

UNIVERSIDAD AUTÓNOMA DE SINALOA

FACULTAD DE CIENCIAS DE LA TIERRA Y EL ESPACIO

FACULTAD DE INFORMÁTICA

MAESTRÍA EN CIENCIAS DE LA INFORMACIÓN



**SISTEMA INTELIGENTE DE MONITOREO DE ESTRUCTURAS
CIVILES EN ALTA RESOLUCIÓN**

TESIS

**COMO REQUISITO PARA OBTENER EL GRADO DE
MAESTRO EN CIENCIAS DE LA INFORMACIÓN**

PRESENTA:

ING. JESÚS RICARDO SALAZAR LÓPEZ

DIRECTORES DE TESIS:

DR. JESÚS ROBERTO MILLÁN ALMARAZ

DR. JOSÉ RAMÓN GAXIOLA CAMACHO

CULIACÁN, SINALOA. FEBRERO DE 2020

Agradecimientos

A mi familia

Por haber recibido tanta ayuda por parte de ellos sin haberla pedido.

A mis amigos.

Por comprender que me encontraba ausente por atender los compromisos del posgrado.

A la Universidad Autónoma de Sinaloa

Por permitirme ser parte de su gran historia.

Al Dr. Jesús Roberto Millán Almaraz y al Dr. José Ramón Gaxiola Camacho

Por asesorarme y aconsejarme en cada una de mis fallas.

Al Dr. Jesús Martín Reyes Blanco y al Dr. Guadalupe Esteban Vázquez Becerra

Por participar de manera activa en él experimento y permitir así validar el proyecto.

Y al compañero M.C. Jesús Rene Vázquez Ontiveros

Por la ayuda prestada durante el posgrado.

Resumen

En este proyecto de tesis se desarrolló un sistema de hardware y software libre, llamado “sistema inteligente de monitoreo”, que cuenta con la capacidad de realizar una serie de mediciones de aceleración en un tiempo definido (monitoreo). Está conformado por una tarjeta de un circuito embebido y un sensor de aceleración, con ello se logró realizar exitosamente una serie de monitoreos en una viga tipo IPR, la cual es ampliamente utilizada en la industria de las estructuras civiles para la construcción de puentes y edificios.

Este dispositivo fue desarrollado con la finalidad de aportar un método viable y confiable para monitorear estructuras civiles según la filosofía SHM (*Structural Health Monitoring*), ya que, se utilizó un sensor de aceleración de uso no comercial y de bajo costo, buscando con esto, reemplazar los dispositivos comerciales que normalmente no son libres. Con base en lo anterior se pretende reducir considerablemente el costo de adquisición de estos dispositivos de medición, liberando de este modo el uso de tecnologías a ser usadas en la filosofía SHM para estructuras civiles como puentes.

En resumen, en este documento de tesis se compara de forma puntal el desempeño del sensor no comercial utilizado en este trabajo (marca Invesense y modelo MPU9250), frente a un sensor comercial de la marca Piezotronics modelo 352C03. Así mismo, se comparan las mismas señales con el comportamiento de unas señales producidas por un sensor de posición denominado Sistema de Posicionamiento Global por sus siglas en inglés GPS.

Gracias a lo anterior se pretende confirmar que los acelerómetros propuestos por esta investigación pueden ser de ayuda en la filosofía SHM.

Abstract

In this research, an embedded device is developed with open-source hardware and software for the Structural Health Monitoring (SHM) of infrastructure as bridges. Such a device is called smart monitoring system. It has the capability of performing a series of measurements of acceleration in a defined time with high accuracy. Basically, the electronic device is formed by a Printed Circuit Board (PCB) card and an acceleration sensor. The implementation of the acceleration sensor is validated by successfully performing acceleration measurements of the structural response of an IPR beam, which is commonly used in civil structures such as bridges and buildings.

The main reason for the assembly of this device was to contribute in the development of a viable and reliable method for the SHM of civil structures to be implemented in bridges. This non-commercial device is able to extract acceleration with high accuracy. In addition, the cost associated to it is relatively low which may represent an alternative to replace the expensive commercial sensors. Hence, the use of the sensor presented in this research will reduce the cost of the SHM technologies for civil infrastructure.

In summary, in this research, the measurement capabilities of the proposed non-commercial sensor are compared to a commercial sensor. Additionally, the performance of the non-commercial sensor is validated by using a GPS sensor that performed measurements in the experiment as well, giving as a result some excellent

response and confirming that the use of this type of non-commercial sensors may be a reliable option in the SHM of infrastructure as bridges.

Tabla de Contenidos

Agradecimientos.....	ii
Resumen.....	iii
Abstract	v
Capítulo 1. Introducción	1
1.1 Objetivos	2
1.1.1 General:.....	2
1.1.2 Específicos:	2
1.2 Hipótesis.....	3
1.3 Planteamiento del problema	4
1.4 Justificación.....	5
Capítulo 2. Antecedentes y Análisis del Estado del Arte.....	6
2.1 Historia de Análisis de estructuras civiles.....	6
2.2 Estado del arte	7
2.3 Fundamentación teórica.....	12
2.3.1 Sensor:.....	12
2.3.2 Sensores de aceleración:.....	12
2.3.3 Protocolos de comunicación.....	15
2.3.4 FPGA.....	22
2.3.5 Microcontroladores	28
2.3.6 Lenguajes de programación.....	32
2.3.7 Tratamiento digital de señales.....	41
Capítulo 3. Materiales y Métodos	53
3.1 Introducción.....	53

3.2	Desarrollo del dispositivo.....	54
3.3	Comunicación dispositivo-sensor.....	57
3.4	Programación del sensor:	58
3.5	Comunicación dispositivo-computadora:	59
3.6	Validación del adquirente de datos de aceleración.....	61
3.6.1	Prueba en estado de reposo positivo:	63
3.6.2	Prueba en estado de reposo negativo.....	64
3.6.3	Prueba en estado de reposo positivo-negativo:	65
3.6.4	Pruebas en laboratorio con excitación a 20Hz	66
3.6.5	Pruebas en laboratorio con excitación a 100Hz	67
3.7	Migración del dispositivo	69
3.8	Algoritmo del Firmware en microcontrolador.....	69
3.9	Importancia del caso de estudio	71
Capítulo 4.	Resultados y discusión	72
4.1	Introducción.....	72
4.1.1	Excitación de la viga por un golpe	73
4.1.2	La viga es excitada con un movimiento forzado.....	77
4.2	Sensores de Aceleración vs GPS	82
4.3	Obtención de los desplazamientos dinámicos.	89
4.4	Análisis de los resultados.	97
4.5	Análisis Estadísticos.....	104
4.5.1	Obtención de la diferencia media entre dispositivos.....	104
4.5.2	Obtención del valor máximo en los experimentos	105
4.5.3	Obtención del valor mínimo de los experimentos.....	106

4.5.4	Obtención de la desviación estándar	107
Capítulo 5.	Conclusiones y Recomendaciones	108
Bibliografía.....		109

Lista de tablas

Tabla 3.1. Especificaciones del acelerómetro MPU 9250.	56
Tabla 3.2. Especificaciones del giroscopio MPU 9250.	56
Tabla 3.3. Especificaciones del magnetometro MPU 9250.	57
Tabla 3.4. Registros de comunicación SPI	58
Tabla 4.1. Medias obtenidas desde el sensor 352C03.....	104
Tabla 4.2. Medias obtenidas desde el sensor MPU9250.....	104
Tabla 4.3. diferencias entre los sensores de aceleración MPU9250 y 352C03.	104
Tabla 4.4. Valores máximos obtenidos desde el sensor 352C03.....	105
Tabla 4.5. Valores máximos obtenidos desde el sensor 352C03.....	105
Tabla 4.7. Valores mínimos del sensor 352C03.	106
Tabla 4.8. Valores mínimos del sensor 352C03.	106
Tabla 4.9. Diferencias de los valores mínimos obtenidos desde ambos sensores de aceleración MPU9250 y 352C03.....	107
Tabla 4.10. desviación estándar de los datos obtenidos por el sensor 352C03.....	107
Tabla 4.11. desviación estándar de los datos obtenidos por el sensor MPU9250.....	107

Lista de Figuras

Figura 2.1: Ubicación de los sensores en el puente del río Erqi Yangtze. (Han et al., 2016)	8
Figura 2.2: Instrumentación del puente Erqi Yangtze (Han et al., 2016).....	8
Figura 2.3: FPGA Altera (fotografía por Mike1024).....	22
Figura 2.4: Ejemplo de una compuerta AND en VHD.	25
Figura 2.5: Ejemplo de una maquina de estados finitos (Romero Troncoso, 2007).....	27
Figura 2.6: Representación gráfica de una señal discreta en el tiempo.....	43
Figura 2.7: Muestreo periódico de una señal analógica.	45
Figura 3.1: Tarjeta FPGA De0-Nano-SOC.....	55
Figura 3.2: Protocolo de comunicación SPI modo 3 (Invensense, 2016).	57
Figura 3.3: Máquina de estados finitos del firmware del dispositivo.	60
Figura 3.4: Mesa vibratoria vertical.	61
Figura 3.5: Circuito controlador de mesa de vibraciones.....	62
Figura 3.6: Señal recibida por el sensor en estado de reposo eje Z positivo.....	63
Figura 3.7: Señal recibida por el sensor en estado de reposo eje Z negativo.....	64
Figura 3.8: Señal en estado de reposo con cambio de positivo a negativo.	65
Figura 3.9: Señal con una excitación mecánica a 20 Hz en el dominio del tiempo.	66
Figura 3.10: Señal con una excitación mecánica a 20 Hz en el dominio de la frecuencia.....	67
Figura 3.11: Señal con una excitación mecánica a 100 Hz en el dominio del tiempo.	67
Figura 3.12: Señal con una excitación mecánica a 100 Hz en el dominio de la frecuencia.....	68
Figura 3.13: Estructura de una viga IPR.	71
Figura 4.1: Diagrama de la ubicación de los sensores.	72
Figura 4.2: Punto de excitación en el experimento.	73
Figura 4.3: Sensor de aceleración Piezotronics 352C03 (Cortesía de pcbpiezotronics.fr).	74

Figura 4.4: Datos de aceleración en “Martillo 1”, MPU9250 vs 352C03.....	75
Figura 4.5: Datos de aceleración en “Martillo 2” MPU9250 vs 352C03.....	76
Figura 4.6: Datos de aceleración en “Excitación de Onda 1” MPU9250 vs 352C03.	77
Figura 4.7: Datos de aceleración en “Excitación de Onda 2” MPU9250 vs 352C03.	78
Figura 4.8: Datos de aceleración en “Excitación de Onda 3” MPU9250 vs 352C03.	79
Figura 4.9: Datos de aceleración de “Excitación de Onda”, MPU9250 vs 352C03.	80
Figura 4.10: Datos de aceleración de “Excitación de Onda 5”, MPU9250 vs 352C03.	81
Figura 4.11: Datos obtenidos “martillo1” MPU9250 vs GPS.....	82
Figura 4.12: Datos obtenidos “martillo2” MPU9250 vs GPS.....	83
Figura 4.13: Datos obtenidos “Excitación de Onda 1” MPU9250 vs GPS.....	84
Figura 4.14: Datos obtenidos “Excitación de Onda 2” MPU9250 vs GPS.....	85
Figura 4.15: Datos obtenidos “Excitación de Onda 3” MPU9250 vs GPS.....	86
Figura 4.16: Datos obtenidos “Excitación de Onda 4” MPU9250 vs GPS.....	87
Figura 4.17: Datos obtenidos “Excitación de Onda” MPU9250 vs GPS.....	88
Figura 4.18: Datos de desplazamiento dinámico “martillo1”, MPU9250 vs GPS.....	90
Figura 4.19: Datos de desplazamiento dinámico “Martillo 2”, MPU9250 vs GPS.	91
Figura 4.20: Datos de desplazamiento dinámico “Excitación de Onda 1”, MPU9250 vs GPS... ..	92
Figura 4.21: Datos de desplazamiento dinámico “Excitación de Onda 2”, MPU9250 vs GPS... ..	93
Figura 4.22: Datos de desplazamiento dinámico “Excitación de Onda 3”, MPU9250 vs GPS... ..	94
Figura 4.23: Datos de desplazamiento dinámico “Excitación de Onda 4”, MPU9250 vs GPS... ..	95
Figura 4.24: Datos de desplazamiento dinámico “Excitación de Onda 5”, MPU9250 vs GPS... ..	96
Figura 4.25: Datos en frecuencia de “martillo1” MPU920(rojo) vs 352C03(azul).	97
Figura 4.26: Datos en frecuencia de “Martillo 2” MPU920(rojo) vs 352C03(azul).	98
Figura 4.27: Datos en frecuencia de “Excitación de Onda 1” MPU920(rojo) vs 352C03(azul).. ..	99

Figura 4.28: Datos en frecuencia de “Excitación de Onda 2” MPU920(rojo) vs 352C03(azul).100

Figura 4.29: Datos en frecuencia de “Excitación de Onda 3” MPU920(rojo) vs 352C03(azul).101

Figura 4.30: Datos en frecuencia de “Excitación de Onda 4” MPU920(rojo) vs 352C03(azul).102

Figura 4.31: Datos en frecuencia de “Excitación de Onda 5” MPU920(rojo) vs 352C03(azul).103

Capítulo 1. Introducción

Las técnicas de monitoreo como SHM pueden ser utilizadas para detectar daños invisibles u ocultos en las estructuras civiles, esta técnica tiene la capacidad de detectar cambios sutiles en las señales de vibraciones, permitiendo localizar y cuantificar la gravedad de los daños en estructuras (Amezquita-Sanchez & Adeli, 2016). En este proyecto de tesis se busca potenciar la calidad de las técnicas de procesamiento de señales en el campo del monitoreo en estructuras civiles, donde los datos obtenidos de los sensores de aceleración complementan los datos de otro tipo de sensores como lo son, por ejemplo, el Sistema de Posicionamiento Global que permite obtener datos de posición a una baja frecuencia de muestreo. El caso de estudio en este trabajo de tesis es el análisis del comportamiento de una viga simplemente sostenida del tipo IPR ya que es comúnmente utilizada por las industrias de construcción de puentes y edificios, el análisis se realizó con la ayuda de instrumentos de medición de aceleración y de desplazamiento, Utilizando un dispositivo de medición de aceleración no comercial y un sensor de aceleración comercial para validar la respuesta del sensor mencionado anteriormente así mismo se complementaron los datos obtenidos con los datos de desplazamiento proporcionados por un sensor del tipo GPS, posteriormente se procesaron los datos obtenidos por los sensores en un programa de procesamiento de datos vectoriales para determinar la frecuencia natural y el desplazamiento dinámico de la estructura.

1.1 Objetivos

1.1.1 General:

- Diseñar y desarrollar un sistema embebido de bajo costo que permita adquirir señales de vibraciones y poder realizar procesamiento digital de dichas señales con el propósito de obtener el comportamiento dinámico de estructuras civiles.

1.1.2 Específicos:

- Desarrollar un sistema de monitoreo de vibraciones de alta frecuencia de muestreo (2 KHz) para aplicaciones en estructuras civiles.
- Adquirir datos de señales de vibraciones en una estructura civil de tamaño real.
- Analizar las señales de vibraciones obtenidas utilizando algoritmos de procesamiento digital de señales en tiempo y frecuencia.
- Comparar el desempeño del dispositivo electrónico en la extracción de desplazamientos con respecto a los obtenidos a partir de un receptor GPS.
- Validar la metodología desarrollada como una nueva alternativa para aplicaciones en SHM.

1.2 Hipótesis

Es posible extraer el comportamiento dinámico de una estructura civil mediante un sistema de monitoreo de bajo costo basado en aceleración y un óptimo algoritmo de procesamiento digital de señales.

1.3 Planteamiento del problema

Actualmente en México y otras partes del mundo hay una gran cantidad de estructuras civiles que presentan deformaciones por causas artificiales o naturales, según la SMIE en el artículo “NORMAS TÉCNICAS COMPLEMENTARIAS PARA DISEÑO Y CONSTRUCCIÓN DE ESTRUCTURAS DE CONCRETO” sección 4 todas las estructuras civiles de concreto deben de ser diseñadas para durar no más de 50 de años. Sin embargo, muchas estructuras se encuentran activas a pesar de haber sobrepasado su periodo de vida útil. Por lo tanto, es necesario realizar un análisis del estado de salud estructural de dichas obras. En este sentido, actualmente existen dispositivos electrónicos que permiten extraer respuestas estructurales, los cuales se basan en aceleración y desplazamientos. Dichos dispositivos son ampliamente utilizados en el estudio del comportamiento de las estructuras siguiendo la filosofía SHM (*Structural Health Monitoring*). Desafortunadamente, estos dispositivos suelen ser de alto costo, por lo cual no son de libre acceso y no se cuenta con la documentación necesaria para el funcionamiento de estos. Por lo tanto, el estudio de la salud estructural se vuelve una tarea muy cara y en ocasiones inaccesible. Por lo tanto, con el fin de proveer de una alternativa para este problema, en esta investigación se presenta el desarrollo e implementación de un sensor de tipo no-comercial; el cual puede ser de fácil acceso y utilizado en el SHM de infraestructura como puentes.

1.4 Justificación

Actualmente, las técnicas detrás de la filosofía SHM son ampliamente utilizadas alrededor del mundo para la detección y cuantificación de posibles fallas en estructuras civiles. Sin embargo, en algunos países, este tipo de análisis no son implementados con frecuencia debido a lo costosos que resultan, causando que se desconozca el estado actual de la mayoría de las estructuras civiles, lo cual, en la mayoría de las ocasiones podría evitar el colapso de estas tanto por cargas artificiales como accidentales (catástrofes naturales). En este sentido, según las normas de construcción mexicanas el tiempo de vida útil en puentes y edificios es generalmente de 45 a 50 años, por lo que las estructuras de dicha edad serían las más susceptibles a sufrir fallas las cuales podrían ocasionar accidentes. Es por ello que es necesario que se desarrollen sistemas de monitoreo accesibles y de alta confiabilidad, lo cual justifica la presente investigación. De esta forma, en este trabajo de tesis, se utilizaron acelerómetros de bajo costo que se pueden usar para instrumentar estructuras como puentes fácilmente, y debido a la alta resolución de muestreo, pueden utilizarse para monitoreo usando la filosofía SHM con una alta precisión.

Capítulo 2. Antecedentes y Análisis del Estado del Arte

Podemos referirnos a la filosofía SHM como una disciplina que se encarga de analizar estructuras civiles utilizando diversidad de métodos; para ello, su base se encuentra en la observación de una estructura y existe desde que la humanidad comenzó a crear estructuras.

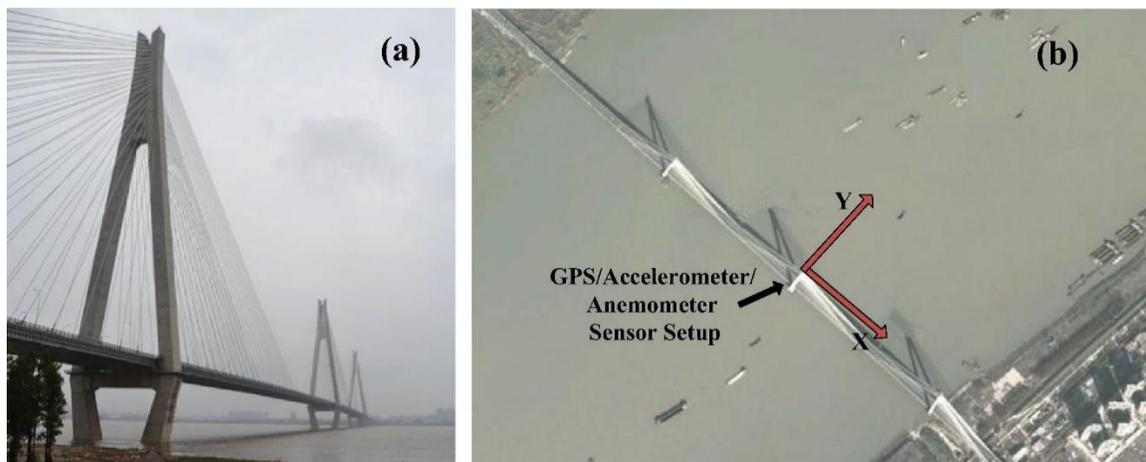
2.1 Historia de Análisis de estructuras civiles

A principios de los 90s (Pandey, 1991), se reportó la curvatura de la forma modal como un parámetro sensible en la rigidez estructural (Stubbs, Kim, & Topole, 1992). En dichas investigaciones, se presentó un procedimiento que calcula el cambio en la energía de deformación almacenada en una estructura. Algunos otros investigadores (Mallat & Hwang, 1992) dedujeron que se pueden obtener coeficientes de ondas (wavelets), para datos discretos mediante la “Transformada Rápida Wavelet (FWT por sus siglas en ingles). Por otro lado, (Kaouk, Kaouk, Zimmerman, & Zimmerman, 1994) consideraron como un caso particular de estudio los cambios en los parámetros estructurales, que dio como resultado “*El modelo basado en el método de detección de cambios en rigidez dinámica*”. (Farrar & Jáuregui, 1996) propusieron que la medición de los cambios de las características en las estructuras puede ser utilizada para detectar y localizar el daño (índice de daño). (MAECK & DE ROECK, 2003), propusieron el método basado en curvaturas que denominó “Calculo Directo de la Rigidez”.

2.2 Estado del arte

La literatura dice que el monitoreo de salud estructural SHM es una técnica que consiste en realizar mediciones de manera constante en estructuras civiles, utilizando sensores de precisión que permitan obtener datos para la detección y la cuantificación de los daños en estructuras. (Amezquita-Sanchez & Adeli, 2016). Para obtener estos datos es necesario realizar monitoreos mediante instrumentos de medición ya sean de aceleración, posición y rotación, unas de las tecnologías que se han estado utilizando recientemente para ello es el uso de los GPS, ya que proporcionan datos de desplazamiento directo sin el uso de la integración doble en el caso de la aceleración. Sin embargo, estos suelen ser de un costo muy elevado y de una baja frecuencia de muestreo, la cual ronda de 1 – 10 Hz, (Roberts & ASHKENAZI, 1997), (Elnabwy, Kaloop, & Elbeltagi, 2013), (Psimoulis, Peppas, Bonenberg, Ince, & Meng, 2016) y (Roberts et al., 2012). Muy por el contrario, los acelerómetros se pueden conseguir fácilmente ya que son de bajo costo y suelen manejar frecuencias de muestreo relativamente altas que rondan desde 1 a 10 KHz (Yi). No obstante, ni la medición con GPS, ni la medición con acelerómetro debe de ser descartada ya que con el desplazamiento directo que proporciona el GPS se puede calcular el desplazamiento semi-estático que se encuentra a bajas frecuencias mediante un filtro media-móvil, y mediante el filtro Chebyshev se puede determinar el desplazamiento dinámico que generalmente se encuentra en altas frecuencias. Por su parte el acelerómetro puede determinar la frecuencia fundamental del puente mediante el uso de la transformada

rápida de Fourier (FFT), y de este modo los análisis se complementan. Como es el caso de (Han, Wang, Meng, & Liu, 2016), donde integraron el uso del GPS, acelerómetros y anemómetros para monitorear el puente El puente del río Erqi Yangtze, es un puente atirantado de tres torres y tiene una longitud de 2922 m.



(a) Puente Erqi Yangtze. (b) Colocación de los receptores GPS / Acelerómetros / Anemómetros

Figura 2.1: Ubicación de los sensores en el puente del río Erqi Yangtze. (Han et al., 2016)

La metodología consistió en tres receptores DL-4 GNSS de doble frecuencia, a una frecuencia de muestreo de 10 Hz, un total de 7 acelerómetros triaxiales, con una velocidad de muestreo de 20 Hz, y 3 anemómetros en cada columna del puente con una frecuencia de muestreo de 1 Hz. Tal y como se puede ver en la siguiente figura:

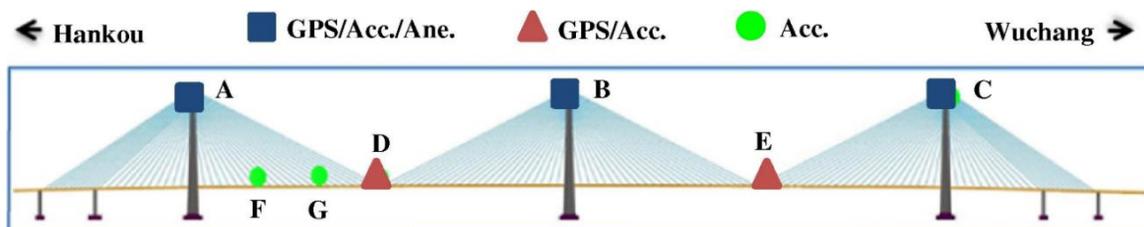


Figura 2.2: Instrumentación del puente Erqi Yangtze (Han et al., 2016).

Así como los casos ilustrados anteriormente, existieron muchos más investigadores que se dedicaron a instrumentar estructuras con el fin de realizar monitoreos y desarrollar diferentes técnicas SHM.

Por ejemplo (Das & Saha, 2018) hicieron pruebas con diferentes tipos de sensores para determinar cuáles son los mejores en SHM y determinaron que los mejores sensores se encuentran entre los que manejan una tensión eléctrica, los cuales son acelerómetros, sensores de presión, sensores de fibra óptica, etc.

Por otro lado (Yunus, Ibrahim, & Ahmad, 2018) hicieron pruebas para determinar cuál es la eficiencia de los acelerómetros frente a los sensores basados en GPS sin embargo encontraron que en lugar de competir entre ellos se complementan ya que el GPS a pesar de ser más exacto en los desplazamientos, el acelerómetro lo compensa con la velocidad de la toma de datos.

En este sentido (Zhu et al., 2018) desarrollaron un dispositivo inalámbrico que poseía un acelerómetro programable en diferentes frecuencias de muestreo, descubrieron que no solo es factible realizar mediciones de esta forma, sino que además puede estar comunicando los datos en tiempos casi reales a otros dispositivos, así se obtienen datos actualizados de la salud estructural.

Adicionalmente, (Vazquez B. et al., 2017) hicieron mediciones de desplazamiento en Puente Juárez, Culiacán Sinaloa, utilizando 6 GPS en puntos estratégicos que estuvieron activos durante varias horas en determinados puntos del día, ellos determinaron que el puente presenta un desplazamiento inusual en el eje vertical

ya que varía entre los 0.011 y los 0.25 m, ellos aclaran que eso no significa que el puente valla a colapsar pero que es prudente que las autoridades tomen medidas en el asunto.

Alternativamente, (Amezquita-Sanchez & Adeli, 2016) utilizaron sensores de audio con una resolución de 10 Hz para monitorear un edificio en puntos específicos con el fin de mostrar varias estrategias de monitoreo, ellos utilizaron una amplia gama de procesamiento de señales y encontraron que las técnicas de procesamiento de señales más comunes utilizadas en SHM son los modelos TS, WT y HHT y, a veces, una combinación de dos de ellos.

También (Pan, Zhao, Zhao, & Mao, 2015) utilizaron acelerómetros para monitorear el movimiento de partículas en tubos de acero rellenos de hormigón, demostraron que la utilización de acelerómetros como forma de identificación cualitativa es efectiva y factible.

Por otra parte (Moschas & Stiros, 2011) realizaron un experimento para calcular los efectos de diferentes fuerzas aplicadas a un puente mediante el salto coordinado de un grupo de personas, aplicaron el filtro Chebyshev para obtener el desplazamiento semi-estático y el filtro media-móvil para obtener el desplazamiento dinámico.

