UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

INSTALACIÓN Y FUNCIONAMIENTO DE SISMÓGRAFOS DE BAJO COSTO RASPBERRY SHAKE 4D V9 David Sebastián Salazar Zambrano

Ingeniería Civil

Trabajo de fin de carrera presentado como requisito

para la obtención del título de

Ingeniero Civil

Quito, 25 de agosto de 2024

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

HOJA DE CALIFICACIÓN **DE TRABAJO DE FIN DE CARRERA**

INSTALACIÓN Y FUNCIONAMIENTO DE SISMÓGRAFOS DE BAJO **COSTO RASPBERRY SHAKE 4D V9**

David Sebastián Salazar Zambrano

Nombre del profesor, Título académico Adrián Patricio Tola Tola, PhD

Quito, 25 de agosto de 2024

© DERECHOS DE AUTOR

Por medio del presente documento certifico que he leído todas las Políticas y Manuales de la Universidad San Francisco de Quito USFQ, incluyendo la Política de Propiedad Intelectual USFQ, y estoy de acuerdo con su contenido, por lo que los derechos de propiedad intelectual del presente trabajo quedan sujetos a lo dispuesto en esas Políticas.

Asimismo, autorizo a la USFQ para que realice la digitalización y publicación de este trabajo en el repositorio virtual, de conformidad a lo dispuesto en la Ley Orgánica de Educación Superior del Ecuador.

Nombres y apellidos:	David Sebastián Salazar Zambrano
Código:	211661
Cédula de identidad:	1725462384
Lugar y fecha:	Quito, 25 de agosto de 2024

ACLARACIÓN PARA PUBLICACIÓN

Nota: El presente trabajo, en su totalidad o cualquiera de sus partes, no debe ser considerado como una publicación, incluso a pesar de estar disponible sin restricciones a través de un repositorio institucional. Esta declaración se alinea con las prácticas y recomendaciones presentadas por el Committee on Publication Ethics COPE descritas por Barbour et al. (2017) Discussion document on best practice for issues around theses publishing, disponible en http://bit.ly/COPETheses.

UNPUBLISHED DOCUMENT

Note: The following capstone project is available through Universidad San Francisco de Quito USFQ institutional repository. Nonetheless, this project – in whole or in part – should not be considered a publication. This statement follows the recommendations presented by the Committee on Publication Ethics COPE described by Barbour et al. (2017) Discussion document on best practice for issues around theses publishing available on http://bit.ly/COPETheses.

RESUMEN

Este trabajo recopila información sobre la instalación y funcionamiento de sismógrafos de bajo costo, específicamente el modelo Raspberry Shake 4D V9, con el fin de monitorear vibraciones provenientes de eventos sísmicos, uso cotidiano de estructuras o la operación de maquinaria.

Se realizaron pruebas de instalación y verificación de las señales obtenidas, así como el desarrollo de scripts en Python y Matlab para procesar y analizar los datos recopilados. Estas herramientas permiten comparar los datos de aceleración, velocidad y desplazamiento obtenidas con los instrumentos y compararlas con curvas de datos conocidos.

Los resultados concluyen que los sismógrafos Raspberry Shake 4D V9 son una opción viable para el monitoreo de vibraciones de diversas fuentes, contribuyendo a la gestión del riesgo y la evaluación de la integridad estructural.

Palabras clave: Monitoreo de vibraciones, Raspberry Shake 4D V9, sismógrafos de bajo costo, aceleración, eventos sísmicos, vibraciones de maquinaria, análisis dinámico, seguridad estructural, procesamiento de señales.

ABSTRACT

This work gathers information on the installation and operation of low-cost seismographs, specifically the Raspberry Shake 4D V9 model, with the aim of monitoring vibrations from seismic events, everyday use of structures, or machinery operation.

Installation tests and verification of the signals obtained were carried out, as well as the development of scripts in Python and Matlab to process and analyze the collected data. These tools allow for the comparison of acceleration, velocity, and displacement data obtained from the instruments with known data curves.

The results conclude that the Raspberry Shake 4D V9 seismographs are a viable option for monitoring vibrations from various sources, contributing to risk management and structural integrity assessment.

Keywords: Vibration monitoring, Raspberry Shake 4D V9, low-cost seismographs, acceleration, seismic events, machinery vibrations, dynamic analysis, structural safety, signal processing.

CAPITU	LO I	14
INTRO)DUCCIÓN:	14
1.1	Objetivos:	
CADITII		16
	DUICCIÓN A LOS SISMÓGRAFOS DE BAIO COSTO RASPRERRY SHAKE AD V	·····10 /9 16
IUSTI	FICACIÓN DEL USO DE SISMÓGRAFOS RASPBERRY SHAKE 4D V9	
,0011		
Capitul		20
FUNCI	ONAMIENTO Y CONFIGURACION DE LOS RASPBERRY SHAKE 4D V9	20
1. 2	Conexión a internet defectuosa	
Ζ.	26	bleauaj.
3	20 Conevión a internet mediante una red inalámbrica (WiFi)	28
Visual	lización de los datos cargados al servidor de Raspherry Shake	30
<i>(</i>		
CAPITU		
EXTRA	ACCION DE DATOS DESDE EL SERVIDOR DE RASPBERRY SHAKE UTILIZAN	DO UN
SCRIP	T DE PYTHUN PRUPURCIUNADU PUR EL FABRICANTE.	32
1. 2	Importación de Dibliotecas necesarias	
2. 2	Inicialización de un chente	
з. Д	Definir el intervalo de tiempo	
	Seleccionar los canales de datos	
6.	Descargar y almacenar las formas de onda	35
7.	Graficar los datos sin procesar	
8.	Adjuntar la información de la respuesta instrumental	
9.	Eliminar la respuesta instrumental y convertirla a unidades físicas	
10.	Graficar el nuevo stream en unidades físicas	
1.	Crear listas para almacenar datos de todos los canales	
2.	Inicialización de la longitud mínima	38
3.	Recopilación de datos de cada canal	
4.	Generación de la columna de tiempo	
5.	Generación del DataFrame y exportación a CSV	40
CAPITU	LO V	40
PROC	ESADO DE SEÑALES MEDIANTE EL USO DE UN CÓDIGO GENERADO EN MA'	TLAB. 40
1.	Importación de Datos y Parámetros Iniciales	
2.	Filtrado de Señales de Aceleración	44
3.	Integración para Obtención de Velocidad	
4.	Visualización de los Resultados	45
CAPITU	LO VI	46
PRUE	BAS DE FUNCIONAMIENTO DE LOS RASPBERRY SHAKE 4D V9	
Pru	ebas de funcionamiento con mesa vibratoria marca Quanser	
1.	Resultados simulación sismo de Pedernales (2016).	
2.	Resultados simulación sismo de Northridge (1994)	55
Lím	ites de Vibración por Tipo de Ocupación en el Análisis con Raspberry Shake	61
Mec	lición de vibraciones en la Universidad San Francisco de Quito	62
Conclus	iones	67

TABLA DE CONTENIDO

Recomendaciones para estudios futuros:	68
Referencias bibliográficas	70
Anexo A: SCRIPT DE PYTHON PARA LA EXTRACCIÓN DE DATOS DESDE EL SERVIDOR DE RASPBERRY SHAKE	72
Anexo B: CÓDIGO DE PYTHON PARA DESCARGA Y EXPORTACIÓN DE DATOS DESDE EL SERVIDOR DE RASPBERRY SHAKE A UN ARCHIVO .CSV	74
Anexo C: CÓDIGO EN MATLAB PARA EL PROCESADO DE SEÑALES DE ACELERACION, VELOCIDAD Y DESPLAZAMIENTO	77

ÍNDICE DE TABLAS

Tabla 1. Atenuación de vibraciones	promedio picos	más relevantes	
------------------------------------	----------------	----------------	--

ÍNDICE DE FIGURAS

Figura 1. Dispositivo Raspberry Shake 4D. (Imagen tomada de Raspberry Shake, 2024.)	16
Figura 2. Puertos laterales del dispositivo Raspberry Shake 4D, incluyendo Ethernet y USI	B.
(Imagen tomada de Botland, 2024.)	17
Figura 3. Puertos fontales del dispositivo Raspberry Shake 4D, incluyendo alimentación,	
HDMI y audio/video. (Imagen tomada de Botland, 2024.)	18
Figura 4. Ubicación de los pernos de nivelación en los 3 puntos designados	20
Figura 5. Vista lateral de la instalación de los pernos de nivelación.	21
Figura 6. Visualización de la correcta nivelación del instrumento en el nivel de burbuja	21
Figura 7. Conexión del cable ethernet en el puerto correspondiente	21
Figura 8. Conexión del cable de poder en su respectivo puerto.	22
Figura 9. Luces de estado en correcto funcionamiento	22
Figura 10. Menú de configuración e información del dispositivo disponible en "rs.local"	22
Figura 11. Diseño del botón de configuraciones de "rs.local"	23
Figura 12. Información que llenar en el menú de configuraciones	23
Figura 13. Mapa donde se escoge la ubicación del instrumento	24
Figura 14. Datos de ubicación del instrumento y datos del edificio	24
Figura 15. Dispositivo funcionando correctamente	25
Figura 16. Símbolo del sistema de una computadora con sistema operativo Windows 10	26
Figura 17. Comando "ssh myshake@IPV4"	27
Figura 18. Bienvenida al sistema del Raspberry Shake	27
Figura 19. Opciones dentro de dhcpcd.conf.	28
Figura 20. Contenido del comando "sudo nano /etc/wpa_supplicant/wpa_supplicant.conf".	.29
Figura 21. Configuración dentro de wpa_supplicant	29
Figura 22. Raspberry Shake Data view vista de la página principal	30
Figura 23. Canal ENN de la estación R03AE visualizado mediante Data View de Raspbern	ry
Shake	31
Figura 24. Acercamiento en un pico de vibración provocado para ejemplificación	
experimental	31
Figura 25. Primeras líneas del código que importan las bibliotecas necesarias para el	
funcionamiento del script	33
Figura 26. Línea de código que crea el cliente de datos sísmicos	33
Figura 27. Líneas para obtener la estación de interés y la respuesta instrumental.	34
Figura 28. Líneas de código donde se establece fecha y hora de inicio con un ejemplo de	
introducción de datos en números menores a 10	34
Figura 29. Canales de datos de los que se quiere extraer la información.	35
Figura 30. Creación de un stream vacío y llenado con los datos de cada canal de datos del	
Raspberry	35
Figura 31. Línea de código que grafica los datos sin procesar.	35
Figura 32. Línea que asocia la respuesta instrumental a los datos del stream.	36
Figura 33. Línea que elimina la respuesta instrumental y convierte los datos a unidades	
físicas	36
Figura 34. Línea de código que grafica la respuesta en unidades físicas.	36
Figura 35. Gráfico ejemplo que contiene la respuesta en counts	37
Figura 36. Gráfico ejemplo que contiene la respuesta en unidades físicas de la prueba de	
control	37
Figura 37. Líneas de código para la creación decanales para guardar la información	38

