

Instituto Tecnológico de Costa Rica

Carrera de Ingeniería Mecatrónica



**Desarrollo de un simulador de un acelerómetro de tres (3) ejes
microelectromecánicos (MEMS)**

**Development of a software-controlled 3-Axis MEMS
Accelerometer**

Informe de Proyecto de Graduación para optar por el título de
Ingeniera en Mecatrónica con el grado académico de Licenciatura

Carla Corina Gómez Carrasquilla

Cartago, 15 de junio 2022



This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-nc-sa/4.0/).

Declaro que el presente Proyecto de Graduación ha sido realizado enteramente por mi persona, utilizando y aplicando literatura referente al tema e introduciendo conocimientos propios. En los casos en que he utilizado bibliografía, he procedido a indicar las fuentes mediante las respectivas citas bibliográficas. En consecuencia, asumo la responsabilidad total por el trabajo de graduación realizado y por el contenido del correspondiente informe final.



Carla Corina Gómez Carrasquilla

Cartago, 15 de junio 2022

Céd: 3-0519-0495

INSTITUTO TECNOLÓGICO DE COSTA RICA
PROGRAMA DE LICENCIATURA EN INGENIERÍA MECATRÓNICA
PROYECTO FINAL DE GRADUACIÓN
ACTA DE APROBACIÓN

El profesor asesor del presente trabajo final de graduación, indica que el documento presentado por el estudiante cumple con las normas establecidas por el programa de Licenciatura en Ingeniería Mecatrónica del Instituto Tecnológico de Costa Rica para ser defendido ante el jurado evaluador, como requisito final para aprobar el curso Proyecto Final de Graduación y optar así por el título de Ingeniero(a) en Mecatrónica, con el grado académico de Licenciatura.

Estudiante: Carla Corina Gómez Carrasquilla

Proyecto: Desarrollo de un simulador de un acelerómetro de tres (3) ejes microelectromecánicos (MEMS).

A handwritten signature in black ink, appearing to read 'M Vilchez', is written over a solid horizontal line.

MSc. -Fís. Marta Vilchez Monge

Asesora

Cartago, 15 de junio 2022

INSTITUTO TECNOLÓGICO DE COSTA RICA
PROGRAMA DE LICENCIATURA EN INGENIERÍA MECATRÓNICA
PROYECTO FINAL DE GRADUACIÓN
ACTA DE APROBACIÓN

Proyecto final de graduación defendido ante el presente jurado evaluador como requisito para optar por el título de Ingeniero(a) en Mecatrónica con el grado académico de Licenciatura, según lo establecido por el programa de Licenciatura en Ingeniería Mecatrónica, del Instituto Tecnológico de Costa Rica.

Estudiante: Carla Corina Gómez Carrasquilla

Proyecto: Desarrollo de un simulador de un acelerómetro de tres (3) ejes microelectromecánicos (MEMS).

Miembros del jurado evaluador

 EDUARDO JOSE ASENJO
CASTILLO (FIRMA)
2022.06.15 15:40:46
-06'00'

Ing. Eduardo José Asenjo Castillo

Jurado

 | Tecnológico de Costa Rica ANA GABRIELA ORTIZ
LEON (FIRMA)
2022.06.15 17:41:29
-06'00'

Dra. -Ing. Gabriela Ortiz León

Jurado

JAIME ALONSO
MORA MELENDEZ
(FIRMA)  Digitally signed by JAIME
ALONSO MORA
MELENDEZ (FIRMA)
Date: 2022.06.15 15:24:29
-06'00'

MSc. -Ing. Jaime Mora Meléndez

Jurado

Los miembros de este jurado dan fe de que el presente proyecto final de graduación ha sido aprobado y cumple con las normas establecidas por el programa de Licenciatura en Ingeniería Mecatrónica.

Cartago, 15 de junio 2022

Resumen

El trabajo que se presenta consiste en la caracterización de un acelerómetro MEMS MPU6050 y la implementación de un simulador de acelerómetro MEMS en la FPGA de una placa de evaluación integrada denominada myRIO, con una secuencia de prueba y verificación de modos. A partir de la búsqueda de información se establece un método para caracterizar el sensor mediante la fuerza centrífuga. Al sensor se le extraen datos de aceleración para obtener su comportamiento. Por medio de la programación en Python de una Raspberry Pi se almacenan los datos de aceleración, revoluciones por minuto y temperatura del sensor en una memoria USB. Con estos datos se obtiene la respuesta del sensor.

Se implementó una API para comunicar al usuario con el simulador de un acelerómetro MEMS de 3 ejes, el cual puede trabajar en un rango de aceleraciones entre 0g y 200g. El sistema simulado corre en una FPGA y trabaja con rangos de temperatura de -40 a 85 °C. Se desarrolló una comunicación mediante el protocolo SPI, el cual envía y recibe datos de la FPGA donde se encuentra el sistema simulado.

El simulador en la FPGA posee tres modos de falla que se asocian a fallas reales de los dispositivos MEMS. Se logra obtener la respuesta del sensor en modo de trabajo, ante una aplicación de un sistema de aceleración centrífuga. La respuesta ante el cambio de temperatura del sistema simulador no sobrepasa los 0.001V con lo que se logra obtener una respuesta estable para cambios de temperatura. Para el error por comunicación se obtiene una respuesta fija de 0.5mv, la respuesta del error mecánico en la masa de prueba es de 0.7mV, el error por vibración da una respuesta de 1V.

Del sistema de caracterización se obtienen valores de aceleración desde 0.010g hasta 16g en el MPU6050. El cambio de aceleración en el eje X tiene un promedio de 15.596g, en el eje Y de 15.277g y el eje Z de 15.241g. En 0 RPM se tiene para el eje X un cambio de 0.002g, para el eje Y una disminución de 0.017g y para el eje Z una disminución de 0.951g.

Finalmente, se realiza un análisis de costos de la implementación y se adjuntan los manuales realizados para los sistemas implementados.

Palabras clave: aceleración, caracterización, FPGA, modo falla, simulación

Abstract

The given work describes the characterization of a MEMS accelerometer model MPU6050, and the implementation of an accelerometer simulator on the myRIO FPGA. First of all, a research is done in order to establish methods and parameters necessary for the characterization in a centrifuge application. The MPU6050 sensor is accelerated in order to get data thru the Raspberry Pi. The data from the sensor was saved in a USB device in order to access it faster.

An API interface was implemented in order to configure the simulation parameters, the simulation works in a range from 0g to 200g, and in a temperature range from -40 to 85°C. The communication protocol implemented is SPI, which can send and receive data from the FPGA.

The simulator has a response similar to a centrifuge application. The 3 failure modes are related to real failures associated with MEMS devices. In working mode, the simulation generates a valid response, and when a temperature change has been applied the deviation does not surpass 0.001V. For the falling mode in the simulation, the communication error has a fixed value of 0.5mV, the proof mass failure has a 0.7mV response and the vibration error has a 1V response.

The MPU6050 sensor acceleration values goes from 0.010g to 16g. The mean value for the X axis is 15.596g, for the Y axis is 15.277 and for the Z axis is 15.241g. At 0 RPM the X-axis has a deviation of 0.002g, the Y-axis of 0.017g, and the Z-axis a 0.0951g deviation.

Finally, a cost analysis is implemented and the necessary documents and manuals are attached to this document.

Keywords: acceleration, characterization, FPGA, Falling mode, simulation,

a mi abuelo Carlos

Agradecimientos

El resultado de este trabajo no hubiese sido posible sin el apoyo de mi mamá, mi papá, mi tía Aracelly, mi hermano y mi abuela.

Se le agradece a la Profesora Marta Vílchez y a mi mentora Mariseth Granados.

Carla Corina Gómez Carrasquilla

Cartago, 17 de junio de 2022

Índice general

Índice de figuras	v
Índice de tablas	ix
Lista de símbolos y abreviaciones	xi
1 Introducción	1
1.1 Contexto y antecedentes del proyecto	1
1.2 Problema a resolver	1
1.3 Síntesis del problema	2
1.4 Objetivos y estructura del documento	2
1.4.1 Objetivo General	3
1.4.2 Objetivos Específicos	3
2 Marco teórico	5
2.1 Dispositivos MEMS	5
2.1.1 Miniaturización de dispositivos	5
2.1.2 Fabricación Paralela	6
2.1.3 Dominio de la energía y sensores	6
2.2 La aceleración	7
2.2.1 Los acelerómetros	8
2.2.2 Limitaciones de Acelerómetros	9
2.2.3 Acelerómetro MEMS	9
2.3 Acelerómetro con masa suspendida	10
2.3.1 Modelo de acelerómetro con masa suspendida	12
2.4 Software y Hardware	16
2.4.1 FPGA	16
2.4.2 <i>Software</i> de programación	16
2.4.3 myRIO	17
2.4.4 Azure DevOps	18
2.4.5 API	18
2.4.6 TestStand	19
2.4.7 Protocolos de comunicación	19
2.4.8 Fricción y vibraciones	21

2.5	Problemática de la fabricación de chips	22
2.6	Repetibilidad	23
3	Metodología	25
3.1	Laboratorios de la empresa	25
3.2	Selección del acelerómetro MEMS	25
3.3	La crisis de contenedores	28
3.3.1	Acelerómetro comercial disponible en Costa Rica	28
3.4	Caracterización del Acelerómetro MEMS	29
3.4.1	Sistema de caracterización	29
3.4.2	Planteamiento	29
3.4.3	Entrevistas y Procesado de Información	30
3.4.4	Jerarquización de necesidades	30
3.4.5	Establecimiento de Métricas	31
3.4.6	Determinar Valores Objetivos	32
3.4.7	Descomposición funcional del problema	36
3.4.8	Proceso de búsqueda de Información	36
3.4.9	Construcción de conceptos	39
3.4.10	Establecimiento y selección del candidato a prototipo	40
3.4.11	Factores de influencia	40
3.4.12	Definición de las pruebas a realizar	41
4	Implementación del sistema de caracterización y del simulador	45
4.1	Sistema de caracterización	45
4.1.1	Calibración de acelerómetros	45
4.1.2	Sistema de aceleración centrífuga	46
4.1.3	Verificación de velocidades con Tacómetro	48
4.1.4	Sistema de toma de datos con Raspberry Pi	49
4.1.5	Guardas de seguridad y cámara de aislamiento térmico	52
4.2	Implementación del simulador en la FPGA	52
4.2.1	Interfaz Simulador-Usuario	57
4.2.2	Modo de Trabajo y selección de Modos de Falla	61
4.2.3	Salidas Digitales y Analógicas	64
4.3	Diseño del experimento y toma de datos	66
4.3.1	Reconocimiento del problema	66
4.3.2	Selección de factores, niveles y variable de respuesta	66
4.3.3	Selección del diseño de experimento	67
5	Resultados y análisis	69
5.1	Sistema de toma de datos	69
5.1.1	Caracterización del acelerómetro	69
5.1.2	Respuesta ante cambio de temperatura	74
5.1.3	Respuestas obtenidas	75
5.2	Simulación	77

5.2.1	Respuesta sistema simulado en la FPGA	79
5.2.2	Sistema real y simulado	83
5.3	Verificación modos en TestStand a través de API	84
5.4	Problemas encontrados e imprevistos	85
5.5	Cumplimiento de las métricas	87
5.6	Síntesis del análisis	88
6	Análisis Económico	91
6.1	Costo Material	91
6.1.1	Costo Implementación simulador	91
6.1.2	Costo Implementación sistema caracterización	92
6.1.3	Costo total del material	92
6.2	Costo del recurso Humano	93
7	Conclusiones, Recomendaciones y Trabajos Futuros	95
7.1	Conclusiones	95
7.2	Recomendaciones y Trabajos Futuros	96
7.3	Valoración personal	97
	Bibliografía	99
A	Apéndice	103
A.1	Tabla completa con los enunciados del cliente	104
A.2	Encuesta al cliente	105
A.3	Lista de métricas	108
A.4	Planos ensamble sistema de caracterización	109
A.5	Planos proceso de manufactura para acople	115
A.6	Código de Arduino	127
A.7	Código de Python para la Raspberry Pi	127
A.8	Gráficas de las otras mediciones	134
A.9	Gráfica de todas las mediciones	136
A.10	Gráfica de todas las mediciones	138
A.11	Manual de usuario para el simulador	140
A.12	Manual de usuario para el sistema caracterización	154
	Anexos	161
A.13	Hoja de Datos myRio	161
A.14	Hoja de Datos MPU-6050	165
	Índice alfabético	170

Índice de figuras

2.1	Diagrama de la configuración de un acelerómetro de masa suspendida [10] .	8
2.2	Diagrama de la configuración de un microacelerómetro de masa suspendida [10]	10
2.3	Diagrama de la configuración de un micro acelerómetro inercial [10]	11
2.4	Acelerómetro de prueba [26]	11
2.5	Acelerómetro con masa suspendida [45]	13
2.6	Acelerómetro con masa suspendida con transductores de capacitancia [48] .	14
2.7	Diagrama de la estructura interna de una FPGA [36]	17
2.8	Diagrama de conexión protocolo I2C [8]	19
2.9	Diagrama de conexión protocolo SPI [15]	21
2.10	Diagrama de conexión protocolo UART [39]	21
3.1	Diagrama de bloques para desarrollar solución	26
3.2	Sensores disponibles comercialmente en Costa Rica	27
3.3	Diagrama de flujo de posibles planes a tomar	27
3.4	Resumen relación métrica necesidad.	35
3.5	Determinación de los valores objetivos.	35
3.6	Simulador de acelerómetro Hansford HS-661 [41]	36
3.7	Simulador de acelerómetro Endevco 4830B [11]	37
3.8	Centrífuga de 20g con 8.84m de radio [3]	38
3.9	Ejemplo de rieles de baja fricción [8]	38
3.10	Primer ronda de selección de prototipos de solución	40
3.11	Selección de prototipo de solución a desarrollar	42
3.12	Factores de influencia y descripción de pruebas a realizar	43
4.1	Diseño propuesto para el sistema de centrifugación.	46
4.2	Diagrama de conexión entre el Arduino Uno, el TB6600 y el motor NEMA 17 [7]	47
4.3	Primer modelo del sistema del circuito de la centrifugadora	48
4.4	Circuito inicial para toma de datos en Raspberry Pi	50
4.5	Diagrama de conexión del sistema de la toma de datos en la Raspberry Pi	51
4.6	Diagrama de conexión del MPU6050 [8]	51
4.7	Placa soldada y conectada a la Raspberry Pi	52
4.8	Primer modelo sistema de toma de datos sobre el sistema de aceleración . .	53