Recientemente (Michel Guzman-Acevedo et al., 2019) realizaron un análisis utilizando tecnología GPS, un acelerómetro profesional y un smartphone, donde se validó el uso del acelerómetro presente en los dispositivos smartphone para el análisis

de salud en estructuras civiles, el análisis se basó en la toma de mediciones en el puente Benito Juárez del municipio de Culiacán en el estado de Sinaloa, México.

2.3 Fundamentación teórica

2.3.1 Sensor:

Se le llama sensor a un dispositivo que tiene la capacidad de detectar magnitudes físicas o químicas y por medio de un transductor transformarlas en magnitudes de variables eléctricas tales como voltaje o corriente, de esta forma permite a otros dispositivos interpretar señales existentes e interactuar con el espacio, existen sensores para detectar la mayoría de las señales presentes en el espacio, solo hay dos formas de detección dependiendo de la construcción del sensor, detección de señal en modo “on/off”, que al detectar encienden o apagan una señal dependiendo de la configuración, y los sensores de detección variable, donde la señal producida por el sensor es directamente proporcional a la señal detectada por este.

2.3.2 Sensores de aceleración:

Se conoce como acelerómetro a cualquier dispositivo que tiene la capacidad de medir aceleraciones, su funcionamiento se basa en monitoreo de una masa con peso conocido como referencia del dispositivo, existen varios tipos de acelerómetros los cuales se presentan a continuación:

2.3.2.1 Acelerómetro mecánico:

Es un acelerómetro que como su nombre lo indica transforma las mediciones de la aceleración a movimiento mecánico, se trata del más simple de este tipo de sensores, se basa en la unión de una masa con peso conocido a un dinamómetro cuyo eje debe estar en la misma dirección de la aceleración a medir.

2.3.2.2 Acelerómetro piezoeléctrico:

Se trata de un acelerómetro basado en un elemento piezoeléctrico, el más común es el piezoeléctrico por compresión, este cuenta con un elemento piezoeléctrico que al ser comprimido, produce una carga eléctrica que es directamente proporcional a la fuerza recibida, normalmente los elementos piezoeléctricos están hechos de zirconato de plomo que se encuentra comprimido por una masa y esta a su vez se encuentra sujeta por un muelle que la une hacia el otro lado de la pared del dispositivo, todo esto dentro de una caja de metal, así que, cuando el dispositivo es sometido a vibración la masa dentro del dispositivo vibra y produce una fuerza sobre el material piezoeléctrico y este produce una señal eléctrica.

2.3.2.3 Acelerómetro de efecto Hall:

Utilizan una masa unida a un muelle y a un elemento magnético y ese es monitoreado todo el tiempo por un sensor de efecto hall, al vibrar este produce cambios en el campo magnético que el sensor traduce a señales eléctricas.

2.3.2.4 Acelerómetro de condensador:

Utilizan un condensador construido de modo tal que existe una masa unida a un muelle entre sus placas, que al vibrar esta hace que cambie la corriente que circula por el capacitor produciendo así una señal que luego puede ser detectada.

2.3.2.5 Acelerómetro MEMS:

Actualmente existe la tecnología para crear sensores triaxiales en un solo chip de silicio, el cual incluye toda la etapa electrónica para procesar las señales y comunicar el dispositivo, en caso de sensores digitales, el funcionamiento de estos dispositivos está basado en el traspaso térmico por convección natural, por lo que la temperatura a la que funciona el sensor suele ser de suma importancia, este tipo de sensores no emplean elementos mecánicos tradicionales tales como la masa unida a un muelle, ya que funcionan por medio de moléculas de gas, estos dispositivos miden el cambio de la transferencia de calor causada por la aceleración, por lo que los elementos mecánicos móviles son eliminados (Serway, Jewett, & Serway, 2004).

2.3.3 Protocolos de comunicación

2.3.3.1 Definición.

El termino red en informática se refiere al conjunto de dispositivos informáticos que se encuentran interconectados por medio de nodos físicos, los cuales se encargan de enviar o recibir impulsos eléctricos con el fin de comunicarse.

Existen varios tipos de redes:

2.3.3.2 Tipos de redes

Red de área personal o PAN por sus siglas en inglés (Personal Area Network), es una red de computadoras usada para la comunicación entre los dispositivos de la computadora cerca de una persona.

Red inalámbrica de área personal o WPAN por sus siglas en inglés (Wireless Personal Area Network), es una red de computadoras inalámbrica, para la comunicación entre distintos dispositivos (tanto computadoras como puntos de acceso a internet, teléfonos celulares, computadoras portátiles, dispositivos de audio, impresoras) cercanos al punto de acceso. Estas redes normalmente son de unos pocos metros y para uso personal, El medio de transporte puede ser cualquiera de los habituales en las redes inalámbricas pero las que reciben esta denominación son habituales en Bluetooth.

Red de Área Local o LAN por sus siglas en inglés (Local Area Network), se refiere a una conexión de los dispositivos dentro de un área limitada como puede ser un cuarto o un edificio pequeño. No utilizan medios ni redes de interconexión públicas.

Red de Área Local inalámbrica o WLAN por sus siglas en inglés (Wireless Local Area Network), Es una conexión de los dispositivos dentro de un área limitada, al igual que la anterior es una LAN, pero con conexión inalámbrica.

Red de Área de Campus o CAN por sus siglas en inglés (Campus Area Network), es una red de alta velocidad diseñada para conectar redes de área local a en áreas geográficas pequeñas, tales como un hospital, una base Militar o una Universidad. No utilizan medios ni redes de interconexión públicas.

Red de área metropolitana o MAN por sus siglas en inglés (Metropolitan Area Network), es una red de alta velocidad que conecta un área geográfica más grande que las redes CAN se utilizan para conectar varios edificios dentro de un área limitada, por ejemplo, la red que conecta varios edificios públicos entre si dentro de un municipio.

Red de área amplia o WAN por sus siglas en inglés (Wide Area Network), son redes de alta velocidad que se utilizan para conectar edificios de áreas geográficas extensas utilizando medios como satélites, cables interoceánicos, internet, fibras ópticas públicas, etc.

Redes inalámbricas de área extensa o WWAN por sus siglas en inglés (Wireless Wide Area Network), tienen el alcance más amplio de todas las redes inalámbricas. Por esta razón, todos los teléfonos móviles están conectados a una red inalámbrica de área extensa.

Las tecnologías principales son:

- GSM (Global System for Mobile Communication)
- GPRS (General Packet Radio Service)
- UMTS (Universal Mobile Telecommunication System)

Red de área de almacenamiento o SAN por sus siglas en inglés (Storage Area Network), diseñadas para conectar servidores, matrices o en inglés (Arrays) de disco y bibliotecas de soporte, permitiendo el tránsito de datos sin afectar los medios por el que acceden los usuarios.

Red de área local virtual o VLAN por sus siglas en inglés (Virtual LAN), es un grupo de computadoras con un conjunto común de recursos a compartir y de requerimientos, que se comunican como si estuvieran adjuntos a una división lógica de redes de computadoras en la cual todos los nodos pueden alcanzar a los otros por medio de una red de difusión o en inglés (broadcast) en la capa de enlace de datos, a pesar de su diversa localización física. Este tipo surgió como respuesta a la necesidad de poder estructurar las conexiones de equipos de un edificio por medio de software, permitiendo dividir un conmutador en varios virtuales.

2.3.3.3 Medios de transmisión.

El internet tiene dos tipos de transmisión por medios guiados y medios no guiados.

Por medios guiados.

Cable de par trenzado: es una forma de conexión en la que dos conductores eléctricos aislados son entrelazados para tener menores interferencias, aumentar la potencia y disminuir la diafonía (Perturbación electromagnética producida en un canal de comunicación por el acoplamiento de este con otro u otros vecinos) de los cables adyacentes. Dependiendo de la red se pueden utilizar, uno, dos, cuatro o más pares trenzados.

Cable coaxial: se utiliza para transportar señales electromagnéticas de alta frecuencia, el cual posee un núcleo sólido (generalmente de cobre) o de hilos, recubierto por un material dieléctrico y una malla o blindaje, que sirven para aislar o proteger la señal de información contra las interferencias o ruido exterior.

Fibra óptica: es un medio de transmisión empleado habitualmente en redes de datos; un hilo muy fino de material transparente, vidrio o materiales plásticos, por el que se envían pulsos de luz que representan los datos a transmitir.

Medios no guiados.

Red por radio es aquella que emplea la radiofrecuencia como medio de unión de las diversas estaciones de la red.

Red por infrarrojos o IRDA, por sus siglas en inglés (Infrared Data Association), permiten la comunicación entre dos nodos, usando una serie de leds infrarrojos para ello. Se trata de emisores/receptores de ondas infrarrojas entre ambos dispositivos, cada dispositivo necesita al otro para realizar la comunicación por ello es escasa su utilización a gran escala. No disponen de gran alcance y necesitan de visibilidad entre los dispositivos.

Red por microondas, es un tipo de red inalámbrica que utiliza microondas como medio de transmisión. Los protocolos más frecuentes son: el IEEE 802.11b y transmite a 2,4 GHz, alcanzando velocidades de 11 Mbps (Megabits por segundo); el rango de 5,4 a 5,7 GHz para el protocolo IEEE 802.11a; el IEEE 802.11n que permite velocidades de hasta 600 Mbps; etc.

2.3.3.4 Tecnologías que utilizan las redes de comunicación.

Red punto a punto o PTP, por sus siglas en inglés (point to point) es aquella en la que existe multitud de conexiones entre parejas individuales de máquinas. Este tipo de red requiere, en algunos casos, máquinas intermedias que establezcan rutas para que puedan transmitirse paquetes de datos. El medio electrónico habitual para la interconexión es el conmutador, o interruptor.

Red de Difusión o en inglés (broadcast) se caracteriza por transmitir datos por un sólo canal de comunicación que comparten todas las máquinas de la red. En este caso, el paquete enviado es recibido por todas las máquinas de la red, pero únicamente la red destino es la que lo puede procesar. Los equipos unidos por un concentrador o en inglés (hub), forman redes de este tipo.

Red multipunto, dispone de una línea o medio de comunicación cuyo uso está compartido por todas las terminales en la red. La información fluye de forma bidireccional. Los terminales pueden estar separados geográficamente.

2.3.3.5 Topologías físicas de la Red.

Independientemente que la tecnología sea inalámbrica o no las topologías de la red son las mismas en los dos casos:

Red en bus (bus o “conductor común”) Red lineal o en inglés (line): se caracteriza por tener un único canal de comunicaciones al cual se conectan los diferentes dispositivos.

Red en anillo, Red circular o en inglés (ring): cada estación está conectada a la siguiente y la última está conectada a la primera.

Red en estrella en inglés (star): las estaciones están conectadas directamente a un punto central y todas las comunicaciones se han de hacer necesariamente a través de éste.

- Red en malla en inglés (mesh): cada nodo está conectado a todos los otros.
- Red en árbol, en inglés (tree): los nodos están colocados en forma de árbol.

Desde una visión topológica, la conexión en árbol es parecida a una serie de redes en estrella interconectadas salvo en que no tiene un nodo central.

Red híbrida o Red mixta: se da cualquier combinación de las anteriores. Por ejemplo, circular de estrella, bus de estrella, etc.

2.3.3.6 Dirección de los datos de red.

- Simplex o unidireccional: un equipo terminal de datos transmite y otro recibe.
- Half-duplex o semidúplex: el método o protocolo de envío de información es bidireccional pero no simultáneamente, sólo un equipo transmite a la vez.
- Full-duplex o dúplex: los dos equipos involucrados en la comunicación lo pueden hacer de forma simultánea, transmitir y recibir (Tanenbaum, 2003).

2.3.4 FPGA

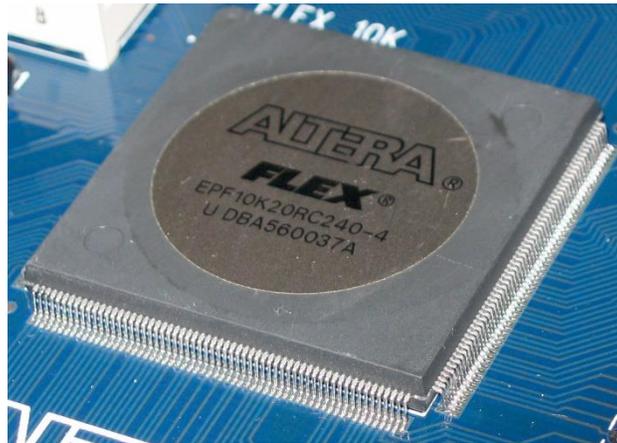


Figura 2.3: FPGA Altera (fotografía por Mike1024).

2.3.4.1 Definición

Se conoce como FPGA a un circuito programable basado en matrices de compuertas digitales por sus siglas en inglés *Field Programmable Gate Array*, tiene la capacidad de programarse para prácticamente cualquier algoritmo que se desee, además se trata de una tecnología que, al basarse en electrónica digital, puede trabajar de forma paralela, la mayoría de los FPGA cuentan con osciladores que manejan frecuencias

muy altas de reloj, mínimo 50 Mhz. Lo que los convierte en unos de los dispositivos programables con más rapidez en el mercado.

Como se comentó anteriormente, los FPGA son circuitos de compuertas digitales que pueden ser programadas, sin embargo, esta programación se trata de simple y llanamente la conexión de dichas compuertas digitales, este tipo de programación se realiza mediante el código VHD que se verá más adelante.

2.3.4.2 Módulo de memoria

Los FPGA cuentan con módulos memoria que básicamente son dispositivos de dos estados flip flop, en el cual se permite o no se permite el paso de una señal, este funcionamiento se conoce como bloque de memoria y cada uno de ellos tiene la capacidad de almacenar 1 bit, lo que permite el almacenamiento de memoria, algunos tipos más complejos de configuración de flip flops son la memoria tipo FIFO, LIFO, RAM, etc., prácticamente todos los tipos de bloques de memoria conocidos.

2.3.4.3 límites en los FPGA

Un dispositivo FPGA se puede programar con básicamente cualquier circuito digital que realice una tarea, sin embargo el límite de estos se podría decir que es la cantidad de bloques de memoria disponible en cada tarjeta, si bien esto podría no ser una limitante ya que se pueden configurar memorias externas para FPGA, resulta bastante complicado realizar este tipo de conexiones ya que cada módulo de memoria, es un mundo diferente, dependiendo del desarrollador de dicha memoria, otra limitante existente es el número de compuertas disponibles como bien su nombre lo indica matriz

de compuertas programables, se trata directamente de eso una matriz de $n \times n$ compuertas programables las cuales no tienen ninguna conexión entre ellas, y todas están listas para conectarse, sin embargo, todos los FPGA cuentan con una cantidad absurda de estas compuertas por lo que pueden realizar prácticamente cualquier tarea.

2.3.4.4 Periféricos en FPGA

Los dispositivos FPGA cuentan también con puertos de pines que pueden ser utilizados como entradas y salidas de datos los cuales son digitales, por lo que es necesario contar con circuitos convertidores de señales analógicas a digitales conocidos como ADS, en estos puertos pueden conectarse prácticamente cualquier modulo que se desee sin embargo para que este funcione correctamente es necesario programar un módulo que interactúe con este dentro de la arquitectura del FPGA.

2.3.4.5 Programación en FPGA VHD

La programación en FPGA es de hecho una conexión interna de las compuertas programables en dicho dispositivo, no se trata de un lenguaje de programación sino de un intérprete para describir conexiones entre circuitos, suele trabajar en capas y permite la conexión entre circuitos internos en el FPGA conocidos como módulos.

Para describir un circuito en FPGA utilizando VHD, siempre se utiliza la misma sintaxis, primero se describen las bibliotecas con las que se van a trabajar en un apartado llamado cabecera, luego se describe la configuración o entidad del circuito a conectar, así como también el número y la cantidad de bits de entradas y salidas del circuito, y por último se coloca la arquitectura del circuito que básicamente describe el

funcionamiento del mismo, esto funciona para todo tipo de circuitos incluyendo los que contienen máquinas de estados finitos y los que unen circuitos entre sí.

```
1  library ieee;
2  use ieee.std_logic_1164.all;
3
4  entity and is
5      port(
6          a,b: in std_logic;
7          y: out std_logic
8      );
9  end and;
10
11 architecture simple of and is
12 begin
13     y <= a and b;
14 end simple;
15
16
```

Figura 2.4: Ejemplo de una compuerta AND en VHD.

En la figura anterior se puede ver un ejemplo de una compuerta lógica AND utilizando el lenguaje VHD utilizado para realizar conexiones en FPGA, esto permite que la salida “y” siempre valga 0 hasta que “a” y “b” sean positivos.

2.3.4.6 Alcance de los FPGA

Los dispositivos basados en FPGA son utilizados frecuentemente en la industria cuando se requiere de velocidad y precisión en las tareas, además al ser un dispositivo basado en compuertas programables una vez configurada su conexión es muy difícil que terceros puedan extraer el programa del circuito, lo que lo convierte en un

dispositivo seguro, debido a esto no es extraño ver este tipo de dispositivos en armamento militar, medicina, exploración espacial, fuerza aérea, etc.

2.3.4.7 Inconvenientes en FPGA

Los dispositivos FPGA al no ser un dispositivo programable no se comporta como un microcontrolador por lo que tareas específicas (ASIC), suelen ser más lentas en FPGA que en los microcontroladores, otro inconveniente, es que la gran mayoría de estos dispositivos demandan una gran cantidad de corriente de entre 2000 a 3000 mA, cuando los microcontroladores consumen un aproximado de 30 mA, además suelen ser más complicados de programar ya que hay que conectar de forma manual todos y cada uno de los módulos presentes en la tarea a realizar, y este tipo de programación en los microcontroladores ya vienen definidos salvo por algunos cambios que se tengan que realizar para que funcionen correctamente.

2.3.4.8 Máquinas de estados finitos

En FPGA todas las tareas se realizan de forma paralela por lo que para serializar y controlar las tareas es necesario que la FPGA cuente con una máquina de estados finitos, las máquinas de estados finitos son simples maquinas que funcionan de forma lógica es decir con el sistema binario, esto quiere decir que tienen la capacidad de activar (1) o desactivar (0) variables lógicas, de carácter booleano. Usual mente se representa por medio de grafos, y al igual que los autómatas tienen un inicio y un ciclo, por lo que se trata de tareas periódicas, las máquinas de estados finitos pueden ser muy sencillas dependiendo de la dificultad de la tarea, pero también pueden llegar a ser muy

confusas ya que normalmente suelen controlar otras máquinas de estados finitos, lo que complican el funcionamiento al ojo del programador ya que se vuelve un proceso abstracto (Romero Troncoso, 2007).

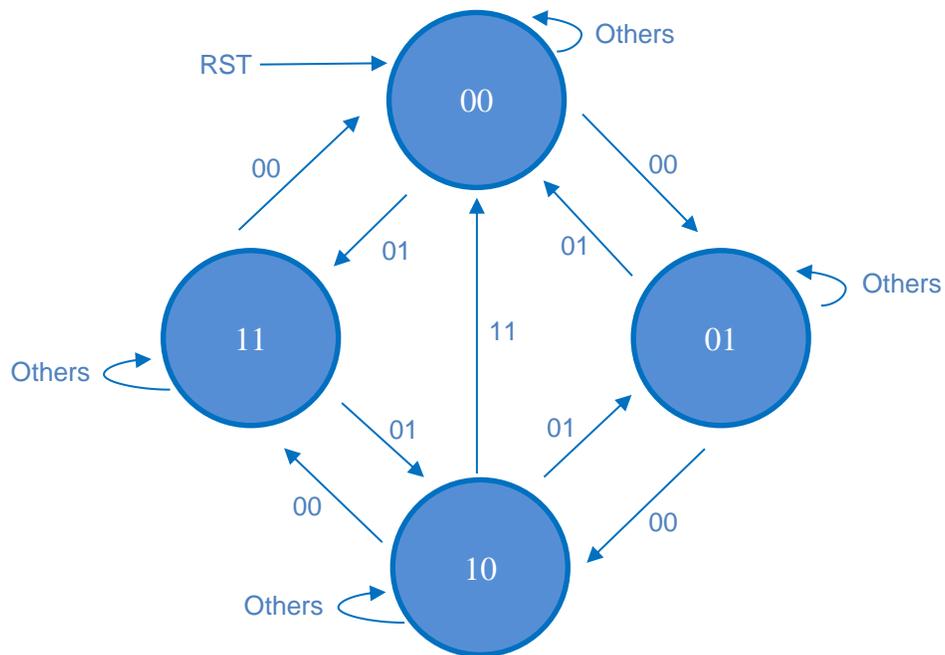


Figura 2.5. Ejemplo de una máquina de estados finitos (Romero Troncoso, 2007).

2.3.5 Microcontroladores

2.3.5.1 Definición

Un microcontrolador es un circuito integrado que está constituido por una unidad central de proceso (CPU), memorias (RAM y ROM), Entradas/salidas y Periferia, todo dentro del mismo dispositivo. Estas partes están interconectadas dentro del microcontrolador, y en conjunto forman lo que se le conoce como una microcomputadora. Se puede decir con toda seguridad que un microcontrolador es una microcomputadora completa encapsulada dentro de un circuito integrado.

Toda microcomputadora requiere de un programa para que realice una función específica. Este se almacena normalmente en la memoria ROM. No está de más mencionar que sin un programa, los microcontroladores carecen de utilidad.

El propósito fundamental de los microcontroladores es el de leer y ejecutar los programas que el usuario le escribe, es por esto por lo que la programación es una actividad básica e indispensable cuando se diseñan circuitos y sistemas que los incluyan. El carácter programable de los microcontroladores simplifica el diseño de circuitos electrónicos.

2.3.5.2 Escritura de los programas.

Los microcontroladores están diseñados para trabajar con programas escritos en código binario, es decir “1” y “0”, a esto se le conoce como lenguaje máquina, y es lo único que son capaces de entender, sin embargo este lenguaje resulta difícil y tedioso para los programadores, así que se desarrollaron lenguajes de programación que se

asemejan más a las palabras que utilizan las personas, a estos lenguajes se le conocen como lenguajes de alto nivel mientras al lenguaje máquina se le conoce como lenguaje de bajo nivel. Los lenguajes de alto nivel tales como el C o BASIC son los más comunes en la programación de estos microcontroladores.

Otro tipo de lenguaje es el ensamblador el cual es más especializado, se trata de una lista con número limitado de instrucciones a las cuales puede responder un microcontrolador. Estas acciones son palabras o abreviaciones que representan las instrucciones en lenguaje máquina.

Las instrucciones en el lenguaje ensamblador, son fáciles de entender y permiten operar directamente con los registros de memoria con las instrucciones intrínsecas del microcontrolador. Es por esto por lo que el lenguaje ensamblador es sin lugar a duda el lenguaje por excelencia en programación de microcontroladores, ya que permite hacer uso eficiente de la memoria y minimizar el tiempo de ejecución de un programa.

2.3.5.3 Compilación.

Todo programa escrito en un lenguaje de alto nivel debe ser transformado en código máquina. Los programas que escribimos los entendemos nosotros, pero el microcontrolador no. Un software de computadora, llamado compilador, traduce y transforma nuestro programa en código máquina, que es lo que realmente puede leer e interpretar el microcontrolador, Una vez compilado el programa, es momento de transferir el código máquina hacia la memoria interna del microcontrolador, usualmente hacia la memoria ROM Para esta tarea se utiliza un programador físico, que es una

pieza de hardware que tiene el propósito de escribir el programa en la memoria interna del microcontrolador.

2.3.5.4 Familias de microcontroladores.

Existen varios fabricantes de microcontroladores tales como Texas Instruments, Motorola, Atmel, Intel, Microchip, Toshiba, etc. Todos ellos ofrecen microcontroladores con características más o menos similares, sin embargo, en términos generales se puede decir que todos sirven para lo mismo: leer y ejecutar los programas. Evidentemente algunos modelos tienen más capacidades que otros, en cuanto a memoria, velocidad, periféricos, etc. En el diseño de un sistema con microcontrolador se pueden elegir dispositivos de cualquiera de los fabricantes y satisfacer la necesidad del sistema en cuestión.

2.3.5.5 Disponibilidad de los microcontroladores en el mercado.

Algunos microcontroladores son más fáciles de encontrar que otros. No es conveniente emprender un proyecto basado en cierto microcontrolador que escasea en el mercado local y/o global ya que podría no satisfacer nuestra demanda y detener el proyecto.

2.3.5.6 Disponibilidad de información y herramientas de desarrollo.

Actualmente la mayoría de los fabricantes de microcontroladores ofrecen información suficiente para entender la operación y funcionamiento de sus dispositivos. El punto débil de algunos fabricantes es la pobre distribución de sus herramientas de desarrollo (programadores, emuladores, software, etc.) o bien su alto costo.

2.3.5.7 Precio.

Hay personas y/o empresas que tienen los recursos para pagar lo que sea necesario cuando se trata de tecnología, sin embargo, la mayoría prefiere ahorrarse dinero, por lo que el precio resulta un punto a favor en la selección de un fabricante de microcontroladores, el modelo de microcontrolador específico que se debe elegir dependiendo de la aplicación, aunque se puede hacer un juicio sobre la capacidad de un microcontrolador tomando en cuenta su capacidad de memoria, la cantidad de puertos de entrada y salida, los periféricos, la velocidad a la cual ejecuta las instrucciones, etc. (Salazar López & Millán Almaraz, 2015).

2.3.6 Lenguajes de programación

2.3.6.1 Definición

Los lenguajes de programación es una forma de comunicación que existe entre hombre máquina, se basa en una serie de instrucciones u órdenes, que permiten al programador escribir algoritmos que cumplen con realizar una tarea en específico, los lenguajes de programación están formados por un conjunto de símbolos llamados alfabeto, reglas gramaticales y reglas semánticas, la programación es el proceso de escritura y compilación del código fuente.

2.3.6.2 Clasificación de los lenguajes de programación

Históricamente siempre han surgido nuevos lenguajes de programación, por lo que se pueden clasificar por generaciones, cada una de estas representando los puntos clave del lenguaje surgido en su época. Los equipos de hardware ya han pasado por cuatro generaciones conocidas.

La primera generación se trata de las primeras computadoras que se programaban utilizando directamente un código basado en sistema binario, esto es llamado lenguaje de bajo nivel, y es necesario que el programador comprenda completamente la lógica de la máquina para programarla, sin embargo, cada computadora tenía su propia estructura de programación por lo que no solían ser computables los lenguajes de programación entre sistemas, eran de uso privativo y cada empresa conocía el funcionamiento de su máquina. Con el tiempo se fue reduciendo este problema con la llegada de la segunda generación.

Los lenguajes de la segunda generación se incorporan con el lenguaje ensamblador que se refiere al uso de estructuras legibles y por lo tanto simplifica las instrucciones por parte del programador, sin embargo, todavía es un lenguaje de bajo nivel que requerirá de cierto conocimiento en lógica de programación para transmitir instrucciones a la máquina, pero esto se reduce con la llegada de la tercera generación.

En esta generación se desarrollan los lenguajes de alto nivel que sustituyen las instrucciones simbólicas (ensamblador), por códigos que utilizan las matemáticas y las instrucciones son palabras muy parecidas al lenguaje humano, fueron creadas para ser comprendido por el usuario común no programador, y así darle solución problemas de procesamiento de datos de una forma rápida y sencilla. Muchos de estos lenguajes se utilizan actualmente, algunos de ellos son C, C++, C#, PHP, Java, Python, etc.