Figura 38. Función para establecer la longitud mínima	
Figura 39. Líneas de código que recopilan los datos de cada canal.	
Figura 40. Líneas que generan la columna de tiempo en base a los datos disp	ponibles
Figura 41. Líneas de código que recopilan los canales dentro del archivo CS	V y lo guardan.
	40
Figura 42. Instrumento instalado lejos de fuentes de ruido	41
Figura 43. Transformada de Fourier de la señal en reposo	41
Figura 44. Comparación señal en reposo sin filtrar y filtrada	
Figura 45: Líneas de código para ingresar los datos de integración y filtrado	43
Figura 46: Filtrado de datos de aceleración.	44
Figura 47: Integración para obtención de velocidades y desplazamientos	45
Figura 48. Resultados de procesamiento de señal correspondiente a la simul	ación del sismo
de Northridge	46
Figura 49: Pesaje del instrumento junto con la base de sujeción	47
Figura 50: Sistema instalado en la mesa vibratoria.	47
Figura 51: Comparación entre aceleraciones y desplazamientos reales vs esc	calados48
Figura 52: Nuevo delta de tiempo generado por el escalado de la informació	on48
Figura 53: Resultados de aceleración procesados y comparativa	49
Figura 54: Superposición de curvas generadas por mesa vibratoria vs Raspb sentido "X" o "ENN"	erry Shake
Figura 55: Superposición de curvas generadas por mesa vibratoria vs Raspb	erry Shake
sentido "Y" o "ENE".	
Figura 56: Detalle de la superposición de curvas generadas por mesa vibrato	oria vs Raspberry
Shake sentido "X" o "ENN"	
Figura 57: Detalle de la superposición de curvas generadas por mesa vibrato	oria vs Raspberry
Shake sentido "Y" o "ENE".	
Figura 58: Gráficos de Velocidad vs Tiempo	51
Figura 59: Velocidad vs tiempo del sentido "X" o "ENN"	
Figura 60: Velocidad vs tiempo del sentido "Y" o "ENE".	
Figura 61: Comparación de velocidades sentido "X" o "ENN"	53
Figura 62: Comparación de velocidades sentido "Y" o "ENE"	53
Figura 63: Comparación Desplazamiento vs Tiempo	54
Figura 64: Detalle de la curva de desplazamiento sentido "X" o "ENN"	54
Figura 65: Detalle de la curva de desplazamiento sentido "Y" o "ENE"	55
Figura 66: Resultados de aceleración procesados y comparativa	56
Figura 67: Superposición de curvas generadas por mesa vibratoria vs. Raspl	perry Shake en el
sentido "X" o "ENN"	56
Figura 68: Superposición de curvas generadas por mesa vibratoria vs. Raspl	berry Shake en el
sentido "Y" o "ENE"	56
Figura 69: Detalle de la superposición de curvas generadas por mesa vibrato	oria vs. Raspberry
Shake en el sentido "X" o "ENN"	57
Figura 70: Detalle de la superposición de curvas generadas por mesa vibrato	oria vs. Raspberry
Shake en el sentido "Y" o "ENE"	57
Figura 71: Velocidad vs. Tiempo	58
Figura 72: Velocidad vs. tiempo del sentido "X" o "ENN"	58
Figura 73: Velocidad vs. tiempo del sentido "Y" o "ENE"	59
Figura 74: Comparación de velocidades en el sentido "X" o "ENN"	59
Figura 75: Comparación de velocidades en el sentido "Y" o "ENE"	59
Figura 76: Comparación de Desplazamiento vs. Tiempo	60

Figura 77: Desplazamiento en el sentido "X" o "ENN".	60
Figura 78: Desplazamiento en el sentido "Y" o "ENE"	61
Figura 79. Estación R9B17 instalada en el piso 3 del edificio Hayek	63
Figura 80: Estación R03AE instalada junto a la rampa de acceso	63
Figura 81. Estación R03AE instalada al lado de la rampa de acceso del S1	64
Figura 82. Gráficas Aceleración vs tiempo obtenidas en el S1 y en el piso 3 edificio Hay	ek.64
Figura 83. Aceleraciones vs tiempo identificando picos mayores que 0.0025g	65

CAPITULO I

INTRODUCCIÓN:

El monitoreo de vibraciones en las estructuras civiles es de gran importancia en el contexto actual, ya que estas estructuras tienen un impacto significativo en la vida moderna de las personas. Entre las principales estructuras civiles se encuentran puentes, centros comerciales, hospitales y centros educativos, entre otras. Este monitoreo es una herramienta fundamental en el campo de la ingeniería civil, especialmente en la ingeniería estructural, ya que permite evaluar la respuesta de las edificaciones frente a diversas fuentes de vibración, tales como sismos, el uso cotidiano de las estructuras, la operación de maquinaria, el flujo de tráfico y otras actividades que ponen a prueba su desempeño.

El monitoreo de vibraciones tradicional se realiza con instrumentos de alta precisión, pero estos equipos suelen tener un costo elevado, lo que dificulta su implementación a gran escala, especialmente en un contexto financiero como el de Ecuador. En respuesta a esta limitación, han surgido nuevos instrumentos de monitoreo de bajo costo, como los sismógrafos Raspberry Shake 4D V9. Estos dispositivos, aunque más asequibles, permiten una recopilación de datos eficaz con una instalación más sencilla y costos operativos considerablemente menores.

Este trabajo se enfoca en evaluar la eficiencia y efectividad de los Raspberry Shake 4D V9 en la recolección de datos de vibraciones provenientes de diversas fuentes. Para ello, se realizan pruebas de instalación y verificación de señales, comparando las curvas de datos obtenidas con las de referencias conocidas. El análisis y procesamiento de señales se lleva a cabo mediante software desarrollado en Python. Los resultados de esta investigación tienen como objetivo validar el uso de los sismógrafos Raspberry Shake 4D V9 como una herramienta accesible y efectiva para el monitoreo de vibraciones en estructuras.

1.1 Objetivos:

Se ha establecido como objetivo general lograr la implementación de los sismógrafos de bajo costo Raspberry Shake 4D V9.

Los objetivos específicos del proyecto son:

- Comprender el funcionamiento de los sismógrafos de bajo costo Raspberry Shake 4D
 V9.
- Realizar la instalación de los sismógrafos Raspberry Shake 4D V9 y procesar las señales obtenidas de los mismos.
- Realizar pruebas de verificación de las señales obtenidas de los sismógrafos.

CAPITULO II

INTRODUCCIÓN A LOS SISMÓGRAFOS DE BAJO COSTO RASPBERRY SHAKE 4D V9

Para este proyecto se utilizan sismógrafos Raspberry Shake 4D V9, que incorporan un acelerómetro de tecnología MEMS con tres sensores de aceleración en un solo módulo, distribuidos en las direcciones X, Y y Z (ver Figura 1).



Figura 1. Dispositivo Raspberry Shake 4D. (Imagen tomada de Raspberry Shake, 2024.)

- **Componente X**: mide la aceleración en la dirección horizontal, representada en el canal de datos "ENE" (Raspberry Shake Technical Specifications, 2021).
- Componente Y: registra la aceleración horizontal perpendicular, representada en "ENN". Esta orientación está indicada con una flecha en la tarjeta Raspberry (Raspberry Shake Technical Specifications, 2021).
- **Componente Z**: detecta la aceleración vertical, registrada en el canal "ENZ" (Raspberry Shake Technical Specifications, 2021).

Además del acelerómetro, el dispositivo cuenta con un sismógrafo vertical independiente, que también mide en la dirección Z (canal "EHZ") pero captura la velocidad del movimiento del suelo, lo que resulta útil para analizar ondas sísmicas de baja frecuencia. Esta combinación permite una caracterización detallada de las vibraciones: el acelerómetro (ENZ) es adecuado para frecuencias altas, mientras que el sismógrafo (EHZ) es más preciso en bajas frecuencias (Raspberry Shake Technical Specifications, 2021).

La gestión de estos sensores se realiza a través de una Raspberry Pi, que incluye:

Conectividad: un puerto Ethernet para conexión a internet y cuatro puertos
 USB 2.0 para conectar dispositivos externos (como almacenamiento, teclados o mouse) para configuración adicional si es necesario. (ver Figura 2).



Figura 2. Puertos laterales del dispositivo Raspberry Shake 4D, incluyendo Ethernet y USB. (Imagen tomada de Botland, 2024.)

- Visualización y almacenamiento: un puerto HDMI para conexión a pantalla y un lector de tarjetas SD (incluye sistema operativo) (ver Figura 3).
- Alimentación y otros puertos: puerto de alimentación USB y un conector de audio/video para monitores sin puerto HDMI (ver Figura 3) (Raspberry Shake Technical Specifications, 2021).



Figura 3. Puertos fontales del dispositivo Raspberry Shake 4D, incluyendo alimentación, HDMI y audio/video. (Imagen tomada de Botland, 2024.)