4.9	Primer modelo sistema de caracterización	53
4.10	Primer modelo del sistema de guardas de protección	54
4.11	Primer modelo de la cámara de aislamiento térmico	54
4.12	Hoja de datos ADXL377 [6]	55
4.13	Relación entre tensión TTL del sensor y la aceleración	56
4.14	Código de la simulación en la FPGA	57
4.15	Componentes de un PXI [18].	58
4.16	Conexión PXI y tarjeta CB-2162 [13].	59
4.17	Código de lectura y escritura en el protocolo SPI	60
4.18	Mapa de pines utilizado para la comunicación SPI en el DPE	60
4.19	Configuración para la comunicación SPI en el DPE	61
4.20	Diagrama de flujo para la selección de los modos	64
4.21	Diagrama de flujo para la selección de los ejes que fallan	65
4.22	Diagrama conexión del conversor de nivel HW-221.	65
4.23	Sección del código de PWM implementado	66
5.1	Primer medición para el eje X en el sistema de centrifugación.	69
5.2	Orientación de los ejes en el MPU6050.	70
5.3	Datos en obtenidos del sensor en Excel.	70
5.4	Segunda y tercer medición para el eje X en el sistema de centrifugación . .	71
5.5	Resultados obtenidos de las mediciones realizadas para el eje X.	72
5.6	Primer medición para el eje Y en el sistema de centrifugación.	73
5.7	Primer medición para el eje Z en el sistema de centrifugación.	74
5.8	Offset en el MPU6050 debido al cambio en al temperatura [22]	75
5.9	Respuesta del eje X, Y y Z ante un aumento de temperatura.	76
5.10	Gráfica con la ecuación de la aceleración en función de las RPM para medición 2 del eje X.	76
5.11	Gráfica con la respuesta de las 16 mediciones para el eje X.	77
5.12	Gráfica con la ecuación de la aceleración en función de las RPM para medición 2 del eje Y Z.	77
5.13	Interfase de usuario del simulador a través de API.	78
5.14	Respuesta del simulador para una aceleración a 20°C.	79
5.15	Respuesta del simulador en un rango de 0g a 16g ante un cambio de tem- peratura.	80
5.16	Respuesta del simulador ante una aceleración en la centrifugadora.	81
5.18	Salida PWM del simulador para una aceleración a 20°C.	82
5.17	Respuesta del simulador ante cambios de temperatura.	82
5.19	Respuesta del simulador ante una aceleración en la centrifugadora.	83
5.20	Respuesta del simulador ante una aceleración en la centrifugadora.	84
5.21	Secuencia de prueba en TestStand.	84
5.22	Ejemplo de errores obtenidos en el reporte	85
5.23	Porcentaje de LUT utilizadas en la FPGA para el simulador.	86
5.24	Primeros datos obtenidos para el eje X en el sistema de centrifugación. . .	87

5.25	Lista con los resultados de las pruebas realizadas.	88
6.1	Costos de la implementación del sistema simulado	91
6.2	Costos de los materiales y componentes previamente disponibles	92
6.3	Costos de la implementación del simulado	92
6.4	Costo de los viáticos	93
A.1	Resultados obtenidos de las mediciones realizadas para el eje Y.	134
A.2	Resultados obtenidos de las mediciones realizadas para el eje Z.	135
A.3	Gráfica con la respuesta de las 16 mediciones para el eje Y.	136
A.4	Gráfica con la respuesta de las 16 mediciones para el eje Z.	136
A.5	Mediciones del simulador.	138
A.6	Mediciones del simulador (Continuación).	139

Índice de tablas

3.1	Declaración de la misión.	30
3.2	Relación necesidad-métrica.	32
3.3	Continuación tabla 3.2 Relación necesidad-métrica.	33
3.4	Continuación tabla 3.3 Relación necesidad-métrica.	34
3.5	Descripción de los criterios de selección.	41
4.1	Velocidades del sistema de caracterización.	49
4.2	Direcciones para la comunicación SPI.	62

Lista de símbolos y abreviaciones

Abreviaciones

I^2C	Inter-Integrated Circuit
ADC	Analog-to-Digital Converter
API	Application Programming Interfaces
CS	Chip Select
DPE	Digital Pattern Editor
DUT	Device Under Test
ESD	Descarga Electroestática
FPGA	Field Programmable Gate Array
LED	Light Emiting Diode
LUT	LookUp Table
MEMS	Microelectromechanical System
MISO	Master Input Slave Output
MOSI	Master Output Slave Input
PS	Professional Services
PXI	PCI Extensions for Instrumentation
SCL	Serial Clock
SDA	Serial Data
SPI	Serial Peripheral Interface
TTL	Transistor-Transistor Logic
UART	Universal Asynchronous Receiver-Transmitter
USB	Universal Serial Bus
VHDL	Very High Speed Description Language

Notación general []

a_n	Aceleración masa sísmica
a_t	Aceleración tangencial
$a(t)$	Aceleración
c	Constante de amortiguamiento
A	Área del electrodo
k_b	Constante de Boltzmann ($k_b=1.380 \cdot 10^{-23}$ [J/K])
Q	Factor de calidad
C	Capacitancia
i_{mot}	Corriente debido al movimiento del acelerómetro

i	Corriente
d	Distancia entre placas del capacitor
ξ	Factor de amortiguamiento
f_0	Frecuencia resonante o natural del sistema
F_e	Fuerza externa
m	Masa
$\dot{x}$	Velocidad
$\ddot{x}$	Aceleración
x	Movimiento de placas desde posición original
ϵ	Permitividad
x_f	Posición del marco de referencia
x_m	Posición de la masa suspendida
P	Presión
r	Radio desde el centro de la mesa giratoria hasta el centro de la masa sísmica
w_0	Frecuencia de resonancia del sistema
K	Constante de restauración del resorte
T	Temperatura [K]
V	Tensión
t_f	Tiempo final
t_i	Tiempo inicial
ν_{xf}	Velocidad final en eje X
ν_{xi}	Velocidad inicial en eje X
f	Velocidad de rotación de la mesa en revoluciones por segundo (rev/s)

Capítulo 1

Introducción

1.1 Contexto y antecedentes del proyecto

La empresa donde se desarrolla el proyecto es de origen estadounidense y durante más de 40 años ha desarrollado sistemas de prueba y medición automatizados, con el fin de ayudar a los ingenieros a resolver problemas. Sus oficinas centrales están en Austin, Texas, Estados Unidos y en Costa Rica están ubicados en Ultrapark II, Lagunilla, Heredia. Sus oficinas llegaron al país en el 2011 y se han mantenido activos en la industria nacional e internacional hasta la fecha. Los sistemas de la empresa integran diversas herramientas conectadas a software, las cuales permiten manejar flujos de trabajo y el envío de datos con el objetivo de realizar una decisión informada de los procesos. Sus soluciones se aplican en campos como los semiconductores, el transporte, los electrónicos, la energía, defensa, el entorno aeroespacial, la academia e investigación. El departamento de Servicios Profesionales (Professional Services - PS) se encarga de desarrollar aplicaciones a la medida para industrias de inspección, automatización, manejo de materiales y prueba de sistemas para clientes globales [16].

1.2 Problema a resolver

La división de PS, busca desarrollar un simulador genérico para Dispositivos Bajo Prueba (Device Under Test - DUT) de un acelerómetro de tres (3) ejes MEMS (Micro-electromechanical Systems). El proyecto procede de la necesidad de la empresa por tener una herramienta que les permita comprobar los conceptos durante el desarrollo de aplicaciones con hardware, ya que el equipo de PS debe ser capaz de realizar pruebas durante fases tempranas de diseño y desarrollo, actualmente el departamento no posee un dispositivo que les permita realizar estas verificaciones ni que apoye el desarrollar pruebas con datos reales durante el prototipado, entrenamiento, prueba de concepto, demostraciones, validaciones y prueba de la aplicación con datos reales.

El motivo por el cual se busca realizar el proyecto recae en la necesidad que poseen los empleados de la empresa de tener un dispositivo capaz de facilitar el proceso de validación y pruebas, cuando no se cuente con el hardware real de la aplicación, por lo que tener un simulador de hardware genérico les permitirá correr las aplicaciones y realizar las correcciones necesarias mientras se trabaja de manera remota o lejos del sitio de implementación. Esta herramienta pueden mejorar los tiempos de respuesta y reducir atrasos durante la implementación de una solución.

Se debe recalcar que todos los inconvenientes relacionados al acelerómetro real se pueden contrarrestar al implementar un acelerómetro simulado, lo que permitirá al equipo de trabajo corroborar con datos reales los prototipos durante el desarrollo de aplicaciones. Además, aunque se posea un acelerómetro real pueden existir problemas relacionados con la toma de datos, por ejemplo: una mala conexión o incompatibilidad con el hardware. Aunque se pueda realizar la conexión del dispositivo, existe la probabilidad de tener problemas durante la adquisición de datos, fallas en el funcionamiento del acelerómetro o errores, dada la inexperiencia de los colaboradores.

Durante el desarrollo de una aplicación, si no se tiene disponible un acelerómetro real de 3 ejes no se pueden realizar pruebas de software que dependan de este hardware. Teniendo como punto de partida estos elementos, se busca mejorar el proceso de pruebas de sistemas DUT y para esto se propone desarrollar el simulador que pueda ser utilizado cuando no se cuente con un acelerómetro real de 3 ejes. Se busca que esta herramienta apoye los servicios brindados por el equipo de PS y mejore los procesos de prueba de DUT.

Dada la naturaleza del trabajo, los equipos a desarrollar deben tener semejanza con los instrumentos de la empresa, se requiere una simulación que refleje el comportamiento del acelerómetro al ser caracterizado con estos instrumentos, por lo que estos deben ser similares a los que son utilizados para desarrollar aplicaciones.

1.3 Síntesis del problema

La empresa no posee un sistema simulado de un acelerómetro MEMS de 3 ejes que muestre un valor confiable de temperatura y el estado del chip ante una solicitud de operación y que pueda ser aplicado durante el proceso de pruebas de las aplicaciones desarrolladas por la empresa.

1.4 Objetivos y estructura del documento

Con el fin de apoyar al departamento de PS se planteo el objetivo general y específico del proyecto, así como la estructura general del documento.

1.4.1 Objetivo General

Desarrollar un software que simule el comportamiento de un acelerómetro MEMS de tres (3) ejes que muestre como resultado un valor confiable de temperatura y el estado del chip ante una solicitud del usuario para la empresa.

1.4.2 Objetivos Específicos

- Desarrollar el proceso de toma de datos y variables de influencia para un acelerómetro real de 3 ejes en al menos 3 rangos establecidos.
- Diseñar un sistema simulado de un acelerómetro MEMS de 3 ejes en una FPGA.
- Validar el simulador para modo de trabajo y modo de falla, mediante API (Application Programming Interface).

El principal aporte del presente informe recae en la implementación de variables y características físicas que caracterizan a un sistema en una plataforma digital que pueda ser accesible y reconfigurable para incluir mejoras en el proceso de pruebas de conceptos, prueba de prototipos y el entrenamiento de usuarios en el campo de las ingeniería.

El documento se dividió en capítulos, el capítulo 1, da una introducción al proyecto. En el capítulo 2 se detalla el marco teórico y se describen las áreas de conocimiento pertinentes para dar solución al problema planteado. Posteriormente, en el capítulo 3 se detalla la metodología utilizada. La propuesta planteada del simulador y del sistema para el proceso de toma de datos se encuentran en el capítulo 4. Los resultados obtenidos tanto del sistema simulado como de la implementación de la caracterización del acelerómetro se describen en el capítulo 5. En el capítulo 6 se describen los costos finales de la implementación mediante un análisis económico. Finalmente se exponen conclusiones y recomendaciones acordes con la implementación del sistema en el capítulo 7.

Capítulo 2

Marco teórico

2.1 Dispositivos MEMS

La tecnología de microfabricación es la base que permite la integración y miniaturización de sensores funcionales. Entre las partes esenciales de los micro sensores está el descubrimiento, el estudio y optimización de la piezoresistividad de los monocristales y los policristales de silicio. Durante los inicios de la investigación sobre la microfabricación se dio un enfoque en dos formas de silicio disponibles: los monocristales, conocidos como sustrato a granel de silicio y los policristales o silicios de película fina. Ambos fueron muy accesibles ya que eran utilizados en la industria de los circuitos integrados. Los monocristales se implementaban como sustrato en circuitos, mientras que los policristales se usan en puertas de transistores [26].

El nombre MEMS (por sus siglas en inglés) o Sistemas Micro Electro Mecánicos, captura la escala (micros), la práctica (electromecánicos) y la aspiración que tiene a ser un sistema. Los dispositivos MEMS son componentes que pertenecen a un sistema más grande, por ejemplo los acelerómetros MEMS forman parte del sistema de detección de choques en un vehículo. Esta tecnología ofrece ventajas en comparación con los sensores electromecánicos macros, estas se relacionan a la alta sensibilidad y el bajo ruido que poseen. Además, este tipo de tecnología disminuye los costos de cada sensor, al eliminar el ensamblaje manual e implementar un proceso de fabricación en paralelo de lotes [26].

La tecnología MEMS también está siendo aplicada en sistemas de microfluidos y la integración de procesadores bioquímicos, con el fin de implementarlos en aplicaciones de diagnóstico médico y monitorización de distribuciones ambientales [26].

2.1.1 Miniaturización de dispositivos

Las dimensiones pequeñas de los dispositivos MEMS dan lugar a ventajas operacionales como: suspensiones suaves, alta frecuencia de resonancia, alta sensibilidad y una baja masa térmica. Por ejemplo, la transferencia de calor hacia y desde una micromáquina es

generalmente rápida. Hay algunos fenómenos que no se escalan favorablemente al reducir el tamaño, para conocer el comportamiento de elementos físicos a diferentes escalas se puede tomar como referencia la Ley del Escalamiento. En muchos casos esta ley se debe evaluar de manera paralela para determinar la figura de mérito variable que se ve más afectada ante la severidad del escalamiento. Por ejemplo, en un acelerómetro analógico, la miniaturización permite obtener vigas soportadas suavemente (lo cual es deseado), también se aumenta la frecuencia de resonancia y el ancho de banda, pero a costas de reducir el valor de capacitancia (no deseado) y generalmente aumentar la complejidad del circuito [26].

En un acelerómetro representativo, el muelle está hecho de una viga de silicio con un piezoresistor dopado. Las dimensiones del piezoresistor, el nivel de dopado, la sensibilidad y el ruido, están relacionados. Para aumentar el efecto piezoresistivo, es deseable dopar al silicio en bajas concentraciones, pero esto tiende a aumentar la sensibilidad a variaciones en la temperatura ambiente, esto también aumentaría el valor de la resistencia para una dimensión dada, lo cual podría aumentar los niveles de ruido [26].

2.1.2 Fabricación Paralela

Los dispositivos MEMS pueden realizar lecturas en dos o tres ejes simultáneamente con precisión aunque sean dispositivos de tamaño pequeño, lo cual no se puede reproducir de manera eficiente con las herramientas de maquinación tradicionales. Al combinar la tecnología MEMS con la fotolitografía, se pueden implementar aplicaciones únicas de tridimensionalidad [26].