La cuarta generación se trata de la programación orientada a objetos, estos tienen una estructura muy similar al idioma inglés, algunas de sus características es que permiten el acceso a base de datos, capacidades gráficas, generación de código automáticamente y la programación que permite una interfaz visual, tal como Visual Basic o SQL, una de sus características más importantes es la reducción a la dificultad por parte del programador, volviéndose intuitivo y produciendo un menor agotamiento.

Actualmente existe un debate entre si llamar 5ta generación a los lenguajes de inteligencia artificial, ya que en 1982 Japón presentó un proyecto de desarrollo de la 5ta generación de máquinas basadas en inteligencia artificial tanto en el hardware como en

el software, sin embargo, este proyecto fracaso y dio como resultado el debate de la 5ta generación.

Los lenguajes de programación también se pueden clasificar por nivel de abstracción, donde siempre salen a relucir dos categorías de esto, los lenguajes de alto y bajo nivel, en general esto refiere a la arquitectura de computación donde los lenguajes con bajo nivel de atracción tienen una alta computabilidad con los computadores, ya que la atracción entre el lenguaje máquina y el lenguaje de programación está muy reducida, muy por el contrario, entre más alto sea el nivel de atracción del lenguaje, más será necesaria la utilización de un intérprete que este traduzca las instrucciones al lenguaje máquina, que como se dijo anteriormente está basado en sistema binario, sin embargo, entre más alto sea el nivel de atracción de un lenguaje, este, será más amigable con el programador que uno de bajo nivel.

También se pueden clasificar por paradigmas, se le llama paradigma de programación al método que consiste en llevar a cabo cómputo y la forma en que las tareas deben estructurarse y organizarse, un paradigma de programación está delimitado por su uso, porque nuevos paradigmas siempre están surgiendo aportando nuevas y mejores soluciones a problemas. Un ejemplo de paradigma de programación es la programación orientada a objetos (Rauch-Hindin, 1989).

2.3.6.3 Lenguaje de programación C

C es un lenguaje de programación de propósito general desarrollado por Dennis Ritchie entre 1969 y 1972 en los laboratorios Bell de AT&T, fue desarrollado como una actualización del lenguaje de programación B de ahí la C, en 1973 el lenguaje C fue tan potente que se volvió el lenguaje de programación con el que se escribió el código de la mayor parte del Kernel Unix, uno de los primeros núcleos de sistemas operativos.

2.3.6.4 C como lenguaje estándar

En la década de los 80, C se había vuelto el lenguaje de programación predominante en microcomputadoras, y fue en este tiempo que IBM comenzó a usar este lenguaje de programación en el desarrollo de sus equipos, esto hizo que la popularidad de C incrementara notablemente.

2.3.6.5 Lenguaje de programación C++

En la década de los 80's, el lenguaje de programación alcanzo su nivel más alto de popularidad, esto hizo que Bjarne Stroustrup comenzara a trabajar en los laboratorios

Bell para añadir funcionalidades de orientación de objetos a C, el resultado de esto fue un lenguaje muy parecido a C que nombraron C++, hoy en día este es uno de los lenguajes de programación más usados, y es tan común que los sistemas operativos de Microsoft lo siguen utilizando.

2.3.6.6 ANSI C

En 1978 Ritchie y Kernighan publicaron la primera edición del lenguaje de programación C, conocido como la biblia de C, en este desarrollo se implementaron características novedosas tales como estructuras y se corrigieron los tipos de datos, también se corrigió un problema de ambigüedad que existía con los operadores aritméticos, esta nueva versión fue conocida como ANSI C, en 1989, el Instituto Nacional Estadounidense de Estándares (por sus siglas en inglés ANSI), estableció un estándar para el lenguaje de programación C llamado “Lenguaje de programación C” ANSI X3.159-1989, esta es la versión conocida oficialmente como ANSI C, en 1990 el estándar ANSI C fue adoptado por la Organización Internacional para la Estandarización (por sus siglas en inglés ISO), el estándar ISO/IEC 9899:1990, cambiando algunas de sus funcionalidades, como los prototipos de las funciones y un procesador mejorado (Ritchie, 1993) & (Kernighan & Ritchie, 1991).

2.3.6.7 Python

Python es un lenguaje de programación interpretado, de alto nivel de abstracción, cuya premisa principal es ofrecer un código que sea legible y una sintaxis

comprensible para el programador. Es un lenguaje de programación multipropósito ya que soporta orientación a objetos, programación imperativa y programación funcional.

Se trata de un lenguaje multiplataforma y dinámico, es administrado por la Python Software Fundación, posee una licencia de código abierto, la Python Software Foundation License y esta es compatible con la Licencia pública general GNU, pero solamente de la versión 2.1.1 en adelante.

Python fue creado a finales de los 80's por Guido Van Rossum en la CWI, como un sucesor al lenguaje de programación ABC. En 1991, Van Rossum publicó el código de la versión 0.9.0, en esta versión aparece un sistema de módulos adaptado de modulo-3, Van-Rossum describe este como una de las mayores unidades de la programación en Python. En 1994 se formó el foro oficial de Python donde se discutían temas relacionados a este lenguaje de programación, marcando un antes y un después de este lenguaje, se volvió tan popular que el crecimiento de usuarios de este lenguaje fue exponencial, actualmente es uno de los lenguajes de programación más utilizados.

Python fue diseñado para ser leído con facilidad, una de las características de este lenguaje es el uso de palabras en lugar de símbolos como se hace de manera tradicional en lenguajes como C, también cuenta la peculiaridad de que no es necesario poner un terminador por cada línea de código ";" pero es necesario que por cada función exista una tabulación en la siguiente línea de código, de modo que obliga al programador a tener todo el espacio organizado, y esto también aplica para funciones creadas por el usuario.

En Python no es necesario declarar el tipo de variables, ya que se declaran de forma automática al escribir asignar un valor a esta, por ejemplo; en C se debe declarar una variable colocando el tipo de variable antes de la variable a asignar el valor, $int\ n = 10$; sin embargo, en Python al declararse la variable de forma automática no es necesario esto; $n = 10$, con solo esta línea el intérprete del lenguaje Python ya sabe que se trata de un valor numérico y de un entero así que le asigna el tipo de dato int , pero, si el usuario requiere que la variable sea de un tipo específico, Python también permite cambiar o forzar el tipo de variable colocando el tipo de variable y entre paréntesis la variable, $int(n)$.

Python también contiene una forma sencilla de acceder a los datos de tipo vector, para ello se manejan dependiendo del modo de acceso, para los datos variables se suelen utilizar listas, para los datos constantes se utilizan las tuplas y los diccionarios contienen índices para acceder a elementos organizados,

Para declarar una lista se utilizan los corchetes $a = []$, para declarar una tupla se utilizan paréntesis $b = ()$, y para declarar un diccionario se utilizan llaves $c = \{\}$, cada uno de este tipo de conjunto de elementos puede almacenar en el mismo conjunto elementos de diferentes tipos de variables, es posible crear una lista con valores del tipo $a = [10, 45.3, "letras"]$.

Así como las listas pueden contener otras listas dentro de sus elementos, las tuplas pueden contener otras tuplas dentro de sus elementos, y los diccionarios pueden contener listas y tuplas en sus elementos, la diferencia entre las listas y las tuplas es que

las listas pueden cambiar de modo que se pueden agregar, cambiar o borrar elementos de la lista, pero, en la tupla una vez declarada, ya no es posible realizar un cambio, por eso, la mayoría de personas prefieren utilizar listas en lugar de tuplas, un dato interesante es que se pueden utilizar listas de listas para simular matrices y se realiza de forma efectiva, sin ninguna biblioteca añadida como math o numpy.

Por defecto en Python al declarar una variable se utiliza lo que se conoce como memoria dinámica, que puede estar cambiando de valor, tanto la declaración como la liberación de memoria es un proceso automático, esto facilita enormemente la escritura de código ya que permite al programador no preocuparse por las direcciones de memoria ya que se asignan automáticamente y permite ahorrarse unas cuantas líneas de código.

Uno de los añadidos de Python es que también permite seleccionar la cantidad de variables necesarias dentro de una cadena, utilizando el símbolo “:”, de tal modo que permite interactuar con varias variables en una sola iteración, esto es una característica muy importante y muy utilizada por los programadores en Python.

El uso de módulos es lo que define al lenguaje de programación Python como un lenguaje multicriterio ya que al añadir o importar bibliotecas o módulos permite transformar el lenguaje de modo que se vuelve compatible con todo tipo de procesos, actualmente los programadores pueden crear sus propios módulos, subirlos a la plataforma y permitir que otros usuarios los importen a sus programas, esta característica ha hecho que Python tenga una gran comunidad de personas dispuestas a

realizar módulos con el fin de minimizar la cantidad de trabajo de otros usuarios, o simplemente cuando no se sabe cómo funciona una u otra función se añade el módulo y se importa al programa (Python.org, 2019).

2.3.7 Tratamiento digital de señales

El tratamiento digital de señales es un área de la ciencia y la ingeniería que provee de herramientas necesarias para realizar análisis de señales, lo cual permite encontrar información de interés dada una señal.

2.3.7.1 Señal

Se define como una señal a cualquier magnitud física que varía con el tiempo, el espacio, cualquier otra variable o variables independientes. Matemáticamente se describe una señal como una función de una o más variables independientes. Por ejemplo:

$$\begin{aligned} S_1(t) &= 5t \\ S_2(t) &= 20t^2 \end{aligned} \quad (2.1)$$

Describen dos señales, una que varía una que varía linealmente con la variable dependiente t (tiempo) y una segunda que varía cuadráticamente con t .

2.3.7.2 Discretización de una señal

Las señales discretas en el tiempo solo están definidas en determinados instantes específicos de tiempo. Dichos instantes de tiempo no tienen que ser equidistantes, aunque, en la práctica, con normalidad están igualmente espaciados para facilitar los cálculos. La señal de la siguiente ecuación

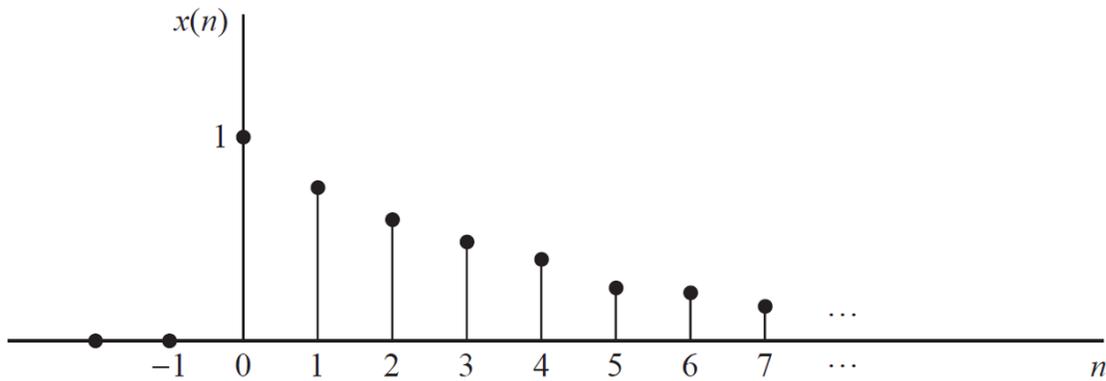
$$x(t_n) = e^{-|t_n|}, n = 0, \pm 1, \pm 2 \quad (2.2)$$

Es un ejemplo de una señal discreta en el tiempo. Si utilizamos el índice n para los instantes de tiempo discretos como la variable independiente, el valor de la señal será una función de una variable entera (es decir, será mediante una secuencia de números). Por tanto, una señal discreta en el tiempo se puede representar matemáticamente mediante una secuencia de números reales o complejos. Con el fin de resaltar la naturaleza discreta de una señal, se denotará dicha señal como $x(n)$ en lugar de como $x(t)$. Si los instantes de tiempo t_n están igualmente espaciados (es decir $t_n = nT$), también se utiliza la notación $x(nT)$. Por ejemplo, la secuencia:

$$x(n) = \begin{cases} 0.8^n, & \text{si } n > 0 \\ 0, & \text{en otro caso} \end{cases} \quad (2.3)$$

Es una señal discreta en el tiempo, que se ha representado gráficamente en la Figura 2.6, las señales discretas en el tiempo pueden originarse de dos formas:

Seleccionando valores de una señal analógica en instantes discretos de tiempo. Este proceso se denomina muestreo que se verá más adelante. Todos los instrumentos de medición realizan medidas en intervalos de tiempo regulares, proporcionan señales discretas en el tiempo, por ejemplo, la señal $x(n)$ de la Figura 2.7 puede obtenerse muestreando la señal de la Figura 2.6. Acumulando una variable en un periodo de tiempo. Por ejemplo, el número de coches que pasan por una calle determinada en una hora, dan lugar a señales discretas en el tiempo.



Donde $x(n) = 0.8^n$ para $n > 0$ y $x(n) = 0$ para $n < 0$.

Figura 2.6: Representación gráfica de una señal discreta en el tiempo.

2.3.7.3 Teorema de muestreo

Dada una señal analógica se puede seleccionar el periodo de muestreo T o lo que es equivalente la frecuencia de muestreo F_s , para ello se debe de disponer de alguna información sobre las características de la señal a muestrear, concretamente se necesita conocer información general acerca del contenido en frecuencia de la señal, normalmente dicha información está disponible, por ejemplo, es sabido que las principales componentes de frecuencia de una señal de voz se encuentran por debajo de los 3000 Hz . Por otro lado, las señales utilizadas por dispositivos como los electrodomésticos contienen componentes de frecuencia de hasta los 5 MHz . La información contenida de tales señales se encuentra en las amplitudes, frecuencias y fases de las diversas componentes de frecuencia, pero la información detallada de las características de dichas señales no estará disponible antes de obtener las señales, de hecho, el propósito del procesamiento de señales a extraer dicha información detallada,

sin embargo, si se conoce la frecuencia máxima de la clase general de señales (por ejemplo la clase de señales de video, etc.), se puede especificar la frecuencia de muestreo necesaria para convertir las señales analógicas en señales digitales.

Por ejemplo, si se tiene cualquier señal analógica, esta puede representarse como una suma de sinusoides de diferentes amplitudes, frecuencias y fases, es decir.

$$x_a(t) = \sum_{i=1}^N A_i \cos(2\pi F_i t + \theta_i) \quad (2.4)$$

Donde N indica el número de componentes de frecuencia. Todas las señales, tales como las de voz y video, se prestan a dicha representación en cualquier intervalo de tiempo pequeño. Normalmente, las amplitudes, las frecuencias y fases varían lentamente con el tiempo de un intervalo a otro.

2.3.7.4 Teorema Nyquist-Shannon

Si la frecuencia máxima de una señal analógica $x_a(t)$ es $F_{max} = B$ y la señal se muestrea con una frecuencia de F_s , esta frecuencia F_s debe ser 2 veces más grande a la F_{max} de $x_a(t)$, esto se puede representar mediante.

$$F_s = 2F_{max} \quad (2.5)$$

Es necesario siempre realizar un muestreo con una frecuencia 2 veces más alta de la frecuencia de la señal original para no perder datos y de este modo poder reconstruir la señal digital con los datos de la señal original.

2.3.7.5 Muestreo de señales analógicas

Una señal analógica puede ser procesada por un sistema discreto realizando un proceso llamado muestreo el cual se trata de la toma de datos de una señal continua en tiempo discreto (véase discretización de una señal) con una frecuencia determinada, la calidad de la señal resultante es proporcional a la frecuencia de la toma de datos, en la siguiente figura se puede ver un ejemplo del muestreo de una señal.

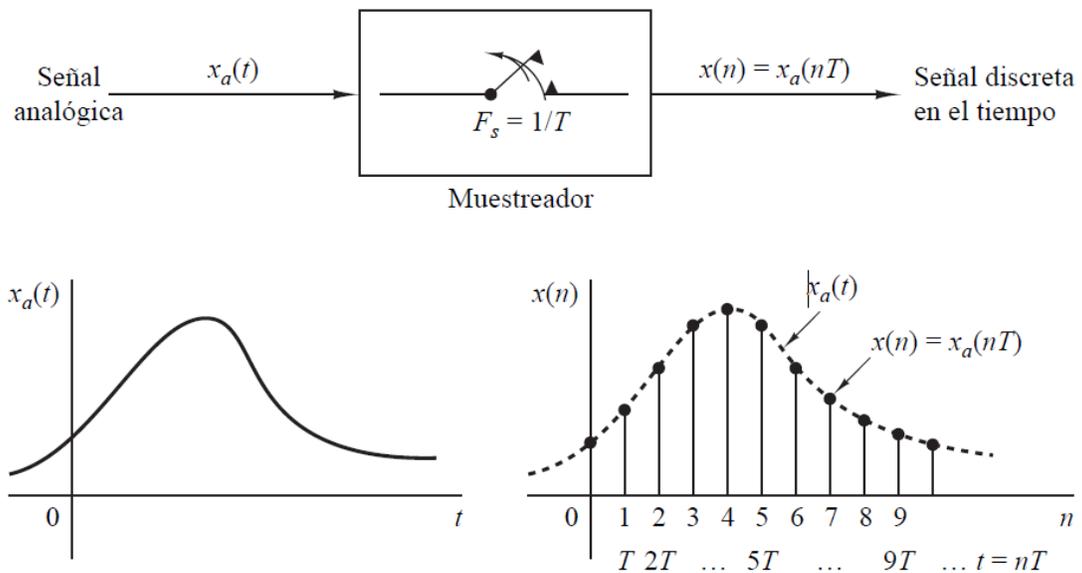


Figura 2.7: Muestreo periódico de una señal analógica.

2.3.7.6 Análisis de frecuencia para señales continuas en el tiempo

Existen diferentes herramientas de análisis de señales continuas en el tiempo, ejemplos de estas señales pueden ser ondas cuadradas, ondas rectangulares, ondas triangulares, y por su puesto sinusoidales, cada una de estas ondas se puede representar matemáticamente mediante la serie de Fourier.

2.3.7.7 Serie de Fourier

La serie de Fourier es una suma lineal ponderada de exponenciales complejas o sinusoidales armónicamente relacionadas. Jean Baptise Joseph Fourier (1768 - 1830), fue un matemático francés, que utilizó estas expansiones de series trigonométricas para describir el fenómeno de conducción del calor y la distribución de la temperatura en los cuerpos. Las técnicas matemáticas que desarrolló se han aplicado en una amplia variedad de problemas, en muchos campos diferentes que incluye disciplinas como la óptica, las vibraciones en sistemas mecánicos, la teoría de los sistemas y el electromagnetismo.

Una combinación lineal de exponenciales complejas armónicamente relacionadas de la forma:

$$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{j2\pi k F_0 t} \quad (2.6)$$

Es una señal periódica de periodo fundamental $T_p = 1/F_0$, por lo tanto, se puede pensar en las siguientes exponenciales:

$$\{e^{j2\pi k F_0 t}, \quad k = 0, \pm 1, \pm 2, \dots\} \quad (2.7)$$

Como en los “bloques de construcción” básicos que nos permiten construir señales periódicas de diversos tipos mediante la correcta elección de la frecuencia fundamental y los coeficientes $\{c_k\}$. F_0 determina el periodo fundamental $x(t)$ y los coeficientes $\{c_k\}$ especifican la forma de la señal.

Si se tiene una determinada señal periódica $x(t)$ de periodo T_p . Se puede representar la señal periódica mediante la serie de la Ecuación 2.6, conocida como serie de Fourier, donde la frecuencia fundamental F_0 se selecciona para ser recíproco del periodo T_p , para determinar la expresión de los coeficientes $\{c_k\}$, en primer lugar, se multiplica ambos lados de la Ecuación 2.7 por la exponencial compleja.

$$e^{-j2\pi l F_0 t} \quad (2.8)$$

Donde l es un entero y luego se integra en ambos lados de la ecuación resultante en un único periodo, es decir, entre 0 y T_p , o de forma más general, entre t_0 y $t_0 + T_p$, donde t_0 es un valor inicial arbitrario y matemáticamente conveniente, de ese modo se tiene:

$$\int_{t_0}^{t_0+T_p} x(t) e^{-j2\pi l F_0 t} dt = \int_{t_0}^{t_0+T_p} e^{-j2\pi l F_0 t} \left(\sum_{k=-\infty}^{\infty} c_k e^{+j2\pi k F_0 t} \right) dt \quad (2.9)$$

Para evaluar la integral del lado derecho de la Ecuación 2.7, se intercambia el orden de la sumatoria y la integral combinando las dos exponenciales, por lo tanto:

$$\sum_{k=-\infty}^{\infty} c_k \int_{t_0}^{t_0+T_p} e^{j2\pi F_0(k-l)t} dt = \sum_{k=-\infty}^{\infty} c_k \left[\frac{e^{j2\pi F_0(k-l)t}}{j2\pi F_0(k-l)} \right]_{t_0}^{t_0+T_p} \quad (2.10)$$

Para $k \neq l$, el lado derecho de la Ecuación 2.8 evaluado para los límites inferior y superior, t_0 y $t_0 + T_p$, respectivamente da cero, por otro lado, si $k = l$, se tiene:

$$\int_{t_0}^{t_0+T_p} dt = t \Big|_{t_0}^{t_0+T_p} = T_p \quad (2.11)$$

Por lo tanto, se puede reducir a:

$$\int_{t_0}^{t_0+T_p} x(t) e^{-j2\pi l F_0 t} dt = c_l T_p \quad (2.12)$$

Y así la expresión para los coeficientes de Fourier en función a la señal periódica dada son:

$$c_l = \frac{1}{T_p} \int_{t_0}^{t_0+T_p} x(t) e^{-j2\pi l F_0 t} dt \quad (2.13)$$

Puesto que t_0 es un valor arbitrario, esta integral puede evaluarse para cualquier intervalo de longitud T_p , es decir, para cualquier intervalo igual al periodo de la señal $x(t)$. En consecuencia, la integral para los coeficientes de la serie de Fourier se escribirá como sigue:

$$c_l = \frac{1}{T_p} \int_{T_p} x(t) e^{-j2\pi l F_0 t} dt \quad (2.14)$$

Una cuestión importante que surge en la representación de la señal periódica $x(t)$ mediante la serie de Fourier es si la serie converge o no a $x(t)$ para todo valor de t , es decir, si la señal $x(t)$ y su representación como serie de Fourier.

$$\sum_{k=-\infty}^{\infty} c_k e^{j2\pi k F_0 t} \quad (2.15)$$

Son iguales para cualquier valor de t . las condiciones de Dirichlet garantizan que la serie dada por la Ecuación 2.10 será igual a $x(t)$, excepto en los valores de t para los que $x(t)$ es discontinua. En dichos valores de t , la Ecuación 2.10 converge al valor medio de la discontinuidad. Las condiciones de Dirichlet son:

- La señal $x(t)$ tiene un numero finito de discontinuidades en cualquier periodo.
- La señal $x(t)$ contiene el numero finito de máximos y mínimos en cualquier periodo.
- La señal $x(t)$ es absolutamente integrable en cualquier periodo, es decir,

$$\int_{T_p} |x(t)| dt < \infty \quad (2.16)$$

Todas las señales periódicas de interés practico satisfacen estas condiciones. La condición menos restrictiva, que establece que la señal tiene energía finita en un periodo.

$$\int_{T_p} |x(t)|^2 dt < \infty \quad (2.17)$$

Garantiza que la energía en la señal diferencia.

$$e(t) = x(t) - \sum_{k=-\infty}^{\infty} c_k e^{j2\pi k F_0 t} \quad (2.18)$$

Es cero, aunque $x(t)$ y su serie de Fourier no sean iguales para todos los valores de t . la Ecuación 2.16 implica a la Ecuación 2.17, pero no al contrario. También tanto la Ecuación 2.17 como las condiciones de Dirichlet son suficientes, pero no necesarias (es decir, existen señales que tienen representación en serie de Fourier, pero no satisfacen estas condiciones).

En resumen, si $x(t)$ es periódica y satisface las condiciones de Dirichlet, se puede representar mediante una serie de Fourier como la dada en la Ecuación 2.6, donde los coeficientes están especificados por la Ecuación de síntesis:

$$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{j2\pi k F_0 t} \quad (2.19)$$

Y la ecuación de análisis:

$$c_k = \frac{1}{T_p} \int_{T_p} x(t) e^{-j2\pi k F_0 t} dt \quad (2.20)$$

En general los coeficientes de Fourier c_k son valores complejos, además, es fácil demostrar que si la señal periódica es real c_k y c_{-k} son complejos conjugados, por lo tanto, si:

$$c_k = |c_k|e^{j\theta_k} \quad (2.21)$$

Entonces

$$c_{-k} = |c_k|^{-j\theta_k} \quad (2.22)$$

En consecuencia, la serie de Fourier también puede representarse en la forma

$$x(t) = c_0 + 2 \sum_{k=1}^{\infty} |c_k| \cos(2\pi k F_0 t + \theta_k) \quad (2.23)$$

Donde c_0 es un valor real cuando $x(t)$ es real.

Por último, se debe indicar que se puede obtener cualquier otra forma de la serie de Fourier expandiendo la función coseno de la Ecuación 2.23 como:

$$\cos(2\pi k F_0 t + \theta_k) = \cos 2\pi k F_0 t \cos \theta_k - \sen 2\pi k F_0 t \sen \theta_k \quad (2.24)$$

En consecuencia, se puede escribir al Ecuación 2.23 de la forma.

$$x(t) = a_0 + \sum_{k=1}^{\infty} (a_k \cos 2\pi k F_0 t - b_k \sen 2\pi k F_0 t) \quad (2.25)$$

Donde:

$$a_0 = c_0, \quad a_k = 2|c_k|\cos\theta_k, \quad b_k = 2|c_k|\sin\theta_k \quad (2.26)$$

Las Ecuaciones 2.19, 2.23 y 2.24 son tres formas equivalentes de la representación de la serie de Fourier de una señal periódica real.

2.3.7.8 Clasificación de señales en el dominio de la frecuencia

Se conoce como frecuencia a la unidad de medida que representa a la cantidad de veces que se produce un fenómeno dentro de un intervalo de tiempo establecido, en análisis de señales en dominio del tiempo se mide en el Hercios y se representa con Hz, se puede conocer la frecuencia dividiendo 1 segundo entre el tiempo transcurrido desde que inicia una onda hasta que termina (periodo “ T ”) Ecuación 2.26.

$$f = \frac{1s}{T} \quad (2.27)$$

De la misma forma como se clasifican las señales en el dominio del tiempo, se pueden clasificar en el dominio de la frecuencia, si una señal es periódica y tiene su espectro de densidad de potencia concentrado alrededor de 0 se dice que es una señal de baja frecuencia si por el contrario el espectro de densidad de potencia se concentra en números elevados, usualmente los miles, se dice que se trata de una señal de alta frecuencia (Romero Troncoso, 2007).

Capítulo 3. Materiales y Métodos

3.1 Introducción

Todas las tarjetas de desarrollo e investigación de los FPGA están divididas en dos partes, la unidad de procesamiento principal FPGA y la etapa de periféricos externos que sirven para realizar experimentos con el dispositivo, la tarjeta FPGA seleccionada para este experimento fue el DE0-Nano-SoC de la marca Altera, ya que cuenta con un tamaño relativamente pequeño y lo mínimo indispensable para que funcione.

