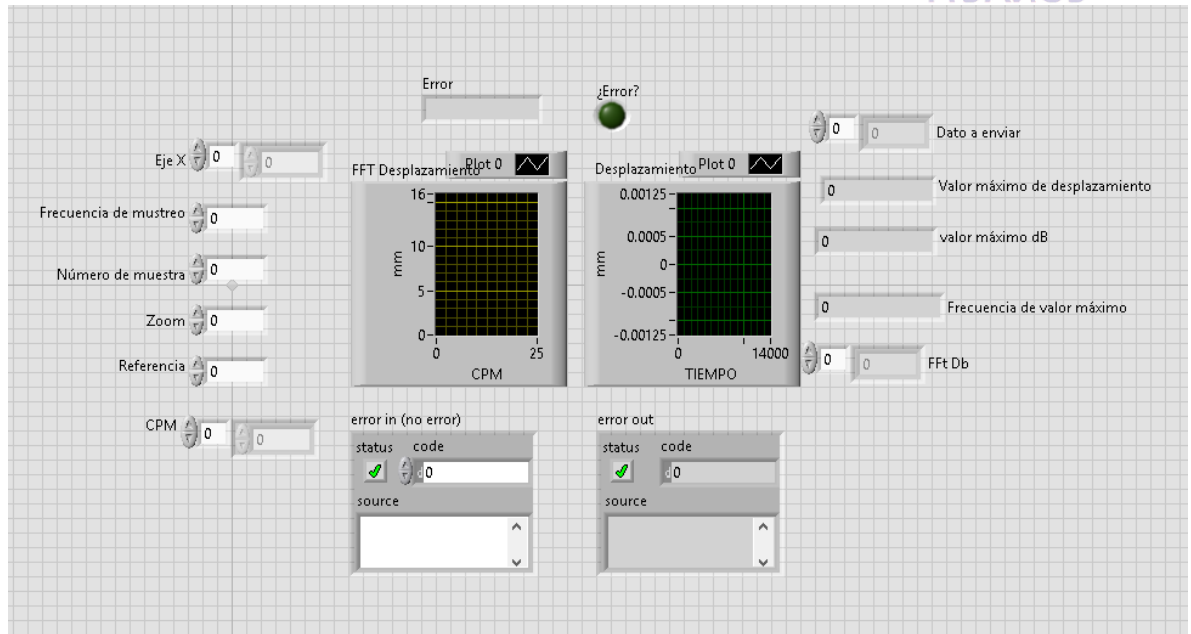


ILUSTRACIÓN 38 PANEL FRONTAL DE SUBVI DE DESPLAZAMIENTO

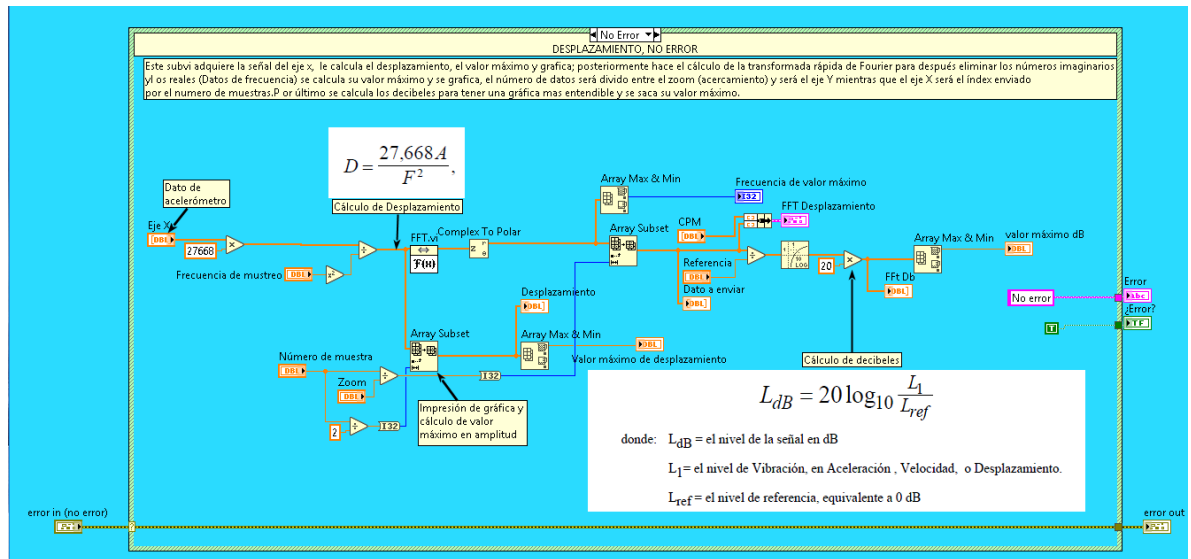


ILUSTRACIÓN 39 DIAGRAMA DE BLOQUES DE SUBVI DE DESPLAZAMIENTO

Este Subvi ejecuta la conversión de los datos del acelerómetro los cuales están en el dominio del tiempo y los pasa al dominio de la frecuencia gracias a la Transformada rápida de Fourier para posteriormente hacer las gráficas en frecuencia, dar las frecuencias altas y bajas así como también hacer el cálculo de decibels y mostrar las gráficas y valores de desplazamiento.

Es importante recalcar que el Subvi cuenta con la estructura estándar para la ejecución, la cual es un Case de error, mientras no exista error que ejecute el programa y cuando exista error que se detenga y que mande el estatus que dicho Subvi está fallando junto con el código del error.

El código del loop del Productor nos queda de la siguiente manera, empezando de arriba abajo y de izquierda a derecha:

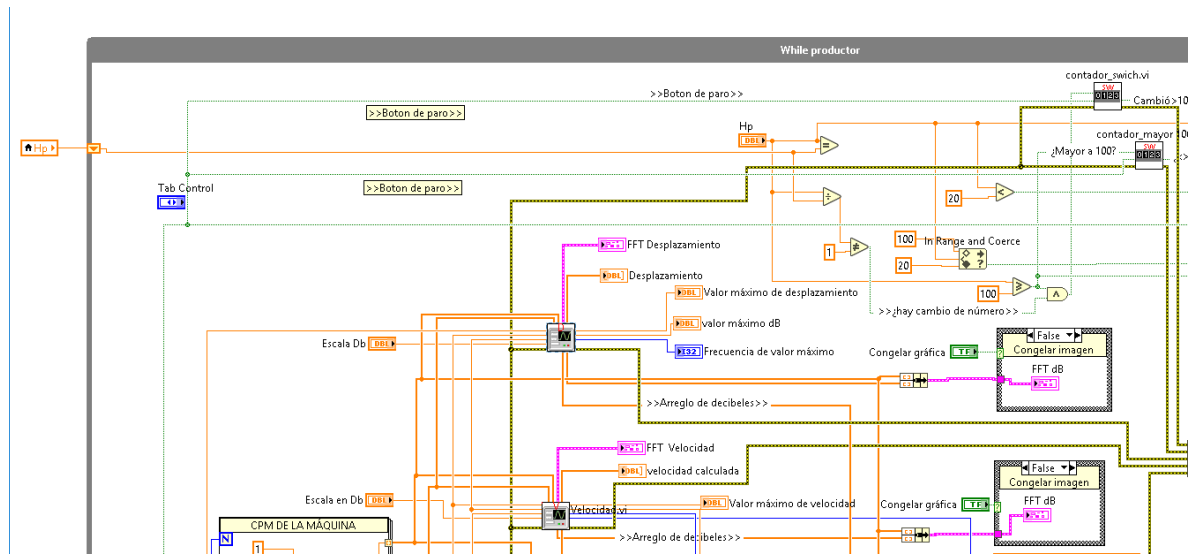


ILUSTRACIÓN 40 PRODUCTOR

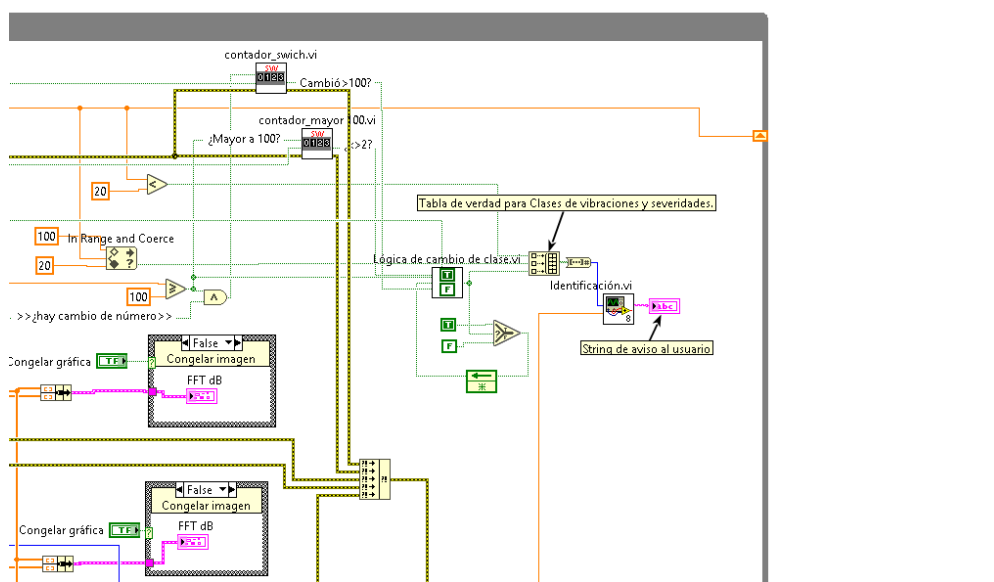


ILUSTRACIÓN 41 PRODUCTOR

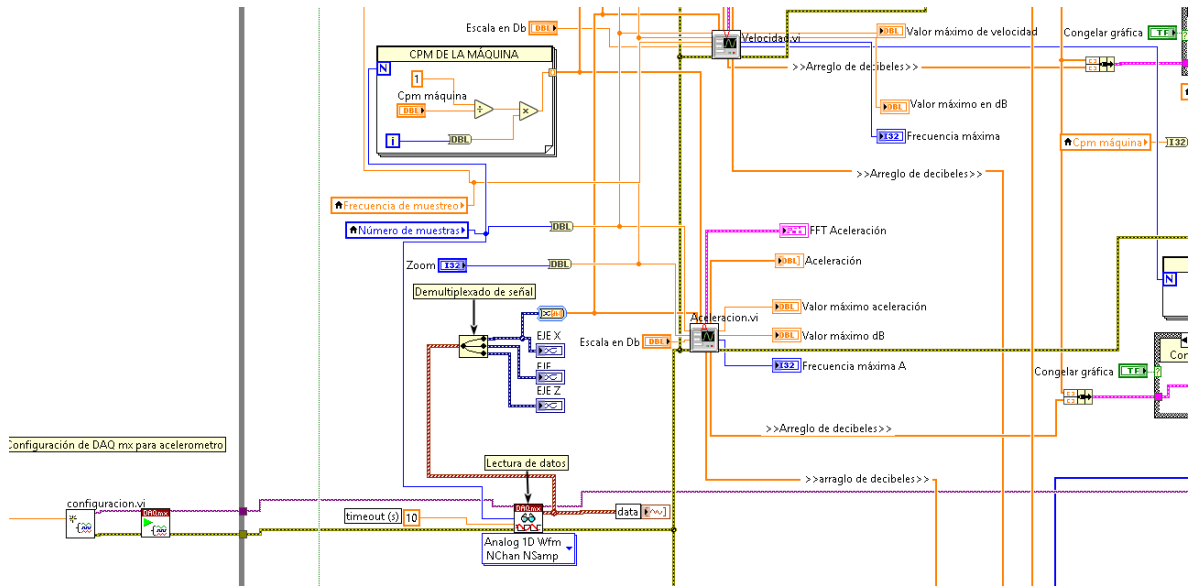


ILUSTRACIÓN 42 PRODUCTOR

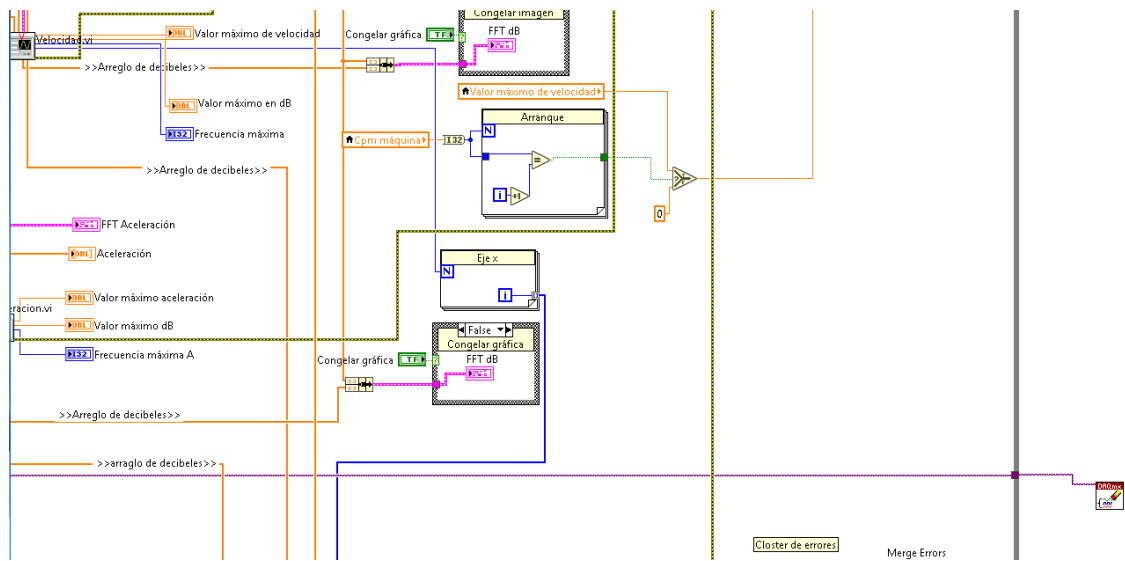


ILUSTRACIÓN 43 PRODUCTOR

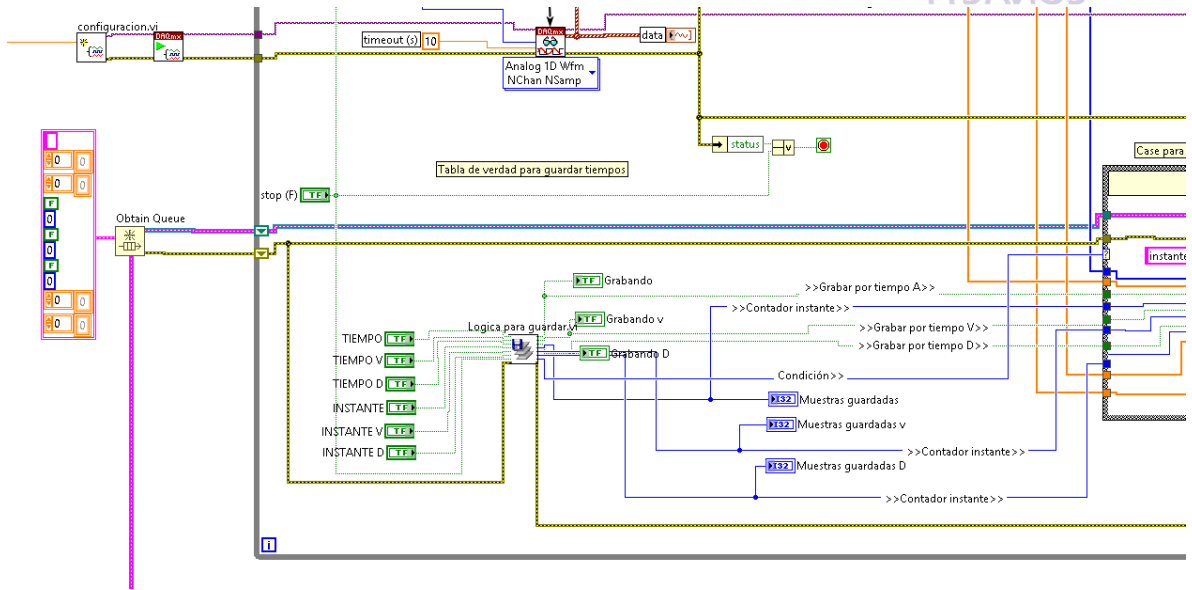


ILUSTRACIÓN 44 PRODUCTOR

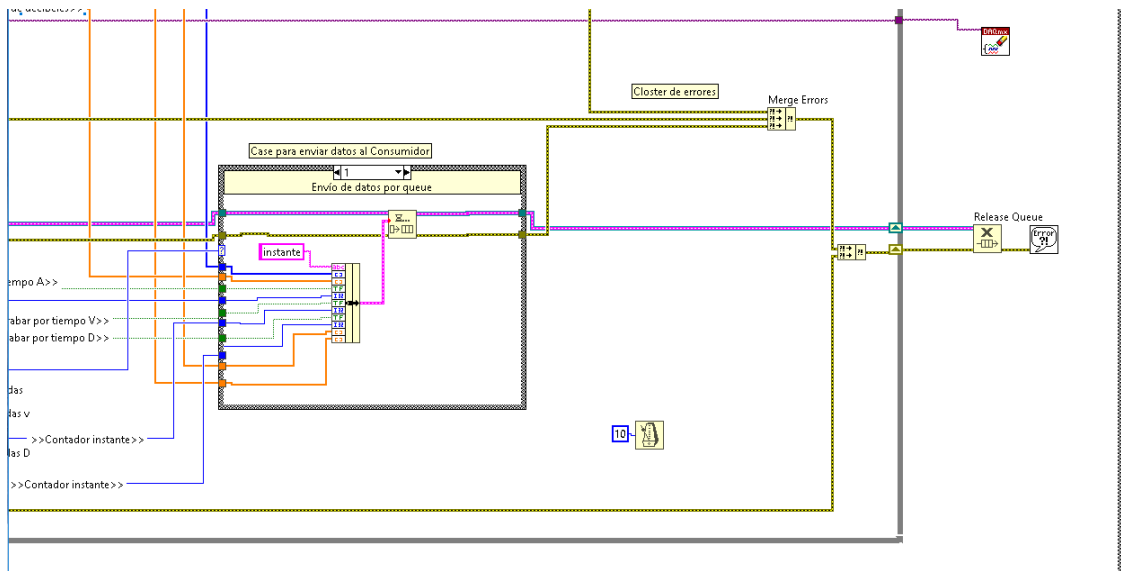


ILUSTRACIÓN 45 PRODUCTOR

Consumidor

El consumidor recibe el dato del productor, como se sabe le mandan los datos en un Cluster entonces para analizar se tiene que separar los datos y eso se hace con un “Unbundle”. Esta función separa los datos en el mismo orden en que se concatenaron.

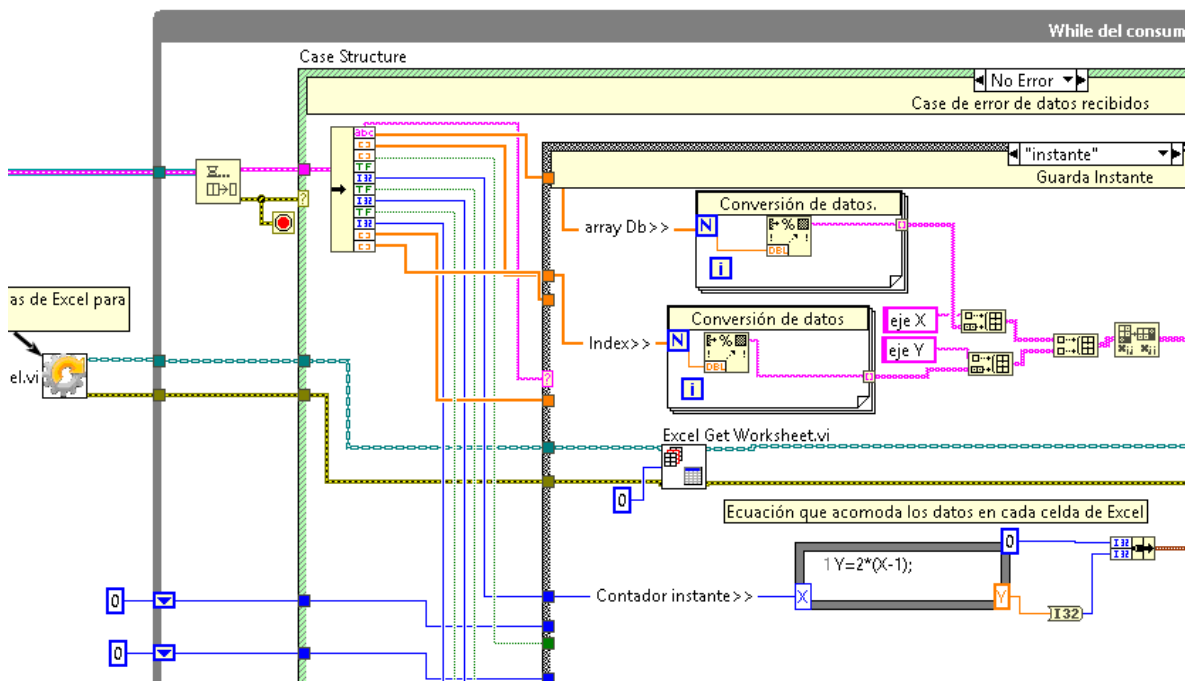


ILUSTRACIÓN 46 CONSUMIDOR, DEMULTIPLEXADO DE SEÑAL

Se observa que viene el cable de “Queue” y entra al While del consumidor mismo que con un “Unbundle” separa los datos de entrada para que se puedan usar en diferentes partes.

Se observa que dentro del While del Consumidor está un Case de error, mismo que si no hay error ejecuta todo el programa y si existe error, se detiene. Posteriormente viene un Case que es la máquina de estados que la gobierna el String enviado por el Productor. Cuenta con el número de casos anteriormente expuestos.