2.1.3 Dominio de la energía y sensores

Los sensores son elementos que permiten detectar fenómenos físicos o químicos, este proceso se puede delimitar como una transducción de la energía, y se considera como un modo de percepción. Por otro lado, los actuadores producen movimiento mecánico, fuerza o torque, en este caso la transducción de la energía se da como una acción. Existen 6 principales dominios de energía: (1) dominio eléctrico, (2) dominio mecánico, (3) dominio químico, (4) dominio radioactivo, (5) dominio magnético y (6) dominio térmico. La energía total de un sistema se distribuye en varios de estos dominios y coexisten mutuamente. Dentro de las principales características que se deben tener en cuenta para el diseño de sensores se tienen [26]:

- Sensibilidad: relación entre la magnitud de la señal de salida y el estímulo en la entrada. Se puede ver afectado por la amplificación de la señal.
- Linealidad: cuando la señal de salida cambia de manera proporcional con respecto a la señal de entrada.
- Exactitud: habilidad del sensor de dar un valor cercano al valor real.

- Precisión: habilidad del sensor de dar la misma lectura cuando se mide repetidamente el mismo factor.
- Resolución: valor mínimo detectable por el sensor.
- Ruido: corresponde en si a una señal que interfiere con la señal por detectar, se considera como una señal aleatoria que opaca la señal principal. En los circuitos MEMS se tiene 3 causas principales de ruido: ruido eléctrico, ruido mecánico y ruido de los circuitos.
- Rango dinámico: relación existente entre la máxima señal detectable y la mínima.
- Ancho de banda: describe la habilidad del sensor de detectar cambios rápidos en la señal, se conoce como el rango de frecuencia efectiva del sensor.
- Desvío: variaciones en el sistema debido a los cambios en las propiedades eléctricas y mecánicas en el sistema por desgastes en el tiempo.
- Confiabilidad: que el sensor pueda funcionar y no verse afectado por cambios en el tiempo o bajo condiciones de trabajo extremas.
- Interferencia: el sensor diseñado para medir una variable puede verse afectado directamente por otra variable física, por ejemplo, la sensibilidad en uno de los ejes de un acelerómetro puede responder a la aceleración en otro eje ortogonal.
- Tiempo de diseño y costo: se busca que el proceso sea rápido y de bajo costo.

2.2 La aceleración

La aceleración corresponde al cambio de velocidad de una partícula mientras está en movimiento, el cambio de la velocidad de la partícula se da con el tiempo. Esta variación de la velocidad se divide entre el intervalo de tiempo con el fin de obtener la aceleración promedio de la partícula [46].

$$a_{x,prom} = \frac{\Delta_{\nu_x}}{\Delta_t} = \frac{\nu_{xf} - \nu_{xi}}{t_f - t_i} \quad (2.1)$$

Al realizar un análisis unidimensional, se utilizan los signos positivo (+) y negativo (-) para indicar la dirección de la aceleración, las unidades de la aceleración en el Sistema Internacional de Unidades (SI) corresponde a metros por segundo cuadrado (m/s^2). En algunas situaciones la aceleración promedio varía según el intervalo de tiempo, por lo que se define la aceleración instantánea como un límite, en el cual el intervalo de tiempo tiende a cero [46].

$$\lim_{\Delta_t \rightarrow 0} \left(\frac{\Delta_{\nu_x}}{\Delta_t} \right) = \frac{d\nu_x}{dt} \quad (2.2)$$

Con esto se define la aceleración instantánea como la derivada de la velocidad respecto al tiempo. Si la aceleración en el eje x es positiva, se tiene una aceleración en la dirección x positiva, pero si la aceleración en el eje x es negativa, la aceleración tiene como dirección el eje x negativo. Para un movimiento rectilíneo, cuando la velocidad y la aceleración del objeto están en la misma dirección, el objeto aumenta la velocidad pero, cuando la velocidad y la aceleración están en direcciones opuestas, el objeto se frena.

$$F_x \propto a_x \quad (2.3)$$

En [46] se menciona que un objeto en caída libre es cualquier objeto que se mueve libremente sólo bajo la influencia de la gravedad, sin importar su movimiento inicial. La magnitud de la aceleración de caída libre se denotará mediante el símbolo "g". El valor de g cerca de la superficie de la Tierra disminuye conforme aumenta la altitud.

2.2.1 Los acelerómetros

Los acelerómetros están compuestos principalmente por una masa soportada por un resorte, además, posee un amortiguador que también está sujeto al encapsulado [10]. La figura 2.1 se muestra un diagrama de la configuración interna de un acelerómetro de masa suspendida.

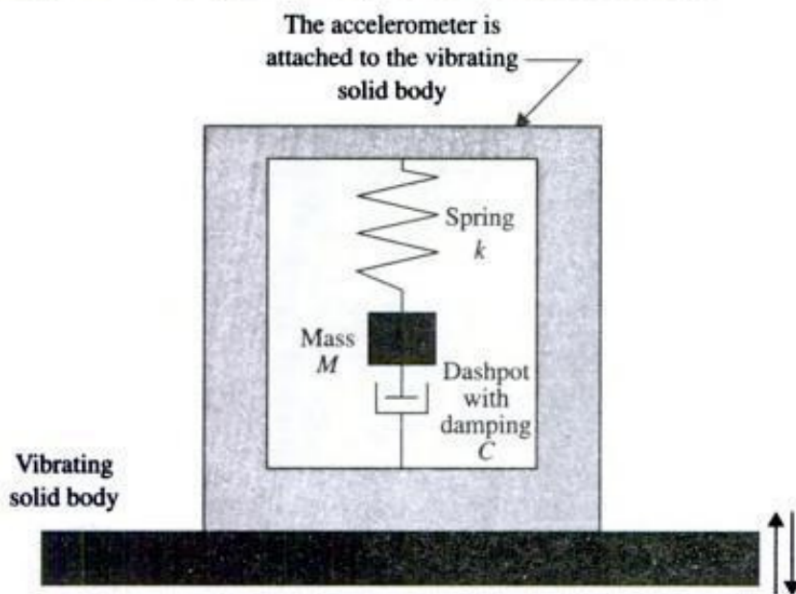


Figura 2.1: Diagrama de la configuración de un acelerómetro de masa suspendida [10]

Hay diversos tipos de tecnología para los acelerómetros, por ejemplo: sensores capacitivos, piezoresistivos, piezoeléctricos, de interferometría óptica y de transferencia térmica. Además, se pueden crear acelerómetros para medir la aceleración rotacional conocidos como giroscopios. Existen las Unidades de Medición de Inercias (IMU), las cuales integran

un paquete de sensores de movimiento como aceleración lineal y velocidad rotacional los cuales poseen suficiente sensibilidad que les permite ser implementados en aplicaciones de navegación [26].

Los acelerómetros se pueden categorizar en vibrómetros, monitores sísmicos, sensores de inclinación y sensores de movimiento según la sensibilidad y la frecuencia característica del dispositivo. Inicialmente, Motorola desarrolló acelerómetros tomando como base los principios de la capacitancia. Un Sensor de medición de g-bajos (low-g sensing) trabaja en rangos menores a los 20g y trabaja con movimientos que un humano puede generar, acelerómetros de mayores rangos de medición son utilizados en la industria y aplicaciones militares. La tecnología de los acelerómetros está dominada por los sensores capacitivos y piezoresistivos [26]. Dado que los piezoresistores deben tener su superficie dopada, solo puede medir movimiento fuera del plano, por lo que tienen poca capacidad al medir la aceleración en múltiples ejes. Los circuitos de los sensores capacitivos están más desarrollados y proveen buena sensibilidad. Al diseñar un sensor se debe considerar las dimensiones, la masa, y la presión de la cámara interna del sensor. La selección de estas características del sensor determinan la dimensión del capacitor, por ejemplo el grosor de la masa interna [26].

2.2.2 Limitaciones de Acelerómetros

Suponiendo que se construye un acelerómetro donde la aceleración de una masa causa su movimiento, es deseable para tener un mayor rango para la respuesta de la frecuencia. Esto normalmente implica una frecuencia de resonancia mayor, lo cual se puede obtener de disminuir la masa o aumentar la constante de fuerza. Al cambiar la constante de fuerza se reduce la sensibilidad del acelerómetro debido a la disminución de la masa y el endurecimiento del muelle, lo que aumenta el ruido debido a un aumento en el rango para la respuesta de la frecuencia. En un acelerómetro representativo, el muelle está hecho de una viga de silicio con un piezoresistor dopado. Las dimensiones del piezoresistor, el nivel de dopado, la sensibilidad y el ruido, están relacionados. Para aumentar el efecto piezoresistivo, es deseable dopar al silicio en bajas concentraciones, pero esto tiende a aumentar la sensibilidad a variaciones en la temperatura ambiente, esto también aumentaría el valor de la resistencia para una dimensión dada, lo cual podría aumentar los niveles de ruido [26].

2.2.3 Acelerómetro MEMS

Los microacelerómetros son utilizados para detectar la dinámica asociada a la fuerzas en sistemas mecánicos en movimiento. Estos tienen una amplia gama de implementaciones en la industria automotriz, por ejemplo en el sistema de suspensión y el sistema de suspensión antibloqueo (ABS), y en los sistemas de las bolsas de aire permite accionar ante una colisión. La notación “g”, como se mencionó anteriormente, representa la aceleración

gravitacional. Los microacelerómetros se clasifican como sensores de inercia, pero contiene un elemento tipo actuador en su interior [10].

En el caso de los microacelerómetros, se debe realizar un ajuste en su configuración física debido al espacio disponible en el interior de la carcasa, este posee una viga de silicio diminuta unida a una masa, conocida como masa sísmica, esta configuración constituyen el sistema de masa-resorte, y el amortiguador se genera mediante el aire encapsulado en la carcasa. La estructura que sostiene la masa funciona como resorte y un piezoresistor se acopla a la viga para poder detectar las deformaciones de la masa, del cual se puede correlacionar la amplitud y la aceleración de la vibración de la masa [10]. La figura 2.2 se muestra un modelo de configuración interna de un microacelerómetro.

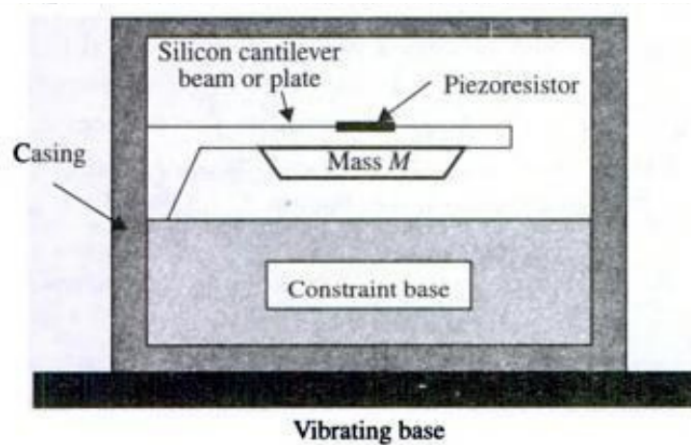


Figura 2.2: Diagrama de la configuración de un microacelerómetro de masa suspendida [10]

Existe otro modelo para la configuración interna de un acelerómetro el cual posee una viga delgada atada en sus dos extremos a resortes, los cuales están formados de un material elástico y están fijos en sus extremos. La viga delgada funciona como una masa sísmica de prueba y tiene acoplado un electrodo, el cual es colocado en medio de dos electrodos fijos. En el caso de una aceleración, la masa de pruebas se desplaza en dirección opuesta a la aceleración, el movimiento de la masa sísmica debido a la aceleración se correlaciona con el cambio en la capacitancia entre el par de electrodos fijos, ver figura 2.3. La masa de pruebas se mueve en dirección opuesta a la aceleración o desaceleración de la unidad, además se pueden realizar mediciones de la aceleración en dos dimensiones a la vez si se duplica el sistema y cambiar la masa de pruebas por una placa cuadrada [10].

2.3 Acelerómetro con masa suspendida

Se debe considerar que los microsistemas son sistemas tridimensionales que involucran la transferencia de calor y la interacción de sólidos y fluidos, por lo que existe un límite para los modelos microscópicos. Algunos de los efectos que se tienen a nivel atómico son los

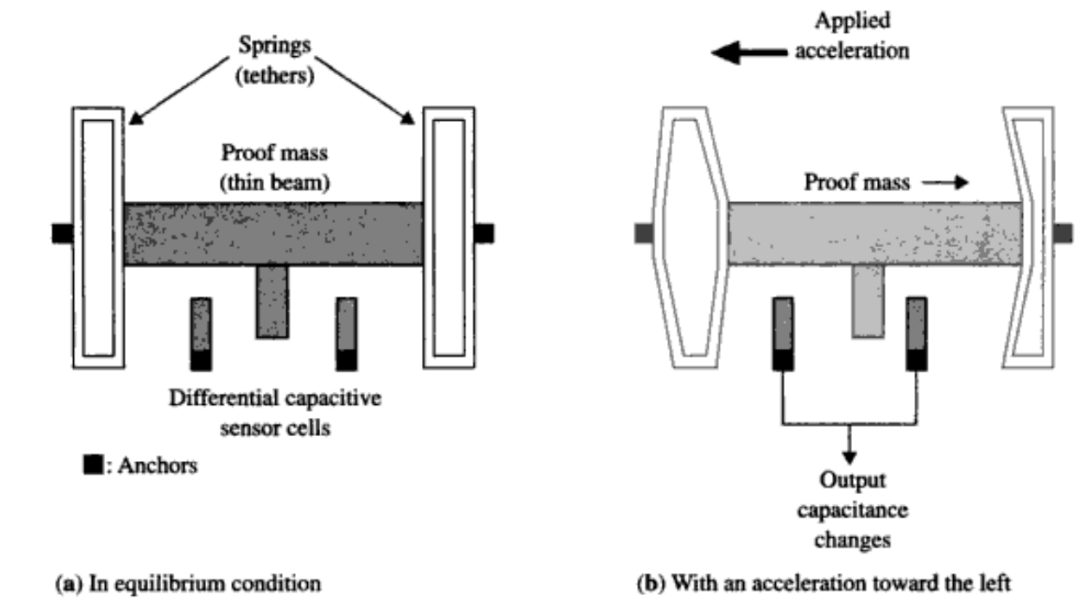


Figura 2.3: Diagrama de la configuración de un micro acelerómetro inercial [10]

efectos de la fuerza de Van der Waals y el efecto del transporte de energía a nivel atómico [10].

Considerando el modelo estático del sensor de la figura 2.4, cuando se tiene una aceleración a , la fuerza inercial ($F = m * a$) causa un desplazamiento estático en el orden de $m * a / K$. En el caso de una entrada dinámica como oscilación, función escalón o un golpe se debe considerar el factor de amortiguamiento. Dentro del encapsulado se tiene una presión P y una temperatura T . La masa de prueba está conectada al encapsulado con resortes o vigas, las cuales tienen una constante del resorte K , este se puede encontrar según la geometría y el material. La fricción y el amortiguamiento se representan con c .