En este estudio se cuenta con cuatro unidades Raspberry Shake 4D V9 con características similares. Dos de ellas poseen certificación IP67, lo que garantiza resistencia al polvo y al agua hasta 1 metro de profundidad por 30 minutos. Las otras dos tienen certificación IP10, ofreciendo protección contra objetos sólidos grandes, pero sin resistencia al polvo o agua (Raspberry Shake Technical Specifications, 2021).

JUSTIFICACIÓN DEL USO DE SISMÓGRAFOS RASPBERRY SHAKE 4D V9

Los sismógrafos Raspberry Shake 4D V9 son adecuados para este proyecto debido a su relación costo-efectividad. Su costo de adquisición es de aproximadamente \$1084.99 USD (Raspberry Shake, 2024), lo cual representa una opción accesible en comparación con otros equipos más costosos. Además, la facilidad de instalación y configuración inicial permite que estos dispositivos sean implementados rápidamente por personas con conocimientos básicos, siguiendo las guías provistas.

La versatilidad de los Raspberry Shake también radica en su alta compatibilidad para la extracción y procesamiento de datos. Los datos obtenidos se pueden extraer fácilmente utilizando un script de Python proporcionado por el fabricante, y se pueden analizar con herramientas estándar como ObsPy o Matlab, sin requerir software especializado adicional.

Raspberry Shake menciona que la respuesta de los Raspberry entre las 0.5 Hz y los 40 Hz es plana, lo que significa que es una medida precisa y confiable, además de que señala que en ese rango de frecuencias el rendimiento del instrumento es comparable con la respuesta de otros sensores sísmicos más avanzados, lo que demuestra que la calidad de las mediciones no se ve comprometida por ser un instrumento de bajo costo.

Segun Raspberry Shake, "From ~2s to 40 Hz where the frequency response of the Raspberry Shake is flat, the performance is the same. Since automated processing systems like SeisComP3 typically filter from 1-5 Hz, this means a broadband and short-period sensors like the Raspberry Shake are essentially indistinguishable for earthquake detection purposes" (Raspberry Shake, 2016-2024).

Los Raspberry Shake son instrumentos más económicos y de menor tamaño que los acelerógrafos avanzados. Sin embargo, su rendimiento es adecuado para los fines de esta investigación, ya que operan en un rango de frecuencias de 0 a 44 Hz (Raspberry Shake Technical Specifications, 2021). En contraste, un acelerógrafo como el ETNA 2 de Kinemetrics tiene un rango de frecuencias más amplio, de 0 a 200 Hz (Kinemetrics, s.f.). Esta diferencia en el rango de frecuencias puede influir en la captura de detalles específicos de las vibraciones de alta frecuencia.

En cuanto a la frecuencia de muestreo, los Raspberry Shake operan a una tasa fija de 100 muestras por segundo, lo que puede limitar la capacidad de capturar detalles finos en comparación con acelerógrafos como el ETNA 2, cuya frecuencia de muestreo es variable y puede ajustarse entre 1 y 500 muestras por segundo. Para esta investigación, la frecuencia de 100 muestras por segundo es suficiente para el análisis de vibraciones en el rango de interés, aunque es importante tener en cuenta que una mayor frecuencia de muestreo podría proporcionar una resolución más detallada en los registros de aceleración, velocidad y desplazamiento.

Es necesario aclarar que los datos registrados por el Raspberry Shake corresponden a la aceleración, mientras que la velocidad y el desplazamiento se obtienen mediante procesos de integración numérica. Las diferencias en la frecuencia de muestreo y el rango de frecuencias entre los dispositivos pueden afectar los resultados del procesamiento de estas señales, especialmente en la precisión de las integraciones para obtener la velocidad y el desplazamiento.

CAPITULO III

FUNCIONAMIENTO Y CONFIGURACIÓN DE LOS RASPBERRY SHAKE 4D V9

La configuración inicial es un paso fundamental que debe realizarse la primera vez que una unidad de Raspberry Shake entra en funcionamiento. A continuación, se detalla la conexión de hardware paso a paso:

- 1. Retirar los envoltorios de los instrumentos.
- Colocar los pernos de nivelación de Raspberry Shake en los agujeros designados y ajustar hasta que la burbuja de nivelación se esté centrada(ver Figuras 4 y 5).



Figura 4. Ubicación de los pernos de nivelación en los 3 puntos designados.



Figura 5. Vista lateral de la instalación de los pernos de nivelación.

3. Verificar la nivelación observando el nivel de burbuja incorporado (ver Figura

6).



Figura 6. Visualización de la correcta nivelación del instrumento en el nivel de burbuja.

4. Conectar el cable Ethernet en el puerto correspondiente (ver Figura 7).



Figura 7. Conexión del cable ethernet en el puerto correspondiente.

 Conectar el cable de alimentación y asegurar que las luces de estado roja y azul estén encendidas (ver Figuras 8 y 9).



Figura 8. Conexión del cable de poder en su respectivo puerto.



Figura 9. Luces de estado en correcto funcionamiento.

- 6. Colocar el instrumento en el lugar designado para el estudio.
- Acceder desde el navegador de una computadora o dispositivo móvil conectado a la misma red a la dirección "rs.local" (ver Figura 10).



Figura 10. Menú de configuración e información del dispositivo disponible en "rs.local".

- Anotar la dirección IP del dispositivo identificada como "Ethernet Local IPV4".
- Ingresar al menú de configuración en la esquina superior izquierda de "rs.local" (ver Figura 11).



Figura 11. Diseño del botón de configuraciones de "rs.local".

 Completar los datos requeridos, incluyendo información personal, ubicación del dispositivo, número de plantas del edificio y el nivel donde se encuentra (ver Figuras 12, 13 y 14).

NERAL	DATOS RED DATACAS	г
¿Recibió e	geófono con el Raspberry Shake? *	
Sí	- O	
¿Cómo pie	nsa usar su Shake?	
¿Cómo pie Educaci	nsa usar su Shake? ón 🚽 🗊	
¿Cómo pie Educac	nsa usar su Shake? Ón 🚽 🗊	
¿Cómo pie Educaci	nsa usar su Shake? Ón 🚽 🗊 nbre	
¿Cómo pie Educaci Primer nor David	nsa usar su Shake? Ón – 1	
¿Cómo pie Educaci Primer nor David	nsa usar su Shake? Ón 🚽 Ŧ	
¿Cómo pie Educaci Primer nor David Apelido	nsa usar su Shake? Ón 🚽 1	
¿Cómo pie Educaci Primer nor David Apelido Salazar	nsa usar su Shake? ón 🖌 T	
¿Cómo pie Educaci Primer nor David Apelido Salazar	nsa usar su Shake? Ón – 1	
¿Cómo pie Educaci Primer nor David Apelido Salazar	nsa usar su Shake? Ón – 1	

Figura 12. Información que llenar en el menú de configuraciones.



Figura 13. Mapa donde se escoge la ubicación del instrumento.

Latitud
Longitud
-70.4304
Elevación (en metros) 2501
Planta en la que se encuentra el instrumento 0
Número total de plantas en el edificio. 4
GUARDAR Y REINICIAR

Figura 14. Datos de ubicación del instrumento y datos del edificio.

11. Selecciona la ubicación en el mapa; las coordenadas de Latitud, Longitud y Elevación se autocompletarán, pero se pueden ajustar manualmente si se dispone de datos más precisos. Indica el número de pisos de la estructura y el nivel donde está instalado el dispositivo, considerando el piso 0 como la planta baja. 12. Hacer clic en "Guardar y Reiniciar" para aplicar los cambios. Espera a que el dispositivo se reinicie automáticamente.

Una vez completado el proceso, el dispositivo está listo para funcionar. En la interfaz, cada servicio debería aparecer como "activado" y la conexión al servidor como "conectado" (ver Figura 15).



Figura 15. Dispositivo funcionando correctamente.

En el caso de que el estado de la conexión al servidor no se encuentre conectado podemos tener algunos escenarios.

1. Conexión a internet defectuosa

Este escenario ocurre cuando la red no tiene acceso a internet. Realizar una prueba de conexión desde otro dispositivo conectado a la misma red. Si la conexión funciona en otros dispositivos, el problema puede ser específico del Raspberry Shake. En ese caso, consultar el siguiente escenario.

2. Conexión a internet activa, pero el servidor sigue desconectado (conexión cableada).

Si otros dispositivos en la red tienen internet, pero el Raspberry Shake no se conecta al servidor, es posible que haya un error en la configuración del dispositivo. Para resolverlo:

- 1. Conectar el Raspberry Shake y seguir el procedimiento de encendido inicial.
- Acceder a "rs.local" para obtener la dirección IP (IPv4) si (aún no se ha realizado).
- Abrir el símbolo del sistema en una computadora con Windows conectada a la misma red presionando Win+R, escribir "cmd" y presionar "Enter" (ver Figura 16).



Figura 16. Símbolo del sistema de una computadora con sistema operativo Windows 10

4. Escribir el siguiente comando en la consola: "ssh

myshake@IPV4_del_dispositivo" y presionar "Enter" (ver Figura 17).



Figura 17. Comando "ssh myshake@IPV4".

5. Introducir la contraseña "shakeme" y presionar "Enter". Si aparece un mensaje de

confirmación, escribir "yes" y presionar "Enter" (ver Figura 18).



Figura 18. Bienvenida al sistema del Raspberry Shake.

- 6. Escribir "sudo nano /etc/dhcpcd.conf" y presionar "Enter".
- 7. Verificar que las primeras dos líneas contengan "interface eth0 y static

domain_name_servers=192.168.(0 o 1).1". El penúltimo dígito puede variar según el proveedor de internet. En una red institucional, consultar al personal de TI para obtener la dirección correcta (ver Figura 19).



Figura 19. Opciones dentro de dhcpcd.conf.

- Revisar que el "static domain_name_servers" sea adecuado para la red. Guardar los cambios con "Ctrl+O", presionar "Enter" y luego "Ctrl+X".
- Reiniciar el servicio con el comando "sudo systemctl restart dhcpcd" y presionar "Enter".
- 10. Revisar nuevamente en "rs.local" que la conexión al servidor aparezca como

"Conectado" después de unos minutos.