Se ilustra un Case, la programación es la misma en los demás, solo varía el arreglo 1D que se va a tomar dependiendo si es aceleración, velocidad o desplazamiento.

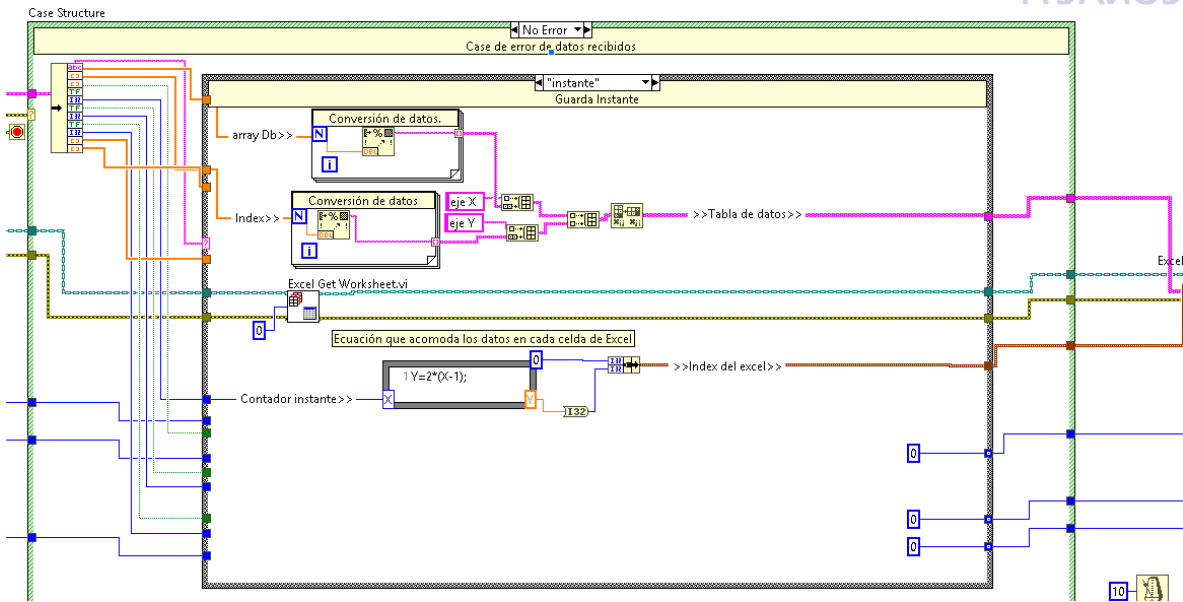


ILUSTRACIÓN 47 CASE DE NO ERROR

Como se observa en la ilustración, se toma el arreglo generado por el “For” para que sea el eje X de la gráfica y como eje Y se toman los datos de desplazamiento; ambos entran a un For para que dato a dato se conviertan a String y sean enviados a concatenarse con un String constante el cual dirá el eje en el cual se va a ingresar el dato, posteriormente se hace un arreglo de String y se genera una tabla de los datos para ser enviada al módulo de Excel.

Por otro lado en el bloque de “Excel Get Work” se coloca el número de la hoja de Excel en el cual se va a escribir los datos y es enviado al módulo de Excel.

Para finalizar se hizo una ecuación en un “Formula node” misma que tiene la función de ir colocando cada dato en una celda de Excel, es como un corrimiento de las celdas.

En seguida se muestra cada caso de la máquina de estados.