Se realizaron varias pruebas para determinar que sensor utilizar en este proyecto, primero se utilizó el sensor de aceleración ADIS16223 de la marca Analog Devices, este sensor es un acelerómetro triaxial digital que tiene una resolución de 16 bits por comunicación SPI, cuenta también con una frecuencia de muestreo de 20 Khz, sin embargo, no se logró una comunicación exitosa con dicho sensor.

Luego se intentó con otro sensor el LIS3DSHTR de la marca STMicroelectronics, mas no se pudo realizar ningún experimento porque el sensor adquirido resulto ser demasiado pequeño para soldarlo por medios convencionales y no se contaba en el laboratorio con la herramienta necesaria para ello.

Por último, se contempló el uso de un sensor de la marca Invensense modelo MPU9250 el cual es no comercial, dicho dispositivo cuenta con la capacidad de realizar monitoreos de aceleración con resolución de 16 bits y una frecuencia de hasta 44 KHz de muestreo.

3.2 Desarrollo del dispositivo

El dispositivo se desarrolló utilizando como base una tarjeta FPGA que tiene la función de realizar tareas programadas mediante software, la tarjeta adquirida fue la DE0-Nano-SoC de la marca Altera que tiene las siguientes especificaciones:

- Dispositivo Altera Cyclone® V SE 5CSEMA4U23C6N
- Configuración de dispositivo serial – EPCS
- USB-Blaster II incluido para la programación; Modo JTAG
- 2 botones tipo push
- 4 interruptores deslizables
- 8 LED verdes
- Tres fuentes de generador de reloj a 50MHz
- 2 puertos de 40 pines GPIO
- Un puerto de expansión de Arduino Uno Versión 3, compatible con la mayoría de los shield de Arduino.
- Un puerto de 10 pines analógicos para Arduino.
- Un conversor A/D, 4 pines para interfaz SPI con FPGA

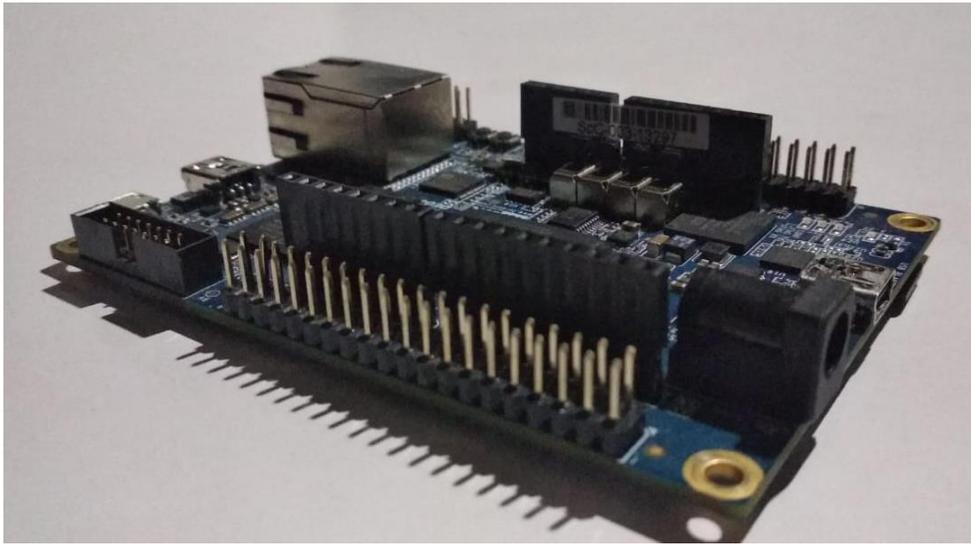


Figura 3.1: Tarjeta FPGA De0-Nano-SOC.

Para realizar la programación es necesario conocer a fondo el dispositivo y el protocolo de comunicación de este, en este caso se utiliza un sensor de la marca Invensense, modelo MPU-9250 que cuenta, con un acelerómetro triaxial con las siguientes características:

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
Full-Scale Range	AFS_SEL=0		±2		g
	AFS_SEL=1		±4		g
	AFS_SEL=2		±8		g
	AFS_SEL=3		±16		g
ADC Word Length	Output in two's complement format		16		bits
Sensitivity Scale Factor	AFS_SEL=0		16,384		LSB/g
	AFS_SEL=1		8,192		LSB/g
	AFS_SEL=2		4,096		LSB/g
	AFS_SEL=3		2,048		LSB/g
Initial Tolerance	Component-Level		±3		%
Sensitivity Change vs. Temperature	-40°C to +85°C AFS_SEL=0 Component-level		±0.026		%/°C
Nonlinearity	Best Fit Straight Line		±0.5		%
Cross-Axis Sensitivity			±2		%
Zero-G Initial Calibration Tolerance	Component-level, X,Y		±60		mg
	Component-level, Z		±80		mg
Zero-G Level Change vs. Temperature	-40°C to +85°C		±1.5		mg/°C
Noise Power Spectral Density	Low noise mode		300		µg/√Hz
Total RMS Noise	DLPFCFG=2 (94Hz)			8	mg-rms
Low Pass Filter Response	Programmable Range	5		260	Hz
Intelligence Function Increment			4		mg/LSB
Accelerometer Startup Time	From Sleep mode		20		ms
	From Cold Start, 1ms V _{DD} ramp		30		ms
Output Data Rate	Low power (duty-cycled)	0.24		500	Hz
	Duty-cycled, over temp		±15		%
	Low noise (active)	4		4000	Hz

Tabla 3.1. Especificaciones del acelerómetro MPU 9250.

Un giroscopio triaxial con las siguientes características:

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
Full-Scale Range	FS_SEL=0		±250		°/s
	FS_SEL=1		±500		°/s
	FS_SEL=2		±1000		°/s
	FS_SEL=3		±2000		°/s
Gyroscope ADC Word Length			16		bits
Sensitivity Scale Factor	FS_SEL=0		131		LSB/(°/s)
	FS_SEL=1		65.5		LSB/(°/s)
	FS_SEL=2		32.8		LSB/(°/s)
	FS_SEL=3		16.4		LSB/(°/s)
Sensitivity Scale Factor Tolerance	25°C		±3		%
Sensitivity Scale Factor Variation Over Temperature	-40°C to +85°C		±4		%
Nonlinearity	Best fit straight line; 25°C		±0.1		%
Cross-Axis Sensitivity			±2		%
Initial ZRO Tolerance	25°C		±5		°/s
ZRO Variation Over Temperature	-40°C to +85°C		±30		°/s
Total RMS Noise	DLPFCFG=2 (92 Hz)		0.1		°/s-rms
Rate Noise Spectral Density			0.01		°/s/√Hz
Gyroscope Mechanical Frequencies		25	27	29	KHz
Low Pass Filter Response	Programmable Range	5		250	Hz
Gyroscope Startup Time	From Sleep mode		35		ms
Output Data Rate	Programmable, Normal mode	4		8000	Hz

Tabla 3.2. Especificaciones del giroscopio MPU 9250.

Un magnetómetro con las siguientes características:

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
MAGNETOMETER SENSITIVITY					
Full-Scale Range			±4800		μT
ADC Word Length			14		bits
Sensitivity Scale Factor			0.6		μT / LSB
ZERO-FIELD OUTPUT					
Initial Calibration Tolerance			±500		LSB

Tabla 3.3. Especificaciones del magnetómetro MPU 9250.

3.3 Comunicación dispositivo-sensor

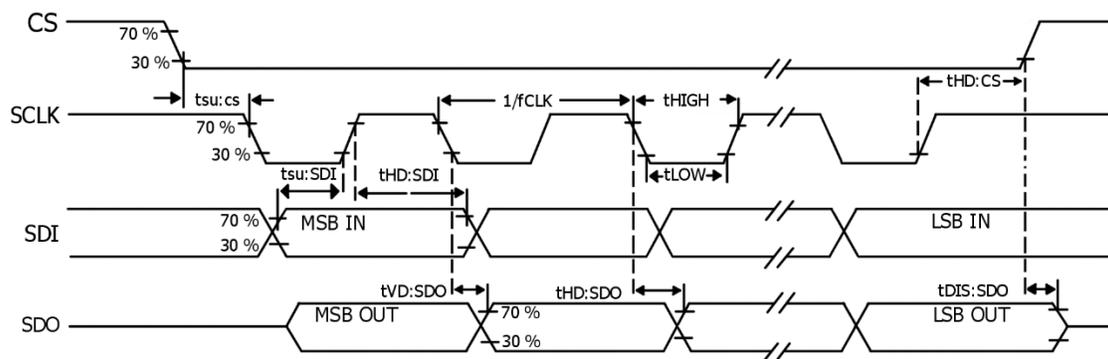


Figura 3.2: Protocolo de comunicación SPI modo 3 (Invensense, 2016).

Este sensor es digital, se comunica mediante el protocolo SPI modo 3 (Figura 3.2), esto quiere decir que la comunicación con el mismo es mediante palabras binarias, el dispositivo maestro, en este caso el FPGA debe mandar una palabra binaria conocida por el sensor para que el sensor pueda responder con el dato solicitado. Esto se logra en FPGA a través de FSM o Maquinas de Estados Finitos (véase máquinas de estados finitos), sin embargo, de esto se hablará en otro momento. Retomando el tema, el sensor cuenta con un glosario de palabras binarias las cuales sirven para comunicarse de forma efectiva con el dispositivo maestro, en la siguiente tabla se pueden ver las más importantes:

Registro (HEX)	Descripción del registro
3B	Aceleración en eje x High
3C	Aceleración en eje x Low
3D	Aceleración en eje y High
3E	Aceleración en eje y Low
3F	Aceleración en eje z High
40	Aceleración en eje z Low
43	Giroscopio en eje x High
44	Giroscopio en eje x Low
45	Giroscopio en eje y High
46	Giroscopio en eje y Low
47	Giroscopio en eje z High
48	Giroscopio en eje z Low

Tabla 3.4. Registros de comunicación SPI

Como se puede apreciar en la tabla anterior el sensor necesita recibir dos palabras de 8 bits (High y Low), en total 16 bits, para responder con el dato solicitado. Donde High significa que son los 8 bits más significativos y Low los 8 bits menos significativos.

3.4 Programación del sensor:

Antes de pasar a hacer nada, es necesario que el dispositivo maestro tenga una correcta programación para comunicarse exitosamente con el sensor, y no solo eso, sino que tenga la capacidad de guardar en memoria los datos obtenidos, para ello se realizó un programa VHD basado en una FSM que controla la comunicación entre el sensor y el FPGA (véase programación en FPGA), el cual, contiene un diccionario con todas las palabras binarias del sensor y que hace selección de cada una por tiempos controlados, también contiene un módulo de memoria FIFO que se encarga de almacenar los datos obtenidos del sensor.

3.5 Comunicación dispositivo-computadora:

Para que todo lo anterior funcione es necesario que se controle todo minuciosamente, por lo que se optó por incluir otro dispositivo el cual se encarga de realizar la petición de los datos de vibraciones por bloques de memoria, en este caso se utilizó una memoria que tiene espacio suficiente para almacenar 120 000 datos, que corresponden a 60 000 para acelerómetro y 60 000 para giroscopio con una frecuencia de 1 Khz (En este punto del experimento si se estaban almacenando los datos de rotación.), por ello, se utilizó una computadora personal con el sistema operativo Ubuntu y un programa de petición de datos basado en Python, sin embargo, la comunicación mediante el PC es por el puerto serial el cual utiliza un protocolo diferente de comunicación (véase protocolo de comunicación serial), por lo que se agregó otro modulo al programa VHD del que se habló en el tema anterior, que cubre la comunicación serial, así como también, un diccionario de palabras reservadas para iniciar y detener el proceso, quedando la máquina de estados de la siguiente manera

Por último, lo que queda es hacer una máquina de estados finitos maestra que se encargue de controlar a todas las otras máquinas y hacer que todas las conexiones entre ellas funcionen correctamente, en este caso la máquina de estados finitos maestra se encarga de realizar las conexiones entre la máquina del sensor, la memoria fifo, el módulo de comunicación por puerto serial y el decodificador, lo que da como resultado la máquina de firmware definitivo:

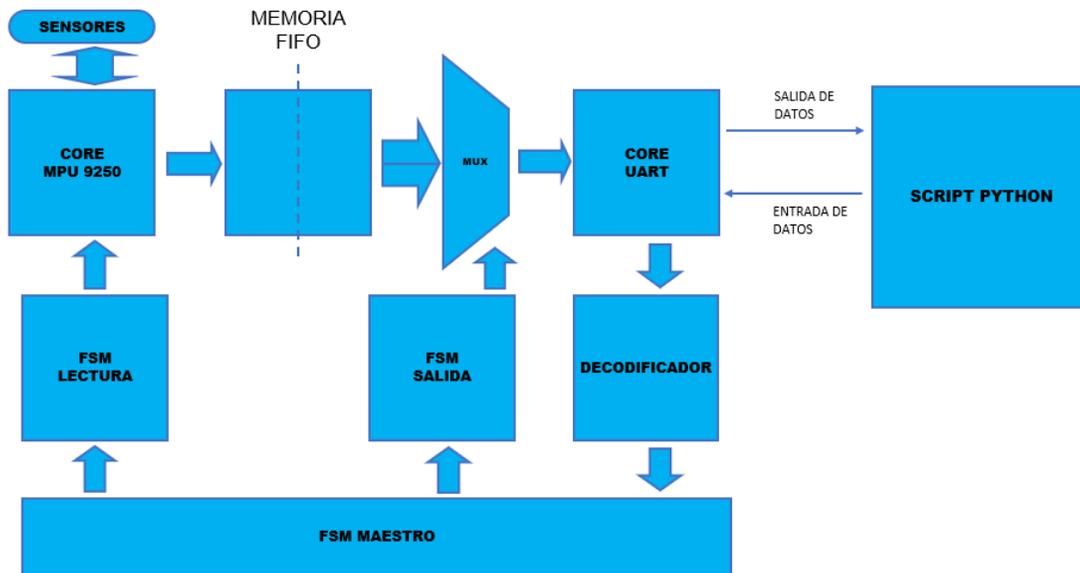


Figura 3.3: Máquina de estados finitos del firmware del dispositivo.

3.6 Validación del adquisidor de datos de aceleración

Luego de todo lo implementado anteriormente, se programó el FPGA con el firmware definitivo y se comenzaron a realizar pruebas de funcionamiento, Las pruebas de laboratorio se realizaron mediante los datos generados con el sensor en estado de reposo y los datos generados con él se sensor con vibraciones controladas, para ello, se realizó una mesa vibratoria que se compone de una bocina, un circuito disparador, una superficie plana y un generador de frecuencia.

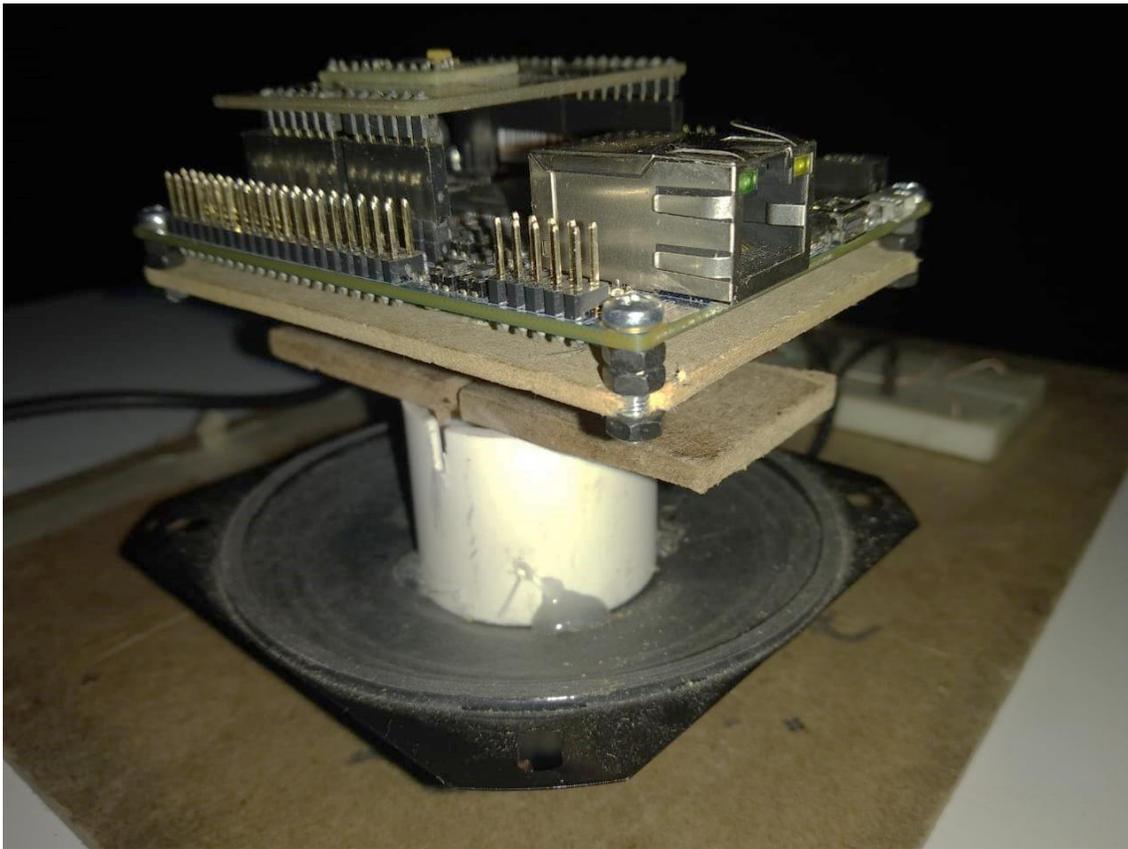


Figura 3.4: Mesa vibratoria vertical.

Esto permite generar un movimiento mecánico que corresponde al eje vertical del dispositivo, ya que en campo solo se tomará en cuenta el eje vertical de las

estructuras, lo cual corresponde al eje Z del sensor, dicho movimiento se comporta como una vibración, cuya frecuencia se controla mediante el generador de frecuencias que se encuentra conectado al circuito de disparo que convierte la señal obtenida del generador a señal en pulsos que hacen que la bocina emita un sonido con una frecuencia controlada.

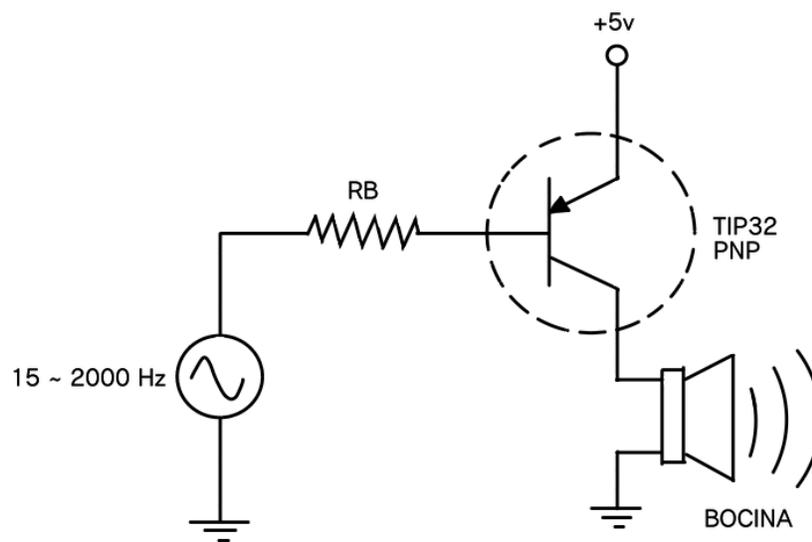


Figura 3.5: Circuito controlador de mesa de vibraciones.

Gracias a esto se tuvo acceso a datos que permitieron observar las frecuencias fundamentales de las vibraciones en cuestión, así como también la señal generada por el sensor en estado reposo.

Observaciones del sensor en estado de reposo:

La meta de realizar monitoreos en estado de reposo es descartar señales sucias tales como ruidos y perturbaciones externas que se puedan estar presentando en el

sensor, se hicieron muchas pruebas en estado de reposo, pero en este capítulo, solo se mostraran 3 tipos particulares.

3.6.1 Prueba en estado de reposo positivo:

La prueba de reposo en positivo es donde se posiciona el sensor en la superficie de la cama de vibraciones con el eje vertical apuntando hacia arriba, es decir al contrario de la fuerza que ejerce la gravedad, esto da como resultado una señal la cual registra

1g:

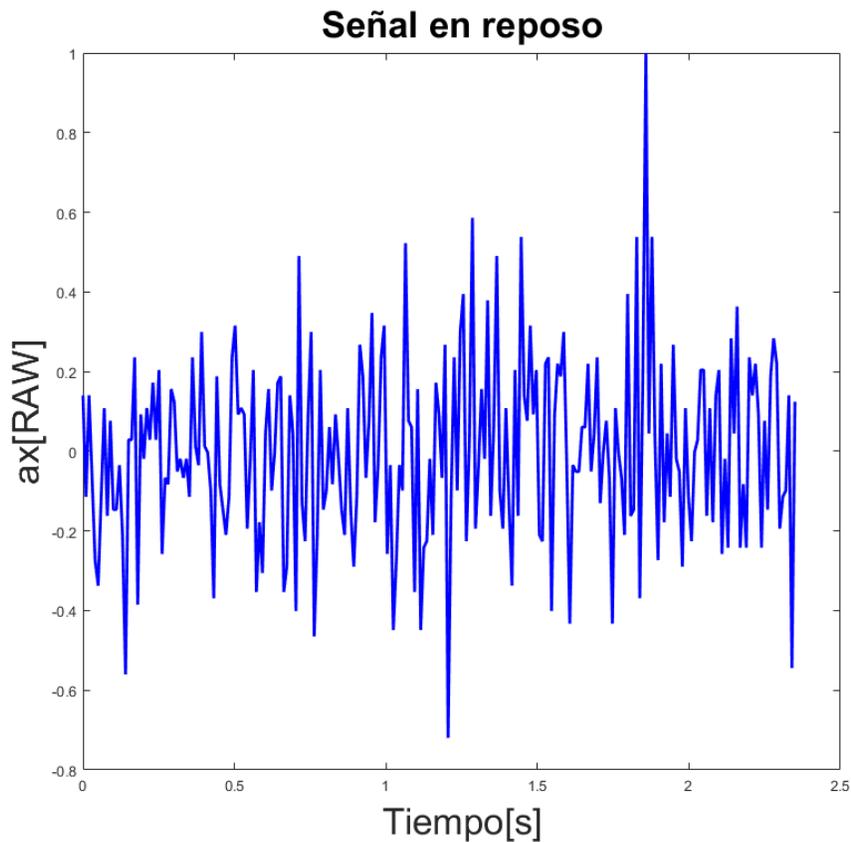


Figura 3.6: Señal recibida por el sensor en estado de reposo eje Z positivo.

3.6.2 Prueba en estado de reposo negativo

El estado de reposo en negativo quiere decir que se coloca el sensor de forma que el eje vertical apunte hacia abajo, exactamente al contrario del experimento anterior, lo que da como resultado una señal de $-1g$:

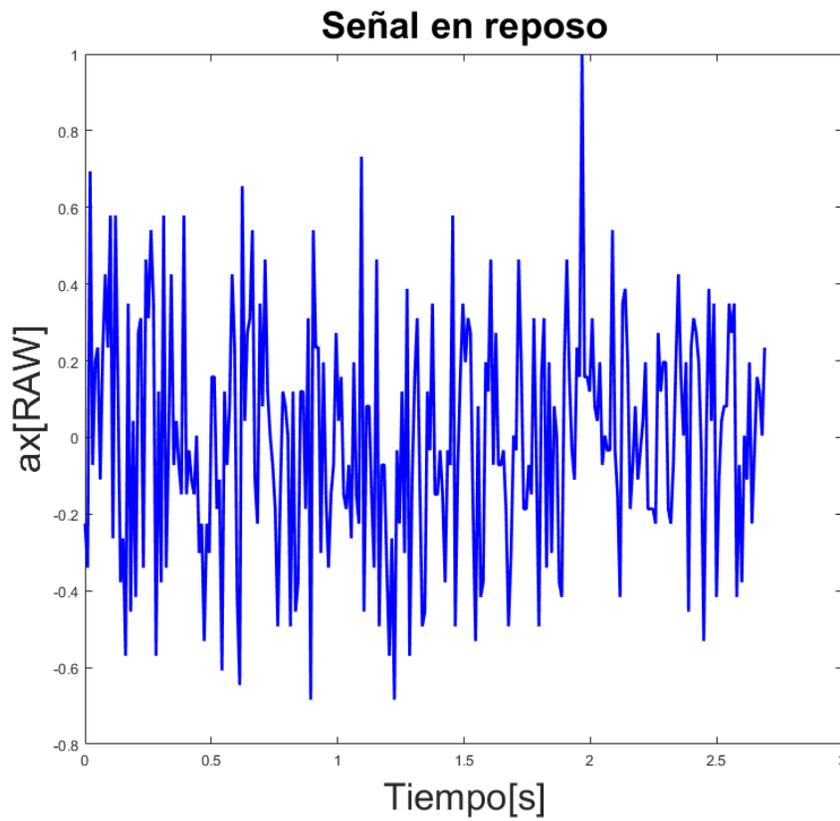


Figura 3.7: Señal recibida por el sensor en estado de reposo eje Z negativo.

3.6.4 Pruebas en laboratorio con excitación a 20Hz

Se programó la mesa de vibraciones con una frecuencia de 20Hz y se colocó el sensor encima de la superficie plana, este se fija mediante tornillos a la mesa para garantizar una sujeción firme, de este modo se impide que se disipen los movimientos creados por la bocina y pasan directamente al sensor, en dicha prueba se obtuvo el siguiente resultado:

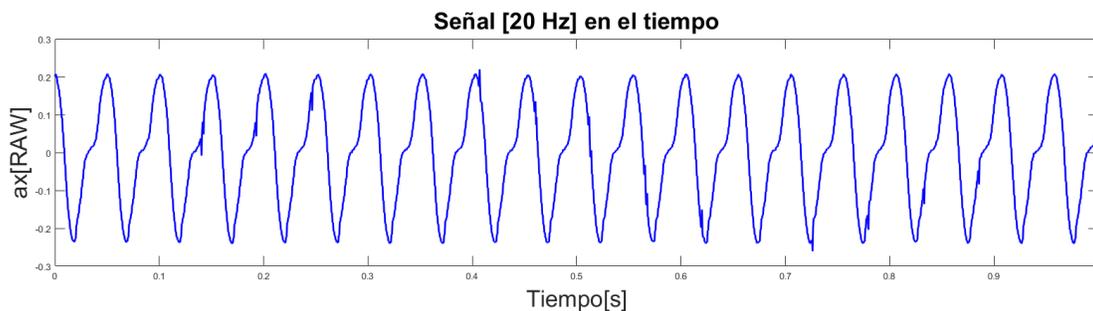


Figura 3.9: Señal con una excitación mecánica a 20 Hz en el dominio del tiempo.