3. Conexión a internet mediante una red inalámbrica (WiFi).

Para conectar el Raspberry Shake a una red WiFi, seguir estos pasos, asegurándose de tener el nombre y la contraseña de la red. Se recomienda usar una conexión cableada para mayor estabilidad, y WiFi solo cuando sea necesario.

- 1. Realizar los pasos de configuración para una red cableada hasta el paso 5.
- Introducir el comando "sudo nano /etc/wpa_supplicant/wpa_supplicant.conf" y
 presionar "Enter" (ver Figura 20).



Figura 20. Contenido del comando "sudo nano /etc/wpa_supplicant/wpa_supplicant.conf".

3. Verificar que solo aparezcan las líneas mostradas en el archivo (ver Figura 20). Si

hay líneas adicionales, eliminarlas.

4. Proporcionar los datos de la red WiFi en el siguiente formato (ver Figura 21):



Figura 21. Configuración dentro de wpa_supplicant.

Modificar el ssid y el psk con el nombre y la contraseña de la red, y ajustar el valor de country según el código del país (por ejemplo, "EC" para Ecuador).

- 5. Guardar los cambios con "Ctrl+O", "Enter" y luego "Ctrl+X".
- 6. Reiniciar el dispositivo con el comando "sudo reboot" y presionar "Enter". Después del reinicio, el Raspberry Shake se conectará automáticamente a la red WiFi configurada. La conexión puede tardar alrededor de 5 minutos en estar operativa. Verificar el estado en "rs.local".

Visualización de los datos cargados al servidor de Raspberry Shake

Para visualizar los datos enviados por los instrumentos al servidor de Raspberry Shake, se pueden utilizar varios métodos. En este proyecto, se empleó el visor de datos en línea de Raspberry Shake, accesible en el siguiente enlace:

https://dataview.raspberryshake.org/#/.

Al ingresar, se accede a la aplicación web de visualización de datos de Raspberry Shake, que muestra la siguiente pantalla (ver Figura 22):

y Shake DataView	R 🕞 🐨	¢ 0
C		
A Filters		
*		
a, united states		

Figura 22. Raspberry Shake Data view vista de la página principal.

Esta aplicación permite visualizar los datos de Raspberry Shakes de todo el mundo. Los instrumentos de interés en este proyecto son las estaciones **R9B17**, **R03AE**, **R53C6** y **R666F**. Para localizar una estación específica, se utiliza la herramienta de búsqueda en la esquina superior izquierda. En el ejemplo, se buscó la estación **R03AE** y se seleccionó el canal **ENN** como canal de estudio (ver Figura 23).



Figura 23. Canal ENN de la estación R03AE visualizado mediante Data View de Raspberry Shake.

En el Data Viewer (Figura 23), se muestra toda la información subida al servidor por el Raspberry Shake en intervalos de tiempo. Los picos en el gráfico representan vibraciones detectadas, y es posible ampliar cualquier sección de interés haciendo clic en ella (ver Figura 24).



Figura 24. Acercamiento en un pico de vibración provocado para ejemplificación experimental.

Cada pico de vibración incluye información sobre la hora de inicio y fin en horario UTC, un aspecto importante al extraer datos, ya que los intervalos deben especificarse en UTC para asegurar la precisión en el análisis de los episodios de vibración.

CAPÍTULO IV

EXTRACCIÓN DE DATOS DESDE EL SERVIDOR DE RASPBERRY SHAKE UTILIZANDO UN SCRIPT DE PYTHON PROPORCIONADO POR EL FABRICANTE.

La extracción de datos desde los servidores de Raspberry Shake se puede realizar mediante un script de Python proporcionado por el fabricante. Para descargar el script, es necesario ingresar al siguiente enlace:

https://manual.raspberryshake.org/developersCorner.html#converting-counts-to-metric-unitsusing-obspy.

En esta página se encuentra el script de Python que, mediante el uso de la herramienta ObsPy, permite procesar datos sísmicos, extrayendo y convirtiendo los datos en unidades métricas. Cabe recalcar que se debe tener instalada la extensión de ObsPy que se puede hacer siguiendo las instrucciones de la siguiente página:

https://pypi.org/project/obspy/1.4.1/

Es importante comprender que los datos enviados por los Raspberry Shake al servidor están en unidades de "counts". Según Raspberry Shake (2016-2024), los "counts" son valores adimensionales que representan la cantidad de voltaje en el circuito de un canal en un momento específico. Este script solo funcionará si el Raspberry Shake funcionó en línea y ha enviado los datos al servidor. A continuación, se presentan los detalles para utilizar el script correctamente, mientras que el código completo se incluye en el Anexo A: SCRIPT DE PYTHON PARA LA EXTRACCIÓN DE DATOS DESDE EL SERVIDOR DE RASPBERRY SHAKE.

Programa propiedad de Raspberry Shake (Raspberry Shake, 2016-2024)

El script proporcionado permite al usuario extraer datos del servidor de Raspberry Shake utilizando la herramienta de procesamiento de datos sísmicos ObsPy, la cual debe estar instalada en el equipo que realizará la extracción. A continuación, se describe el funcionamiento del script:

1. Importación de bibliotecas necesarias

El programa comienza importando las bibliotecas de ObsPy necesarias para el procesamiento de datos sísmicos. Es importante no modificar estas líneas (ver Figura 25).

```
from obspy.clients.fdsn import Client
from obspy.core import UTCDateTime, Stream
```

Figura 25. Primeras líneas del código que importan las bibliotecas necesarias para el funcionamiento del script.

2. Inicialización de un cliente

El script crea un cliente de datos sísmicos que permite la conexión con la base de datos de Raspberry Shake. Esta línea no debe modificarse (ver Figura 26).



Figura 26. Línea de código que crea el cliente de datos sísmicos.

3. Descargar la información de la estación

Se establece el nombre de la estación de interés en la línea stn="nombre del instrumento". Para este proyecto, los nombres de los instrumentos son: **R9B17, R03AE, R53C6 y R666F**. Solo se debe ingresar el nombre de la estación entre comillas sin modificar otras partes del código. También se obtiene la información de respuesta instrumental, necesaria para la corrección de los datos (ver Figura 27).



Figura 27. Líneas para obtener la estación de interés y la respuesta instrumental.

4. Definir el intervalo de tiempo

En esta sección se establece la fecha y hora de inicio y finalización de los datos de interés. Se recomienda utilizar el visor de datos en línea de Raspberry Shake para obtener las horas exactas en formato UTC. Al ingresar horas o minutos menores a 10, no es necesario agregar un cero inicial (ver Figura 28).



Figura 28. Líneas de código donde se establece fecha y hora de inicio con un ejemplo de introducción de datos en números menores a 10.

5. Seleccionar los canales de datos

Este apartado permite especificar los canales de datos a extraer, útil en dispositivos distintos al modelo 4D. Para este proyecto, todos los dispositivos son modelo 4D, por lo que no es necesario realizar cambios en estas líneas (ver Figura 29).



Figura 29. Canales de datos de los que se quiere extraer la información.

6. Descargar y almacenar las formas de onda

Se crea un "stream" en ObsPy para almacenar los datos de cada canal (ver Figura 30).



Figura 30. Creación de un stream vacío y llenado con los datos de cada canal de datos del

Raspberry.

7. Graficar los datos sin procesar

El programa genera un gráfico con los datos sin procesar en "counts", sin unidades físicas (ver Figura 31).



Figura 31. Línea de código que grafica los datos sin procesar.

8. Adjuntar la información de la respuesta instrumental

Estas líneas asocian la respuesta instrumental al stream, permitiendo que los datos se conviertan a unidades físicas conocidas (m/s para velocidad y m/s² para aceleración) (ver Figura 32).



Figura 32. Línea que asocia la respuesta instrumental a los datos del stream.

9. Eliminar la respuesta instrumental y convertirla a unidades físicas

El código elimina la respuesta instrumental, convirtiendo los datos de "counts" a aceleración en m/s² (ver Figura 33).



Figura 33. Línea que elimina la respuesta instrumental y convierte los datos a unidades

físicas.

10. Graficar el nuevo stream en unidades físicas

Finalmente, se grafica el stream convertido a unidades físicas (ver Figura 34).

resp_removed.plot()

Figura 34. Línea de código que grafica la respuesta en unidades físicas.

Con estos pasos, el script extrae y visualiza los datos en el formato requerido para el análisis. A continuación, se presentan ejemplos de gráficos de control de la descarga de datos en counts y en unidades físicas (ver Figuras 35 y 36).



Figura 35. Gráfico ejemplo que contiene la respuesta en counts.



Figura 36. Gráfico ejemplo que contiene la respuesta en unidades físicas de la prueba de control.
ADICIÓN DE CÓDIGO PARA LA EXPORTACIÓN DE LA INFORMACIÓN DEL STREAM EN UN ARCHIVO .CSV PARA POSTERIOR PROCESADO DE DATOS.

Una vez descargados los datos, se añade un bloque de código adicional para guardar los datos en un archivo .CSV, facilitando su procesamiento posterior. A continuación, se explican los pasos para la creación del archivo .CSV:

1. Crear listas para almacenar datos de todos los canales

Se crea un diccionario data para almacenar los datos de tiempo y las cuatro componentes de aceleraciones de los distintos canales. Cada canal tendrá su propia lista para los valores de aceleración (ver Figura 37).



Figura 37. Líneas de código para la creación decanales para guardar la información.

2. Inicialización de la longitud mínima

La variable "min_length" se utiliza para determinar la longitud mínima entre los diferentes canales de datos, lo cual asegura una columna de tiempo compatible con todas las columnas de aceleración (ver Figura 38).



Figura 38. Función para establecer la longitud mínima.