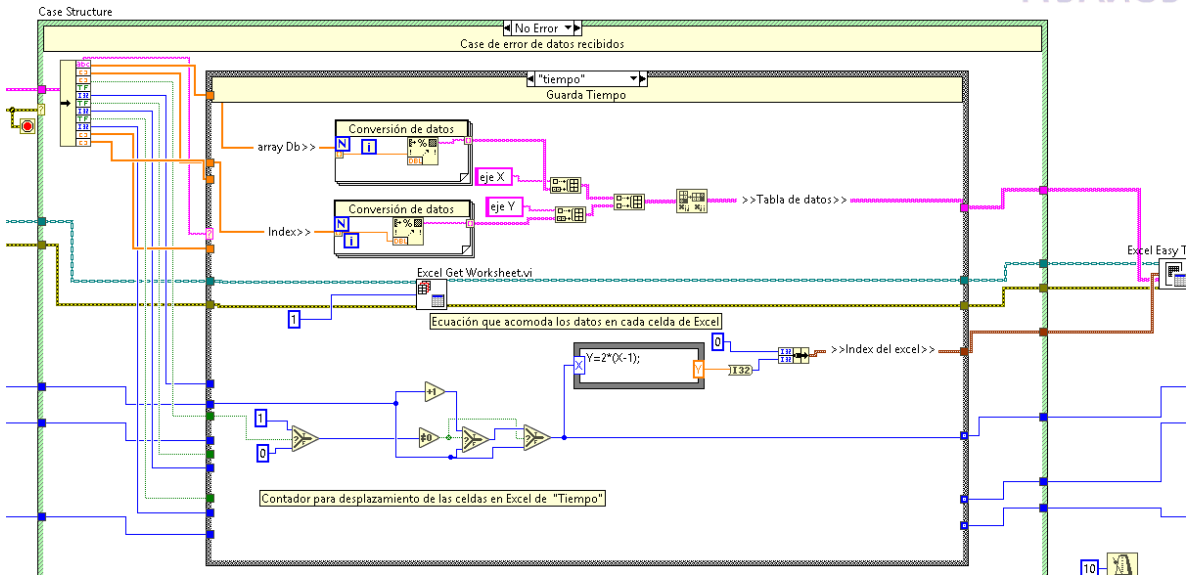


ILUSTRACIÓN 48 PROGRAMACIÓN DE MÁQUINA DE ESTADOS

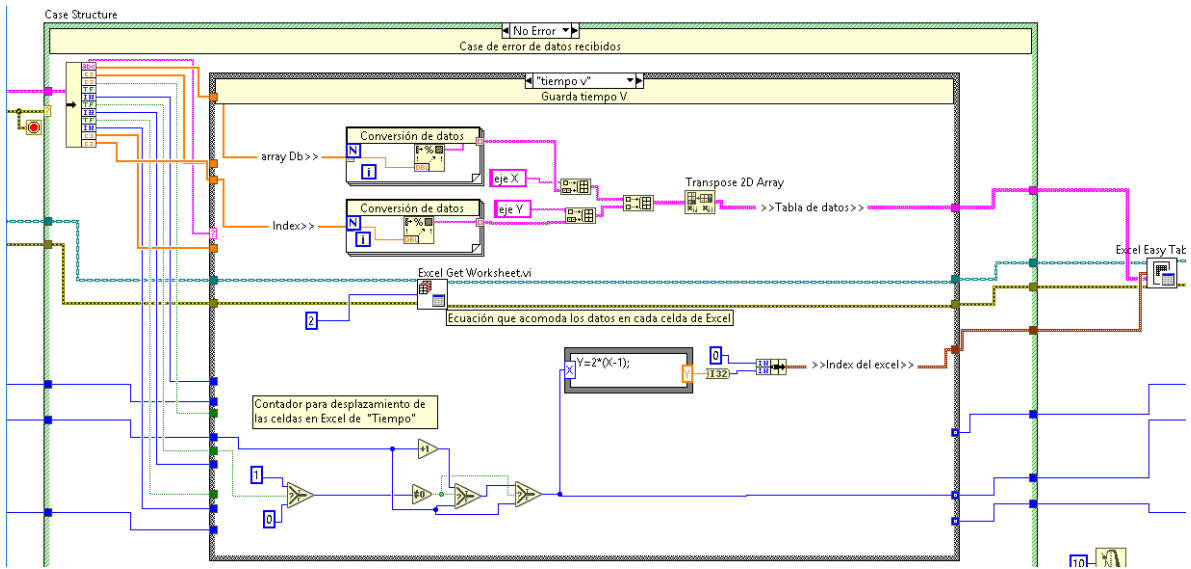


ILUSTRACIÓN 49 PROGRAMACIÓN DE GUARDADO DE TIEMPO EN VELOCIDAD

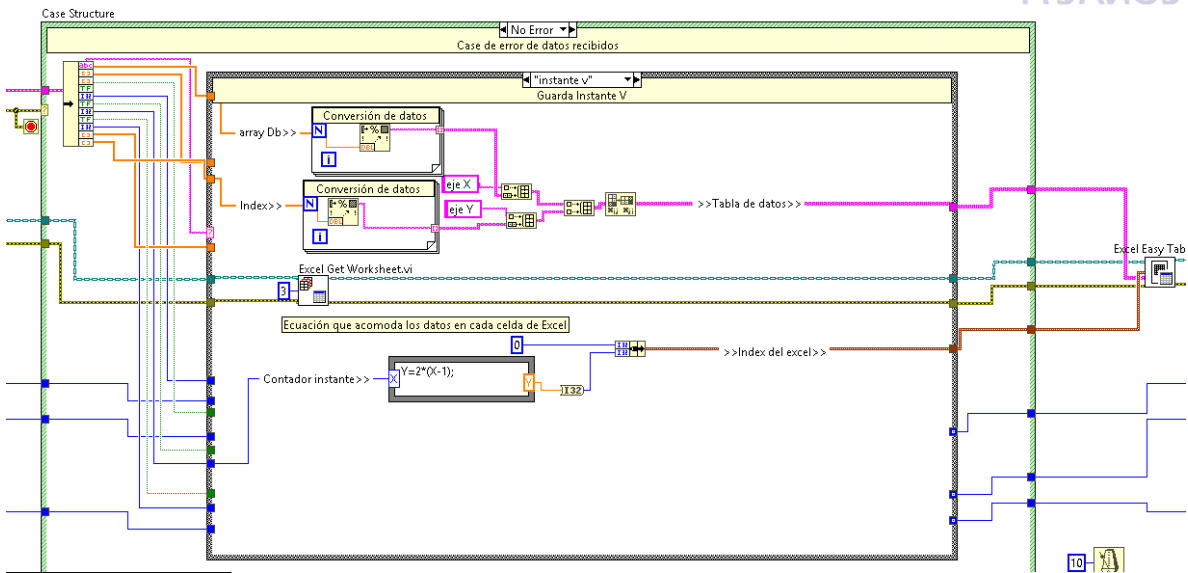


ILUSTRACIÓN 50 PROGRAMACIÓN DE GUARDADO DE INSTANTE EN VELOCIDAD

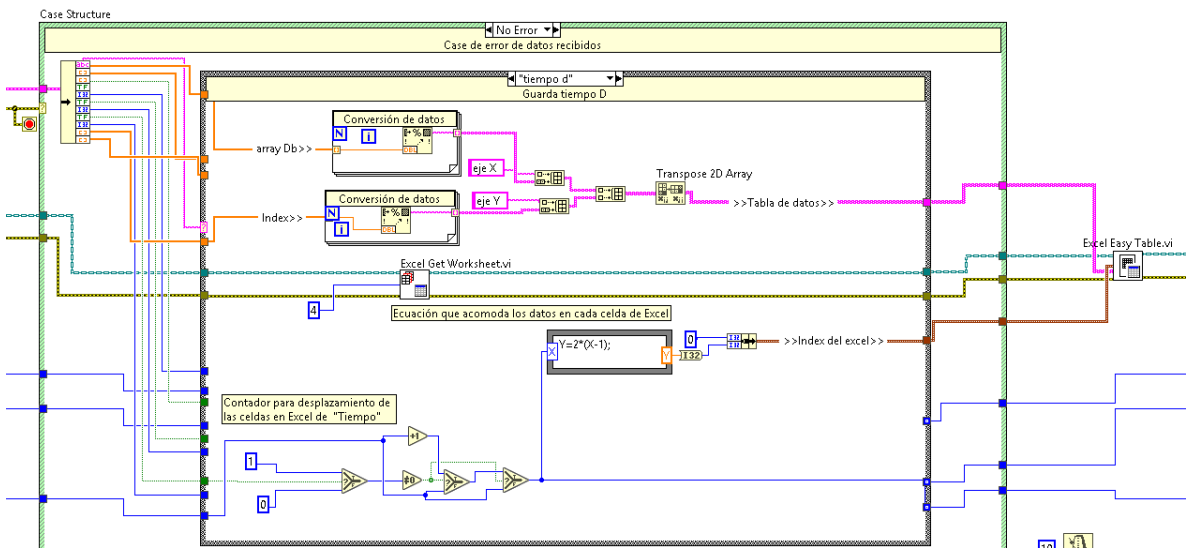


ILUSTRACIÓN 51 PROGRAMACIÓN DE GUARDADO DE TIEMPO EN DESPLAZAMIENTO

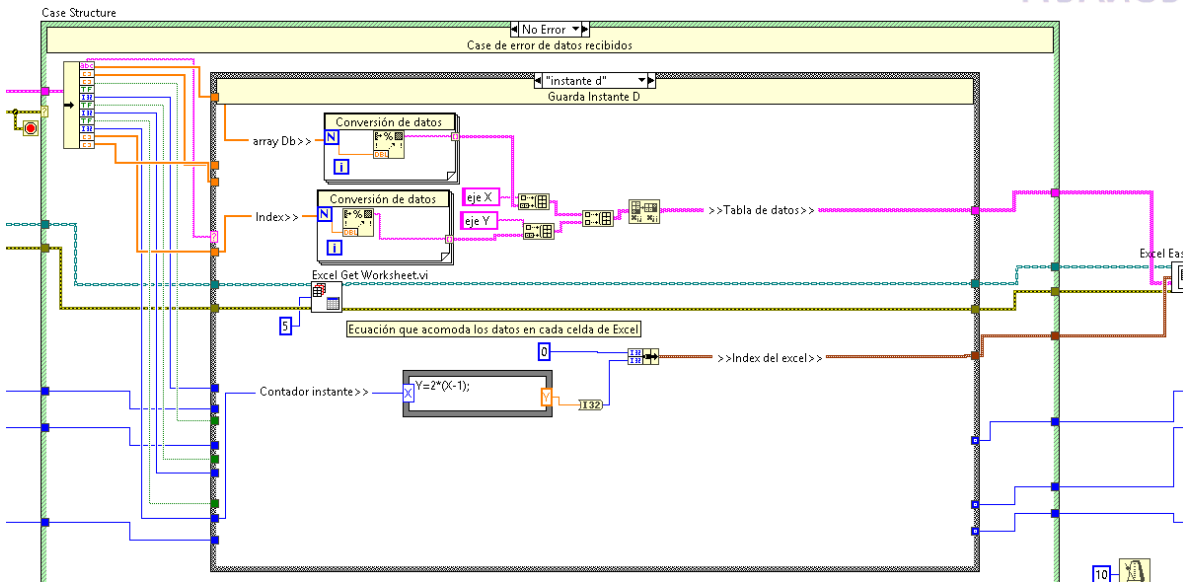