Lo que se puede apreciar en la imagen anterior es un segundo de la forma de onda de la señal inducida al sensor en el dominio del tiempo, en este caso al ser una señal con una frecuencia de 20 Hz y una frecuencia de muestreo de 2000 Hz se puede apreciar muy bien la curvatura de la señal, al cambiar el dominio del tiempo por el dominio de la frecuencia se puede observar lo siguiente:

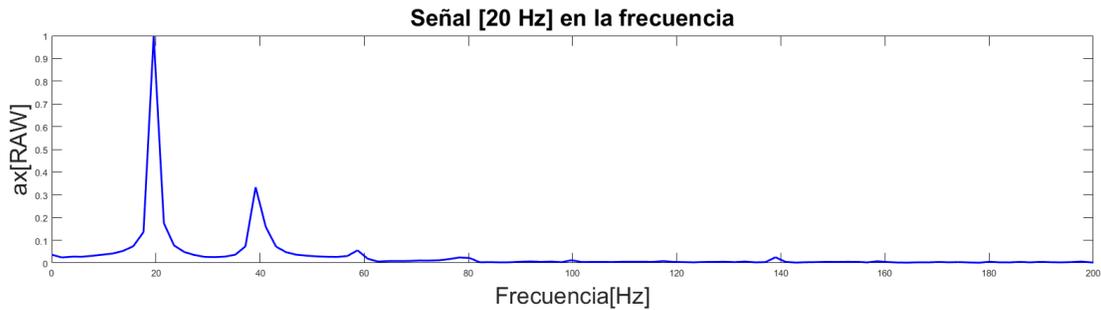


Figura 3.10: Señal con una excitación mecánica a 20 Hz en el dominio de la frecuencia.

Como se puede observar en la figura anterior se puede apreciar perfectamente que la vibración mecánica de la mesa tiene una frecuencia de 20 Hz pues el pico más alto de la señal indica esa frecuencia.

3.6.5 Pruebas en laboratorio con excitación a 100Hz

Se realizó una prueba con la mesa de vibraciones con una excitación de 100 Hz y se puede observar el resultado en la siguiente figura:

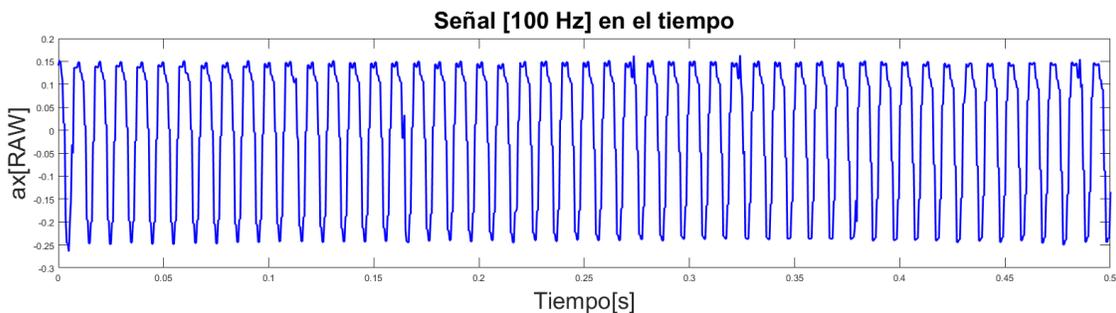


Figura 3.11: Señal con una excitación mecánica a 100 Hz en el dominio del tiempo.

Aquí se muestra grafica donde se aprecia una señal un poco más rápida que la anterior, por lo que es normal que las ondas se encuentren más cercanas entre sí, al

igual que la Figura 3.9 fue muestreada a 2000 Hz, en la Figura 3.11 anterior se muestran solamente 0.5 segundos ya que un segundo de muestreo, las ondas se ven demasiado cercanas entre sí y esto no es agradable a la vista. Si se cambia al dominio de la frecuencia se obtiene lo siguiente:

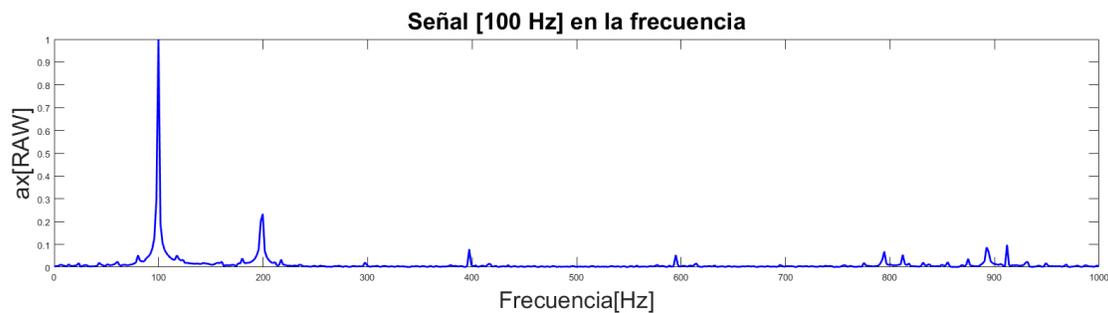


Figura 3.12: Señal con una excitación mecánica a 100 Hz en el dominio de la frecuencia.

Como se puede apreciar en la imagen anterior la frecuencia fundamental que aparece se encuentra muy cercana a los 100 Hz, esta señal se pudo obtener después de un simple análisis de la señal de la Figura 3.11, aplicando una transformada rápida de Fourier, permite obtener la frecuencia del objeto monitoreado.

3.7 Migración del dispositivo

Como se puede apreciar en el tema anterior se realizaron varias pruebas de laboratorio que avalan el funcionamiento del sensor que puede monitorear por un tiempo determinado una señal artificial con una frecuencia igual o menor a la mitad de la frecuencia de muestreo (véase teorema de Nyquist-Shannon), por lo tanto, es el momento idóneo para registrar señales reales, pero por cuestiones de procesamiento de datos, el manejo de memoria resulta inviable pues el tiempo que tarda registrando y entregando la información merma la frecuencia de muestreo con la que se monitorean los datos, ya que, en laboratorio se estaban haciendo los monitoreos con una frecuencia de 5 minutos entre los muestreos, esto es ineficiente ya que al llegar a campo es requerida la toma de datos por más de 20 segundos de forma continua además existe otro problema que es presentado por el adaptador de corriente, ya que para que el FPGA funcione correctamente es necesario que se le suministren 2A con una tensión de 5v por lo que se descartó el uso del FPGA y se optó por utilizar un microcontrolador.

El microcontrolador seleccionado fue el microcontrolador de la marca Atmel modelo Atmega328p por razones muy simples, es de bajo costo, fácil de conseguir y se programa relativamente fácil utilizando la interfaz de Arduino.

3.8 Algoritmo del Firmware en microcontrolador

Uno de los problemas comunes de los desarrolladores que trabajan con sistemas mínimos basados en Arduino, es programar el microcontrolador utilizando las bibliotecas que se pueden encontrar en la base de datos de arduino.com, el

inconveniente de esto es que al no ser un código propio, se trata de una “caja negra”, ya que los desarrolladores que crean estas bibliotecas suelen hacerlo atacando un problema en específico y se desconoce el funcionamiento de este, sin embargo la mayoría de programadores ya dan por sentado que las bibliotecas de la comunidad pueden solventar todos los problemas que se presenten, para mitigar este problema, en este proyecto se desarrolló un algoritmo propio que cumple específicamente con la lectura del sensores MPU9250 de forma eficiente sin recurrir a las bibliotecas de terceros, ya que funciona de forma nativa como se programó por primera vez en el dispositivo FPGA.

Como se comentó en el anterior párrafo el algoritmo es exactamente el mismo que en el FPGA salvo con algunos cambios menores como que ahora no se maneja espacio de memoria ya que el dispositivo tiene la capacidad de iniciar y cerrar el ciclo de monitoreo mediante palabras binarias recibidas por el puerto serial de este.

Esto permite que el mismo sensor funcione de la misma forma que en el FPGA, pero ahora en un microcontrolador, el dispositivo funciona con una frecuencia de muestreo de 2Khz y transmite a una velocidad de 1Mbps, para lograr esta hazaña fue necesario adquirir un cable de alta velocidad que fuera compatible con el dispositivo.

3.9 Importancia del caso de estudio

En el experimento de este proyecto se lleva a cabo el estudio de una viga IPR W12x22 de 12 m. de largo, Las vigas IPR son estructuras de acero que ofrecen una gran resistencia antes los movimientos estructurales, por ello son una excelente elección para la construcción de estructuras civiles, muchos puentes y edificios están contruidos con vigas IPR como base estructural, por ello se seleccionó una viga para llevar a cabo el experimento, las vigas IPR tienen una estructura como se presenta en la siguiente figura:

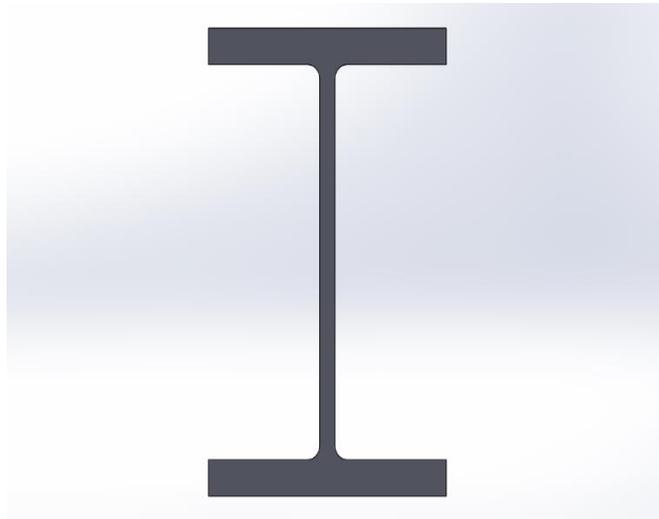


Figura 3.13: Estructura de una viga IPR.

Capítulo 4. Resultados y discusión

4.1 Introducción

Como se mencionó anteriormente el experimento consistió en realizar monitoreos de aceleración y GPS en una viga IPR en dos casos específicos; uno cuando recibe un golpe directo utilizando un martillo de caucho y el otro es donde la viga recibe una excitación externa moviéndola de forma controlada, en la viga se colocaron cinco sensores, tres de GPS y dos de aceleración, estos sensores se colocaron de la forma en cómo se presenta en la siguiente figura:

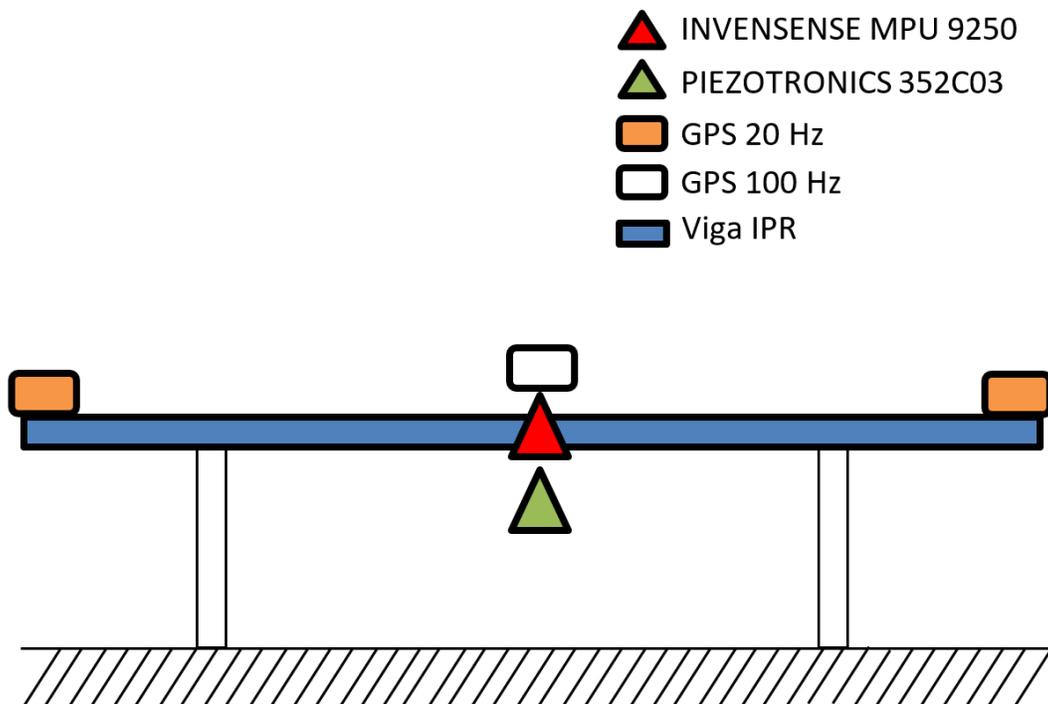


Figura 4.1: Diagrama de la ubicación de los sensores.

Cómo se puede apreciar en la figura anterior los dispositivos de medición de la aceleración se encuentran en el centro de la viga y los de GPS en los extremos para los

que tienen una frecuencia de muestreo de 20 Hz, por otra parte se encuentra el GPS con una frecuencia de muestreo de 100 Hz colocado al centro de la viga justo encima de los sensores de aceleración, todos estos instrumentos de medición permiten obtener los movimientos resultado de la deflexión que presenta la viga en estos dos casos particulares, cada uno de estos casos consistió en realizar un monitoreo con todos los sensores sincronizados mientras se llevaba a cabo el acto para la excitación de la viga.

4.1.1 Excitación de la viga por un golpe

Este caso en particular consiste en lo siguiente; la viga es excitada externamente con el golpe de un martillo de caucho, esto se lleva a cabo cada que los operadores de los sensores se sincronizan con los tiempos de medición, mientras que otra persona golpea en el extremo derecho con el martillo (véase la figura 4.2), esto ocasiona que la viga vibre por el impacto y dichas vibraciones son obtenidas por los sensores.

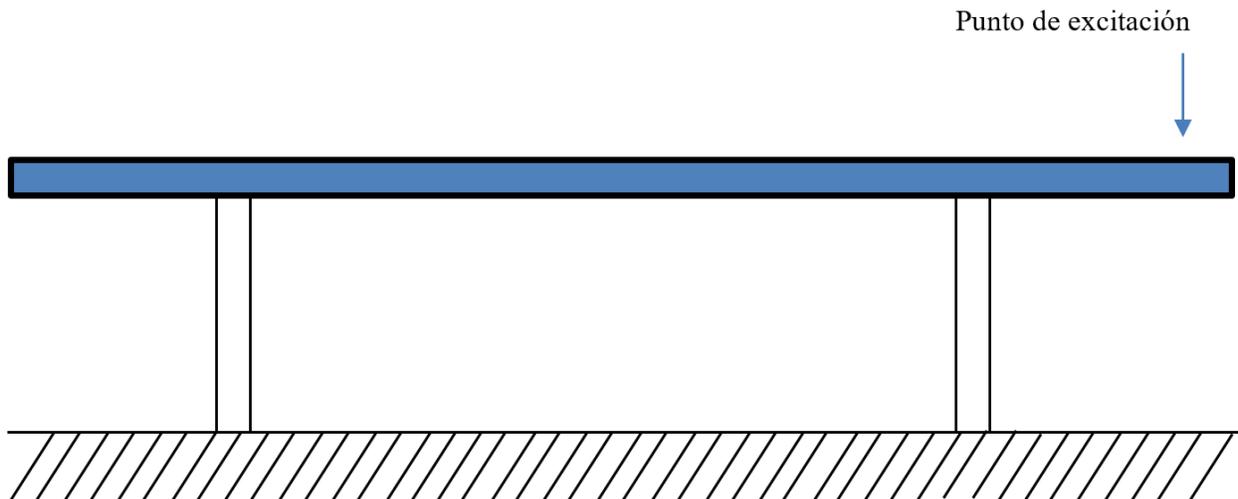


Figura 4.2: Punto de excitación en el experimento.

En este experimento hubo varios monitoreos, sin embargo, debido a problemas de sincronización entre los operadores de los estos, solo se analizaron los resultados que se obtuvieron en dos monitoreos específicos, los cuales fueron llamados; Martillo1 y Martillo 2 respectivamente.

El análisis de los datos fue el siguiente; en primera se necesitaba validar la viabilidad de utilizar un sensor no comercial como lo es el sensor MPU9250 por ello los datos obtenidos de este fueron comparados con los datos obtenidos por un sensor comercial de la marca Piezotronics 352C03 (véase Figura 4.3), del cual se tuvo acceso gracias al Dr. Jesús Martin Reyes Blanco de la universidad de Massachusetts – Campus Lowell de la facultad de Ingeniería Mecánica.



Figura 4.3: Sensor de aceleración Piezotronics 352C03 (Cortesía de pcbiezotronics.fr).

Gracias a ello se pudo realizar una lectura con ambos sensores funcionando al mismo tiempo, obteniendo los siguientes resultados:

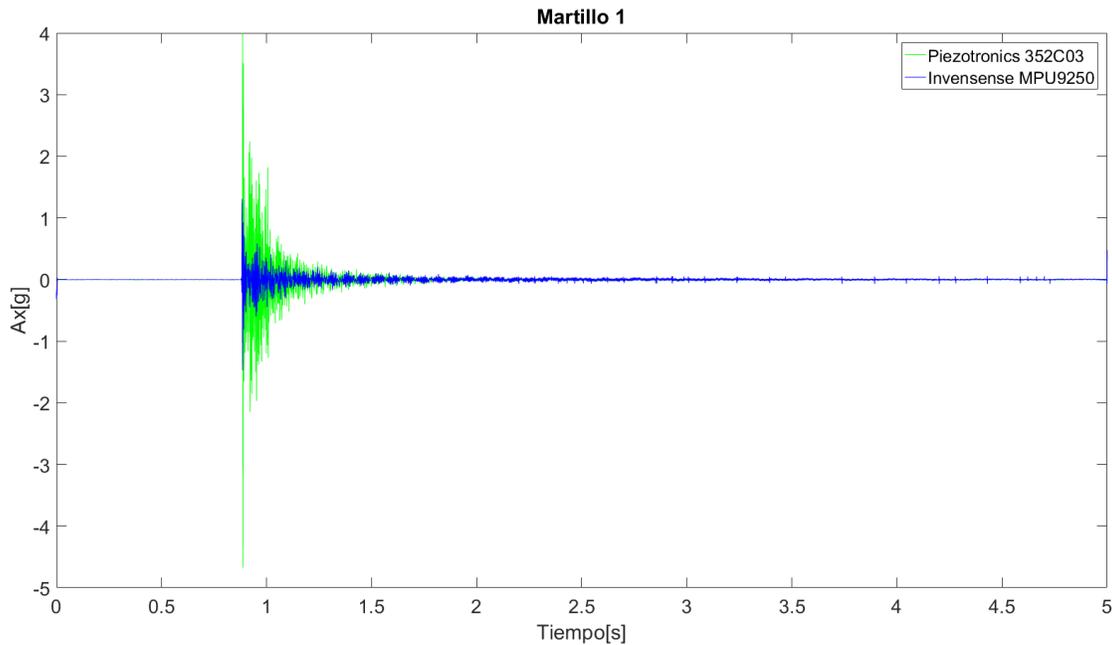


Figura 4.4: Datos de aceleración en “Martillo 1”, MPU9250 vs 352C03.

La gráfica que se muestra en la Figura 4.4 representa el impacto del martillo de caucho contra la viga en el ámbito del tiempo, donde los datos de la aceleración se representan en cantidad de g que cada una equivale a 9.81 m/s^2 ya que es el valor de la de la gravedad, el tiempo transcurrido está en segundos, todas las gráficas en este experimento reúnen datos de alrededor de 5 segundos de muestreo.

En la Figura 4.4 se puede observar algo muy interesante y es que en el momento en que la viga se encuentra en reposo la aceleración se mantiene oscilando entre valores constantes muy cercanos a cero, sin embargo, se puede apreciar con exactitud el momento en que la viga recibe un golpe, porque la señal de aceleración se satura y a continuación comienza a vibrar de forma libre amortiguada hasta que la viga vuelva al punto de reposo.

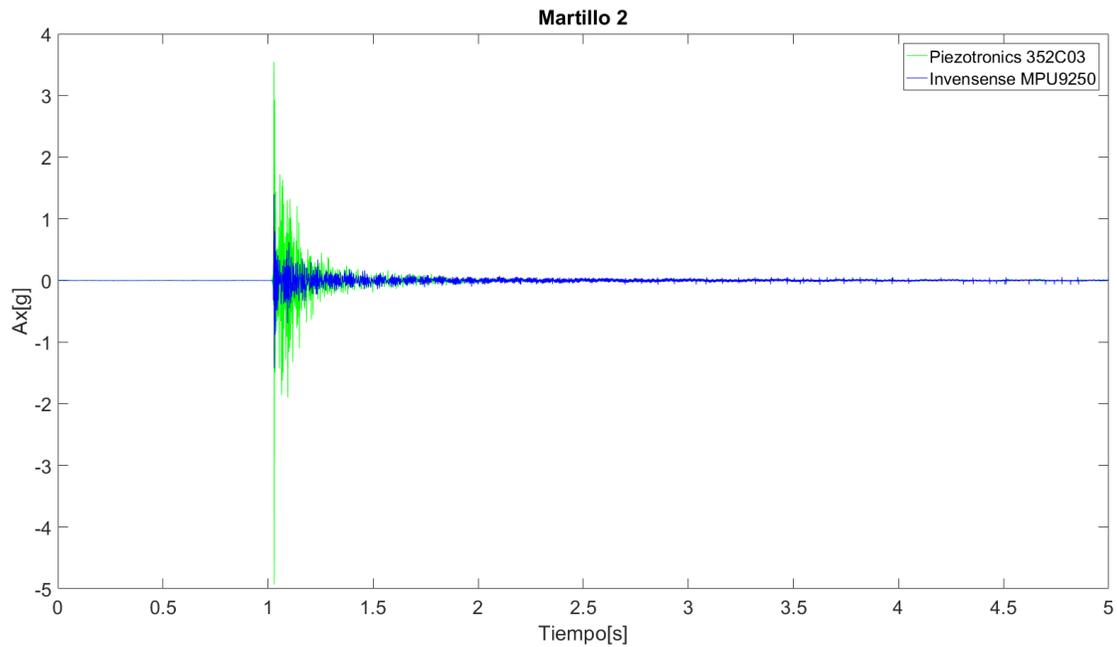


Figura 4.5: Datos de aceleración en “Martillo 2” MPU9250 vs 352C03.

En la Figura 4.5 se puede observar el comportamiento de la viga en la fase del experimento “martillo 2” donde es golpeada con el martillo de caucho en el extremo derecho de la viga mientras se realiza un monitoreo de cinco segundos, la línea verde representa los datos arrojados por el sensor Invensense MPU9250 mientras que la línea azul representa los datos del sensor Piezotronics 352C03, al igual que la Figura 4.4 aquí se observar el mismo comportamiento, calma, ruido y movimiento libre amortiguado.

4.1.2 La viga es excitada con un movimiento forzado.

En este caso la viga sufre una deflexión forzada por el movimiento que introduce manualmente una persona, levantando y bajando el extremo derecho de la viga simulando el movimiento natural de una señal sinusoidal en el tiempo, se le llamaran “Excitación de Onda” a cada uno de estos experimentos.

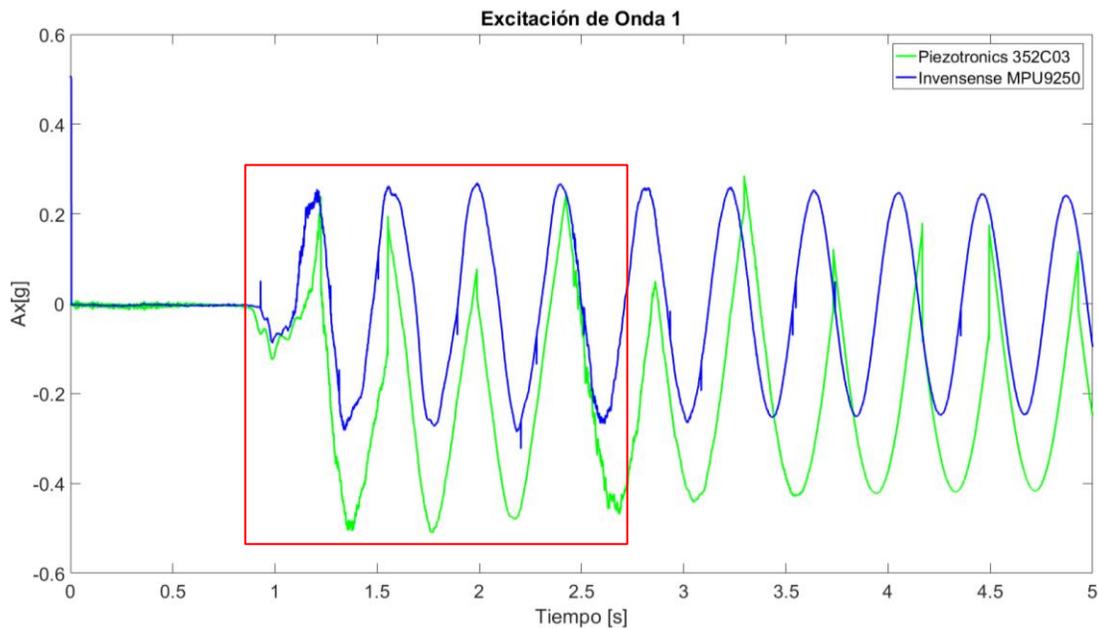


Figura 4.6: Datos de aceleración en “Excitación de Onda 1” MPU9250 vs 352C03.

En la Figura anterior se puede apreciar una gráfica correspondiente a la respuesta de la viga cuando se somete a la movimiento provocados por un una persona “Estación de Onda 1” donde la señal de color azul representa al sensor MPU9250 mientras que la verde a al sensor 352C03, la señal que se encuentra a la izquierda del cuadro rojo se encuentra en perfecto reposo, lo que se encuentra encerrado en el cuadro rojo representa a “Excitación de Onda 1” y el resto es el movimiento libre amortiguado de la viga.

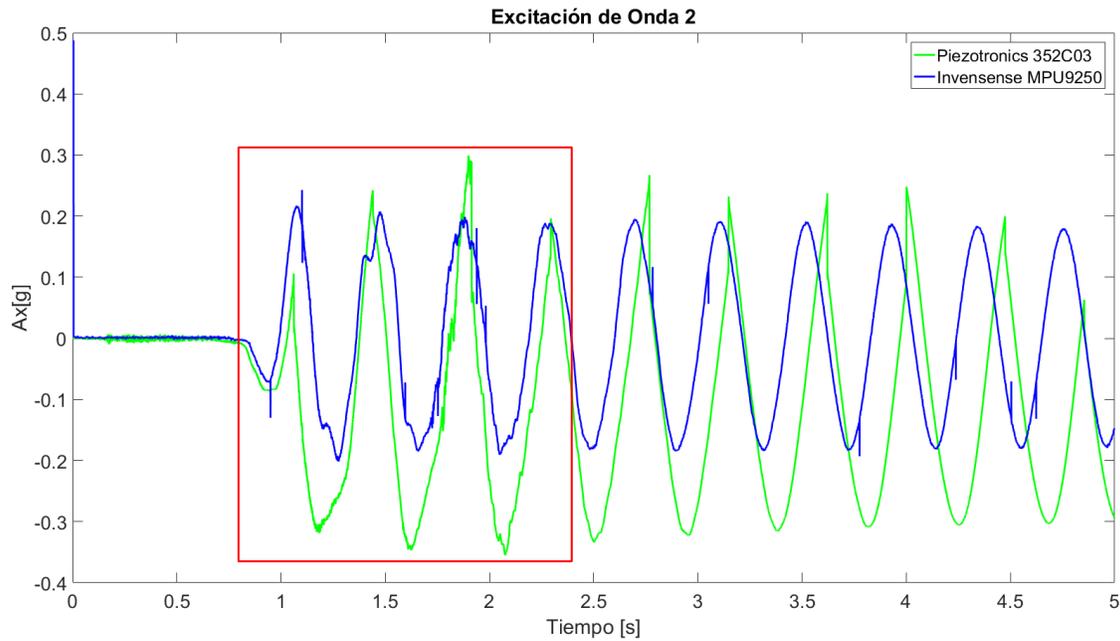


Figura 4.7: Datos de aceleración en “Excitación de Onda 2” MPU9250 vs 352C03.