3. Recopilación de datos de cada canal

Mediante un bucle "for", el script recorre cada trazado (trace) en el stream

"resp_removed", que ya contiene los datos convertidos a unidades físicas, y extrae los valores de cada canal (ver Figura 39).

```
for trace in resp_removed:
    channel = trace.stats.channel
    values = trace.data
```

Figura 39. Líneas de código que recopilan los datos de cada canal.

4. Generación de la columna de tiempo

Esta función genera una lista de tiempo compatible con todos los canales de datos.

Utiliza la frecuencia de muestreo del primer canal para crear una lista de tiempo que inicia en 0 y aumenta en intervalos según la frecuencia, cubriendo la duración completa de la muestra (ver Figura 40).





5. Generación del DataFrame y exportación a CSV

Estas líneas de código utilizan Pandas para crear un DataFrame con una columna para cada canal de datos: "time", "EHZ", "ENE", "ENN" y "ENZ". Luego, el archivo .CSV se guarda en la misma ubicación donde se encuentra el script de Python, por lo que es necesario conocer esta ruta (ver Figura 41).



Figura 41. Líneas de código que recopilan los canales dentro del archivo CSV y lo guardan.

Todo el proceso se resume en el programa completo que se encuentra en el Anexo B, permitiendo la extracción de datos desde el servidor de Raspberry Shake y la creación de un archivo .CSV con las lecturas de velocidad y aceleración medidas.

CAPITULO V

PROCESADO DE SEÑALES MEDIANTE EL USO DE UN CÓDIGO GENERADO EN MATLAB.

El procesamiento de las señales de aceleración, velocidad y desplazamiento fue realizado a través de un código en MATLAB diseñado específicamente para extraer y analizar los datos registrados por el sensor Raspberry Shake 4D.

Primeramente, se realizó una prueba con los instrumentos instalados en un lugar alejado de fuentes externas de ruido. Las mediciones se llevaron a cabo durante toda una noche, de tal forma que el único ruido registrado provenía únicamente del propio dispositivo. La franja de datos seleccionada fue de 2 a 4 am (ver Figura 42).



Figura 42. Instrumento instalado lejos de fuentes de ruido.

Una vez finalizado el proceso de medición, los datos se descargaron desde el servidor utilizando el script descrito en el Anexo B: CÓDIGO DE PYTHON PARA DESCARGA Y EXPORTACIÓN DE DATOS DESDE EL SERVIDOR DE RASPBERRY SHAKE A UN ARCHIVO .CSV. Posteriormente, se realizó la transformada de Fourier de la señal obtenida, con el objetivo de identificar las frecuencias de filtrado asociadas al ruido instrumental. El análisis reveló que las frecuencias de corte se encuentran por debajo de 0.5 Hz, por lo que se implementó un filtro pasa-altas configurado para permitir únicamente las frecuencias superiores a este valor, eliminando así toda información no deseada con frecuencias menores a 0.5 Hz (ver Figura 43).



Figura 43. Transformada de Fourier de la señal en reposo.

Como se muestra en la Figura 43, el conjunto de frecuencias dominantes se encuentra por debajo de 0.5 Hz. Luego se aplicó un filtro pasa altas para permitir que los datos con frecuencias por encima de 0.5 Hz se mantengan. Posteriormente, se procede a realizar una prueba comparando el registro de aceleraciones de la señal sin filtrar que se muestra en color azul y la filtrada que se muestra en color rojo, en la cual se obtiene lo siguiente (ver Figura 44).



Figura 44. Comparación señal en reposo sin filtrar y filtrada.

Como se observa en la Figura 44 el filtrado de la señal logra eliminar los picos ubicados alrededor de 0.5 Hz en la transformada de Fourier.

El procesamiento de las señales de aceleración, velocidad y desplazamiento fue realizado mediante un código desarrollado en MATLAB. Este código permite filtrar y procesar los datos obtenidos por los sensores Raspberry Shake 4D, asegurando la eliminación de ruido instrumental y la obtención precisa de velocidades y desplazamientos. A continuación, se describe cada etapa del procesamiento aplicado: El código completo puede encontrarse en el Anexo C: CÓDIGO EN MATLAB PARA EL PROCESADO DE SEÑALES DE ACELERACION, VELOCIDAD Y DESPLAZAMIENTO.

1. Importación de Datos y Parámetros Iniciales

Los datos de aceleración se importan desde un archivo .CSV generado por el Raspberry Shake 4D. Para cada componente (ENN, ENE, ENZ), los valores se normalizan dividiendo entre 9.81 para convertir de metros por segundo cuadrado (m/s²) a unidades de gravedad (g). Además, se establecen parámetros clave para el procesamiento, como:

- dt: Intervalo de muestreo, definido en 0.01 segundos para el modelo de Raspberry Shake utilizado.
- order: Orden del filtro Butterworth, en este caso 2, recomendado para un balance entre suavizado y precisión.
- fr: Frecuencia de corte alta para eliminar ruido instrumental. Se estableció en 0.5 Hz para todas las componentes (ver Figura 45).

```
%% Integrador para obtener velocidad y desplazamiento
% Datos
lw = 0.1; fsa = 6; fsl = 6; fst = 14;
ARCH = 'Prueba_4.csv'; % Nombre del archivo .CSV que se desea leer
dt = 0.01; % Intervalo de muestreo (s) para RSShake 4D V9
order = 2; % Orden del filtro (recomendado entre 1 y 6)
fr = 0.5; % Frecuencia de corte alta para el filtro (Hz)
%% Importar los datos recopilados con el Raspberry Shake
locFolder1 = 'C:\Users\david\OneDrive\Escritorio\David\USFQ\Ultimo_Semestre\Tesis\Matlab';
cd(locFolder1);
M = readtable(ARCH, 'NumHeaderLines', 1);
TSX = table2array(M(:,1)); % Tiempo extraído del archivo CSV
ASENN = table2array(M(:,4)) / 9.81; % Aceleración ENN en g
ASENE = table2array(M(:,5)) / 9.81; % Aceleración ENZ en g
```



2. Filtrado de Señales de Aceleración

Las señales de aceleración crudas contienen ruido de baja frecuencia que puede distorsionar los resultados de las integraciones subsiguientes. Para mitigar este efecto, se aplicó un filtro pasa-altas Butterworth de orden 2 con una frecuencia de corte de 0.5 Hz, eliminando efectivamente el ruido no deseado en las tres componentes: ENN, ENE y ENZ (ver Figura 46).

```
%% Filtrado de las señales de aceleración
fs = 1 / dt; % Frecuencia de muestreo (Hz)
[b, a] = butter(order, fr / (fs / 2), 'high'); % Filtro pasa-altas
% Filtrar las señales
ASENN_filtered = filtfilt(b, a, ASENN);
ASENE_filtered = filtfilt(b, a, ASENE);
ASENZ_filtered = filtfilt(b, a, ASENZ);
```

Figura 46: Filtrado de datos de aceleración.

3. Integración para Obtención de Velocidad

Luego del filtrado, las señales de aceleración se integran numéricamente usando el método trapezoidal (cumtrapz) para calcular la velocidad en cada componente (ENN, ENE y ENZ) en unidades de cm/s. Para evitar tendencias no deseadas, se utilizó la función detrend, lo que asegura que las velocidades comiencen desde cero y estén libres de deriva acumulada (ver Figura 47).

```
%% Integración de la aceleración para obtener la velocidad en cm/s
VELENN = cumtrapz(TSX, ASENN filtered * 9.81 * 100); % Velocidad ENN
VELENE = cumtrapz(TSX, ASENE_filtered * 9.81 * 100); % Velocidad ENE
VELENZ = cumtrapz(TSX, ASENZ_filtered * 9.81 * 100); % Velocidad ENZ
% Corregir tendencia y valores iniciales de velocidad
VELENN = detrend(VELENN) - VELENN(1);
VELENE = detrend(VELENE) - VELENE(1);
VELENZ = detrend(VELENZ) - VELENZ(1);
%% Integración de la velocidad para obtener el desplazamiento en mm
DESENN = cumtrapz(TSX, VELENN * 10); % Desplazamiento ENN
DESENE = cumtrapz(TSX, VELENE * 10); % Desplazamiento ENE
DESENZ = cumtrapz(TSX, VELENZ * 10); % Desplazamiento ENZ
% Corregir tendencia y valores iniciales de desplazamiento
DESENN = DESENN - mean(DESENN);
DESENE = DESENE - mean(DESENE);
DESENZ = DESENZ - mean(DESENZ);
```

Figura 47: Integración para obtención de velocidades y desplazamientos.

4. Visualización de los Resultados

Finalmente, se generaron gráficos para visualizar las señales procesadas de

aceleración, velocidad y desplazamiento en función del tiempo para las tres componentes:

- Aceleración: Representada en g, mostrando la señal cruda filtrada.
- Velocidad: Obtenida a partir de la integración de la aceleración, en cm/s.
- **Desplazamiento**: Derivado de la integración de la velocidad y ajustado con un filtrado adicional, en mm.

Los gráficos se presentan en una matriz de 3x3, con cada columna correspondiente a

una de las componentes (ENN, ENE, ENZ), facilitando una comparación visual clara (ver Figura 48).



Figura 48. Resultados de procesamiento de señal correspondiente a la simulación del sismo de Northridge.

CAPITULO VI

PRUEBAS DE FUNCIONAMIENTO DE LOS RASPBERRY SHAKE 4D V9

Pruebas de funcionamiento con mesa vibratoria marca Quanser

Se realizaron varias pruebas de funcionamiento con los Raspberry Shake 4D V9 para corroborar la precisión de los datos recopilados por los instrumentos. Para las pruebas de funcionamiento se utilizó una mesa de vibraciones de la marca Quanser, modelo Shake Table III XY, propiedad de la Facultad de Ingeniería Civil de la Universidad Central del Ecuador. La mesa tiene capacidad de simular los registros de aceleración de sismos pasados (como el sismo de Pedernales del 2016), pero ajustando el intervalo de tiempo de cada data de la señal, de tal forma que el desplazamiento generado no exceda los límites de desplazamiento de la mesa vibratoria (Límite de 3 cm recomendado por el operador de la máquina).