ILUSTRACIÓN 52 PROGRAMACIÓN DE GUARDADO DE INSTANTE EN DESPLAZAMIENTO

Para hacer la configuración de las hojas de cálculo de Excel se usó un Subvi que se muestra ahora:

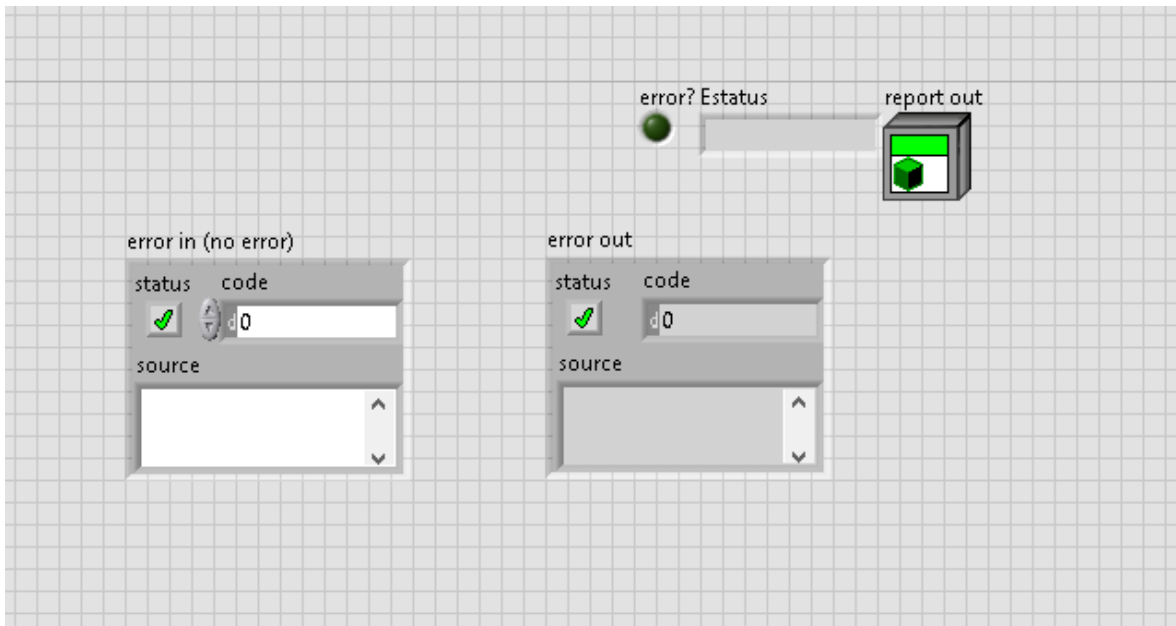


ILUSTRACIÓN 53 PANEL FRONTAL DE SUBVI DE CONFIGURACIÓN DE EXCEL

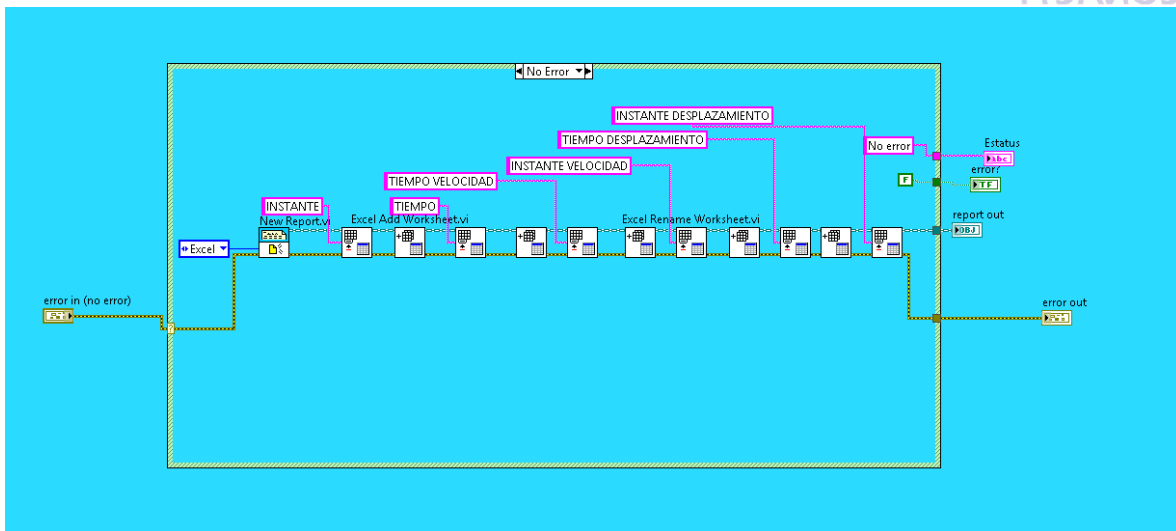


ILUSTRACIÓN 54 DIAGRAMA DE BLOQUES DE SUBVI DE CONFIGURACIÓN DE EXCEL

Y para terminar la configuración es el siguiente código:

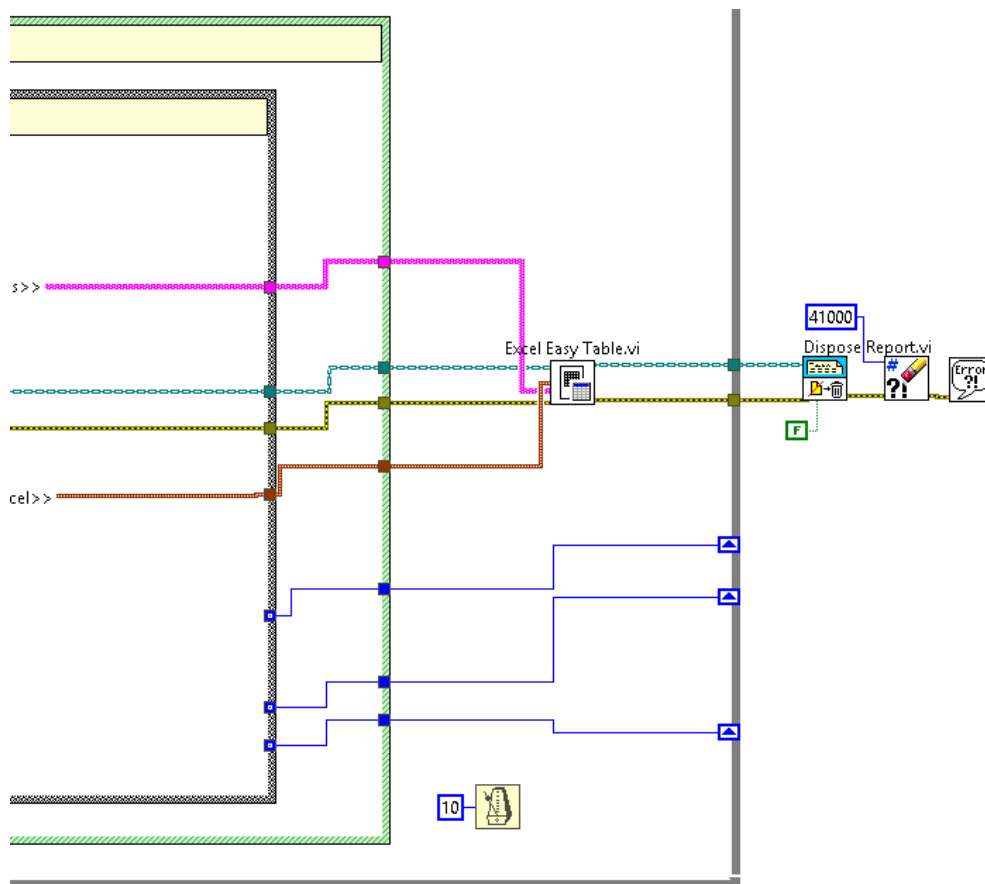


ILUSTRACIÓN 55 FIN DE CONFIGURACION DE EXCEL

Se observa que tanto en el Productor como en el Consumidor se usa un “Elapsed Time Delay” de 10 ms, esto se calculó tomando el tiempo que tarda cada While. De esta manera nos cercioramos que no tarda más de lo que debería y lo hará lo más rápido de manera segura.

Arranque de programa

Teniendo en cuenta el panel frontal y el algoritmo de la arquitectura de programación, se va a mostrar el inicio del programa el cual tiene ciertas condiciones:

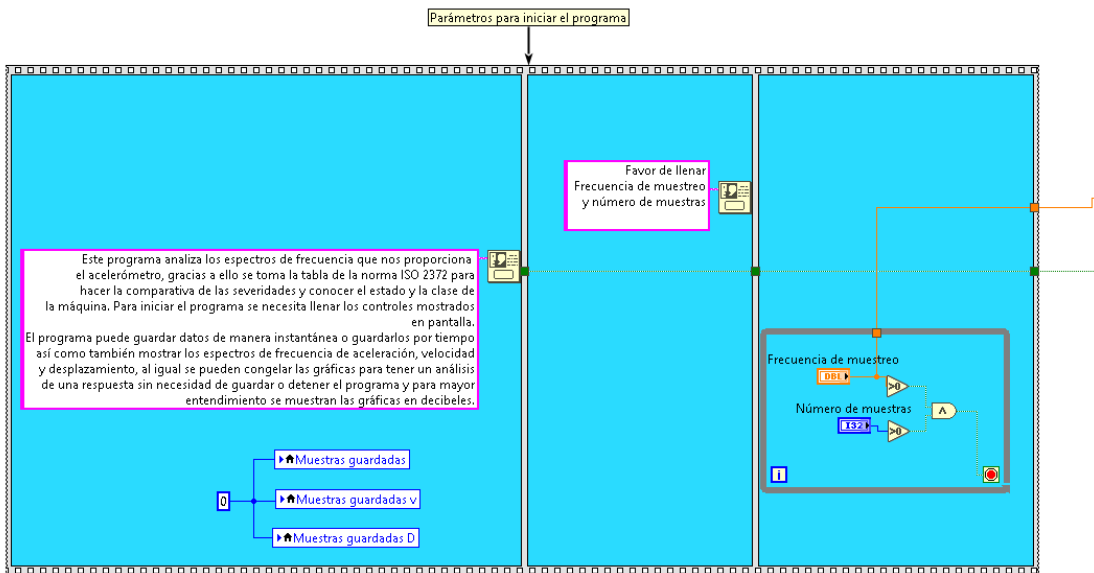


ILUSTRACIÓN 56 CONDICIONES DE ARRANQUE

1. Las variables locales se inician en cero y nos dice lo que el programa es capaz de hacer.
2. Te fuerza a que llenes los datos de frecuencia de muestreo y número de muestras.
3. Se va a estar preguntando si ya se metió valores en las entradas que pidió, hará esto de manera infinita hasta que se pongan.

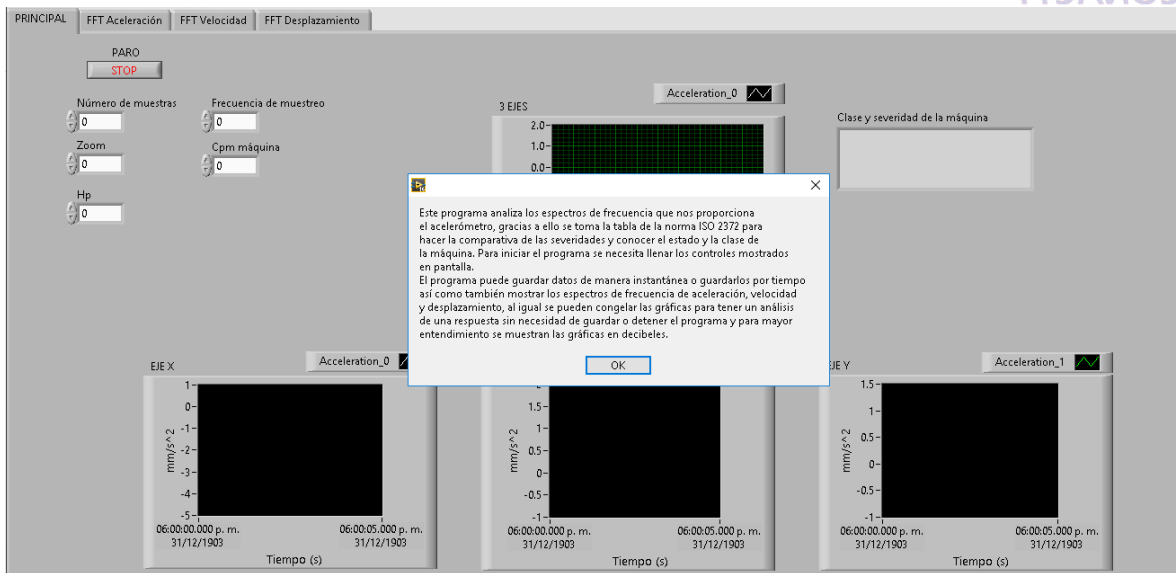


ILUSTRACIÓN 57 CONDICIÓN DE ARRANQUE, EXPLICACIÓN DEL PROGRAMA

No avanzará hasta que se hallan llenado las casillas.