Los datos que se pueden ver en la Figura anterior corresponden a la respuesta de la viga en el movimiento “Estación de Onda 2” que al igual que la Figura 4.8 se puede observar el mismo comportamiento, a la izquierda del cuadro rojo la viga está en reposo, dentro esta la perturbación externa “Excitación de Onda 2” y a la derecha el movimiento libre amortiguado.

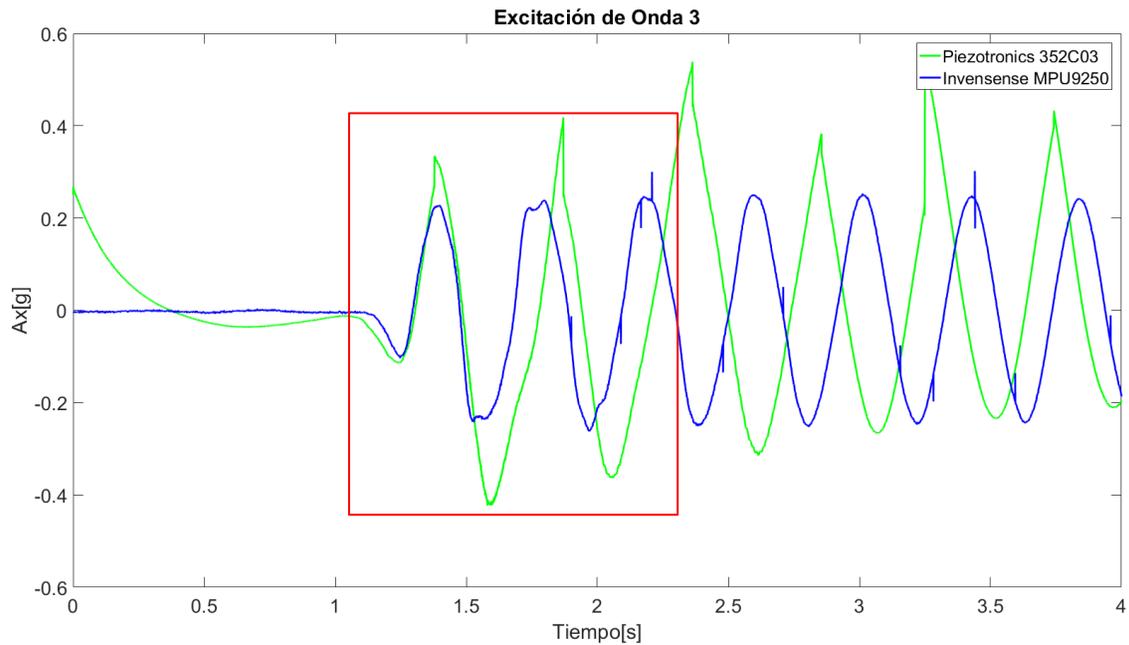


Figura 4.8: Datos de aceleración en “Excitación de Onda 3” MPU9250 vs 352C03.

En la Figura 4.8, se puede observar el comportamiento de la viga en el movimiento “Excitación de Onda 3”, aquí el sensor 352C03 comenzó a dar señales extrañas ya el comportamiento de su señal comenzó a tener una tendencia hacia arriba tal y como se puede ver en la señal verde de la Figura 4.8, pero, en el sensor MPU9250 no se observa dicho fenómeno. Sin embargo, se sabe que este tipo de dispositivos (sensores electrónicos) son muy sensibles a los cambios electromagnéticos y pueden presentar fallas.

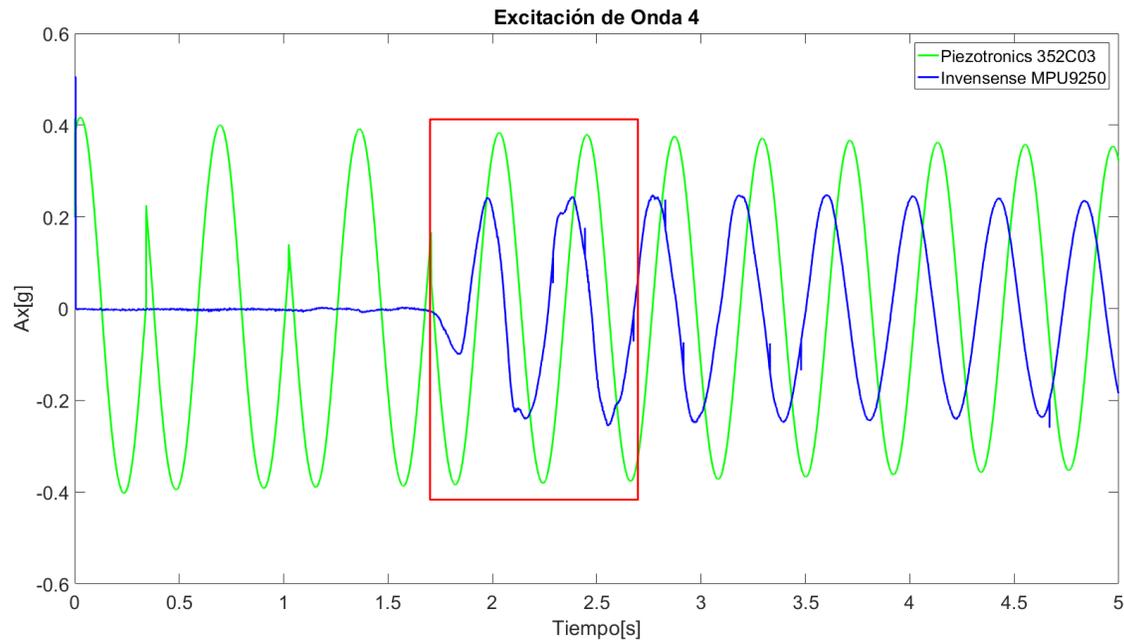


Figura 4.9: Datos de aceleración de “Excitación de Onda”, MPU9250 vs 352C03.

En este caso en particular (Figura 4.9) se puede observar el comportamiento de la viga solamente en los datos de la señal de color azul que corresponde al sensor MPU9250, ya que el sensor 352C03 solamente presentaba una señal causada por ruido electromagnético.

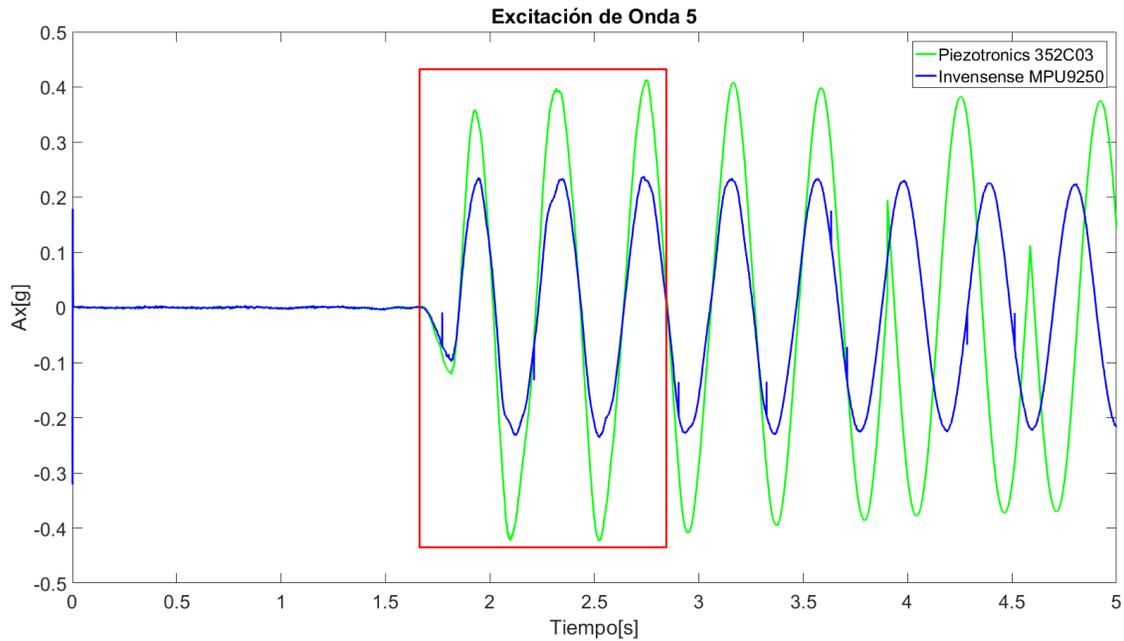


Figura 4.10: Datos de aceleración de “Excitación de Onda 5”, MPU9250 vs 352C03.

Como se puede apreciar en la Figura 4.10, ambos sensores estaban funcionando en condiciones óptimas, y el comportamiento de la viga se comportó exactamente igual que los casos presentados en las Figuras 4.6, 4.7, 4.8 y 4.9. gracias a estos experimentos se puede obtener el comportamiento mecánico de la viga.

4.2 Sensores de Aceleración vs GPS

Al mismo tiempo que se estaban registrando las lecturas de aceleración por los acelerómetros se encontraban funcionando tres GPS gracias al Dr. Guadalupe Esteban Vázquez Becerra, de los cuales solamente se tomarán en cuenta los datos de desplazamiento que es estaban obteniendo por el GPS a 100 Hz de muestreo, este análisis consiste en ver cuál es el comportamiento de la viga tanto en el sensor de aceleración MPU9250 como el sensor de desplazamiento GPS, a continuación, se presentarán figuras de las gráficas obtenidas.

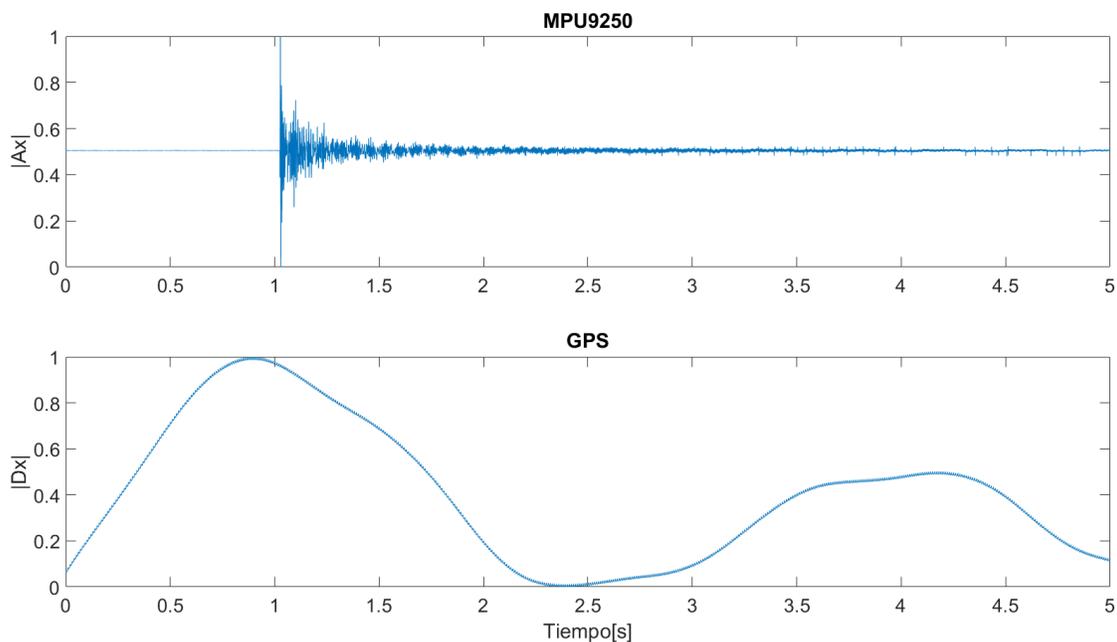


Figura 4.11: Datos obtenidos “martillo1” MPU9250 vs GPS.

En la figura anterior se puede observar el comportamiento de la viga con la perturbación “martillo1” donde los datos de la parte superior corresponden a la

aceleración leída por el sensor MPU9250 y la parte inferior son datos de Desplazamiento correspondientes al sensor GPS.

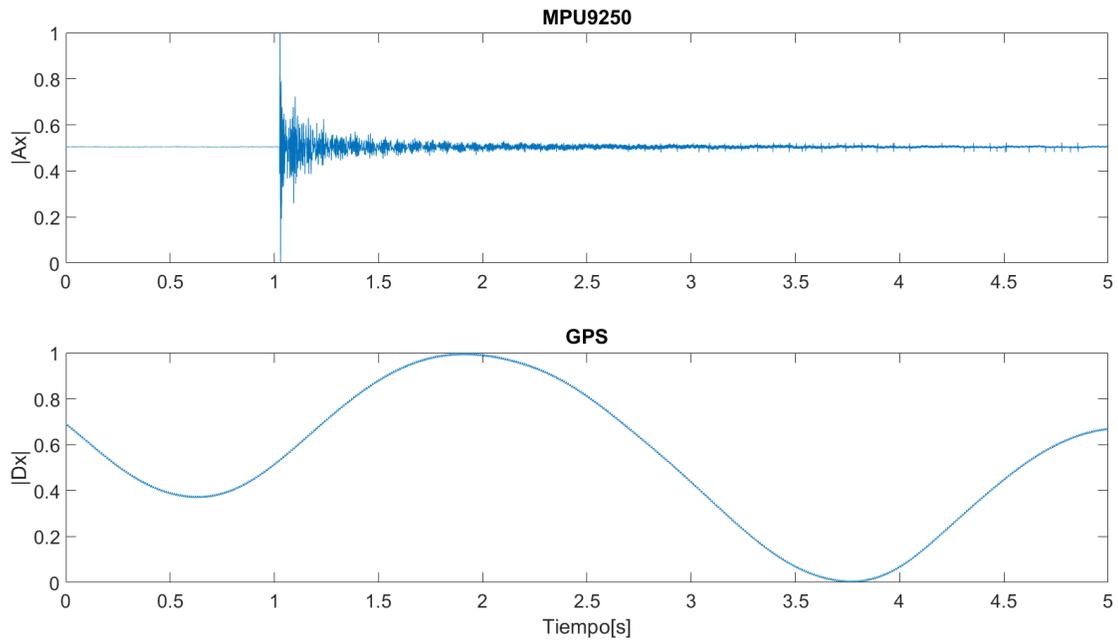


Figura 4.12: Datos obtenidos “martillo2” MPU9250 vs GPS.

Los datos presentados en la figura anterior corresponden a la gráfica obtenida por la lectura de los datos de aceleración (MPU9250) en parte superior y desplazamiento (GPS) en la parte inferior del experimento “martillo 2”.

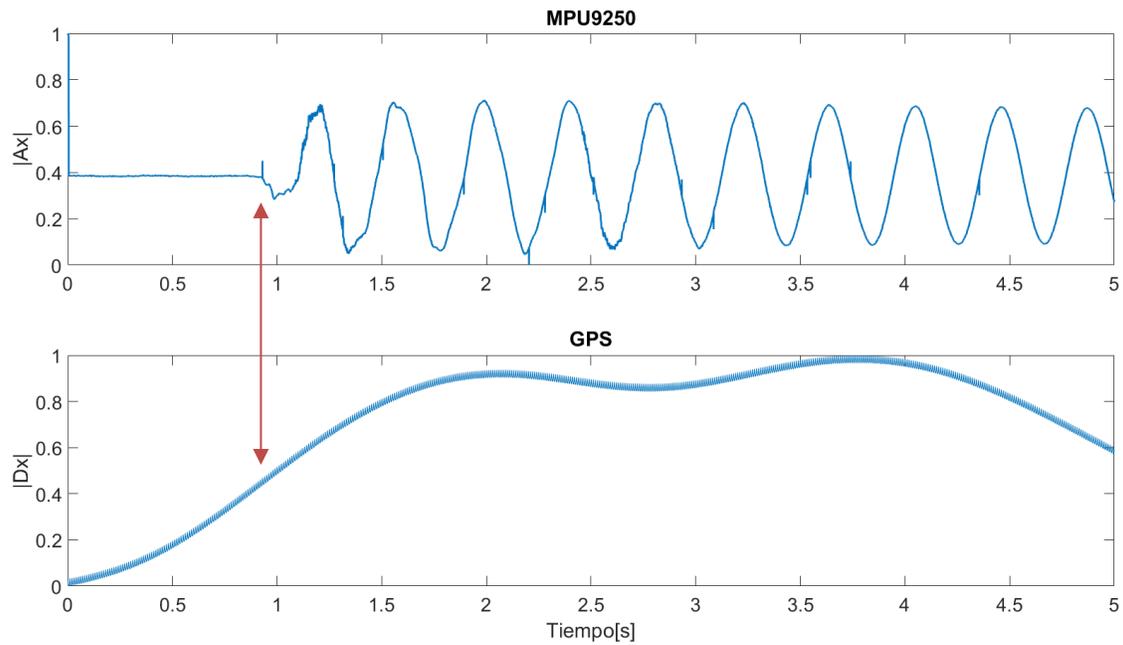


Figura 4.13: Datos obtenidos “Excitación de Onda 1” MPU9250 vs GPS.

Los datos de la figura interior representan al movimiento externo “Excitación de Onda 1” aceleración vs desplazamiento de la viga, si se puede mencionar un parecido es que la señal de desplazamiento gana altura justo cuando la aceleración comienza a presentarse como se muestra en la Figura 4.13.

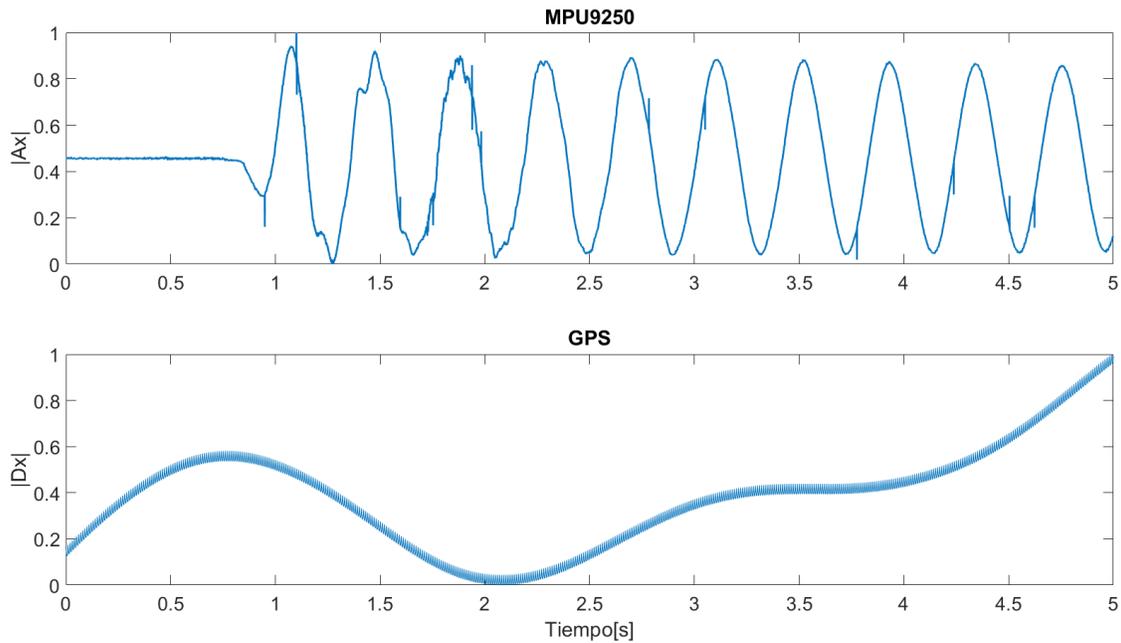


Figura 4.14: Datos obtenidos “Excitación de Onda 2” MPU9250 vs GPS.

En la Figura 4.14 tenemos el mismo comportamiento, el sensor de aceleración tiene la capacidad de detectar el movimiento pues la curva del desplazamiento comienza al mismo tiempo. Sin embargo, cabe aclarar que si bien, existe una diferencia entre el tiempo de la lectura del desplazamiento y la aceleración, esto es debido a que los equipos de medición de los sensores eran activados por personas que se sincronizaban con un reloj de bolsillo, esto causa que exista cierto desfase entre las señales.

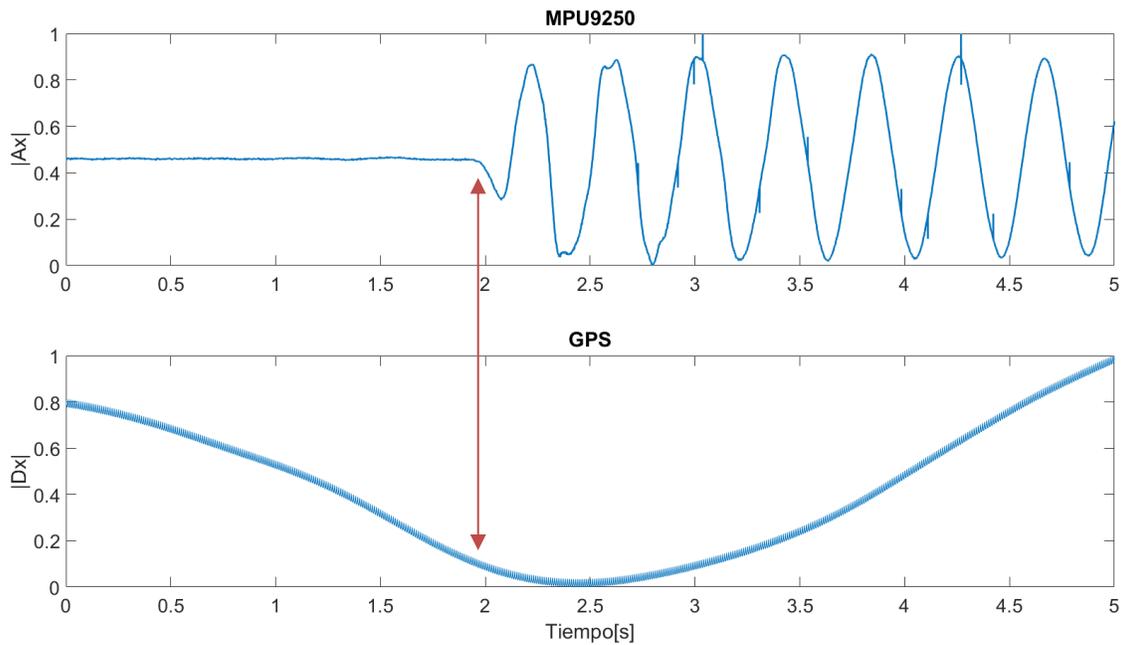


Figura 4.15: Datos obtenidos “Excitación de Onda 3” MPU9250 vs GPS.

Esta vez le toca a la Figura 4.15 la cual representa la señal obtenida por los sensores en “Excitación de onda 3” donde nuevamente se puede apreciar un comportamiento donde el desplazamiento se presenta en el mismo tiempo donde se presenta la perturbación de la viga debido al movimiento externo provocado manualmente.

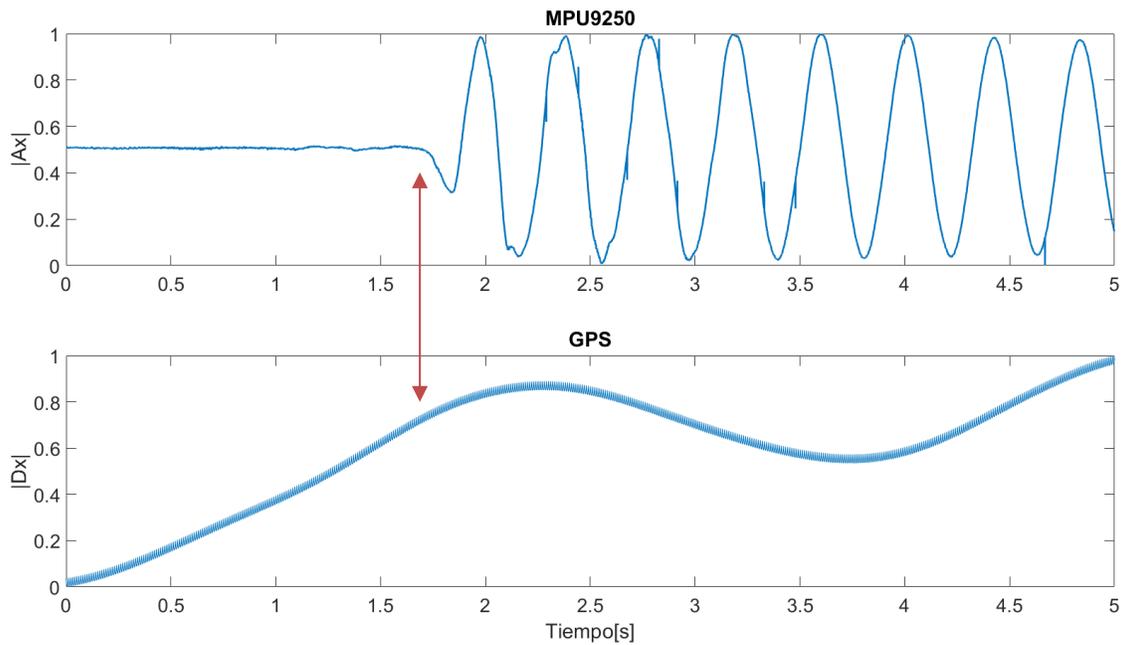


Figura 4.16: Datos obtenidos “Excitación de Onda 4” MPU9250 vs GPS.

Los que se puede observar en la figura anterior son los datos obtenidos por los sensores de aceleración MPU9250 y desplazamiento GPS en “Excitación de Onda 4”. Nuevamente el comportamiento del desplazamiento de la viga comienza a variar cuando la perturbación externa se presenta.

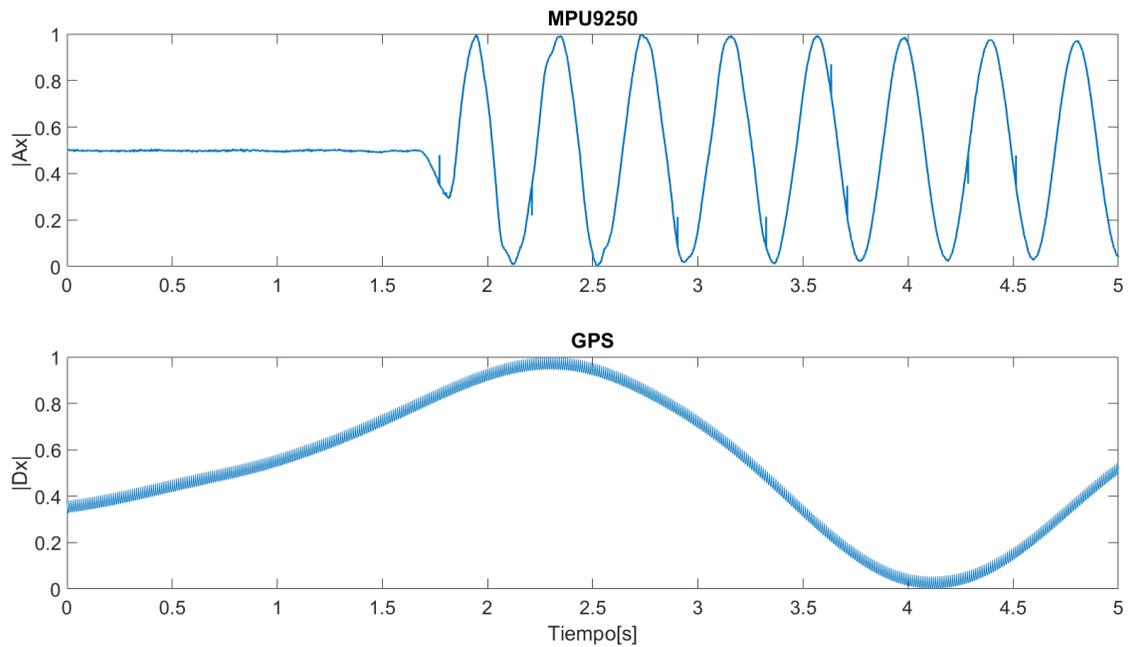


Figura 4.17: Datos obtenidos “Excitación de Onda” MPU9250 vs GPS.