La prueba realizada se llevó a cabo siguiendo el siguiente proceso:

1. Pesar el instrumento junto con la base de sujeción (ver Figura 49).



Figura 49: Pesaje del instrumento junto con la base de sujeción.

2. Fijar el conjunto instrumento-base a la mesa vibratoria (ver Figura 50).



Figura 50: Sistema instalado en la mesa vibratoria.

3. Comprobar la configuración de la mesa vibratoria (ver Figura 51y 52).



Figura 51: Comparación entre aceleraciones y desplazamientos reales vs escalados.



Figura 52: Nuevo delta de tiempo generado por el escalado de la información.

4. Ejecutar la prueba.

El proceso anterior fue realizado en un total de 2 veces. La primera prueba se realizó con la simulación del sismo de Pedernales (2016). La segunda prueba se realizó la simulación del sismo de Northridge (1994). Una vez realizadas las pruebas, se procede a realizar el procesamiento de los datos recopilados mediante los instrumentos Raspberry Shake, y se los compara con los datos medidos por la mesa vibratoria.

1. Resultados simulación sismo de Pedernales (2016).

La primera parte de los resultados obtenidos corresponde a la réplica del sismo de Pedernales. A continuación, se muestra los resultados del procesamiento de la aceleración en el sentido "X" de la mesa que corresponde a la componente "ENN" del Raspberry Shake, y en el sentido "Y" que corresponde a la componente "ENE". Se realizó un ajuste en la señal medida por el Raspberry Shake haciendo recortes de datos para que coincidan las dos curvas en el tiempo. Durante el procesamiento de los datos obtenidos con el Raspberry Shake se realizaron ajustes en la componente temporal. Esto se llevó a cabo con el propósito de alinear las dimensiones de las matrices de datos provenientes del Raspberry Shake con las obtenidas de la mesa vibratoria, garantizando así su compatibilidad para el análisis comparativo. Una vez ajustadas las curvas se graficaron simultáneamente para comprobar su ajuste (ver Figura 53).



Figura 53: Resultados de aceleración procesados y comparativa.

A continuación, se muestra más a detalle la comparativa entre la curva generada por la mesa vibratoria y la curva generada por el Raspberry Shake (ver Figura 54 y 55).



Figura 54: Superposición de curvas generadas por mesa vibratoria vs Raspberry Shake

sentido "X" o "ENN".



Figura 55: Superposición de curvas generadas por mesa vibratoria vs Raspberry Shake sentido "Y" o "ENE".

El análisis visual de los resultados revela una alta similitud entre las curvas. Para una mejor apreciación de esta similitud, se presenta a continuación un acercamiento a un segmento de la curva, lo cual permite una inspección visual más detallada (ver Figuras 56 y 57).



Figura 56: Detalle de la superposición de curvas generadas por mesa vibratoria vs Raspberry Shake sentido "X" o "ENN".



Figura 57: Detalle de la superposición de curvas generadas por mesa vibratoria vs Raspberry Shake sentido "Y" o "ENE".

Tras la comprobación visual de las curvas de aceleración, se procede a integrar los datos bajo la curva de aceleración para obtener las curvas de velocidad correspondientes a ambas señales. Posteriormente, se realiza una inspección y comparación visual de estas curvas de velocidad. Los resultados son los siguientes (ver Figura 58):



Figura 58: Gráficos de Velocidad vs Tiempo.

En la Figura 58, se presenta la comparación entre la curva de velocidad obtenida de la mesa vibratoria y la generada por el Raspberry Shake. A continuación, se muestra un análisis más detallado de la comparativa de estas curvas de velocidad (ver Figuras 59 y 60).



Figura 59: Velocidad vs tiempo del sentido "X" o "ENN".



Figura 60: Velocidad vs tiempo del sentido "Y" o "ENE".

El análisis visual de los resultados revela una alta similitud entre las curvas. Para una mejor apreciación de esta similitud, se presenta a continuación un acercamiento a un segmento de la curva, lo cual permite una inspección visual más detallada (ver Figuras 61 y 62).



Figura 61: Comparación de velocidades sentido "X" o "ENN".



Figura 62: Comparación de velocidades sentido "Y" o "ENE".

Después de la comparación de las curvas de velocidad, se procede a evaluar las curvas de desplazamiento correspondientes a ambas señales. Posteriormente, se realiza una inspección y comparación visual de estas curvas de desplazamiento. Los resultados se presentan a continuación (ver Figura 63):



Figura 63: Comparación Desplazamiento vs Tiempo

En la Figura 63, se presenta la comparación entre la curva de desplazamiento obtenida de la mesa vibratoria y la generada por el Raspberry Shake. A continuación, se muestra un análisis más detallado de la comparativa de estas curvas de desplazamiento (ver Figuras 64 y 65).



Figura 64: Detalle de la curva de desplazamiento sentido "X" o "ENN".



Figura 65: Detalle de la curva de desplazamiento sentido "Y" o "ENE".

En las figuras 64 y 65 se observa una gran similitud en la forma de las curvas de desplazamiento obtenidas, lo que nos demuestra que el proceso de integración de los datos es correcto.

2. Resultados simulación sismo de Northridge (1994).

La segunda parte de los resultados obtenidos corresponde a la simulación del sismo de Northridge. Se realizó un ajuste en la señal medida por el Raspberry Shake haciendo recortes de datos para que coincidan las dos curvas en el tiempo. Durante el procesamiento de los datos obtenidos con el Raspberry Shake se realizaron ajustes en la componente temporal. Esto se llevó a cabo con el propósito de alinear las dimensiones de las matrices de datos provenientes del Raspberry Shake con las obtenidas de la mesa vibratoria, garantizando así su compatibilidad para el análisis comparativo. Una vez ajustadas las curvas se graficaron simultáneamente para comprobar su ajuste (ver Figura 66).



Figura 66: Resultados de aceleración procesados y comparativa.

A continuación, se muestra en mayor detalle la comparación entre la curva generada

por la mesa vibratoria y la curva generada por el Raspberry Shake (ver Figuras 67 y 68).



Figura 67: Superposición de curvas generadas por mesa vibratoria vs. Raspberry Shake en el

sentido "X" o "ENN".



Figura 68: Superposición de curvas generadas por mesa vibratoria vs. Raspberry Shake en el sentido "Y" o "ENE".

El análisis visual de los resultados revela una alta similitud entre las curvas. Para una mejor apreciación de esta similitud, se presenta a continuación un acercamiento a un segmento de la curva, lo cual permite una inspección visual más detallada (ver Figuras 69 y 70).



Figura 69: Detalle de la superposición de curvas generadas por mesa vibratoria vs. Raspberry Shake en el sentido "X" o "ENN".



Figura 70: Detalle de la superposición de curvas generadas por mesa vibratoria vs. Raspberry Shake en el sentido "Y" o "ENE".

Tras la comprobación visual de las curvas de aceleración, se procede a integrar los datos bajo la curva de aceleración para obtener las curvas de velocidad correspondientes a ambas señales. Posteriormente, se realiza una inspección y comparación visual de estas curvas de velocidad. Los resultados son los siguientes (ver Figura 71):



Figura 71: Velocidad vs. Tiempo.

En la Figura 71, se presenta la comparación entre la curva de velocidad obtenida de la mesa vibratoria y la generada por el Raspberry Shake. A continuación, se muestra un análisis más detallado de la comparativa de estas curvas de velocidad (ver Figuras 72 y 73).



Figura 72: Velocidad vs. tiempo del sentido "X" o "ENN".



Figura 73: Velocidad vs. tiempo del sentido "Y" o "ENE".

El análisis visual de los resultados revela una alta similitud entre las curvas. Para una mejor apreciación de esta similitud, se presenta a continuación un acercamiento a un segmento de la curva, lo cual permite una inspección visual más detallada (ver Figuras 74 y 75).



Figura 74: Comparación de velocidades en el sentido "X" o "ENN".



Figura 75: Comparación de velocidades en el sentido "Y" o "ENE".

Después de la comparación de las curvas de velocidad, se procede a evaluar las curvas de desplazamiento correspondientes a ambas señales. Posteriormente, se realiza una inspección y comparación visual de estas curvas de desplazamiento. Los resultados se presentan a continuación (ver Figura 76):



Figura 76: Comparación de Desplazamiento vs. Tiempo.

En la Figura 76, se presenta la comparación entre la curva de desplazamiento obtenida de la mesa vibratoria y la generada por el Raspberry Shake. A continuación, se muestra un análisis más detallado de la comparativa de estas curvas de desplazamiento (ver Figuras 77 y 78).



Figura 77: Desplazamiento en el sentido "X" o "ENN".



Figura 78: Desplazamiento en el sentido "Y" o "ENE".

En las Figuras 77 y 78 se observa una gran similitud en la forma de las curvas de desplazamiento obtenidas, lo que demuestra que el proceso de integración de los datos es correcto. Actualmente, se está buscando una forma de optimizar y mejorar el proceso de filtrado de la señal para obtener los mejores resultados.

Límites de Vibración por Tipo de Ocupación en el Análisis con Raspberry Shake

Para el análisis de vibraciones en sistemas de piso, es esencial contar con límites de vibración definidos para evaluar la comodidad y seguridad de los usuarios en función del tipo de ocupación. Los valores de aceleración permitidos, expresados como un porcentaje de la gravedad (%g), varían según la ocupación del espacio y las expectativas de confort de los usuarios (American Institute of Steel Construction [AISC], 2016). Aunque estos límites se desarrollaron en el contexto de estructuras de acero, son aplicables como referencia general en el análisis de vibraciones de distintos sistemas de piso:

Oficinas y residencias: En estos entornos, el límite recomendado de aceleración pico es de aproximadamente 0.5%g, lo cual minimiza la percepción de vibración y asegura un nivel de comodidad adecuado para los ocupantes (AISC, 2016, p. 11).

Restaurantes y áreas de comidas: Los usuarios en estos espacios toleran mayores niveles de vibración que en oficinas y residencias. Se recomienda un límite de aceleración entre 1.5%g y 2.5%g para evitar incomodidades en estos entornos (AISC, 2016, p. 12).