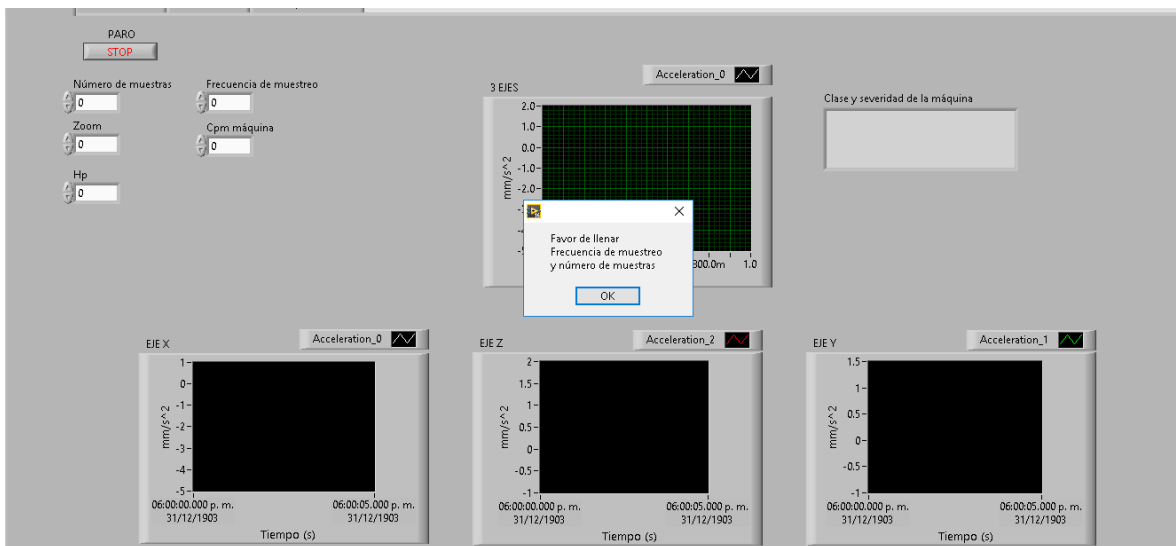


ILUSTRACIÓN 58 CONDICIÓN DE ARRANQUE, LLENADO DE DATOS.

ENTREGABLES

1. Estructura principal del algoritmo de programación.
2. Programa con cálculos de velocidad y desplazamiento aplicados en frecuencia y en decibeles.
3. Programa que ejecute la comparativa.
4. Interfaz que muestre la clase y el rango de severidad de la máquina.
5. Datos guardados en hojas de cálculo de Excel.
6. Documento de Tesis.

FACTIBILIDAD DEL PROYECTO

✓ Económica:

1. Apoyo económico obtenido de proyectos del área de energía tales como:
 - Diablo Instrumentado.
 - Inspector de tuberías.

✓ Recursos humanos:

1. Especialistas con conocimiento en LabVIEW y vibraciones.

✓ Infraestructura:

1. Computadora.
2. Tarjeta de adquisición de datos.
3. Acelerómetro.
4. Generador de señales.
5. Shaker.

RESULTADOS

La ejecución del algoritmo nos da una serie de gráficas con respuestas dadas por el acelerómetro y nos genera el estatus del sistema, así como también la hoja de Excel que nos genera al guardar los datos. Teniendo estos datos se puede saber el estatus del sistema y con los datos guardados se puede dar un análisis posterior.

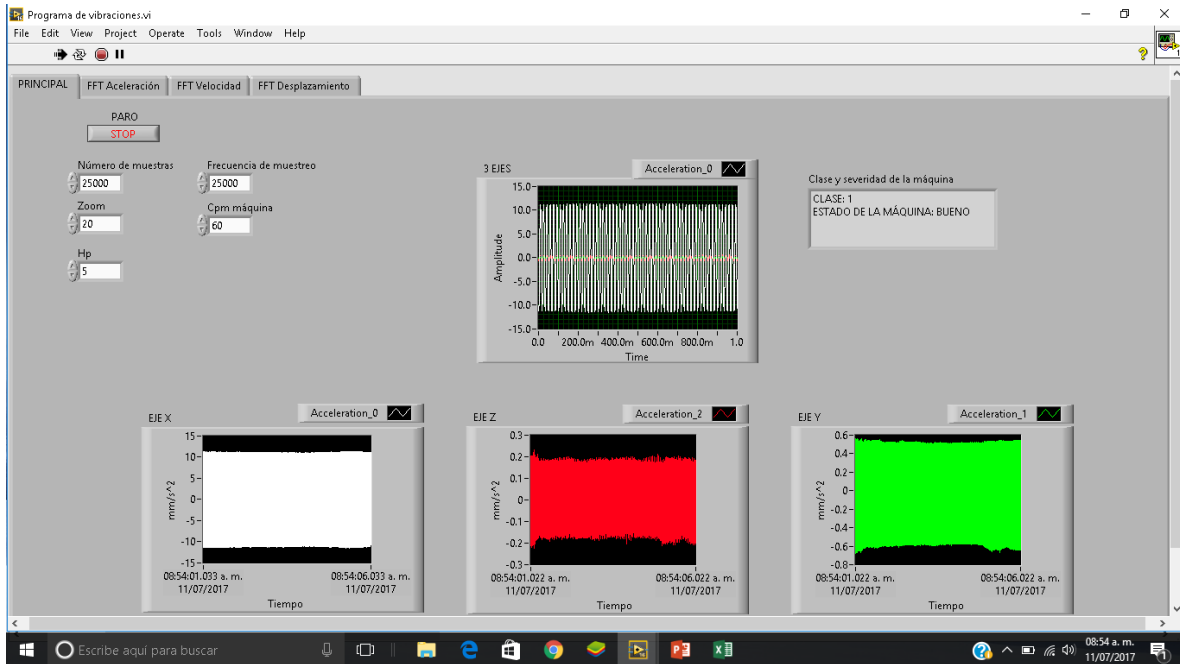


ILUSTRACIÓN 59 PANEL FRONTAL DEL PROGRAMA EJECUTÁNDOSE.

Se puede observar que en el indicador tipo String el cual nos proporción la clase y el estado de la máquina; éste mismo da el estatus del sisma.

Las gráficas nos muestran el estado del sistema en el dominio del tiempo, esa señal viene directamente del acelerómetro. Cabe mencionar que son solo para visualización puesto que no tiene importancia alguna.

Se tienen los controles necesarios para que el programa funcione, son de suma importancia para hacer a adquisición de datos, zoom de gráficas y clasificación de máquinas.

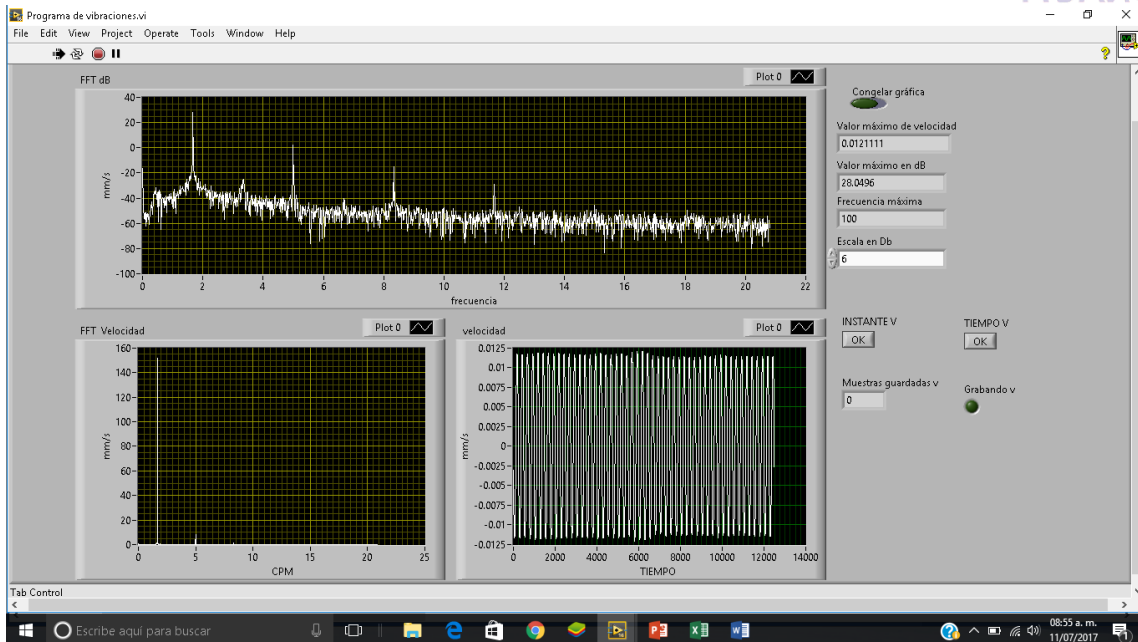


ILUSTRACIÓN 60 PESTAÑA DE VELOCIDAD

Esta es la ventana más importante ya que en la velocidad viene implícita la posición y la frecuencia, por lo tanto nos da el comportamiento más importante de severidad del sistema. Partiendo de estas gráficas se hacen los análisis pertinentes, es por ello que es la más importante.

Ahora se muestra la gráfica de aceleración, en esta ilustración se muestran las gráficas en decibeles y frecuencias del sistema, también nos da los valores máximos. Este dato es importante para cuando se tiene una máquina rotativa con altas revoluciones, entonces, es perfecta para dicho análisis.