Por último, en la Figura 4.17 se pueden apreciar los datos correspondientes a la aceleración por parte del sensor MPU9250 y desplazamiento por parte del GPS, todos estos datos es importante tenerlos puesto que permitirán entender lo que se sucede en el en temas posteriores.

4.3 Obtención de los desplazamientos dinámicos.

Para obtener el desplazamiento dinámico según (Hwang, Yun, Park, Lee, & Hong, 2012) es necesario primero obtener los datos de la posición de la viga, en este caso se parte desde datos de aceleración, por lo que para obtener la posición y luego el desplazamiento es necesario realizar el método de la doble integral.

$$f(x) = \iint_0^n A(x) + C \quad (4.1)$$

Que consiste en integrar dos veces la aceleración para de este modo obtener la posición, como el desplazamiento es la diferencia entre dos posiciones es posible obtenerlo al pasar la ecuación al tiempo discreto quedando de la siguiente manera.

$$f(x) = \sum_{i=0}^n \sum_{j=0}^n A(x) \quad (4.2)$$

de esta forma queda como una doble sumatoria y se puede implementar en un programa computacional, esto permite obtener los datos de desplazamiento de la viga por cada punto de la gráfica pero aun es necesario que los datos sean de la señal en desplazamiento dinámico, pues según (Michel Guzmán-Acevedo et al., 2019) para obtener el desplazamiento dinámico es necesario aplicar un filtro pasa bajas, en este proyecto se utilizó el LPF Chebyshev 2 y se obtuvo lo siguiente:

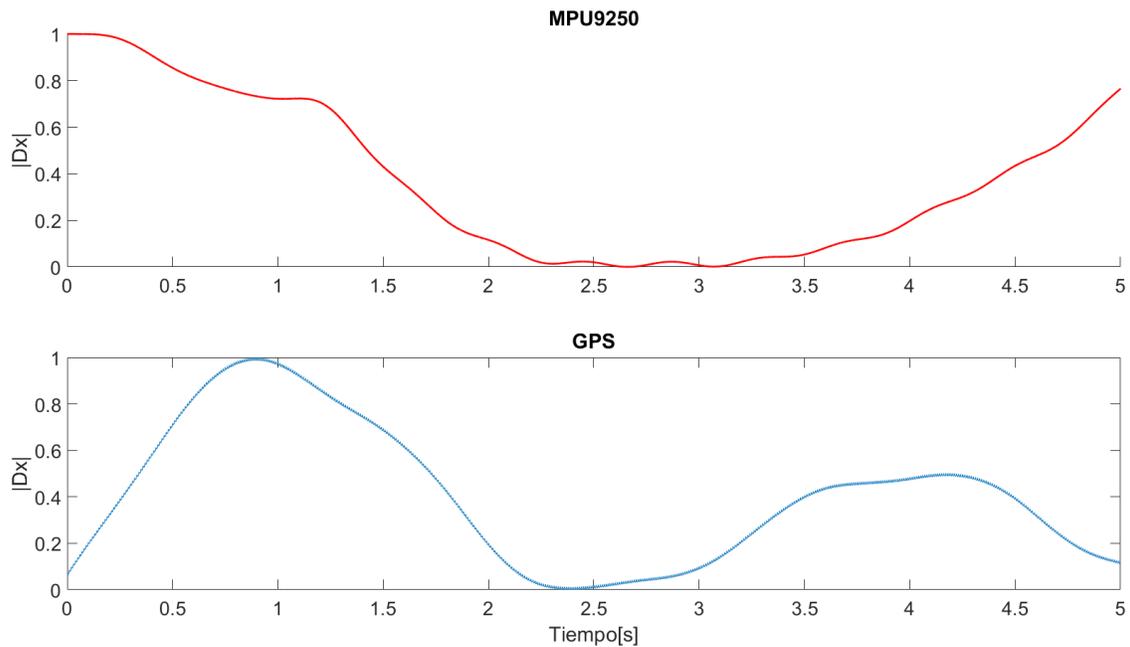


Figura 4.18: Datos de desplazamiento dinámico “martillo1”, MPU9250 vs GPS.

Como se puede observar en la figura anterior si bien no es posible obtener los datos tal cual los presentan las lecturas del GPS es posible obtener unos datos con el comportamiento muy similar. Los datos de la figura anterior corresponden a “martillo1”, la parte superior con la señal en rojo son datos de desplazamiento obtenidos del acelerómetro MPU9250, mientras que la azul representa al desplazamiento del GPS.

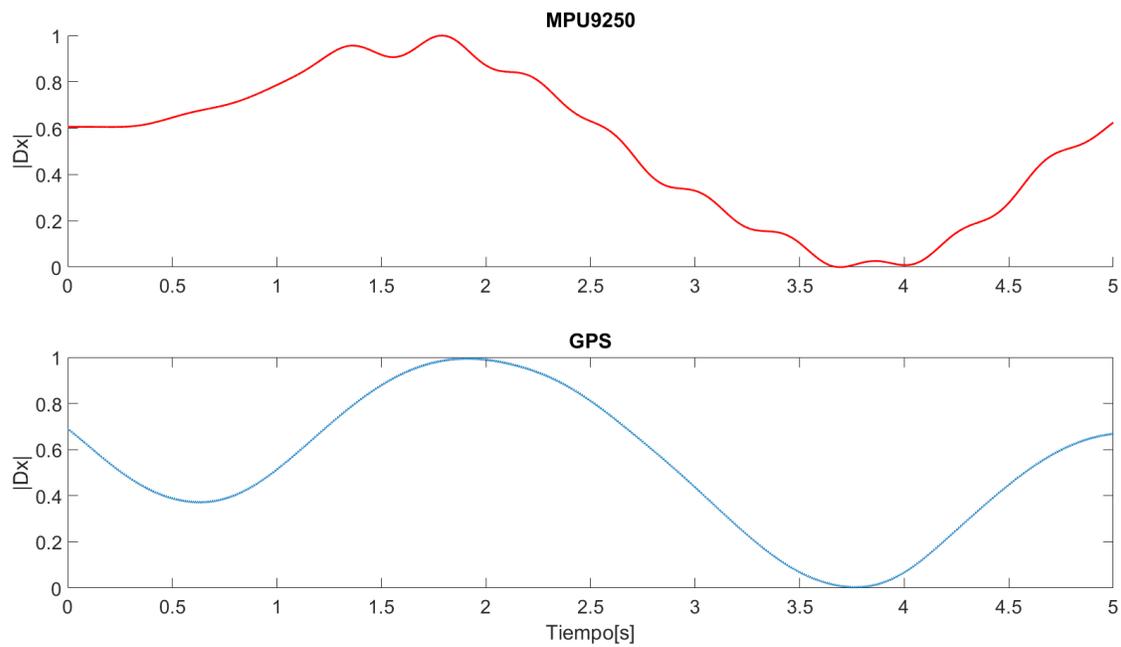


Figura 4.19: Datos de desplazamiento dinámico “Martillo 2”, MPU9250 vs GPS.

En este caso tenemos el mismo tipo de análisis una vez convertidos los datos de aceleración a datos de desplazamiento del sensor Invensense MPU9250 se procedió a la comparación entre los dispositivos, como se puede ver hay un comportamiento muy similar en los datos.

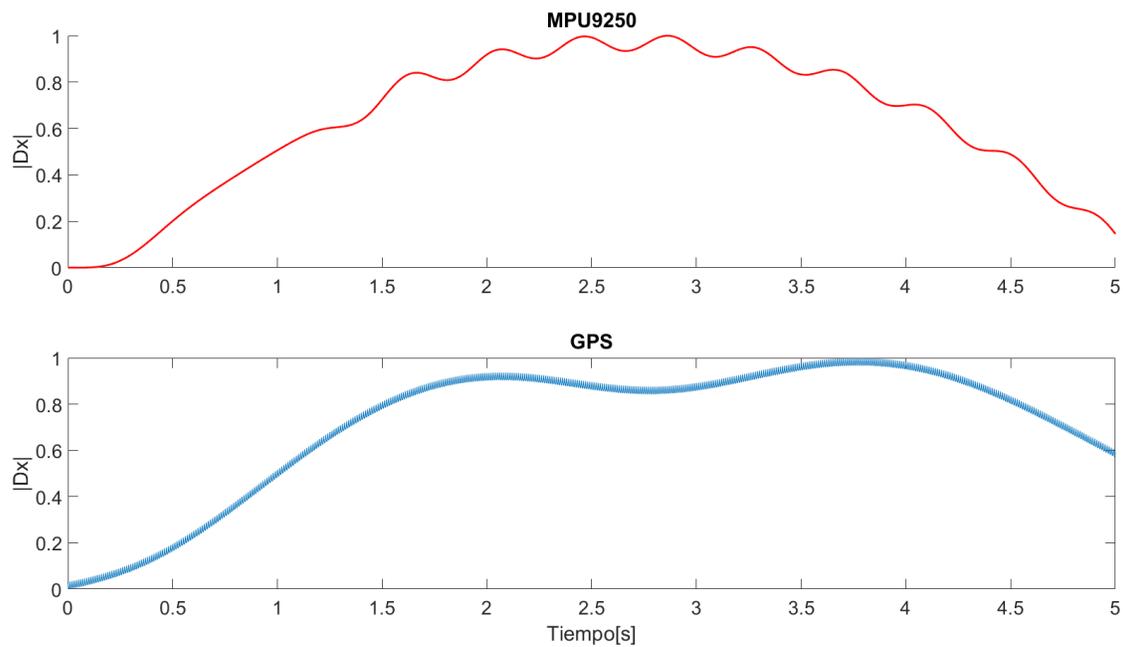


Figura 4.20: Datos de desplazamiento dinámico “Excitación de Onda 1”, MPU9250 vs GPS.

En este tercer análisis también se obtuvo un comportamiento muy similar en ambos dispositivos, por lo que gracias a esto se puede decir que efectivamente los datos obtenidos por aceleración del Invensense MPU9250 pueden utilizarse para obtener el desplazamiento dinámico de las estructuras.

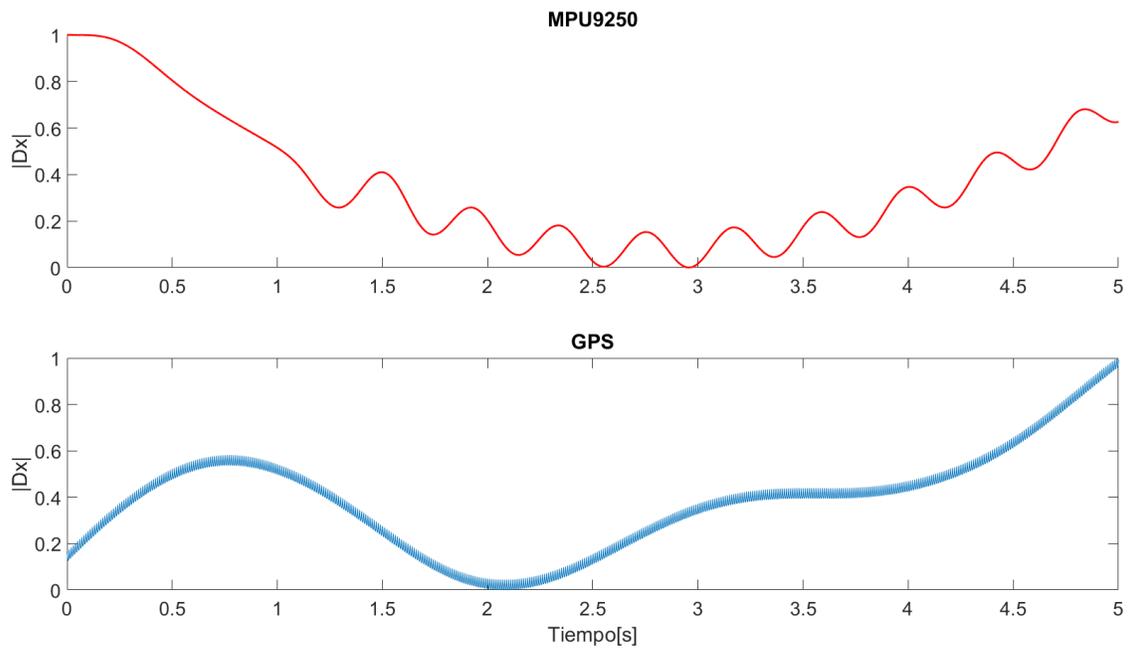


Figura 4.21: Datos de desplazamiento dinámico “Excitación de Onda 2”, MPU9250 vs GPS.

En la Figura 4.21 se presenta el comportamiento del desplazamiento dinámico de ambos dispositivos de medición, donde presentan un comportamiento muy similar. La señal de color rojo corresponde al MPU9250 y la de color azul corresponde al proporcionado por el Sistema Global de Posicionamiento.

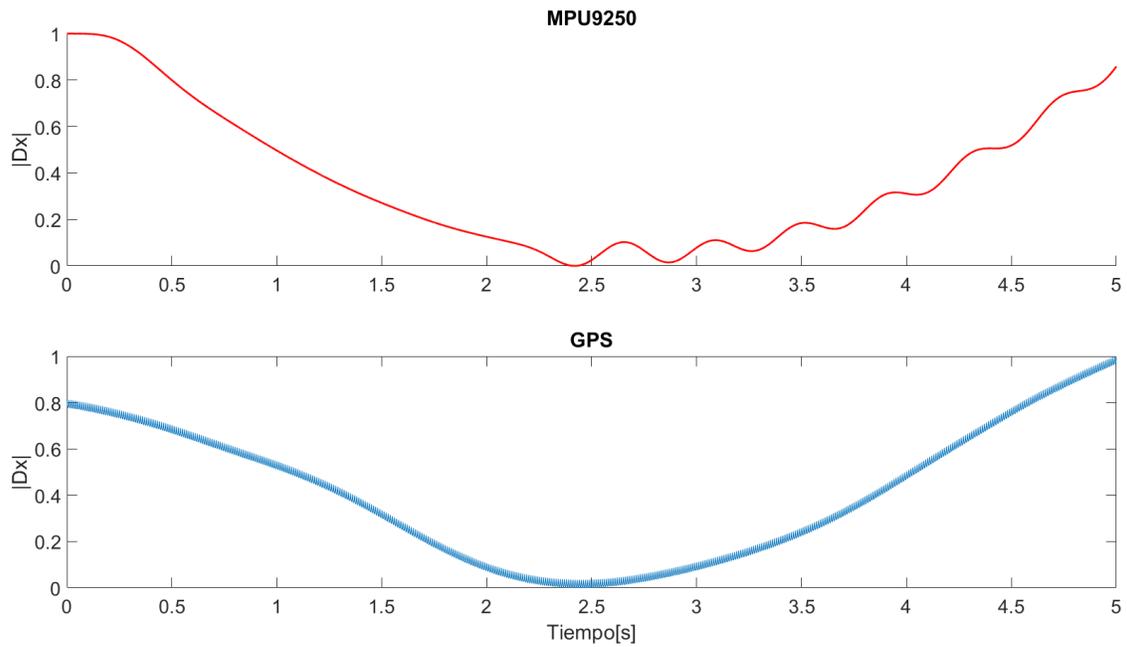


Figura 4.22: Datos de desplazamiento dinámico “Excitación de Onda 3”, MPU9250 vs GPS.

En la figura anterior se puede observar el desplazamiento dinámico obtenido por ambos dispositivos el rojo corresponde al MPU9250 y el azul al GPS, como se observa en este caso también se pudo obtener el mismo comportamiento en ambos dispositivos.

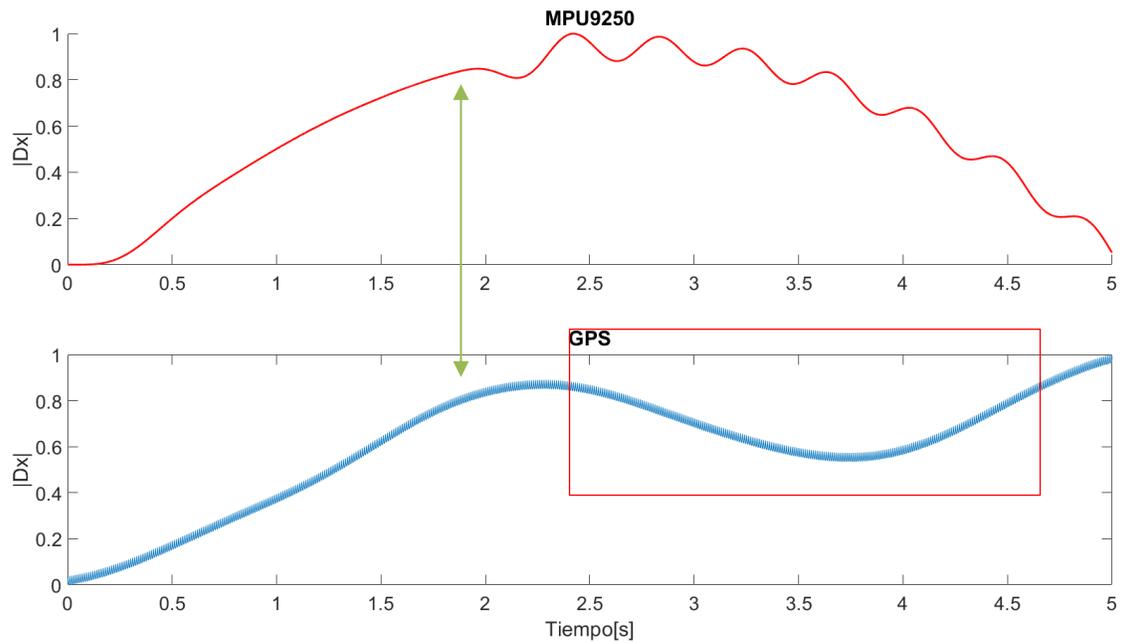


Figura 4.23: Datos de desplazamiento dinámico “Excitación de Onda 4”, MPU9250 vs GPS.

En la “Excitación de Onda 4” (figura 2.23). se obtiene una pérdida de datos ya que en la gráfica del GPS hay un comportamiento que no fue registrado por el MPU9250 (caja roja), sin embargo, el comportamiento de la primera curva (flecha verde) es prácticamente el mismo.

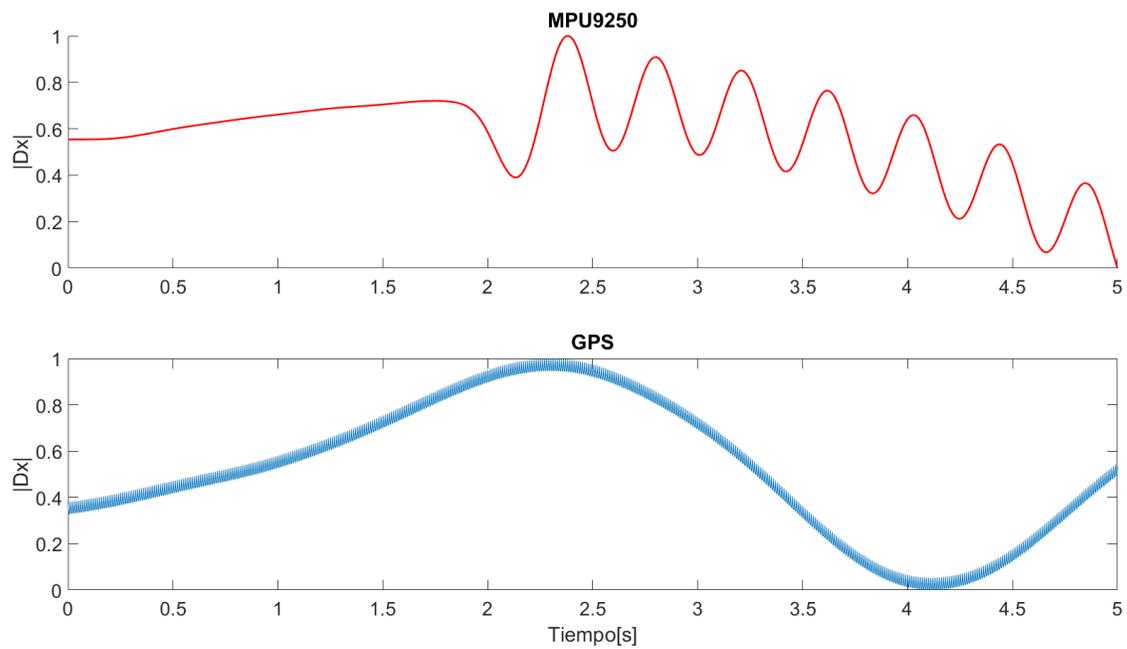


Figura 4.24: Datos de desplazamiento dinámico “Excitación de Onda 5”, MPU9250 vs GPS.

En la Figura 4.24 se muestra el único caso donde la curva de ambas graficas no se parecen, sin embrago parecen seguir el mismo patrón de movimiento.

4.4 Análisis de los resultados.

A continuación, se presentarán los datos obtenidos por el sensor Invensense MPU9250 (no comercial) vs el sensor Piezotronics 352C03 (comercial) en el dominio de la frecuencia, para ello se realizó un análisis de datos que los convierte de tiempo a frecuencia utilizando una herramienta llamada transformada rápida de Fourier, los datos obtenidos fueron los siguientes.

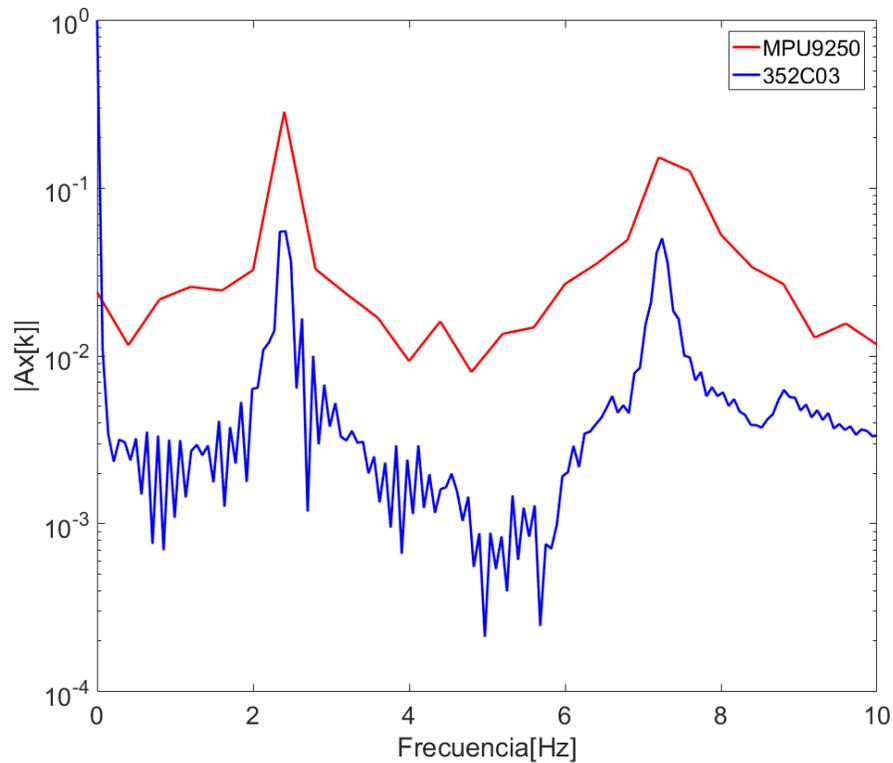


Figura 4.25: Datos en frecuencia de “martillo1” MPU920(rojo) vs 352C03(azul).

Los datos que se observan en la figura 4.25 son muy interesantes pues demuestra la frecuencia natural de la viga, la cual se representa como el pico más alto de la frecuencia (2.4 Hz), como se puede ver esto es muy explícito pues ambos dispositivos

obtuvieron exactamente la misma frecuencia. Esto da a entender que el dispositivo es capaz de encontrar la frecuencia natural de las estructuras, siempre y cuando dicha frecuencia sea menor que la mitad de la frecuencia de muestreo.

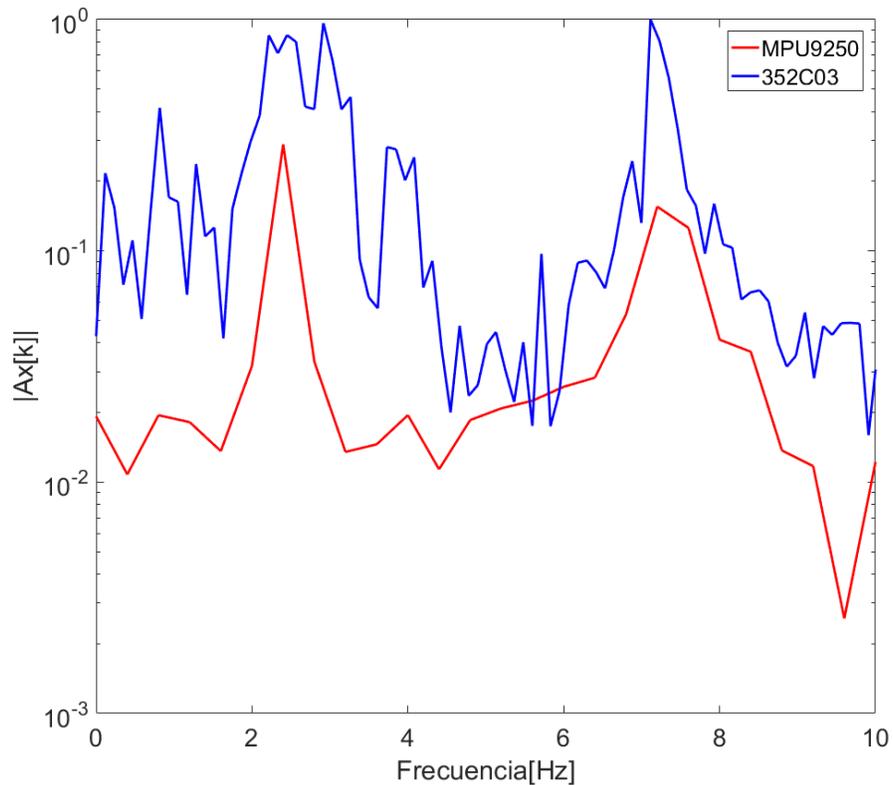


Figura 4.26: Datos en frecuencia de “Martillo 2” MPU920(rojo) vs 352C03(azul).

en el caso de la Figura 4.26 se observa como una especie de ruido en la detección de frecuencia con el sensor 352C03 esto se debe a que el sensor se encontraba saturado, sin embargo, la frecuencia obtenida por el MPU9250 es la misma de la figura 4.25 que corresponde a 2.4 Hz.