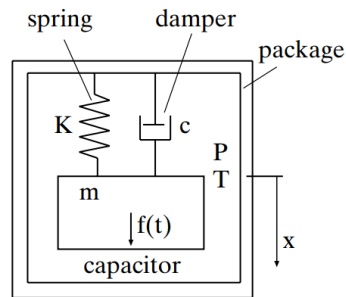


Figura 2.4: Acelerómetro de prueba [26]

El movimiento de la masa se representa en la coordenada X , que corresponde al eje X del plano coordenado de encapsulado. Mientras está en reposo la masa tiene un desplazamiento $x=0$. En resumen, la masa, m , experimenta 3 fuerzas, la fuerza inercial $f(t)$, la fuerza de restauración del resorte $K * x$ y la fuerza de resistencia del amortiguador $c\dot{x}$. De esto se obtiene la siguiente ecuación característica que describe la aceleración $a(t)$ [26].

$$\ddot{x} + 2\xi w_n \dot{x} + w_n^2 = a(t) \quad (2.4)$$

$$\ddot{x} + \frac{w_0}{Q} \dot{x} + w_0^2 = a(t) \quad (2.5)$$

Donde ξ corresponde al factor de amortiguamiento, Q corresponde al factor de calidad del sistema y w_0 a la frecuencia resonancia del sistema.

Los sistemas de segundo orden responden según el tipo de amortiguamiento, cuando se posee un c muy grande se tiene un sistema sobreamortiguado y la solución decae y se extingue, en el caso de amortiguamiento crítico, el sistema regresa al reposo rápidamente y la solución muere fácilmente. Este caso es el deseado en los acelerómetros ya que la respuesta se estabiliza rápido. Cuando el sistema es subamortiguado el sensor puede no obtener la lectura real por un largo tiempo. Cuando la entrada es compleja, se puede analizar la salida en función del tiempo, la solución se puede encontrar al resolver la ecuación diferencial, puede ser con la transformada de Laplace [26]. La aceleración ante una entrada escalón su salida en estado estable se define mediante:

$$x_{s.s}(t) = \frac{a}{w_0^2} = \frac{a}{\frac{K}{m}} = \frac{ma}{K} \quad (2.6)$$

La sensibilidad es inversamente proporcional a la frecuencia resonancia w_0^2 del sistema, para un acelerómetro críticamente amortiguado, cuanto mayor sea la frecuencia de resonancia menor será la sensibilidad. Tiene sentido, ya que para que un dispositivo obtenga una frecuencia de resonancia grande la masa debe ser pequeña para que se mueva rápido, pero una masa pequeña puede disminuir la fuerza de inercia y la sensibilidad [26].

2.3.1 Modelo de acelerómetro con masa suspendida

En esta sección se obtienen el modelo matemático de un acelerómetro de masa suspendida, las ecuaciones se derivan de [24] y se aplican y explican pasos intermedios.

En la figura 2.5 se muestra el modelo del acelerómetro de masa suspendida. Donde la masa de prueba m está conectada al marco mediante un resorte k , se tiene un amortiguador c que evita que la masa choque con las paredes.

Se tiene que $x_m = x(t)$ es la posición de la masa suspendida y $x_f = y(t)$ es la posición del frame o marco de referencia. F_E corresponde a la fuerza externa o ruido Browniano que actúa en la masa. Se tiene inicialmente la ecuación del movimiento de la masa de prueba:

$$m \frac{\partial^2 x_m}{\partial t^2} + c \frac{\partial (x_m - x_f)}{\partial t} + k(x_m - x_f) = F_E \quad (2.7)$$

Restando el término $m \cdot a_f = m \cdot \frac{\partial^2 x_f}{\partial t^2}$ a ambos lados de la ecuación para simplificar se obtiene

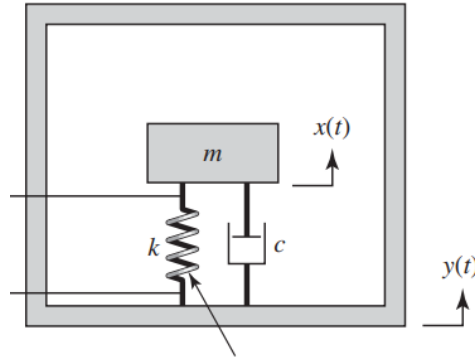


Figura 2.5: Acelerómetro con masa suspendida [45]

$$m \frac{\partial^2(x_m - x_f)}{\partial t^2} + c \frac{\partial(x_m - x_f)}{\partial t} + k(x_m - x_f) = -m \frac{\partial^2 x_f}{\partial t^2} + F_E \quad (2.8)$$

Debido a la inercia de la masa de prueba, el movimiento no sigue el movimiento del marco por lo que se tiene una diferencia de desplazamiento $x = x_m - x_f$ la cual se utiliza para calcular la aceleración, este término representa la diferencia entre la posición del marco de referencia y la masa, lo que genera un modelo de resonancia amortiguado con un grado de libertad gobernada por:

$$m \frac{\partial^2 x}{\partial t^2} + c \frac{\partial x}{\partial t} + kx = F \quad (2.9)$$

Donde F es la suma de la inercia y la fuerza externa $F = m \frac{\partial^2 x_f}{\partial t^2} - F_E = m \ddot{x}_f - F_E$

Resolviendo la ecuación 2.9 con la Transformada de Laplace

$$ms^2 X(s) + csX(s) + kX(s) = F(s) \quad (2.10)$$

$$X(s)(ms^2 + cs + k) = m \ddot{x}_f - F_E \quad (2.11)$$

Donde se asume que no hay fuerzas externas ($F_E=0$) y que la aceleración del marco de referencia es $\ddot{x} = \frac{\partial^2 x_f}{\partial t^2}$

$$X(s) = \frac{m \ddot{x}_f}{(ms^2 + cs + k)} \quad (2.12)$$

Si la respuesta del acelerómetro para $w < w_0$, donde $w_0 = 2\pi * f_0$ corresponde a la frecuencia de resonancia, se tiene

$$\frac{x}{\ddot{x}_f} = \frac{m}{k} = \frac{1}{w_0^2} \quad (2.13)$$

Sacando a factor común m del denominador y definiendo un factor de calidad como $Q = w_0 m / c$ se tiene

$$x = \frac{\frac{F}{m}}{s^2 + \frac{sw_0}{Q} + w_0^2} = \frac{\ddot{x}_f}{s^2 + \frac{sw_0}{Q} + w_0^2} \equiv H(s)\ddot{x} \quad (2.14)$$

La respuesta en frecuencia de un acelerómetro críticamente amortiguado $H(s)$ con $Q=0.5$ en función de la frecuencia de resonancia w_0 , indica que la respuesta en bajas frecuencias aumenta mientras que la frecuencia de resonancia w_0 disminuye, y el desplazamiento debido a altas frecuencias se considera independiente de la frecuencia de resonancia w_0 . Al reducir la frecuencia de resonancia se aumenta la sensibilidad del sistema pero se reduce el ancho de banda por -3dB. La respuesta del acelerómetro se puede definir con la ecuación 2.15 cuando $w \ll w_0$

$$H(s) = \frac{x}{\ddot{x}_f} = \frac{1}{s^2 + \frac{sw_0}{Q} + w_0^2} \approx \frac{1}{w_0^2} \quad (2.15)$$

Con lo que se obtiene la función de transferencia (Salida/Entrada) que describe el movimiento mecánico del acelerómetro. La entrada corresponde a la aceleración del marco de referencia $\ddot{x}_f$ y la salida corresponde a la posición de la masa de prueba x .

En la figura 2.6 se muestra el modelo del acelerómetro de masa suspendida con transductores de capacitancia, el cual detecta pequeños cambios en la capacitancia debido a el movimiento relativo de la masa de pruebas y el marco de referencia, este cambio en la capacitancia se utiliza para deducir la aceleración [24].

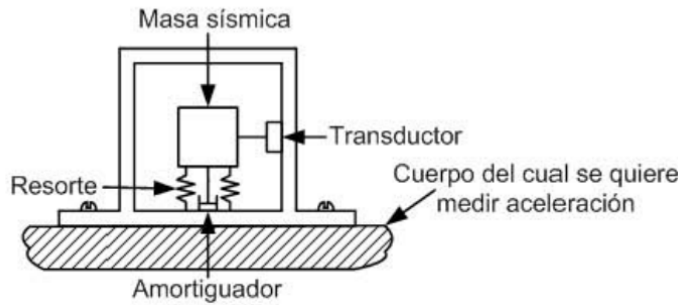


Figura 2.6: Acelerómetro con masa suspendida con transductores de capacitancia [48]

Tomando en cuenta que las mediciones de capacitancia se basan en detectar el cambio de la corriente en el capacitor, mediante la siguiente ecuación se puede obtener el valor de la corriente

$$i = \frac{\partial CV}{\partial t} = C \frac{\partial V}{\partial t} + V \frac{\partial C}{\partial t} \quad (2.16)$$

El término $C \frac{\partial V}{\partial t}$ representa la corriente debido al cambio de la tensión en el tiempo, este método para medir la capacitancia se conoce como el método de la medición del despla-

zamiento. El termino $V \frac{\partial C}{\partial t}$ corresponde a la corriente debido al cambio de la capacitancia en el tiempo, este método para medir la capacitancia se conoce como el método de la medición de la velocidad [24].

Tomando un enfoque en el método de la medición de la velocidad, si la tensión $V = V_{dc}$ se mantiene constante ($\frac{\partial V}{\partial t} = 0$) la corriente del capacitor derivada de la ecuación 2.16

$$i = V_{dc} \frac{\partial C}{\partial t} \equiv i_{mot} \quad (2.17)$$

El capacitor de placas paralelas es el modelo más común utilizado en los acelerómetros MEMS, en este se tiene una capacitancia de

$$C = \epsilon \frac{A}{d - x} \quad (2.18)$$

Donde ϵ corresponde a la permitividad, A es el área del electrodo, d es la distancia inicial entre placas y x es el movimiento de las placas desde la posición original. Con esto se puede obtener la corriente generada por el movimiento para $x \ll d$

$$i_{mot} = V_{dc} \frac{\partial C}{\partial t} = V_{dc} \frac{\partial C}{\partial x} \frac{\partial x}{\partial t} = V_{dc} \epsilon \frac{A}{(d - x)^2} \frac{\partial x}{\partial t} \approx V_{dc} \epsilon \frac{A}{d^2} \frac{\partial x}{\partial t} = V_{dc} \epsilon \frac{A}{d^2} \dot{x} \quad (2.19)$$

Se observa que la corriente es proporcional a la velocidad de las placas del capacitor.

Tomado la ecuación 2.19 y aplicando la transformada de Laplace

$$i_{mot} = V_{dc} \epsilon \frac{A}{d^2} \dot{x} \Rightarrow I_{mot} = V_{dc} \epsilon \frac{A}{d^2} sX(s) \quad (2.20)$$

Despejando X(s) e igualando con la ecuación 2.14 en el dominio de S, se puede obtener una función de transferencia que relacione el modelo mecánico con el modelo eléctrico del acelerómetro MEMS capacitivo.

$$X(s) = \frac{I(s)}{V_{dc} \epsilon \frac{A}{d^2} s} = \frac{\ddot{x}_f}{s^2 + \frac{sw_0}{Q} + w_0^2} \quad (2.21)$$

Considerando I(s) como la salida del sistema y la aceleración $\ddot{x}_f$ como la entrada, se obtiene la función de transferencia en función de la corriente que relaciona el modelo mecánico con el modelo eléctrico del acelerómetro [24]

$$\frac{I(s)}{\ddot{x}_f} = \frac{V_{dc} \epsilon \frac{A}{d^2} s}{s^2 + \frac{sw_0}{Q} + w_0^2} \quad (2.22)$$

Finalmente, se busca relacionar la temperatura con la aceleración, esto se da mediante la ecuación que describe aceleración equivalente al ruido dado por el cambio de temperatura

y el ruido Browniano (ruido mecánico), con unidades de $[m/s^2/\sqrt{(Hz)}]$

$$\ddot{x}_n = \sqrt{\frac{4 * k_b * T * w_0}{m * Q}} \quad (2.23)$$

Donde k_b corresponde a la constante de Boltzmann ($k_b=1.380*10^{-23}$ [J/K]) y T corresponde a la temperatura en (Kelvin) ,

2.4 Software y Hardware

En la siguiente sección se mencionan elementos de *Software* y *Hardware* relevantes para la implementación del proyecto

2.4.1 FPGA

Las FPGA (Field Programmable Gate Arrays, por sus siglas en inglés) son circuitos integrados reconfigurables, están compuestas por interconexiones programables, bloques lógicos, memorias embebidas y circuitos para el procesamiento de señales digitales, ver figura 2.7. Las FPGA son consideradas como sistemas de “hardware programable”. Aunque estas operan a frecuencias de reloj más bajas y tienen picos de rendimiento inferiores, la capacidad que se tiene para adaptar el hardware en cada aplicación particular le permite a las FPGAs lograr mejores tasas de rendimiento que otros procesadores en la mayoría de los casos. Algunas de las desventajas recaen en el alto costo de programación y de desarrollo, ya que la programación se realiza a través de lenguajes de descripción de hardware (HDL), como Verilog o VHDL (Very High Speed Description Language, por sus siglas en inglés) [36].

La empresa Xilinx ofrece FPGAs de bajo costo y rendimiento medio y otras de alto costo con rendimiento alto. Para poder recibir y transmitir señales digitales, se disponen de un bloque de Entrada y Salida (E/S - I/O blocks) que puede ser utilizado con muy diversos rangos de tensiones, frecuencias de trabajo y estándares de señales digitales. Existe un bloque E/S por cada terminal de E/S del FPGA. Así cada terminal puede ser configurado como entrada, como salida o bidireccional. En cada bloque E/S existe un buffer configurable por el diseñador que permiten adaptar el FPGA según las necesidades de la aplicación [43]. Las FPGA utilizan Conversores Analógicos-Digitales (ADC) en sus E/S, además existen problemas potenciales debido a ruido entre comunicaciones de la FPGA y los ADC [42].