Espacios de actividad rítmica: En gimnasios y áreas de ejercicio, donde se anticipa un alto nivel de movimiento, se permite una mayor aceleración. Los límites de vibración aceptables en estos espacios oscilan entre 4%g y 7%g (AISC, 2016, pp. 12-13).

Estos límites de vibración sirven como base para evaluar la respuesta de estructuras existentes. El uso de sismógrafos de bajo costo, como el Raspberry Shake, permite monitorear y registrar las vibraciones en tiempo real, facilitando la comparación de las aceleraciones medidas con los valores límite recomendados. De esta forma, es posible determinar si las vibraciones en una estructura existente se encuentran dentro de los rangos aceptables para cada tipo de ocupación, proporcionando una herramienta accesible y precisa para la comprobación de condiciones de confort y seguridad en sistemas de piso en uso (AISC, 2016, p. 12).

Medición de vibraciones en la Universidad San Francisco de Quito

Se llevó a cabo la medición de las vibraciones generadas por el ingreso vehicular a través de la rampa de acceso al nivel S1 del centro comercial Paseo San Francisco. El análisis se enfocó en determinar la influencia de estas acciones en las vibraciones registradas en las oficinas del Campus Hayek de la Universidad San Francisco de Quito, específicamente en el tercer piso de la edificación.

Para esta prueba, se utilizó la estación Raspberry Shake 4D V9 identificada como base R9B17, la cual fue instalada en el tercer piso del edificio Hayek. La ubicación específica de la base fue al lado de la oficina H326, dentro de la sala de reuniones del bloque de oficinas Hayek (ver Figura 79).



Figura 79. Estación R9B17 instalada en el piso 3 del edificio Hayek.

Adicionalmente, se instaló la estación base R03AE en el subsuelo 1, al lado de la rampa de ingreso vehicular. En este lugar, se dispuso un monitor que permitió observar en tiempo real las vibraciones generadas por los vehículos durante su ingreso (ver Figura 80 y 81).



Figura 80: Estación R03AE instalada junto a la rampa de acceso.



Figura 81. Estación R03AE instalada al lado de la rampa de acceso del S1.

Con ambas estaciones instaladas, se procedió a realizar el monitoreo durante las horas pico de ingreso al parqueadero, entre las 9:45 am y las 12:00 pm. Durante este periodo, se observó que el ingreso de los vehículos genera vibraciones que se transmiten directamente hacia las oficinas ubicadas en el tercer piso. Posteriormente, se llevó a cabo un análisis detallado de las vibraciones para determinar la atenuación de estas a través de la estructura. Los resultados obtenidos se presentan a continuación. La curva de aceleración medida en el Piso 3 se muestra en color azul y la curva de aceleración medida en el S1 se muestra en color rojo (ver Figura 82).



Figura 82. Gráficas Aceleración vs tiempo obtenidas en el S1 y en el piso 3 edificio Hayek.

Como se muestra en la Figura 82, las vibraciones generadas por el ingreso vehicular al subsuelo 1 se transmiten directamente al tercer piso del edificio. Estas vibraciones llegan con un nivel de atenuación que se analiza identificando, en primer lugar, la amplitud de las vibraciones en instantes de tiempo similares. Para ello, se identificaron los picos de aceleración mayores a 0.0025 g, los cuales se utilizaron como referencia para realizar la comparación en las gráficas. En las gráficas de Aceleración vs Tiempo de color azul, se usaron puntos rojos para resaltar los picos de aceleraciones superiores a 0.0025g(ver Figura 83).



Figura 83. Aceleraciones vs tiempo identificando picos mayores que 0.0025g.

En la Figura 83 se observa que los picos con amplitud superior a 0.0025 g son significativamente más frecuentes en la señal registrada en el subsuelo 1. Esto evidencia que la estructura atenúa las vibraciones antes de que estas lleguen al tercer piso. Al comparar los valores de los picos más relevantes en ambas señales, se puede calcular la atenuación promedio de las vibraciones generada por la estructura (ver Tabla 1).

ID Pico	Amplitud Piso 3	Amplitud S1	%Diferencia	%Atenuación
1	0.00339703	0.00587668	57.81%	42.19%
2	0.00366049	0.00535488	68.36%	31.64%
3	0.00338062	0.00498486	67.82%	32.18%
4	0.00284236	0.00453332	62.70%	37.30%
5	0.00263621	0.00414359	63.62%	36.38%
Promedio				35.94%

Tabla 1. Atenuación de vibraciones promedio picos más relevantes.

En la Tabla 1 se observa que la atenuación promedio de las vibraciones generada por la estructura del edificio Hayek es del 35.94%. Este nivel de atenuación permite que, aunque en el subsuelo 1 se registren vibraciones que superan los 0.005 g, las vibraciones que llegan al bloque de oficinas con esta atenuación no superan el límite de 0.005g propuesto por la norma.

CONCLUSIONES

Este estudio ha demostrado la viabilidad de los sismógrafos de bajo costo Raspberry Shake 4D V9 para el monitoreo de vibraciones en estructuras civiles y eventos sísmicos, validando su uso como una herramienta accesible y precisa. Las pruebas realizadas confirmaron que estos dispositivos, aunque presentan limitaciones en su rango de frecuencias y diseño, pueden capturar datos con un alto grado de precisión cuando se emplean metodologías de procesamiento adecuadas.

Los resultados obtenidos en simulaciones controladas, como las pruebas en la mesa vibratoria con réplicas de los sismos de Pedernales (2016) y Northridge (1994), mostraron una correlación significativa entre las curvas generadas por el Raspberry Shake y los registros reales. Esto valida la capacidad del dispositivo para medir con precisión aceleraciones en estructuras sometidas a condiciones dinámicas. Además, el análisis detallado de las mediciones de vibraciones en la Universidad San Francisco de Quito evidenció que el dispositivo es útil para evaluar la transmisión de vibraciones en diferentes niveles estructurales, destacando su aplicación en el monitoreo de estructuras existentes.

Un hallazgo clave fue la identificación de parámetros óptimos para el procesamiento de señales, como la frecuencia de corte a 0.5 Hz en filtros pasa-altas, lo que permitió eliminar ruido instrumental sin comprometer la calidad de los datos. Asimismo, la integración de herramientas de software como Python y MATLAB resultó esencial para optimizar el análisis y maximizar el potencial del dispositivo, facilitando procesos como la conversión de datos de aceleración en unidades métricas y la creación de gráficos comparativos.

Desde una perspectiva aplicada, este proyecto resalta la importancia de disponer de tecnologías económicas para el monitoreo estructural, particularmente en países sísmicamente activos como Ecuador, donde la implementación de estos dispositivos podría mejorar significativamente la gestión del riesgo y la seguridad estructural. Por ejemplo, su uso en redes colaborativas de monitoreo puede proporcionar datos valiosos para la prevención de desastres y la optimización de diseños estructurales.

En términos personales, este trabajo me permitió fortalecer mis competencias en el procesamiento de señales y el manejo de herramientas avanzadas, al tiempo que desarrollé soluciones para desafíos técnicos específicos, como la calibración de dispositivos y la eliminación de deriva en los datos integrados. Estos aprendizajes subrayan la necesidad de continuar investigando para mejorar las capacidades de los Raspberry Shake, incluyendo el aprovechamiento del sensor vertical de velocidad y la implementación de filtros más avanzados.

En conclusión, los Raspberry Shake 4D V9 no solo representan una alternativa viable y económica para el monitoreo de vibraciones, sino que también abren la puerta a futuras investigaciones y desarrollos tecnológicos en ingeniería civil, posicionándose como una solución valiosa en contextos locales e internacionales.

Recomendaciones para estudios futuros:

Uso de la componente del sismógrafo vertical EHZ: Investigar métodos para aprovechar el sensor de velocidad vertical que dispone el Raspberry Shake.

Estudios de impacto socioeconómico: Evaluar cómo la implementación de estos dispositivos puede influir en la reducción de riesgos y costos asociados a fallos estructurales en sectores sísmicamente activos.

Desarrollo de redes de monitoreo colaborativo: Explorar la creación de redes de monitoreo colaborativo que integren múltiples estaciones Raspberry Shake en una zona sísmicamente activa. En conclusión, este trabajo no solo demuestra la aplicabilidad de los Raspberry Shake 4D V9 en el ámbito de la ingeniería civil, sino que también abre la puerta a nuevas oportunidades de investigación y desarrollo tecnológico, posicionando estos dispositivos como un recurso valioso tanto en el contexto nacional como internacional.

REFERENCIAS BIBLIOGRÁFICAS

- American Institute of Steel Construction (AISC). (2016). Design guide 11: Vibrations of steel-framed structural systems due to human activity (2nd ed.). Chicago, IL: American Institute of Steel Construction.
- Arias, D. (2023). Análisis de vibraciones estructurales mediante Raspberry Shake. [Tesis de maestría, Universidad Autónoma del Estado de México].
- Botland. (2024). *Raspberry Shake 4D Pi Full Kit: Overlay with a seismograph*. Recuperado de https://botland.store/withdrawn-products/10010-raspberry-shake-4d-pi-full-kit-overlay-with-a-seismograph-5904422338701.html
- Ferrández Vega, D. (2023). *Monitoreo de estructuras mediante sensores sísmicos*. [Tesis de maestría, Universidad Politécnica de Valencia].
- IRIS. (n.d.). What is a count in time series data?. Recuperado de https://ds.iris.edu/ds/support/faq/6/what-is-a-count-in-timeseries-data/.
- International Standards Organization (ISO). (1989). ISO 2631-2: Evaluation of human exposure to whole-body vibration—Part 2: Human exposure to continuous and shockinduced vibrations in buildings (1 to 80 Hz). Geneva, Switzerland: ISO.
- International Standards Organization (ISO). (2007). *ISO 10137: Bases for design of structures—Serviceability of buildings and walkways against vibrations*. Geneva, Switzerland: ISO.