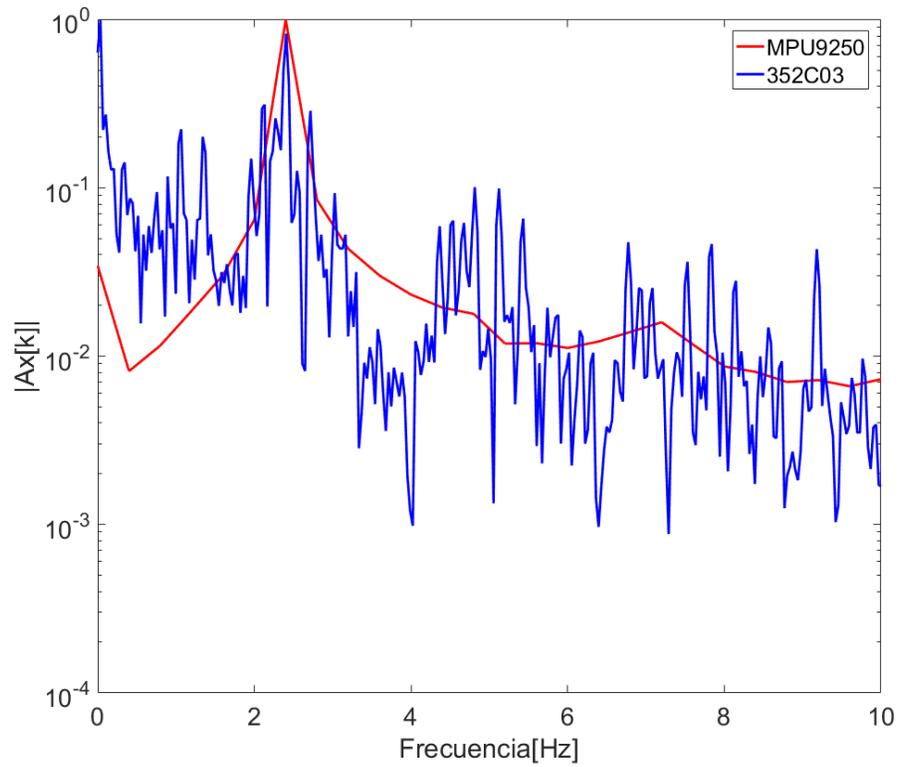


Figura 4.27: Datos en frecuencia de “Excitación de Onda 1” MPU920(rojo) vs 352C03(azul).

En el caso de las Excitaciones de Onda la frecuencia natural o fundamental de la viga esta exactamente en el mismo lugar 2.4 Hz, la cual fue detectada por ambos sensores.

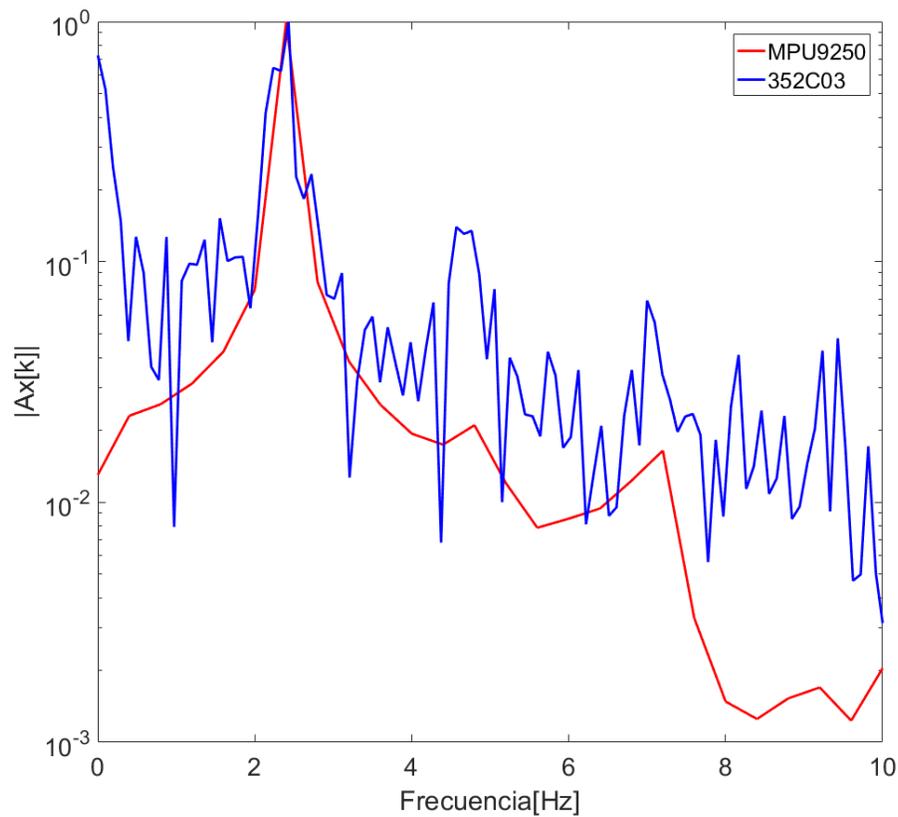


Figura 4.28: Datos en frecuencia de “Excitación de Onda 2” MPU920(rojo) vs 352C03(azul).

En esta figura se obtuvo el mismo comportamiento, donde ambos sensores fueron capaces de detectar el punto exacto de la frecuencia a la que vibra la viga de este experimento.

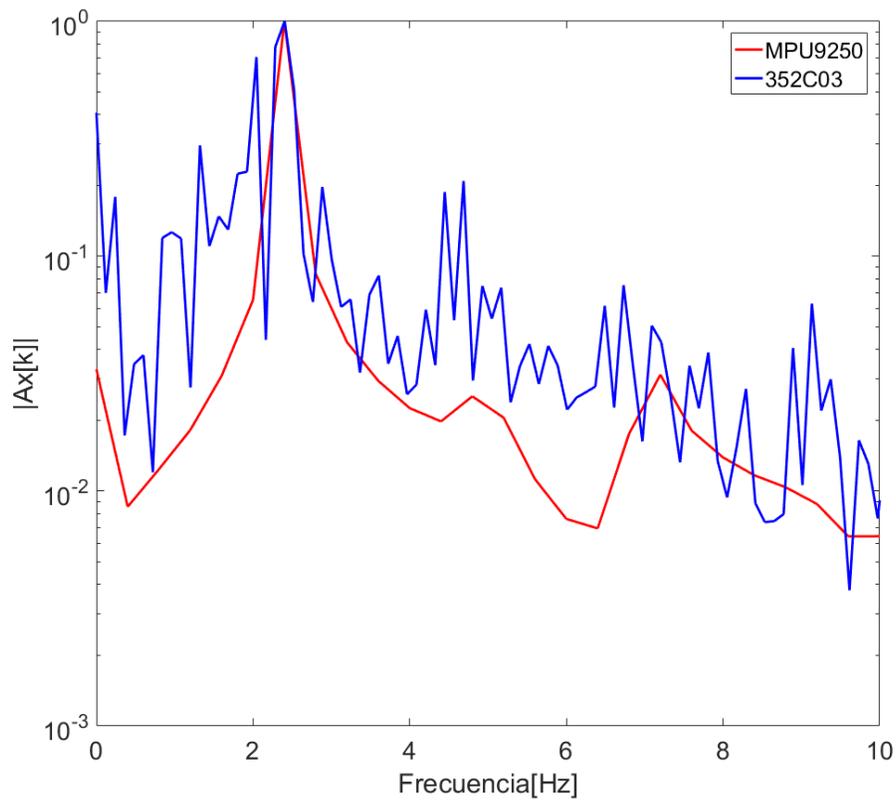


Figura 4.29: Datos en frecuencia de “Excitación de Onda 3” MPU920(rojo) vs 352C03(azul).

Lo mismo sucede con la figura anterior donde los datos de aceleración fueron capaces de obtener la frecuencia natural o fundamental de la viga.

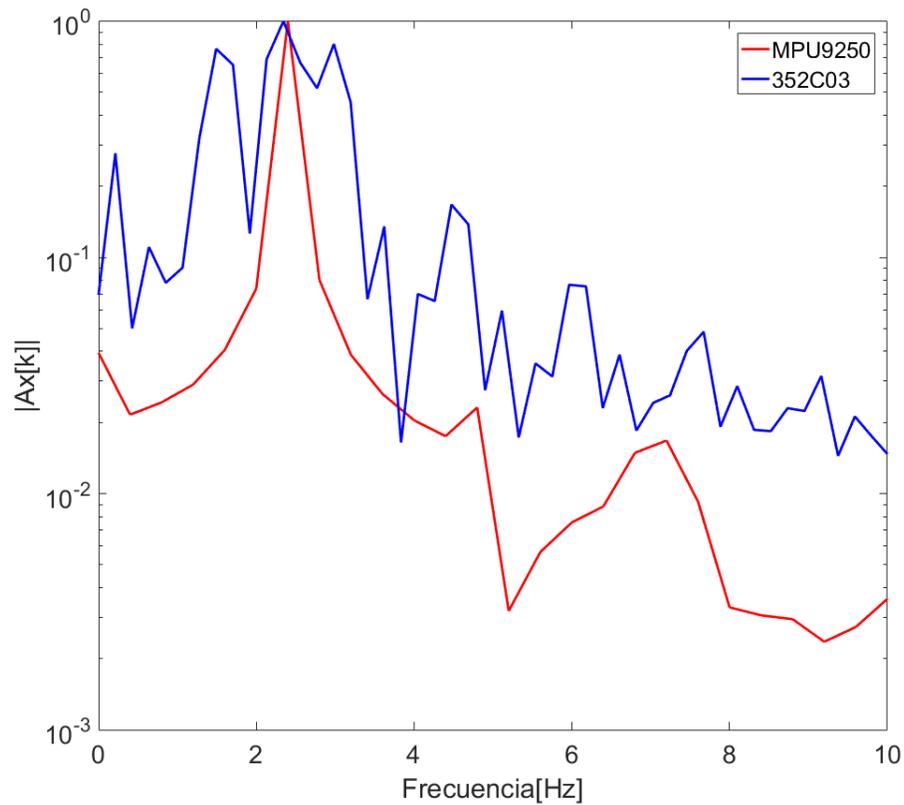


Figura 4.30: Datos en frecuencia de “Excitación de Onda 4” MPU920(rojo) vs 352C03(azul).

En esta figura se obtiene la frecuencia fundamental de la viga por el sensor MPU9250, sin embargo, los datos del sensor 352C03 fueron un montón de ruido ya que el sensor se encontraba desfasado en ese preciso momento.

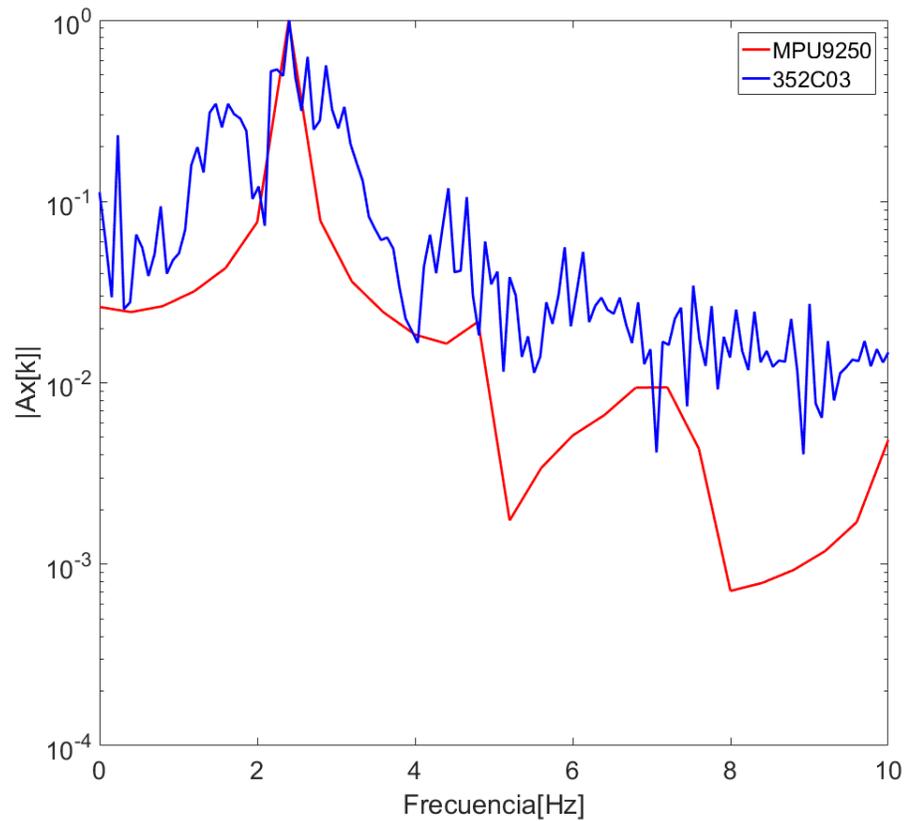


Figura 4.31: Datos en frecuencia de “Excitación de Onda 5” MPU920(rojo) vs 352C03(azul).

Esta es la última figura donde nuevamente ambos sensores fueron capaces de obtener la frecuencia fundamental o natural de la viga lo que demuestra la viabilidad del dispositivo MPU9250 para la detección y análisis de datos en aceleración en estructuras civiles.

4.5 Análisis Estadísticos

4.5.1 Obtención de la diferencia media entre dispositivos

Los análisis consistieron en obtener la media de cada uno de los experimentos y se obtuvieron los siguientes valores:

Martillo 1	-0.000857870105569955	g
Martillo 2	-0.000683566447312822	g
Excitación de Onda 1	-0.173656988183337	g
Excitación de Onda 2	-0.0994458745306587	g
Excitación de Onda 3	-0.211262227290530	g
Excitación de Onda 4	-0.0262125942040192	g
Excitación de Onda 5	-0.0147070815464522	g

Tabla 4.1. Medias obtenidas desde el sensor 352C03.

Martillo 1	- 2.07340811186896e-15	g
Martillo 2	0.000187432467009953	g
Excitación de Onda 1	0.00352808618426979	g
Excitación de Onda 2	-0.000309042634356615	g
Excitación de Onda 3	-0.00253430585649997	g
Excitación de Onda 4	0.00282354808192117	g
Excitación de Onda 5	0.00117901547289034	g

Tabla 4.2. Medias obtenidas desde el sensor MPU9250.

Posteriormente se calcularon las diferencias entre ambos dispositivos sensores de aceleración utilizando los valores de las medias anteriores:

Martillo 1	0.000857870105572028	g
Martillo 2	0.000870998914322775	g
Excitación de Onda 1	0.177185074367607	g
Excitación de Onda 2	0.0991368318963020	g
Excitación de Onda 3	0.208727921434030	g
Excitación de Onda 4	0.0290361422859404	g
Excitación de Onda 5	0.0158860970193425	g

Tabla 4.3. diferencias entre los sensores de aceleración MPU9250 y 352C03.

Por último, se calculó la media total de las diferencias entre ambos sensores donde se obtuvo un valor de 0.4423 g, esta es la distancia media que existe entre las mediciones obtenidas por los acelerómetros MPU9250 y el 352C03.

4.5.2 Obtención del valor máximo en los experimentos

En esta ocasión se obtuvieron los valores máximos de cada uno de los experimentos con la finalidad de observar las diferencias entre ambos sensores de aceleración MPU9250 y el 352C03. Donde se obtuvieron los siguientes resultados:

Martillo 1	3.35744868861958	g
Martillo 2	3.54213464174773	g
Excitación de Onda 1	0.285169001701640	g
Excitación de Onda 2	0.299097105600085	g
Excitación de Onda 3	0.336875472959239	g
Excitación de Onda 4	0.417004018896468	g
Excitación de Onda 5	0.413465290853540	g

Tabla 4.4. Valores máximos obtenidos desde el sensor 352C03.

Martillo 1	1.02492080446751	g
Martillo 2	1.39675832420651	g
Excitación de Onda 1	0.511477401043710	g
Excitación de Onda 2	0.487306741538730	g
Excitación de Onda 3	0.992449110385510	g
Excitación de Onda 4	0.508791772209810	g
Excitación de Onda 5	0.239008148442510	g

Tabla 4.5. Valores máximos obtenidos desde el sensor 352C03.

Lo que permitió obtener las diferencias de los valores máximos por cada experimento:

Martillo 1	2.33252788415207	0%
Martillo 2	2.14537631754122	0%
Excitación de Onda 1	0.226308399342070	0%
Excitación de Onda 2	0.188209635938645	0%
Excitación de Onda 3	0.655573637426271	0%
Excitación de Onda 4	0.0917877533133423	0%
Excitación de Onda 5	0.174457142411030	0%

Tabla 4.6. Diferencias de los valores máximos obtenidos desde ambos sensores de aceleración MPU9250 y 352C03.

4.5.3 Obtención del valor mínimo de los experimentos

Al igual que en el título 4.5.2 que se obtuvieron los valores máximos del sensor de aceleración MPU9250 y el sensor de aceleración 352C03. Este apartado se obtuvieron los valores mínimos con la finalidad de ver las diferencias entre los valores mínimos. Donde se obtuvieron los siguientes:

Martillo 1	-4.36804751152802	0%
Martillo 2	-4.93582282216743	0%
Excitación de Onda 1	-0.510660236546865	0%
Excitación de Onda 2	-0.354873771428558	0%
Excitación de Onda 3	-0.641956185458009	0%
Excitación de Onda 4	-0.403236602373942	0%
Excitación de Onda 5	-0.423695622413042	0%

Tabla 4.7. Valores mínimos del sensor 352C03.

Martillo 1	-1.17680654316349	0%
Martillo 2	-1.42119876679349	0%
Excitación de Onda 1	-0.319358500930490	0%
Excitación de Onda 2	-0.201923276467490	0%
Excitación de Onda 3	-1.19511764885349	0%
Excitación de Onda 4	-0.256856593523490	0%
Excitación de Onda 5	-0.319846797082490	0%

Tabla 4.8. Valores mínimos del sensor 352C03.

Estos datos permitieron encontrar las diferencias entre los valores máximos de ambos sensores de aceleración MPU9250 y 352C03. Donde se obtuvo lo siguiente:

Martillo 1	3.19124096836453	0%
Martillo 2	3.51462405537394	0%
Excitación de Onda 1	0.191301735616375	0%
Excitación de Onda 2	0.152950494961068	0%
Excitación de Onda 3	0.553161463395481	0%
Excitación de Onda 4	0.146380008850452	0%
Excitación de Onda 5	0.103848825330552	0%

Tabla 4.9. Diferencias de los valores mínimos obtenidos desde ambos sensores de aceleración MPU9250 y 352C03.

4.5.4 Obtención de la desviación estándar

A continuación, se procedió a obtener la desviación estándar de cada uno de los experimentos, donde se obtuvieron los siguientes resultados:

Martillo 1	0.139851939239796	0%
Martillo 2	0.153240694275979	0%
Excitación de Onda 1	0.187102613749460	0%
Excitación de Onda 2	0.150916369345148	0%
Excitación de Onda 3	0.198585102926429	0%
Excitación de Onda 4	0.265403893085136	0%
Excitación de Onda 5	0.216439784169764	0%

Tabla 4.10. desviación estándar de los datos obtenidos por el sensor 352C03.

Martillo 1	0.0529440420650070	0%
Martillo 2	0.0609556343939003	0%
Excitación de Onda 1	0.159882689524226	0%
Excitación de Onda 2	0.119196179168197	0%
Excitación de Onda 3	0.133713067780604	0%
Excitación de Onda 4	0.137098829466744	0%
Excitación de Onda 5	0.128905277402934	0%

Tabla 4.11. desviación estándar de los datos obtenidos por el sensor MPU9250.

Capítulo 5. Conclusiones y Recomendaciones

Con base de los resultados de este experimento se puede determinar que es posible extraer el comportamiento dinámico en estructuras civiles calculando la doble integral de los datos de sensores de aceleración tipo MEMS propuestos esta investigación de tesis. Esto se validó, debido a que el comportamiento del sensor de aceleración no comercial MPU9250, es muy similar al comportamiento del sensor de aceleración comercial 35C203, sin embargo, gracias a las pruebas estadísticas se pudo determinar que existe una diferencia entre las señales de 0.4423 g. por lo que todavía es necesario realizar un ajuste de calibración en el sensor MPU9250.

Concluyendo que este dispositivo de medición es apto para encontrar el comportamiento dinámico en estructuras civiles.

Como trabajo a futuro se plantea la posibilidad de realizar un dispositivo que tenga la capacidad de realizar un cálculo de la salud estructural en estructuras civiles utilizando este tipo de sensores no comerciales y de bajo costo, que sea capaz de emitir una alerta utilizando comunicación inalámbrica para indicar alguna posibilidad de falla, que se pueda controlar de forma remota y que aloje los datos en una base de datos en la nube para que cualquiera sea capaz de ver las condiciones de las estructuras en tiempo real.

También se recomienda utilizar más dispositivos de medición como el sensor tipo LVDT que proporcionen datos de desplazamiento directo como el Sistema de posicionamiento Global (GPS).

Bibliografía

- Amezquita-Sanchez, J. P., & Adeli, H. (2016). Signal Processing Techniques for Vibration-Based Health Monitoring of Smart Structures. *Archives of Computational Methods in Engineering*, 23(1), 1-15.
<https://doi.org/10.1007/s11831-014-9135-7>
- Das, S., & Saha, P. (2018). A review of some advanced sensors used for health diagnosis of civil engineering structures. *Measurement*, 129, 68-90.
<https://doi.org/10.1016/j.measurement.2018.07.008>
- Elnabwy, M., Kaloop, M., & Elbeltagi, E. (2013). Talkha Steel Highway bridge Monitoring and Movement Identification using RTK-GPS Technique. En *Measurement*. <https://doi.org/10.1016/j.measurement.2013.08.014>
- Farrar, C., & Jáuregui, D. (1996). *Damage detection algorithms applied to experimental modal data from the I-40 Bridge*. <https://doi.org/10.2172/201810>
- Han, H., Wang, J., Meng, X., & Liu, H. (2016). Analysis of the dynamic response of a long span bridge using GPS/accelerometer/anemometer under typhoon loading. *Engineering Structures*, 122, 238-250.
<https://doi.org/10.1016/j.engstruct.2016.04.041>
- Hwang, J., Yun, H., Park, S.-K., Lee, D., & Hong, S. (2012). Optimal Methods of RTK-GPS/Accelerometer Integration to Monitor the Displacement of Structures. *Sensors (Basel, Switzerland)*, 12(1), 1014-1034.
<https://doi.org/10.3390/s120101014>

- Kaouk, M., Kaouk, M., Zimmerman, D. C., & Zimmerman, D. C. (1994). Structural damage assessment using a generalized minimum rank perturbation theory . *AIAA journal* , Vol. 32, pp. 836-842. <https://doi.org/10.2514/3.12061>
- Kernighan, B. W., & Ritchie, D. M. (1991). *El lenguaje de programación C*. Pearson Educación.
- MAECK, J., & DE ROECK, G. (2003). DAMAGE ASSESSMENT USING VIBRATION ANALYSIS ON THE Z24-BRIDGE. *Mechanical Systems and Signal Processing*, 17(1), 133-142. <https://doi.org/10.1006/mssp.2002.1550>
- Mallat, S., & Hwang, W. L. (1992). Singularity detection and processing with wavelets. *IEEE Transactions on Information Theory*, 38(2), 617-643. <https://doi.org/10.1109/18.119727>
- Michel Guzman-Acevedo, G., Vazquez Becerra, G., Millan-Almaraz, J., Rodríguez-Lozoya, H., Reyes-Salazar, A., Gaxiola-Camacho, J. R., & A. Martinez-Felix, C. (2019). GPS, Accelerometer, and Smartphone Fused Smart Sensor for SHM on Real-Scale Bridges. *Advances in Civil Engineering*, 2019, 1-15. <https://doi.org/10.1155/2019/6429430>
- Moschas, F., & Stiros, S. (2011). Measurement of the dynamic displacements and of the modal frequencies of a short-span pedestrian bridge using GPS and an accelerometer. *Engineering Structures*, 33(1), 10-17. <https://doi.org/10.1016/j.engstruct.2010.09.013>

- Pan, S.-S., Zhao, X.-F., Zhao, H., & Mao, J. (2015). *Experiment on interface separation detection of concrete-filled steel tubular arch bridge using accelerometer array* (Vol. 9435). <https://doi.org/10.1117/12.2084114>
- Pandey, M. D. (1991). *Structure-material interaction in stability of fibrous composite elements*. University of Waterloo (Canada), Ann Arbor.
- Psimoulis, P., Peppas, I., Bonenberg, L., Ince, S., & Meng, X. (2016). *Combination of GPS and RTS measurements for the monitoring of semi-static and dynamic motion of pedestrian bridge*.
- Python.org. (2019). Python documentation. Recuperado de <https://docs.python.org/3/>
- Rauch-Hindin, W. B. (1989). *Aplicaciones de la inteligencia artificial en la actividad empresarial, la ciencia y la industria*. Díaz de Santos.
- Ritchie, D. M. (1993). The Development of the C Language. *SIGPLAN Not.*, 28(3), 201-208. <https://doi.org/10.1145/155360.155580>
- Roberts, G., & ASHKENAZI, V. (1997). Experimental monitoring of the Humber Bridge using GPS. En *Proceedings of The Institution of Civil Engineers-civil Engineering—PROC INST CIVIL ENG-CIVIL ENG* (Vol. 120). <https://doi.org/10.1680/icien.1997.29810>
- Roberts, G., Brown, C., Meng, X., Ogundipe, O., Atkins, C., & Colford, B. (2012). Deflection and frequency monitoring of the Forth Road Bridge, Scotland, by GPS. En *Bridge Engineering* (Vol. 165). <https://doi.org/10.1680/bren.9.00022>

- Romero Troncoso, R. de J. (2007). *Electrónica digital y lógica programable*. Universidad de Guanajuato.
- Salazar Lopez, J. R., & Millan Almaraz, J. R. (2015). *Red inalámbrica de sensores de monitoreo ambiental para manejo de virus en Chile*. Instituto Tecnológico de Culiacán, Sinaloa, Mexico.
- Serway, R. A., Jewett, J. W., & Serway, R. A. (2004). *Physics for scientists and engineers*. Belmont, CA: Thomson-Brooks/Cole.
- Stubbs, N., Kim, J.-T., & Topole, K. (1992). An Efficient and Robust Algorithm for Damage Localization in Offshore Platforms. En *Proc. ASCE Tenth Structures Congress*. <https://doi.org/10.2307/3716590>
- Tanenbaum, A. S. (2003). *Redes de computadoras*. Editorial Alhambra S. A. (SP).
- Vazquez B., G. E., Gaxiola-Camacho, J. R., Bennett, R., Guzman-Acevedo, G. M., & Gaxiola-Camacho, I. E. (2017). Structural evaluation of dynamic and semi-static displacements of the Juarez Bridge using GPS technology. *Measurement*, *110*, 146-153. <https://doi.org/10.1016/j.measurement.2017.06.026>
- Yunus, M. Z. M., Ibrahim, N., & Ahmad, F. S. (2018). A review on bridge dynamic displacement monitoring using global positioning system and accelerometer. *AIP Conference Proceedings*, *1930*(1), 20039. <https://doi.org/10.1063/1.5022933>

Zhu, L., Fu, Y., Chow, R., Spencer, B. F., Park, J. W., & Mechitov, K. (2018).
Development of a High-Sensitivity Wireless Accelerometer for Structural
Health Monitoring. *Sensors* (14248220), 18(1), 262.