2.4.2 Software de programación

LabVIEW es un entorno de programación gráfica que permite a los ingenieros desarrollar sistemas de pruebas automatizadas de investigación, validación y producción. Posee

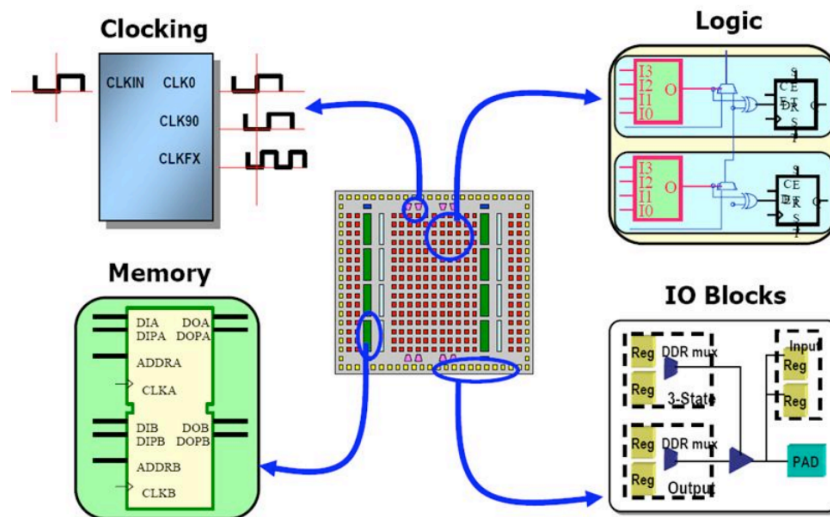


Figura 2.7: Diagrama de la estructura interna de una FPGA [36]

elementos de visualización interactivos y configurables, controladores para automatizar los instrumentos y el hardware de adquisición de datos. Permite tener conectividad con lenguajes y protocolos estándares de la industria como acceder, ver y analizar datos, además permite crear secuencias de pruebas automatizadas con TestStand. TestStand es un software de administración de pruebas que ayuda a desarrollar, depurar e implementar sistemas de pruebas y proporciona visibilidad completa del proceso de prueba y los resultados. LabVIEW tiene como objetivo incrementar la productividad con nuevas características que mejoran la interoperabilidad con Python y MATLAB, desarrollando aplicaciones en tiempo real y FPGA, e ingeniería de software [17].

2.4.3 myRIO

El myRIO es un dispositivo embebido para estudiantes, entrenamiento e implementación de proyectos, cuenta con módulos E/S en ambos lados del dispositivo en forma de conectores MXP y SMP. Además, incluye entradas analógicas, salidas analógicas, líneas de E/S digitales, LEDs, un botón, un acelerómetro interno, una FPGA Xilinx y un procesador dual-core ARM Cortex-A9. Este dispositivo se puede programar con LabVIEW o C, esta plataforma reconfigurable permite enseñar e implementar múltiples conceptos de diseño con un dispositivo [16]. En [12] se tiene el enlace a la hoja de especificaciones del myRIO y en el anexo A.13 se adjunta la hoja de datos. La FPGA del myRIO posee aproximadamente 28000 celdas lógicas. El acelerómetro interno es un NXP MMA8452Q, en [40] se muestra la hoja de datos del acelerómetro embebido en su tarjeta.

2.4.4 Azure DevOps

Corresponde a una herramienta para el control de versiones, en el cual se establecen las tareas que se deben completar para cumplir con una funcionalidad del sistema por desarrollar. Se puede actualizar el sistema de control de versiones mediante GIT, con el fin de almacenar los cambios en la plataforma Azure DevOps y que todos los involucrados en el proyecto puedan ver y acceder a cambios realizados. En NI se utiliza esta plataforma como medio para la distribución de tareas y registro de horas implementadas en el desarrollo de un objetivo.

2.4.5 API

Las siglas API (Application Programmer Interface, por sus siglas en inglés) Interfaz de Programación de Aplicaciones, actualmente corresponde a uno de los servicios de intercambio de información y acceso a datos más comunes. Funciona como una interfaz que facilita a programas distintos comunicarse e intercambiar información. Su uso depende de los permisos que el propietario de la API brinda a los desarrolladores de terceros. No es necesario desarrollar un nuevo código para lograr la interacción entre diversos sistemas de información de una organización. Una ventaja que poseen las API es que no es necesario conocer la estructura interna para generar las conexiones y permiten aplicar diferentes operaciones durante el acceso y recuperación de los datos ante una descarga de archivos [5].

Las API permiten combinar funcionalidades que aporta un software e integrarlo a otro software para mejorar sus resultados, sirve como canal de enlace entre un software ya creado y otro que lo puede necesitar, permite tener una interacción “software-to-software” [37]. Las **API internas o privadas** son utilizadas únicamente dentro de una misma compañía, las **API externas o públicas** pueden ser utilizadas por cualquiera que la necesite. Las **API Partner** corresponden a un paso intermedio entre las privadas y las públicas, ya que pueden ser utilizadas por varias compañías que colaboran entre ellas. Las API se pueden clasificar según su orientación [37]:

- **APIs de servicios Web** proporcionan acceso a su servicio mediante una dirección web e implica comunicación en una red.
- **APIs basadas en librerías** tienen gran importancia según el lenguaje de programación, al hacer uso de sus librerías se facilita la programación.
- **APIs de sistemas operativos** permiten conocer la estructura de las funcionalidades del Sistema Operativo.

2.4.6 TestStand

TestStand es un software de administración de pruebas que ayuda a desarrollar, depurar e implementar sistemas de pruebas y proporciona visibilidad completa del proceso de prueba y los resultados. TestStand es un software de aplicación que permite desarrollar sistemas de validación y pruebas automatizadas que integran módulos de código escritos en cualquier lenguaje de programación [16].

2.4.7 Protocolos de comunicación

I^2C

I^2C es un protocolo de comunicación serial que permite transferir bits entre 2 dispositivos digitales. Se tienen dos cables de comunicación: Datos Seriales (Serial Data – SDA) y Reloj Serial (Serial Clock – SCL), en este se pueden conectar hasta 127 dispositivos esclavos con esas dos líneas, con un velocidades de hasta 100, 400 y 1000 kbits/s. Los mensajes que se envían mediante un puerto I^2C incluyen una dirección tanto del registro como del sensor. Para la información que se envía siempre existe una confirmación de recepción por parte del dispositivo [14]. Se diferencian dos elementos básicos, un maestro y un esclavo. La figura 2.8, muestra una conexión de tres dispositivos, el bus consiste de dos líneas: SDA y SCL. Normalmente al bus se le conectan dos resistencias en arreglo pull-up, de entre 2.2K y 10K.

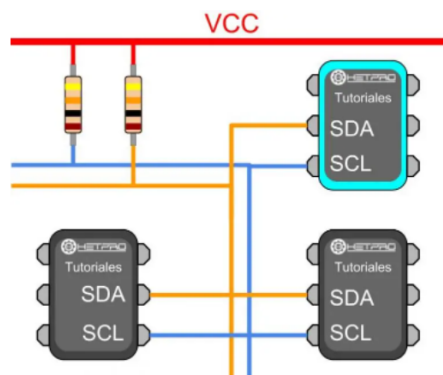


Figura 2.8: Diagrama de conexión protocolo I^2C [8]

El maestro se encarga de controlar al cable de reloj e iniciar y parar la comunicación. La información binaria serial se envía por la línea de datos seriales. Dos Maestros no pueden hacer uso de un mismo puerto I^2C . Las funciones principales del maestro son:

- Iniciar la comunicación – S
- Enviar 7 bits de dirección – ADDR
- Generar 1 bit de Lectura ó Escritura – R/W

- Enviar 8 bits de dirección de memoria
- Transmitir 8 bits de datos
- Confirmar la recepción de datos – ACK – Acknowledged
- Generar confirmación de No-recepción, NACK – No-ACKnowledged
- Finalizar la comunicación

El esclavo suministra la información solicitada por el maestro. Puede actuar como esclavo-transmisor ó esclavo-receptor. Un dispositivo I2C esclavo, no puede generar a la señal SCL. Sus funciones principales son: Enviar información en paquetes de 8 bits y enviar confirmaciones de recepción (ACK) [14].

SPI

El protocolo de comunicación SPI fue desarrollada por Motorola en 1985. Se trata de una interfaz serial síncrona para la comunicación entre dispositivos a corta distancia. El SPI es una interfaz direccionada de hardware simple que ofrece completa flexibilidad para la cantidad de bit transferidos. Usa un maestro simple y puede manejar varios dispositivos secundarios usando comunicaciones dúplex que operan a velocidades de reloj de hasta 50 MHz. No usa un protocolo estándar y transfiere solo paquetes de datos, lo que la hace ideal para transferir flujos de datos largos.

SPI usa un máximo de cuatro líneas de señal, la figura 2.9, muestra una conexión de tres dispositivos. El dispositivo maestro suministra y controla el reloj (SCL) y líneas de selección de chip (CS). La operación multiplexor completa se maneja a través de las líneas de datos Master Out Slave In (MOSI) y Master In Slave Out (MISO). Si el dispositivo secundario solo puede enviar datos (comunicación semidúplex), la línea MOSI puede eliminarse, y así reducir el conteo de señales. Los datos salen a través de la señal del reloj de tal forma que la transferencia de datos se asemeja a un registro de turnos con un bit cambiado para cada reloj [15].

UART

UART significa Transmisor-Receptor Asíncrono Universal y define un protocolo para intercambiar datos en serie entre dos dispositivos, la figura 2.9 muestra la conexión entre dispositivos. El UART fue uno de los primeros protocolos en serie. En los últimos años, la popularidad del UART ha disminuido y los protocolos con SPI e I2C han empezado a reemplazarlo. El UART aún se utiliza para aplicaciones de menor velocidad y de menor rendimiento, porque es muy simple, de bajo costo y de fácil implementación.

Este protocolo solo utiliza dos cables entre el transmisor y receptor para transmitir y recibir en ambas direcciones. Ambos terminales también tienen una conexión a tierra. La

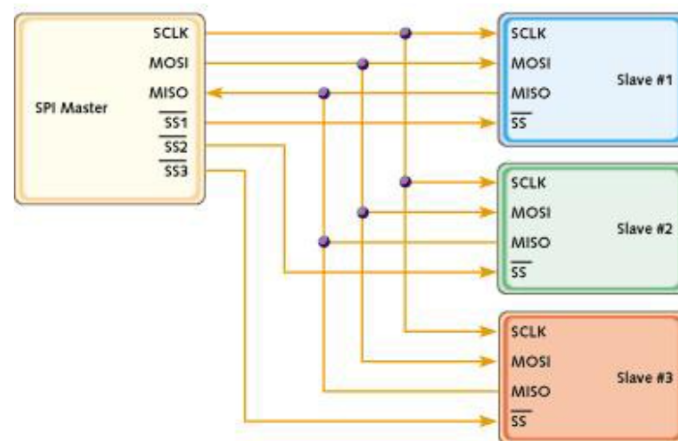


Figura 2.9: Diagrama de conexión protocolo SPI [15]

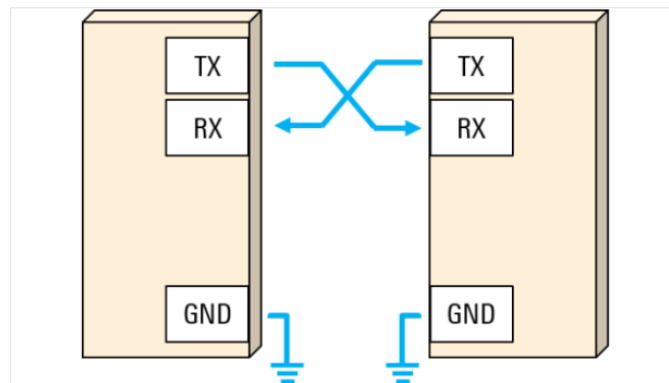


Figura 2.10: Diagrama de conexión protocolo UART [39]

comunicación en el UART puede ser simplex (los datos se envían en una sola dirección), semidúplex (cada lado transmite pero solo uno a la vez), o dúplex completo (ambos lados pueden transmitir en simultaneo). Los datos en el UART se transmiten en la forma de tramas [39]. Una de las grandes ventajas del UART es que es asíncrono: el transmisor y el receptor no comparten una señal de reloj común, lo que simplifica el protocolo, pero pone ciertos requisitos al transmisor y al receptor. Ambos terminales deben transmitir a la misma velocidad preestablecida para que tengan la misma sincronización de bits. Las velocidades de baudios del UART más comunes que se utilizan en estos días son 4800, 9600, 19.2K, 57.6K y 115.2K. Además de tener la misma velocidad en baudios, ambos lados de una conexión UART también deben utilizar los mismos parámetros y estructura de trama.

2.4.8 Fricción y vibraciones

La fricción corresponde a una fuerza que resiste el movimiento de dos superficies en contacto que se deslizan relativamente entre sí. Esta fuerza actúa siempre tangencialmente

a la superficie en los puntos de contacto y está dirigida en sentido opuesto al movimiento entre las superficies [9].

Cualquier movimiento que se repite después de un intervalo de tiempo se llama vibración u oscilación. Por ejemplo el vaivén de un péndulo y el movimiento de una cuerda pulsada son representaciones comunes de vibración. Este se basa en el estudio de los movimientos oscilatorios de los cuerpos y las fuerzas asociadas con ellos. La vibración de un sistema implica la transformación de su energía potencial en energía cinética y de ésta en energía potencial, de manera alterna [35].

2.5 Problemática de la fabricación de chips

Durante el periodo de la pandemia, se dio a conocer que la renaciente industria automotriz se enfrentaba a un "Armagedón de los chips". Los vehículos nuevos a menudo incluyen más de 100 microprocesadores y los fabricantes no fueron capaces de abastecer a todos sus clientes. Desde entonces, los fabricantes de chips han advertido que también enfrentan limitaciones. Las restricciones impulsaron la venta de computadoras y otros dispositivos para realizar teletrabajo y la industria automotriz inicialmente vio una gran caída en la demanda y recorta sus pedidos [25].

El problema recae en que las fábricas de chips llevan muchos meses produciendo al límite de su capacidad (especialmente las de semiconductores de alta integración), por lo que no es posible obtener todos los chips con la misma frecuencia. El impacto que está teniendo el déficit de circuitos integrados se aplica tanto en sectores industriales como en consumidores independientes. Algunas de las industrias más afectadas son las relacionadas con manufactura de automóviles, las empresas de tecnologías de la información y fabricación de electrodomésticos, esto debido a su dependencia de la producción de semiconductores [27].

La escasez de semiconductores continuará al menos hasta el 2023, y puede que durante varios años más. La brutal escasez de semiconductores provocada por el aumento de la demanda durante la pandemia ha generado costos adicionales que el consumidor debe pagar. El problema es que cada vez menos compañías se dedican a la manufactura. Las empresas 'pure fab' no generan diseños, ni conceptualizan, solo se encargan de fabricar los semiconductores [30].

Para poder validar el modelo implementado en la FPGA se propone caracterizar un acelerómetro MEMS, pero dada la situación de la producción de chips se debe considerar la posibilidad no poder obtener el mismo debido al desabastecimiento que está sufriendo la industria de venta de componentes electrónicos. En este caso, se propone realizar una búsqueda de sensores que se encuentren o se puedan obtener en Costa Rica en un corto plazo, esto con el fin de poder realizar la caracterización del sensor y no tener retrasos en la entrega del proyecto.