Kinemetrics. (2017). *ETNA2 Accelerograph Datasheet*. Recuperado de https://kinemetrics.com/wp-content/uploads/2017/04/datasheet-etna2-accelerograph-kinemetrics.pdf

ObsPy Development Team. (n.d.). *ObsPy Documentation: A Python toolbox for seismology*. Recuperado de https://docs.obspy.org/tutorial/index.html.

Python Package Index (PyPI). (n.d.). *ObsPy Package*. Recuperado de https://pypi.org/project/obspy/.

- Raspberry Shake. (2021). *RS Global Technical Specifications Document*. Recuperado de https://raspberryshake.org/.
- Raspberry Shake. (n.d.). *Developer's Corner: Converting counts to metric units using ObsPy*. Recuperado de https://manual.raspberryshake.org/developersCorner.html#converting-counts-to-metric-units-using-obspy.
- SIATA. (2023). *Aplicación de Raspberry Shake 4D en el monitoreo sísmico*. Documento técnico SIATA.
- Vanstone, W. M., & Sheeny72. (n.d.). *RPiShakeCode & RPiSandB GitHub Repositories*. Recuperado de https://github.com/wmvanstone/RPiShakeCode y https://github.com/sheeny72/RPiSandB.

ANEXO A: SCRIPT DE PYTHON PARA LA EXTRACCIÓN DE DATOS DESDE EL SERVIDOR DE RASPBERRY SHAKE

from obspy.clients.fdsn import Client

from obspy.core import UTCDateTime, Stream

rs = Client('RASPISHAKE')

set the station name and download the response information

stn = 'R24FA' # your station name

inv = rs.get_stations(network='AM', station=stn, level='RESP')

set data start/end times

start = UTCDateTime(2020, 3, 7, 2, 40, 45) # (YYYY, m, d, H, M, S)

end = UTCDateTime(2020, 3, 7, 2, 42, 0) # (YYYY, m, d, H, M, S)

set the FDSN server location and channel names

channels = ['EHZ', 'ENE', 'ENN', 'ENZ'] # ENx = accelerometer channels; EHx or SHZ =

geophone channels

get waveforms and put them all into one Stream

stream()

for ch in channels:

trace = rs.get_waveforms('AM', stn, '00', ch, start, end)

stream += trace

here you will plot raw counts for all four channels
stream.plot()

attach the response info

stream.attach_response(inv)

remove the response and plot

resp_removed = stream.remove_response(output='ACC') # convert to acceleration in M/S/S

now plot all channels in acceleration units (m/s/s)
resp_removed.plot()

Programa autoría de Raspberry Shake
ANEXO B: CÓDIGO DE PYTHON PARA DESCARGA Y EXPORTACIÓN DE DATOS DESDE EL SERVIDOR DE RASPBERRY SHAKE A UN ARCHIVO .CSV

import pandas as pd import matplotlib.pyplot as plt from obspy.clients.fdsn import Client from obspy.core import UTCDateTime, Stream

Configuración del cliente y la estación

rs = Client('RASPISHAKE')

stn = 'R666F' # Nombre de tu estación

inv = rs.get_stations(network='AM', station=stn, level='RESP')

Configuración de tiempos y canales

start = UTCDateTime(2024, 9, 9, 16, 0, 53)

end = UTCDateTime(2024, 9, 9, 16, 1, 4)

channels = ['EHZ', 'ENE', 'ENN', 'ENZ']

Obtener y combinar las formas de onda

stream()

for ch in channels:

trace = rs.get_waveforms('AM', stn, '00', ch, start, end)

stream += trace

Graficar datos crudos

stream.plot()

Adjuntar respuesta y removerla para convertir a aceleración/velocidad

stream.attach_response(inv)

resp_removed = stream.remove_response(output='ACC')

Graficar datos en aceleración/velocidad

resp_removed.plot()

Crear diccionario para almacenar los datos

data = {

'time': [], # Tiempo en segundos

'EHZ': [], # Velocidad (m/s) para EHZ

'ENE': [], # Aceleración (m/s²) para ENE

'ENN': [], # Aceleración (m/s²) para ENN

'ENZ': [] # Aceleración (m/s²) para ENZ

}

Variable para la longitud mínima de los datos

min_length = None

Recopilar datos para cada canal

for trace in resp_removed:

channel = trace.stats.channel

values = trace.data

if len(data['time']) == 0:

min_length = len(values)

Generar tiempos en función del intervalo de muestreo

sampling_rate = trace.stats.sampling_rate

data['time'] = [i / sampling_rate for i in range(min_length)]

if channel in data:

if len(values) < min_length:

min_length = len(values)

data[channel] = values[:min_length].tolist()

Asegurarse de que el tiempo tenga la misma longitud que los datos data['time'] = data['time'][:min_length]

Crear DataFrame y guardar en CSV df = pd.DataFrame(data) df.to_csv('accelerations.csv', index=False) print("Datos exportados a 'accelerations.csv'") Programa de autoría propia. (Salazar. D, 2024)

ANEXO C: CÓDIGO EN MATLAB PARA EL PROCESADO DE SEÑALES DE ACELERACION, VELOCIDAD Y DESPLAZAMIENTO

%% Integrador para obtener velocidad y desplazamiento

% Datos

lw = 0.1; fsa = 6; fsl = 6; fst = 14;

ARCH = 'Prueba_4.csv'; % Nombre del archivo .CSV que se desea leer

dt = 0.01; % Intervalo de muestreo (s) para RSShake 4D V9

order = 2; % Orden del filtro (recomendado entre 1 y 6)

fr = 0.5; % Frecuencia de corte alta para el filtro (Hz)

%% Importar los datos recopilados con el Raspberry Shake

locFolder1 =

'C:\Users\david\OneDrive\Escritorio\David\USFQ\Ultimo_Semestre\Tesis\Matlab';

cd(locFolder1);

M = readtable(ARCH, 'NumHeaderLines', 1);

TSX = table2array(M(:,1)); % Tiempo extraído del archivo CSV

ASENN = table2array(M(:,4)) / 9.81; % Aceleración ENN en g

ASENE = table2array(M(:,3)) / 9.81; % Aceleración ENE en g

ASENZ = table2array(M(:,5)) / 9.81; % Aceleración ENZ en g

%% Filtrado de las señales de aceleración

fs = 1 / dt; % Frecuencia de muestreo (Hz)

[b, a] = butter(order, fr / (fs / 2), 'high'); % Filtro pasa-altas

% Filtrar las señales

ASENN_filtered = filtfilt(b, a, ASENN);

ASENE_filtered = filtfilt(b, a, ASENE);

ASENZ_filtered = filtfilt(b, a, ASENZ);

%% Integración de la aceleración para obtener la velocidad en cm/s

VELENN = cumtrapz(TSX, ASENN_filtered * 9.81 * 100); % Velocidad ENN VELENE = cumtrapz(TSX, ASENE_filtered * 9.81 * 100); % Velocidad ENE VELENZ = cumtrapz(TSX, ASENZ_filtered * 9.81 * 100); % Velocidad ENZ

% Corregir tendencia y valores iniciales de velocidad

VELENN = detrend(VELENN) - VELENN(1);

VELENE = detrend(VELENE) - VELENE(1);

VELENZ = detrend(VELENZ) - VELENZ(1);

%% Integración de la velocidad para obtener el desplazamiento en mm DESENN = cumtrapz(TSX, VELENN * 10); % Desplazamiento ENN DESENE = cumtrapz(TSX, VELENE * 10); % Desplazamiento ENE DESENZ = cumtrapz(TSX, VELENZ * 10); % Desplazamiento ENZ

% Corregir tendencia y valores iniciales de desplazamiento

DESENN = DESENN - mean(DESENN);

DESENE = DESENE - mean(DESENE);

DESENZ = DESENZ - mean(DESENZ);

%% Gráficos
figure;
% Aceleración
subplot(3,3,1); plot(TSX, ASENN_filtered, 'k', 'LineWidth', lw);
ylabel('Aceleración (g)'); xlabel('Tiempo (s)');
title('Aceleración ENN'); grid on;

subplot(3,3,2); plot(TSX, ASENE_filtered, 'k', 'LineWidth', lw); ylabel('Aceleración (g)'); xlabel('Tiempo (s)'); title('Aceleración ENE'); grid on;

subplot(3,3,3); plot(TSX, ASENZ_filtered, 'k', 'LineWidth', lw); ylabel('Aceleración (g)'); xlabel('Tiempo (s)'); title('Aceleración ENZ'); grid on;

% Velocidad

subplot(3,3,4); plot(TSX, VELENN, 'r', 'LineWidth', lw); ylabel('Velocidad (cm/s)'); xlabel('Tiempo (s)');

title('Velocidad ENN'); grid on;

subplot(3,3,5); plot(TSX, VELENE, 'r', 'LineWidth', lw); ylabel('Velocidad (cm/s)'); xlabel('Tiempo (s)'); title('Velocidad ENE'); grid on;

subplot(3,3,6); plot(TSX, VELENZ, 'r', 'LineWidth', lw);

ylabel('Velocidad (cm/s)'); xlabel('Tiempo (s)'); title('Velocidad ENZ'); grid on;

% Desplazamiento subplot(3,3,7); plot(TSX, DESENN, 'm', 'LineWidth', lw); ylabel('Desplazamiento (mm)'); xlabel('Tiempo (s)'); title('Desplazamiento ENN'); grid on;

subplot(3,3,8); plot(TSX, DESENE, 'm', 'LineWidth', lw); ylabel('Desplazamiento (mm)'); xlabel('Tiempo (s)'); title('Desplazamiento ENE'); grid on;

subplot(3,3,9); plot(TSX, DESENZ, 'm', 'LineWidth', lw); ylabel('Desplazamiento (mm)'); xlabel('Tiempo (s)'); title('Desplazamiento ENZ'); grid on;

sgtitle('Aceleración, Velocidad y Desplazamiento');

Programa de autoría propia. (Salazar. D, 2024)