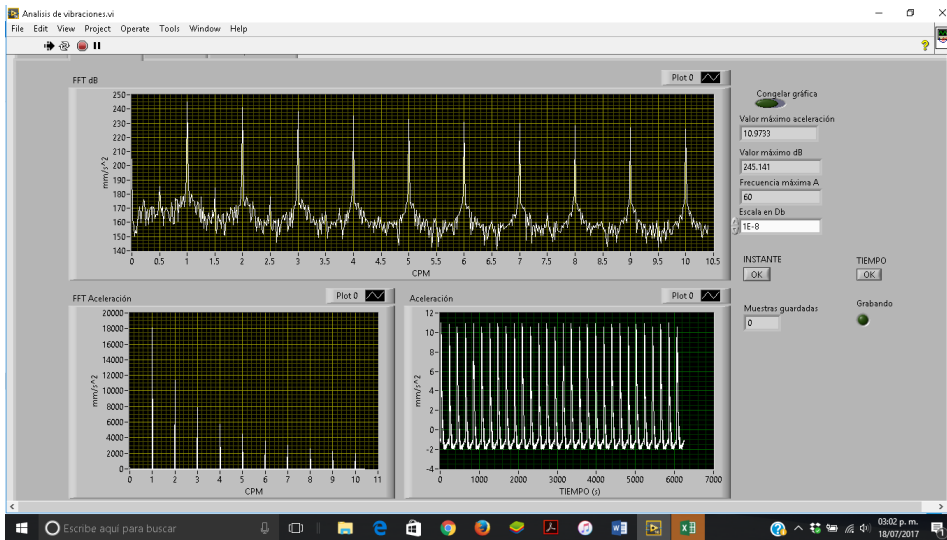


ILUSTRACIÓN 61 GRÁFICA DE ACCELERACIÓN

Por último en gráficas, viene el desplazamiento que se ocupa de bajas revoluciones y severidades fuertes del sistema, se manejan las mismas gráficas que en las anteriores. Como se ha mencionado a lo largo de la Tesis, será importante para cierto tipo de máquinas.

	TIEMPO DESPLAZAMIENTO		INSTANTE DESPLAZAMIENTO	
	aje X	aje Y	aje X	aje Y
1	0	0.006501	0	0.006501
2	1	0.000128	1	0.000064
3	2	0.000089	2	0.000058
4	3	0.000021	3	0.000044
5	4	0.000006	4	0.000049
6	5	0.000038	5	0.000046
7	6	0.000059	6	0.000017
8	7	0.000067	7	0.00002
9	8	0.000093	8	0.000164
10	9	0.000074	9	0.000046
11	10	0.000095	10	0.000066
12	11	0.000153	11	0.000105
13	12	0.000116	12	0.000115
14	13	0.0001	13	0.000037
15	14	0.000141	14	0.000083
16	15	0.000018	15	0.000061
17	16	0.000143	16	0.000013
18	17	0.00006	17	0.00008
19	18	0.000031	18	0.00002
20	19	0.000092	19	0.000167
21	20	0.000274	20	0.00017
22	21	0.000271	21	0.000211

Ahora se tiene la hoja de cálculo de Excel que son los datos que el usuario guarda para un análisis a posteriori. Esta sección es de suma importancia ya que en cualquier momento se puede hacer un análisis a detalle de las diferentes partes del sistema y en diferentes tiempos.

CONCLUSIÓN

Es importante aplicar los algoritmos realizados en LabVIEW ya que nos ayuda a reducir pérdidas económicas, monetarias y sobre todo se tiene mayor seguridad con los empleados. Teniendo en cuenta lo anterior se mantiene un programa riguroso de mantenimiento preventivo y correctivo que nos ayuda a saber con antelación la falla del sistema y ¿acaso no es importante predecir una futura falla y conocer el estado del sistema en un futuro?

Desarrollando los algoritmos y con el paso del tiempo se han estado mejorando para dar un análisis certero gracias a una buena elección de los sensores y la adquisición de datos. El procesamiento es parte fundamental es por ello que las bibliografías de estudio deben ser de alta calidad para tener mayor fidelidad en el cálculo de los datos para las gráficas e interpretarlas.

El costo computacional no es tan elevado pero el costo en componentes si lo es, pero va de la mano con la fidelidad, precisión y exactitud el que un componente sea de una compañía de alto prestigio ya que se aseguras que no fallará el sistema.

REFERENCIAS

- i. Analysis, S. A. (n.d.). *No Title*.
- ii. De, D., & Academia, R. (1830). Introducción . Señales y análisis de Fourier, 1–21.
- iii. Departamento de Ingeniería Mecánica, E. y de M. (2007). Vibraciones en máquinas: Mantenimiento predictivo. *Elementos de Máquinas Y Vibraciones.*, 1–54. <https://doi.org/10.1017/CBO9781107415324.004>
- iv. Eléctricos, P., & Vibración, Y. M. D. E. (2009). Detectando fallas en rodamientos utilizando métodos de prueba eléctricos y mecánicos de vibración., 1–14.
- v. Espino, P., Rendón, J., Valdez, J., Davizón, Y., Nuñez, J., & Hinojosa, C. (2015). Design of a Labview Application for the Teaching of Mechanical Vibrations in a Mechatronics Engineering Program. *IEEE Latin America Transactions*, 13(7), 2347–2355. <https://doi.org/10.1109/TLA.2015.7273797>
- vi. Galindo, G. F. (2010). Construcción y Validación de un Sensor de Vibraciones usando un Sistema Micro-Electrico-Mecanico (MEMS), 67.
- vii. Iso, N. (1800). Norma ISO 10816, 1–4.
- viii. Iso 2372 – Vibration Severity. (n.d.), 2372.
- ix. Karnovsky, I. A., & Lebed, E. (2016). *Theory of vibration protection. Theory of Vibration Protection*. <https://doi.org/10.1007/978-3-319-28020-2>
- x. Lee, J., & Chang, J. (2017). Analysis of the vibration characteristics of coaxial magnetic gear. *IEEE CEFC 2016 - 17th Biennial Conference on Electromagnetic Field Computation*, 53(6). <https://doi.org/10.1109/CEFC.2016.7816237>
- xi. Mosquera, G., Piedra, M., & Armas, R. (2001). Las vibraciones mecánicas y su aplicación al mantenimiento predictivo. *Instituto Superior de Investigación Y Desarrollo*, 205. <https://doi.org/10.1177/004057368303900411>
- xii. Nacional, C., En, M., En, C., Opci, M. C. O. N., & Digitales, E. N. S. (n.d.). Norma Angélica Benítez González.
- xiii. National Instruments. (2004). Sound and Vibration Toolkit User Manual, (322194), 1–6.
- xiv. Ocampo, J. R. (2013). VIBRACIÓN EN MÁQUINAS ROTATORIAS, 14–31.
- xv. Pozo, P. P. (2013). En Baja Frecuencia De Motor Eléctrico.
- xvi. Spending, T. P. (2008). Mp Le Mp Le. *Construction*, (January), 18–25.
- xvii. Sumathi, S., & Surekha, P. (2007). LabVIEW LabVIEW LabVIEW based based based Advanced Advanced Advanced Instrumentation Instrumentation Instrumentation Systems Systems Systems. Retrieved from <http://download.springer.com.gcu.idm.oclc.org/static/pdf/930/bok%3A978-3-540-48501-8.pdf?originUrl=http://link.springer.com/book/10.1007/978-3-540-48501-8&token2=exp=1457223739~acl=/static/pdf/930/bok%253A978-3-540-48501-8.pdf?originUrl=http%3A%2F%2Flink>.
- xviii. Torres, F., Royo, J., & Rabanaque, G. (2015). Análisis de vibraciones e interpretación de datos. *DIDYF Universidad de Zaragoza*, 1–14. Retrieved from <http://www.guemisa.com/articulo/pdf/vibraciones.pdf>
- xix. Version, C. S., & Number, E. P. (2014). LabVIEW Core 2 Participant Guide, (November).
- xx. Vishwakarma, M., Purohit, R., Harshlata, V., & Rajput, P. (2017). Vibration Analysis & Condition Monitoring for Rotating Machines: A Review. *Materials Today*:

- Proceedings*, 4(2), 2659–2664. <https://doi.org/10.1016/j.matpr.2017.02.140>
- xxi. Wang, W., Li, C., Tollner, E. W., & Rains, G. C. (2012). Development of software for spectral imaging data acquisition using LabVIEW. *Computers and Electronics in Agriculture*, 84, 68–75. <https://doi.org/10.1016/j.compag.2012.02.010>
- xxii. White, G. (2010). Introducción al Análisis de Vibraciones, 1–148.