2.6 Repetibilidad

En estadística la palabra experimento describe cualquier proceso que genere un conjunto de datos y es de gran interés las observaciones que se obtienen al repetir varias veces un experimento. En la mayoría de los casos los resultados dependen del azar. En la mayoría de las aplicaciones de probabilidad opera la interpretación de frecuencia relativa de probabilidad, la cual se basa en el experimento estadístico en vez de en la subjetividad, por lo que muchas aplicaciones de probabilidad en ciencia e ingeniería se deben basar en experimentos que se puedan repetir [34].

Para saber cuántas muestras tomar para cada situación se utiliza la ecuación 2.24 de [34]. En este, si usamos $\bar{x}$ como una estimación de μ , podemos tener $100(1-\alpha)\%$ de confianza en que el error no excederá a una cantidad específica e cuando el tamaño de la muestra sea

$$n = \left(\frac{z_{\alpha/2}\sigma}{e} \right)^2 \quad (2.24)$$

Donde n corresponde a la cantidad de medidas a realizar por situación, σ corresponde a la desviación estándar y e corresponde al error. Además, $z_{\alpha/2}$ es el valor z que deja una área de $\alpha/2$ a la derecha de la curva normal.

Capítulo 3

Metodología

En la figura 3.1 se muestra una diagrama de bloques que indica los pasos realizados para obtener la solución del problema.

3.1 Laboratorios de la empresa

Se realiza una visita inicial a los laboratorios de la empresa. En este se verifican los equipos disponibles y herramientas que pueden ser implementadas en la solución. Para esto la empresa tiene un sistema de administración de equipos, con el cual cada usuario puede solicitar el equipo por un periodo de tiempo determinado, permite verificar la ubicación y estado de los equipos disponibles, y verificar los equipos y elementos existentes en el laboratorio de PS de la empresa.

3.2 Selección del acelerómetro MEMS

Como se mencionó en la problemática de la fabricación de chips, se procede a realizar una búsqueda de sensores comerciales disponibles en Costa Rica con el fin de realizar la caracterización y evitar retrasos por el periodo de entrega de los mismos. En la figura 3.2 se muestra una tabla con el nombre del sensor, el rango de medición, el número de ejes que posee, su precio unitario y la disponibilidad en Costa Rica durante la búsqueda.

Dentro de los principales criterios de selección se verifica el rango de medición, el cual debe ser lo suficientemente grande para abarcar diversas aceleraciones; el precio unitario, ya que se deben comprar al menos 6 sensores para tener de repuesto y de respaldo ante fallas; finalmente la disponibilidad en Costa Rica, con el fin de evitar retrasos en la entrega de los sensores y problemas de disponibilidad.

Algunos de estos acelerómetro MEMS están embebidos (encapsulado) en placas que se pueden conectar directamente a microcontroladores, mientras que otros son solo los chips

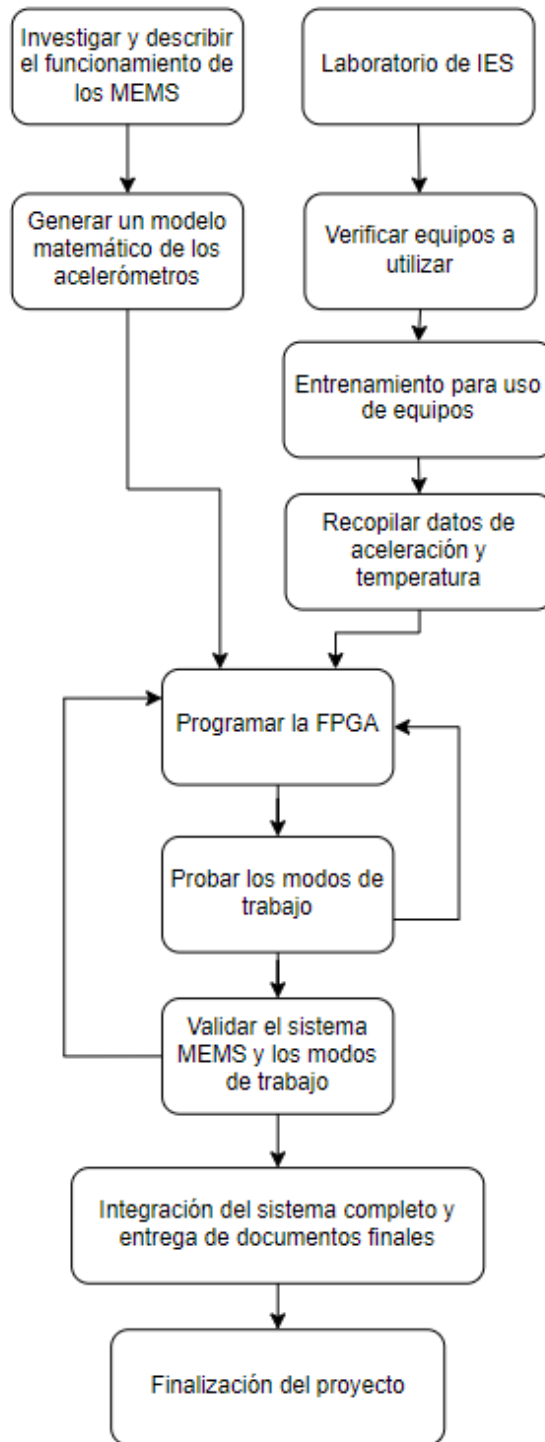


Figura 3.1: Diagrama de bloques para desarrollar solución

(micro chips) que deben ser acoplados y soldados a placas para poder obtener datos de los mismos. Debido a esto, en la figura 3.3 se muestra una diagrama de bloques con los planes a seguir según el tipo de sensor que se logre obtener para la caracterización.

En el Plan A, si se tiene el chip MEMS del acelerómetro, se construye una placa de montaje para poder realizar la caracterización. Este sería el caso ideal para el proyecto,

Nombre	Rango	Num. de ejes	Precio unitario	Disponibilidad	Tiempo de entrega
Acelerómetros Triaxial, 200g, Tray 830M1 Accelerometer	200	3	€107 035	40	Desconocido
Acelerómetros Micropower Three-Axis +/-400g Accel	-+400	3	€12 194	747	Desconocido
H3LIS331DLTR Acelerómetro MEMS, Digital, X, Y, Z, $\pm 100g$, $\pm 200g$, $\pm 400g$, 2.16 V, 3.6 V, TFLGA	$\pm 100g$, $\pm 200g$, $\pm 400g$	3	\$14.08	No disponibles	Desconocido
MPU6050 Module Triple-Axis Gyroscope Accelerometer For Arduino	± 2 , ± 4 , ± 8 , $\pm 16g$	3	\$6.95	6 en Costa Rica	Inmediato
LSM6DS3 - SparkFun 6 Degrees of Freedom Breakout [SEN-13339]	$\pm 2/\pm 4/\pm 8/\pm 16$ g full scale	3	\$14,50	No disponibles	Desconocido
ICM-20948 (Qwiic) - SparkFun 9DoF IMU Breakout [SEN-15335]	$\pm 2g$, $\pm 4g$, $\pm 8g$, $\pm 16g$	3	\$23.90	2 en Costa Rica	Inmediato
MPU-9250 IMU Breakout [Generic]	$\pm 2g$, $\pm 4g$, $\pm 8g$, $\pm 16g$	3	\$9.50	No disponibles	Desconocido
Accelerometer module ADXL335 Three Axis Low-g	± 3 g	3	\$9.95	17 en Costa Rica	Inmediato
MMA8452Q, 3-axis, 12-bit/8-bit digital accelerometer	± 2 g/ ± 4 g/ ± 8	3	\$1,200.00	1 en Costa Rica	Inmediato

Figura 3.2: Sensores disponibles comercialmente en Costa Rica

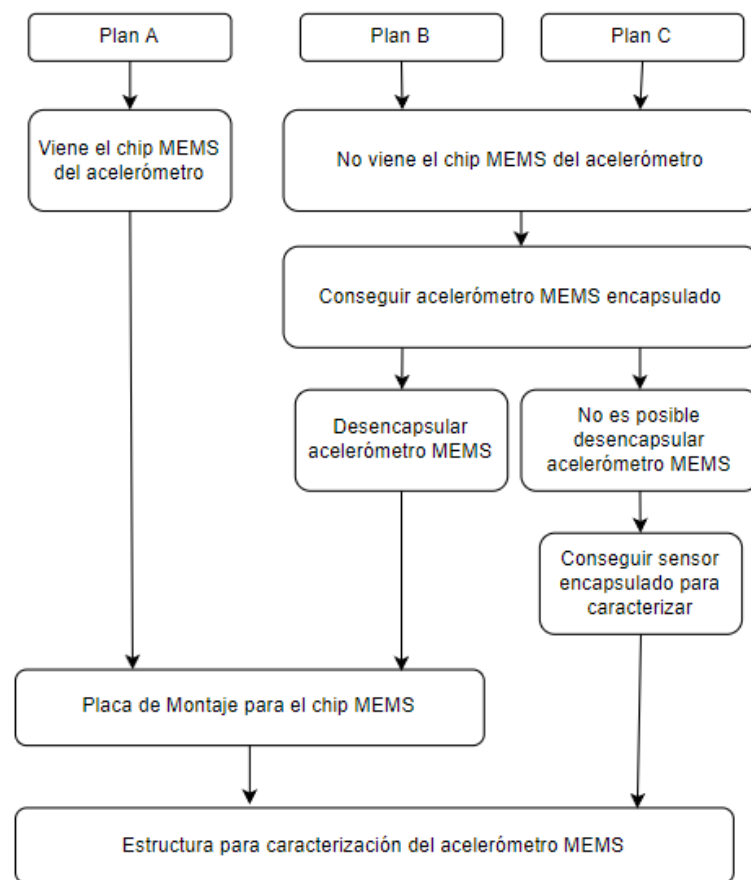


Figura 3.3: Diagrama de flujo de posibles planes a tomar

se consigue un micro chip del acelerómetro MEMS, se diseña el sistema de acople y se suelda para poder realizar su caracterización, En la figura 3.2 se puede observar que el tiempo de entrega de los sensores es desconocido. Al verificar las páginas de ventas se indica que los sensores de micro chip no están disponibles y se aproxima que podrían

ingresar al inventario hasta dentro de 30 semanas y las entregas se hacen bajo la lista de pedidos ya en marcha. Debido a esto se proponen dos alternativas con los acelerómetros disponibles en el país.

En el Plan B, no se tiene el chip MEMS del acelerómetro pero se consigue el acelerómetro encapsulado, se intenta desencapsular el acelerómetro para obtener el chip y se construye una placa de montaje para poder realizar la caracterización. Finalmente, en el Plan C, no se tiene el chip MEMS del acelerómetro pero se consigue el acelerómetro encapsulado, no es posible desencapsular el acelerómetro para obtener el chip por lo que se utiliza el sensor encapsulado para realizar la caracterización.

3.3 La crisis de contenedores

Durante el inicio de la pandemia se invirtió más en bienes que en servicios, lo que aumentó el tráfico de buques y el volumen de productos a desembarcar por una poca cantidad de mano de obra [32].

Las cadenas de suministro global está pasando por una crisis que ha llevado a una escasez de productos. Los estragos de la pandemia del Covid-19 llevaron al cierre de fábricas, pero actualmente la problemática no es debido a la producción, sino corresponde a un problema de logística y distribución. El sistema de transporte marítimo sustenta el 90% del comercio internacional y se tiene que los buques están atascados en las costas por lo que los dispositivos electrónicos no han podido llegar a su punto final. Debido a los problemas de logística se están utilizando los contenedores para almacenar y no para transportar productos. La demanda de productos se mantiene igual, pero las cadenas de suministro están desatendidas, por lo que la distribución se hace más lenta [31]. Esta crisis de logística se ve influenciada por la escasez de contenedores y el poco espacio para poder transportar carga. Con esto los precios de traslado aumentaron por lo que los consumidores son los que pagarán el precio del transporte [29].

3.3.1 Acelerómetro comercial disponible en Costa Rica

Dado los problemas de atraso de contenedores y las pocas unidades de acelerómetros MEMS disponibles en Costa Rica, se utiliza la lista de sensores en la figura 3.2 para tener diversas alternativas de sensores a utilizar.

Al evaluar los criterios de selección de los diversos sensores se tiene:

- **Precio unitario:** El precio unitario del sensor no debe sobrepasar los 15 dólares la unidad, esto dado el presupuesto de compra y que se debe comprar al menor 6 sensores para tener repuesto y respaldo ante fallas. El H3LIS331DLTR cuesta \$14.08, el MPU-6050 cuesta \$6.95, el LSM6DS3 cuesta \$14.50, el MPU-9250 cuesta \$6.50 y el ADXL335 cuesta \$9.95

- **Rango de medición:** De los sensores anteriores se extrae el ADXL335 ya que este posee un único rango de medición y es muy pequeño en comparación con los otros sensores. Todos los otros sensores tiene al menos 3 rangos de medición.
- **Disponibilidad:** Al verificar la disponibilidad del H3LIS331DLTR, el MPU-6050, el LSM6DS3 y el MPU-9250, se tiene que solamente el el MPU-6050 tiene unidades disponibles en Costa Rica al momento de hacer este análisis.

Mediante esta selección se logra obtener 6 unidades del sistema de orientación con acelerómetro embebido MPU-6050, el cual contiene un acelerómetro de masa suspendida en cada eje, un rango desde $\pm 2g$ hasta un máximo de $\pm 16g$. Este acelerómetro posee un modo de lectura para baja corriente y tiene funciones de detección de orientación y modo de prueba. La hoja de datos del MPU-6050 se encuentra en [22] y en el anexo A.14.

Aunque este acelerómetro no pueda llegar a rangos de aceleración de $\pm 100g$ se podría considerar extrapolar la respuesta del sensor a otros rangos más amplios para poder cumplir con los requerimientos del proyecto.

3.4 Caracterización del Acelerómetro MEMS

3.4.1 Sistema de caracterización

Con el fin de poder comparar la respuesta del acelerómetro simulado, se propone obtener datos del MPU-6050 bajo diversas condiciones de aceleración y temperatura, modelar su comportamiento y plantear las ecuaciones que describen al sensor. Para esto se procede a seleccionar un sistema que permite caracterizar al sensor, donde sea posible variar la aceleración del sistema y tomar datos del sensor para generar el modelo.

3.4.2 Planteamiento

Seguidamente, se describe el proceso desarrollado para caracterizar los acelerómetros y la implementación del simulador en la FPGA. Como se mencionó anteriormente, la no posee un sistema simulado de un acelerómetro de 3 ejes que muestre un valor confiable de temperatura y el estado del chip ante una solicitud de operación y que pueda ser aplicado durante el proceso de pruebas de las aplicaciones desarrolladas por la empresa.

Con base en esta información se plantea la misión del proyecto, como se muestra en la siguiente tabla:

Tabla 3.1: Declaración de la misión.

Declaración de la Misión: Simulador de un acelerómetro MEMS	
Descripción del sistema	Simulador, MEMS, Acelerómetro
Propuesta de Valor	Apoyar desarrollos del departamento de Servicios Profesionales (PS)
Metas claves	Reducir tiempos de prueba, utilizar datos reales en las pruebas, apoyar el proceso de prototipado, mostrar a clientes funcionamiento real de aplicación
Suposición	Confiable Mecatrónico Configurable - a modos de prueba y falla
Usuarios	Integrantes del departamento de Servicios Profesionales (PS)

3.4.3 Entrevistas y Procesado de Información

Seguidamente se recopiló información sin procesar a través de entrevistas y encuestas al cliente, con base a esto se plantearon los requerimientos y necesidades del proyecto. Con esta información se construye un conjunto de necesidades, y una categorización y jerarquización de las mismas a través de un cuestionario al cliente. La entrevista se hace mediante dos reuniones, en la primera se hace una explicación general del proyecto por parte del cliente, en el cual se indica la lista de requerimientos que tiene sobre el sistema. En la segunda reunión, se realizan preguntas de acuerdo a los rangos de medición y se realizan cambios. Como encargado de la aplicación el desarrollador puede elegir los rangos, pero estos deben ser consistentes y validables.

En el apéndice [A.1](#) se tiene la tabla completa con los enunciados del cliente, la necesidad interpretada de la misma, el enunciado correcto de la necesidad, la escala de importancia dada por el cliente y los comentarios del cliente respecto a los mismos.

3.4.4 Jerarquización de necesidades

Con base a las necesidades establecidas se realiza una encuesta al cliente, en el apéndice [A.2](#) se encuentra la encuesta completada por el cliente. Según la información de la calificación se procede a jerarquizar las necesidades, un * indica que la necesidad es más importante que las otras:

El sistema simula de manera confiable un acelerómetro MEMS

*El sistema simulado varía su respuesta según la temperatura ingresada y pueden agregarse nuevas funcionalidades.

*El sistema simulado trabaja en un rango de temperatura entre -40 y +125 celsius.

*El sistema simulado se configura en al menos 3 rangos de diferentes aceleraciones e indica la sensibilidad en mv/g.

*El sistema simulado es alimentado con una fuente de entre 3.3V y 5V.

*El sistema simulado indica si el chip está presente, el modelo del chip y el número de serie del mismo.

El sistema simulado se configura en modo de trabajo o modo de falla a través de la comunicación con el chip.

*El sistema simula un acelerómetro MEMS en la FPGA del myRio.

*El sistema simulado puede dar respuesta en el modo de prueba o el modo de falla en una, alguna o todas las salidas.

*El sistema simulado utiliza API para comunicarse, medir y probar el chip.

El sistema simulado tiene salidas analógicas y digitales para cada uno de sus ejes, mediante algún protocolo IC.

*El sistema simulado es compatible con tecnología TTL en sus salidas.

*El sistema simulado muestra al usuario la temperatura actual y el estado del chip.

*El sistema simulado posee un canal individual para cada salida.

El sistema simulado posee un manual de operación con los modo de trabajo y las especificaciones del sistema.

*El sistema simulado valida el chip mediante una secuencia

*El sistema simulado posee una tarjeta electrónica con las conexiones necesarias para su funcionamiento.

*El sistema simulado está dentro de un encapsulado que lo aisle y resguarde.

El sistema de caracterización varía la aceleración hasta los rangos de trabajo establecidos.

*El sistema de caracterización está aislado del medio ambiente y en un medio controlado para la medición.

*El sistema de caracterización guarda los datos obtenidos y permite accesarlos.

**El sistema de caracterización se puede utilizar repetidamente y se puede cambiar los sensores a utilizar.

***El sistema de caracterización compensa los errores de ruido y vibraciones en el sistema.

El precio de los acelerómetros a utilizar no sobrepasa los 15 dólares la unidad.

*Los acelerómetros a utilizar están disponibles en Costa Rica y el tiempo de adquisición es menor o igual a 2 semanas.

3.4.5 Establecimiento de Métricas

Se establecen las métricas para cada necesidad. Es importante recalcar que las métricas deben ser cuantificables y deben responder a una necesidad planteada. Es posible que

una métrica atiende a más de una necesidad, sin embargo no es posible a lo inverso. En el apéndice A.3 se encuentra la lista de métricas completa. En la tabla 3.2 se especifica la correspondencia entre las necesidades y las especificaciones establecidas para el proyecto.

Tabla 3.2: Relación necesidad-métrica.

	Necesidad	Métrica
1	El sistema simula de manera confiable un acelerómetro MEMS	Validar el comportamiento de la simulación, comparar con la respuesta de un acelerómetro real mediante el grado de semejanza entre simulación y acelerómetro real (Documento/TEST)
2	El sistema simulado varía su respuesta según la temperatura ingresada y pueden agregarse nuevas funcionalidades	Grado de semejanza entre simulación y acelerómetro real ante variaciones en temperatura y modificaciones en funciones (Documento/TEST)
3	El sistema simulado trabaja en un rango de temperatura entre -40 y +125 celcius	Rango de temperatura de trabajo a la cual se simula la respuesta del sistema (Numérico)
4	El sistema simulado se configura en al menos 3 rangos de diferentes aceleraciones e indica la sensibilidad en mv/g	Cantidad, tres (3) rangos de aceleración diferentes
5	El sistema simulado es alimentado con una fuente de entre 3.3V y 5V	Tensión de entrada del sistema (Volts - rango)
6	El sistema simulado indica si el chip está presente, el modelo del chip y el número de serie del mismo	Chip validado mediante secuencia, Cumplimiento de la lectura del chip, modelo del chip y el número de serie (TEST / Binario Pasa no pasa)
7	El sistema simulado se configura en modo de trabajo o modo de falla a través de la comunicación con el chip	Validar modo trabajo y modo falla para cada salida (Documento/TEST)
8	El sistema simulado puede dar respuesta en el modo de prueba o el modo de falla en una, alguna o todas las salidas	Validar modo trabajo y modo falla para cada salida (Documento/TEST)

3.4.6 Determinar Valores Objetivos

Los valores objetivo se obtuvieron según las necesidades del cliente y comparando en algunos casos con los requerimientos establecidos en la descripción del proyecto dada por el cliente.

Las métricas en sí dan una idea de que factores y necesidades tomar en cuenta. Sin embargo, es necesario cuantificar las métricas para valorar su desempeño. Para ello se

Tabla 3.3: Continuación tabla 3.2 Relación necesidad-métrica.

	Necesidad	Métrica
9	El sistema simulado es compatible con tecnología TTL en sus salidas	Tensión de salida del sistema es compatible con TTL (Volts - rango)
10	El sistema simulado tiene salidas analógicas y digitales para cada uno de sus ejes, mediante algún protocolo IC	La lectura de la salida es tanto analógica como digital para cada eje. (cumple o no cumple / binario)
11	El sistema simulado muestra al usuario la temperatura actual y el estado del chip	Visualización de los datos y estado chip en la interfaz (cumple o no cumple / binario)
12	El sistema simulado posee un canal individual para cada salida.	Cantidad de salidas son 6, 1 analógica y 1 digital por cada eje. (Numérico)
13	El sistema simula un acelerómetro MEMS en la FPGA del myRIO	Grado de cumplimiento, FPGA del myRIO simula el sistema, se usa la FPGA del myRIO o no (cumple o no cumple / binario)
14	El sistema simulado utiliza API para comunicarse, medir y probar el chip	Grado de cumplimiento, API se comunica, mide y prueba el chip, se usa API o no (cumple o no cumple / binario)
15	El sistema simulado posee una tarjeta electrónica con las conexiones necesarias para su funcionamiento	Grado de cumplimiento, Tarjeta electrónica posee conexiones necesarias para su funcionamiento, se tiene tarjeta o no (cumple o no cumple / binario)
16	El sistema simulado está dentro de un encapsulado que lo aisle y resguarde	Grado de cumplimiento, Sistema simulado está dentro de un encapsulado que lo aisle y resguarde, está dentro de un encapsulado o no (cumple o no cumple / binario)

toma en cuenta rangos de valores dentro de los cuales el parámetro puede ser:

- Inaceptable: El valor es crítico y no permitiría la creación de un prototipo aceptable ya que contradice directamente una necesidad del cliente o un principio fundamental de funcionamiento del sistema.
- Marginal: Se tiene un valor que no cumple la expectativa ideal del parámetro, sin embargo no afecta gravemente el sistema y se puede tomar como aceptable.
- Ideal: Es el valor buscado o al que se aspira a llegar.

El valor ideal, normalmente implica el detrimento de otros valores y por esto el diseño de experimentos se vuelve complejo e implica una búsqueda de una solución que contenga

Tabla 3.4: Continuación tabla 3.3 Relación necesidad-métrica.

	Necesidad	Métrica
17	El sistema simulado posee un manual de operación con los modo de trabajo y las especificaciones del sistema	Grado de cumplimeinto, Manual de operación con los modo de trabajo y las especificaciones del sistema, posee un manual o no (cumple o no cumple / binario)
18	El sistema simulado valida el chip mediante una secuencia	Chip validado mediante secuencia, Cumplimiento de la lectura del chip, modelo del chip y el número de serie (TEST / Binario Pasa no pasa)
19	El sistema de caracterización considera los errores de ruido y vibraciones en el sistema	Grado de cumplimiento, Ruido y vibraciones son consideradas en el sistema aislado y controlado del medio, se considera el ruido y vibraciones o no (cumple o no cumple / binario)
20	El sistema de caracterización está aislado del medio ambiente y en un medio controlado para la medición	Grado de cumplimiento, Ruido y vibraciones son consideradas en el sistema aislado y controlado del medio, se considera el ruido y vibraciones o no (cumple o no cumple / binario)
21	El sistema de caracterización varía la aceleración hasta los rangos de trabajo establecidos.	Cantidad de rangos de aceleración que el sistema logra (3 rangos de aceleración diferentes (g))
22	El sistema de caracterización guarda los datos obtenidos y permite accesarlos	Grado de cumplimiento, Sistema de medición guarda los datos de aceleración obtenidos, guarda los datos o no (cumple o no cumple / binario)
23	El sistema de caracterización se puede utilizar repetidamente y se puede cambiar los sensores a utilizar	Grado de cumplimiento, Sistema de medición se puede reusar y cambiar acelerómetros, se puede reutilizar o no (cumple o no cumple / binario)
24	Los acelerómetros a utilizar están disponibles en Costa Rica y el tiempo de adquisición es menor o igual a 2 semanas	Tiempo de adquisición que tardaría en llegar el sensor es menor o igual a 2 semanas (Tiempo en días hábiles)
25	El precio de los acelerómetros a utilizar no sobrepasa los 15 dólares la unidad.	Costo de cada acelerómetro no debe superar los \$15 por unidad.

valores marginales para así cumplir con la mayor cantidad de necesidades, de la mejor manera posible.

En la figura 3.4 se muestra un resumen de la relación métrica necesidad y en la figura 3.5 se muestran los valores objetivo, para los parámetros establecidos.

Número Métrica	Número Necesidad	Métrica	Importancia
1	1	Grado de semejanza entre simulación y acelerómetro real ante cambios en temperatura y modificaciones en funciones	5
2	2	Sistema varía ante cambios temperatura y permite modificar funciones	5
3	3	Rango de temperatura de trabajo	5I
4	4	Cantidad de rangos de trabajo de la simulación	5
5	5	Valor de tensión entrada del sistema	5
6	6, 18	Chip validado mediante secuencia: chip presente, modelo del chip y número de serie	5
7	7, 8	Validar modo trabajo y modo falla para cada salida	5
8	9	Valor tensión de salida del sistema es compatible con TTL	5
9	10	Salida es analógica y digital para cada salida	5
10	11	Datos y estado chip se visualizan en la interfaz	5
11	12	Cantidad de canales individuales	5
12	13	FPGA del myRio simula el sistema	5
13	14	API se comunica, mide y prueba el chip	5
14	15	Tarjeta electrónica posee conexiones necesarias para su funcionamiento	5
15	16	Sistema simulado está dentro de un encapsulado que lo aisle y resguarde	5
16	17	Manual de operación con los modo de trabajo y las especificaciones del sistema	5
17	19, 20	Ruido y vibraciones son consideradas en el sistema de caracterización aislado y controlado del medio	4
18	21	Cantidad de rangos de aceleración que el sistema de caracterización logra	5
19	22	Sistema de medición guarda los datos de aceleración obtenidos	4
20	23	Sistema de medición se puede reusar y cambiar acelerómetros	4
21	24	Tiempo de adquisición es menor o igual a 2 semanas	4
22	25	Costo de cada acelerómetro es menor o igual a 15 dólares	5I

Figura 3.4: Resumen relación métrica necesidad.

Métrica	Importancia	Unidades	Valor Marginal	Valor Ideal
Grado de semejanza entre simulación y acelerómetro real ante cambios en temperatura y modificaciones en funciones	5	Documento / Prueba	N.A.	N.A.
Sistema varía ante cambios temperatura y permite modificar funciones	5	Documento / Prueba	N.A.	N.A.
Rango de temperatura de trabajo	5I	Número	-40 a 60	-40 a 125
Cantidad de rangos de trabajo de la simulación	5	Número	<2	>3
Valor de tensión entrada del sistema	5	Voltios	3.3V a 4V	3.3V y 5V
Chip validado mediante secuencia: chip presente, modelo del chip y número de serie	5	Binario	Pasa	Pasa
Validar modo trabajo y modo falla para cada salida	5	Documento / Prueba	N.A.	N.A.
Valor tensión de salida del sistema es compatible con TTL	5	Voltios	2.2 a Vcc	4.75V a 5.25V
Salida es analógica y digital para cada salida	5	Binario	Cumple	Cumple
Datos y estado chip se visualizan en la interfaz	5	Binario	Cumple	Cumple
Cantidad de canales individuales	5	Número	3	6
FPGA del myRio simula el sistema	5	Binario	Cumple	Cumple
API se comunica, mide y prueba el chip	5	Binario	Cumple	Cumple
Tarjeta electrónica posee conexiones necesarias para su funcionamiento	5	Binario	Cumple	Cumple
Sistema simulado está dentro de un encapsulado que lo aisle y resguarde	5	Binario	Cumple	Cumple
Manual de operación con los modo de trabajo y las especificaciones del sistema	5	Documento	N.A.	N.A.
Ruido y vibraciones son consideradas en el sistema de caracterización aislado y controlado del medio	4	Binario	Cumple	Cumple
Cantidad de rangos de aceleración que el sistema de caracterización logra	5	Número	3	4
Sistema de medición guarda los datos de aceleración obtenidos	4	Binario	Cumple	Cumple
Sistema de medición se puede reusar y cambiar acelerómetros	4	Binario	Cumple	Cumple
Tiempo de adquisición es menor o igual a 2 semanas	4	Días hábiles	18	7
Costo de cada acelerómetro es menor o igual a 15 dólares	5I	\$USD	<20	<15

Figura 3.5: Determinación de los valores objetivos.

3.4.7 Descomposición funcional del problema

Esta descomposición del problema en subproblemas es necesaria, ya que simplifica la búsqueda posterior de conceptos. Además, al dividir un problema se puede puntualizar de mejor manera en cada una de las necesidades de forma individual.

3.4.8 Proceso de búsqueda de Información

Para el simulador

Se procede a buscar fuentes externas de información por ejemplo: sistemas existentes que simulen un acelerómetro. Se realiza una búsqueda bajo los siguientes términos:

- simulador y acelerómetro
- generador de señales de acelerómetro

De esta búsqueda se obtienen productos que simulan la respuesta de un acelerómetro. Estos son sistemas que tienen como salida una señal que represente una aceleración. Se investiga sobre dispositivos similares a un simulador de acelerómetro en el mercado actual. Los elementos encontrados que cumplen estas características son:

- Hansford Sensors HS-661: equipo portátil alimentado por batería diseñada para permitir a los ingenieros de instalación probar sistemas de monitorización de vibraciones que emplean acelerómetros industriales de corriente constante de 100 mV/g. El equipo proporciona dos niveles conmutados de señales de velocidad de vibración simuladas, 5 mm/s y 20 mm/s, lo que permite la verificación del monitor de vibración y los circuitos de alarma. Los niveles de vibración conmutados y la potencia del acelerómetro del sistema de monitorización se indican mediante LEDs [41]. En la figura 3.6 se muestra el simulador.



Figura 3.6: Simulador de acelerómetro Hansford HS-661 [41]

- Direct Industry - Endevco 4830B: es un generador de señales de mano a batería diseñado específicamente para simular la salida eléctrica de los tipos comunes de acelerómetros. El instrumento contiene un oscilador de alta precisión con un nivel de salida ajustable y es ideal para configurar, probar y el diagnóstico de fallos en los sistemas de adquisición de datos, sistemas de pruebas ambientales o simplemente como un flexible generador de señales. El 4830B proporciona señales de salida de CA que imitan las de los acelerómetros de modo de voltaje (genérico IEPE, Iso-tron®, etc.) o acelerómetros de modo de carga (tanto de un solo extremo como diferenciales). Las salidas de la simulación están convenientemente escaladas en unidades de aceleración, es decir, "g", como mV/g (milivoltio) o pC/g (picoculombios) según corresponda, aunque las salidas pueden configurarse para ser proporcional a las unidades de velocidad o desplazamiento. El 4830B también tiene la capacidad de analizar una señal de vibración usando una función de Transferencia Rápida de Fourier. Este permite introducir una señal de vibración proporcional al voltaje en el instrumento, proporcionando así una indicación tanto de la frecuencia como de la magnitud [11]. En la figura 3.7 se muestra el simulador.



Figura 3.7: Simulador de acelerómetro Endevco 4830B [11]

Para el sistema de caracterización

Búsqueda externa

En el caso del sistema de caracterización se buscan métodos de caracterización de acelerómetros, con el fin de conocer y poder replicar la respuesta que tienen los mismos.

Centrifugación Una centrífuga es una máquina que utiliza fuerza para mover el contenido lejos del eje de rotación. Se utiliza normalmente para separar sustancias de diferente densidad, remover líquidos o simular los efectos gravitacionales. Algunos juegos rotatorios utilizan esta fuerza para mantener a los usuarios sujetos a las paredes del sistema [2]. En la NASA (National Aeronautics and Space Administration), utilizan una centrífuga que permite producir fuerzas de hasta 20 veces la gravedad de la tierra (20g) con 8.84m de

radio, para realizar investigación y entrenar para resolver problemas relacionados con los efectos de la aceleración en sistemas. Este sistema permite variar el radio con el fin de estudiar los cambios en los gradiente de aceleración o efecto coriolis. Además, se pueden colocar diversas masa de prueba para poder obtener todos los datos a la vez. Su velocidad máxima es de 50 RPM, para un peso máximo de 1200 lb y una aceleración máxima de 20g [3]. En la figura 3.8 se muestra la centrífugadora mencionada.

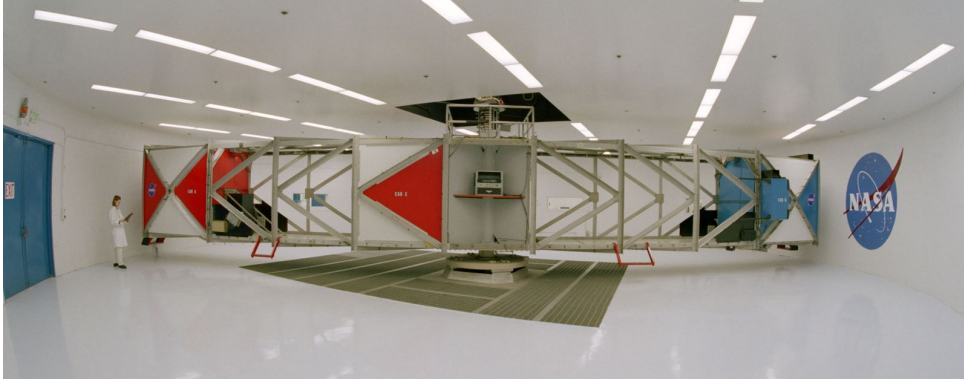


Figura 3.8: Centrífuga de 20g con 8.84m de radio [3]

Búsqueda interna

Sistema Mecánico

- **Rieles de baja fricción**, permiten desplazar linealmente elementos de manera suave. Se pueden acoplar elementos como rodamientos lineales que facilitan el desplazamiento con suavidad. Existen rieles con diversas geometrías, pueden ser redondos, rectangulares o acanalados.



Figura 3.9: Ejemplo de rieles de baja fricción [8]

- **Motores**, permiten convertir la energía eléctrica en movimiento, el cual puede ser rotacional o lineal. En este caso se implementaría el motor para acelerar el sistema de caracterización y poder variar la aceleración ejercida sobre el acelerómetro MEMS.
- **Impulsión con aire a presión**, se puede utilizar el aire a presión para impulsar objetos, esto mediante un compresor y un canal o ducto sellado que permita aplicar la presión de aire sobre el elemento a impulsar.

Sistema de adquisición de datos Búsqueda externa

Con el fin de poder controlar los diversos elementos del sistema de caracterización se proponen controladores con los que se ha tenido experiencia anteriormente.

- **Raspberry Pi** es un procesador ultra compacto con procesador Broadcom BCM2837B0 Cortex-A53 64-bit SoC @ 1,4 GHz, una RAM de 1 GB LPDDR2 SDRAM, bluetooth 4.2 BLE, Wi-Fi Dual Band b/g/n/ac, lector de tarjetas micro-SD, y un header con 40 pines de salidas y entradas digitales (General Purpose Input Output GPIO), además posee 4 puertos USB 2.0 y ocupa una alimentación de 5V / 2,5 A y un peso de 50g. Este se puede programar mediante la plataforma de Python [33].
- **Feather Huzzah ESP8266**, posee un microcontrolador ESP8266, un voltaje de operación: 3.3V, comunicación USB (CP2104), I²C, SPI, posee 9 pines de salidas y entradas digitales, además de 1 pin de entrada analógica, una memoria Flash de 4 MB y un peso de 9.7g. Este se puede programar mediante la plataforma de Arduino [1].
- **myRio**, como se mencionó en la sección 2.4.3 este dispositivo permite realizar la adquisición de datos y se puede programar mediante la plataforma de LabView.

De entre estos controladores se descarta el Arduino, ya que según las recomendaciones del mentor este método se sale de los rangos de precisión y confiabilidad que busca la implementación para la empresa.

3.4.9 Construcción de conceptos

Construcción de conceptos propios a partir de la búsqueda interna realizada y el asesoramiento de la búsqueda externa. Se proponen 7 conceptos de características variadas y combinadas.

1. Centrifugadora con Raspberry Pi
2. Centrifugadora con Feather Huzzah
3. Centrifugadora con myRio
4. Riel de baja fricción con vehículo impulsado por un motor mediante Raspberry Pi

5. Riel de baja fricción con vehículo impulsado por un motor mediante myRio
6. Riel de baja fricción con vehículo impulsado por aire a presión mediante Raspberry Pi
7. Riel de baja fricción con vehículo impulsado por aire a presión mediante myRio

3.4.10 Establecimiento y selección del candidato a prototipo

Dados los 7 conceptos mencionados anteriormente, se procede a valorar cada uno de ellos en función de cada parámetro. La descripción de los criterios de selección se mencionan en la tabla 3.5. En la primer fase de selección se asignó al concepto 1 como la referencia, en la figura 3.10 se muestra el desarrollo de la primer ronda de selección. A partir del mismo se valoran los parámetros de los demás conceptos en comparación a este. El resultado del mismo se reporta, ya sea como mejor (+), igual (0) o peor(-) en comparación con el concepto de referencia.

En la figura 3.11 se muestra la selección del prototipo a desarrollar para la caracterización del sensor. En este se realiza una evaluación ponderada de cada criterio, para obtener una puntuación y así elegir cual prototipo a desarrollar. Los pesos de cada criterio se asignan según la importancia dada por el cliente y las consideraciones del diseñador. La calificación es un valor de 1 a 5, donde 1 es una nota baja y 5 es una nota perfecta. El prototipo a desarrollar corresponde al que tenga el mayor puntaje y sea el más cercano a 5.

Criterios de Selección	Concepto						
	1 (Referencia) Centrifugadora con Raspberry Pi	2 Centrifugadora con Feather Huzzah	3 Centrifugadora con myRio	4 Riel de baja fricción con vehículo impulsado por motor mediante Raspberry Pi	5 Riel de baja fricción con vehículo impulsado por motor mediante myRio	6 Riel de baja fricción con vehículo impulsado por aire a presión mediante Raspberry Pi	7 Riel de baja fricción con vehículo impulsado por aire a presión mediante myRio
1 Distancia de desplazamiento es fácil de variar y ensamblar	0	0	0	-	-	-	-
2 Configurable en 3 o más rangos de aceleración	0	0	0	-	0	-	0
3 Peso del sistema de sensores	0	+	-	-	-	-	-
4 Sistema se puede aislar del medio	0	0	0	-	0	-	0
5 Facilidad de controlar velocidad y aceleración del sistema	0	0	+	0	+	0	+
6 Ruido y vibraciones del sistema es bajo	0	0	-	-	-	-	-
7 Facilidad reusar y cambiar acelerómetros	0	0	0	0	+	0	+
8 Facilidad de programar la adquisición de datos	0	0	-	0	-	0	-
9 Facilidad acceder y almacenar los datos de aceleración	0	-	0	0	0	0	0
Suma +	0	1	1	0	2	0	2
Suma 0	10	8	6	4	3	4	3
Suma -	0	1	3	6	5	6	5
Evaluación Neta	0	0	-2	-5	-3	-5	-3
Lugar	1	2	3	5	4	5	4
¿Continuar?	SI	SI	SI	NO	NO	NO	NO

Figura 3.10: Primer ronda de selección de prototipos de solución

3.4.11 Factores de influencia

Con base en las métricas previamente establecidas, se adecuaron los rangos y factores de influencia como se muestra en la figura 3.12. En esta tabla se da una justificación de factores, en este se describe el porqué de cada factor.

Tabla 3.5: Descripción de los criterios de selección.

	Criterios de Selección	Descripción
1	Distancia de desplazamiento es fácil de variar y ensamblar	La distancia que se debe desplazar el sensor para medir la aceleración del rango se puede variar de ser necesario y el tiempo de ensamblar este cambio es menor a 5min.
2	Configurable en 3 o más rangos de aceleración	El sistema de desplazamiento se puede utilizar para 3 o más rangos de aceleración
3	Peso del sistema de sensores	El sistema de sensores tiene un peso menor a 250kg, contando el controlador, los sensores y componentes necesarios para su funcionamiento
4	Sistema se puede aislar del medio	Durante el proceso de toma de datos se puede aislar el sistema del medio
5	Facilidad de controlar velocidad y aceleración del sistema	El control de velocidad y aceleración del sistema es controlable y configurable con facilidad
6	Ruido y vibraciones del sistema es bajo	El ruido introducido por las posibles vibraciones del sistema mecánico es bajo
7	Facilidad reusar y cambiar acelerómetros	El sistema permite cambiar los acelerómetros de ser necesario caracterizar estos nuevos sensores
8	Facilidad de programar la adquisición de datos	El sistema de adquisición de datos no requiere de mucha experiencia para su programación
9	Facilidad acceder y almacenar los datos de aceleración	El sistema de adquisición de datos no requiere de códigos complejos ni muchos recursos para poder almacenar los datos recopilados

3.4.12 Definición de las pruebas a realizar

Para la verificación de las métricas expuestas, se propone realizar las pruebas de la figura 3.12. En este se tiene una prueba para cada factor, con lo que se permitirá verificar los rangos de trabajo dados por los valores marginales e ideales de cada necesidad. Se debe recalcar que las pruebas de los factores de influencia pueden ser aplicados únicamente a necesidades que posean una prueba numérica o un rango de valores para validarlo. Las pruebas a realizar de la figura 3.12 son las siguientes:

- Comparar la respuesta del acelerómetro real y el simulado bajo una temperatura

		Concepto						
		1 Centrifugadora con Raspberry Pi		2 Centrifugadora con Feather Huzzah		3 Centrifugadora con myRio		
Criterios de Selección		Peso	Calificación	Evaluación Ponderada	Calificación	Evaluación Ponderada	Calificación	Evaluación Ponderada
1	Distancia de desplazamiento es fácil de variar y ensamblar	21	5	1.05	5	1.05	5	1.05
2	Configurable en 3 o más rangos de aceleración	28	5	1.4	5	1.4	5	1.4
3	Peso del sistema de sensores	7	4	0.28	5	0.35	3	0.21
4	Sistema se puede aislar del medio	5	5	0.25	5	0.25	5	0.25
5	Facilidad de controlar velocidad y aceleración del sistema	10	5	0.5	5	0.5	5	0.5
6	Ruido y vibraciones del sistema es bajo	3	4	0.12	5	0.15	3	0.09
7	Facilidad reusar y cambiar acelerómetros	5	5	0.25	4	0.2	5	0.25
8	Facilidad de programar la adquisición de datos	10	5	0.5	3	0.3	5	0.5
9	Facilidad acceder y almacenar los datos de aceleración	11	5	0.55	3	0.33	5	0.55
Total Puntos		100		4.9		4.53		4.8
Lugar				1		3		2
¿Continuar?				Desarrollar		No		No

Figura 3.11: Selección de prototipo de solución a desarrollar

específica.

- Probar la simulación en una temperatura con respuesta conocida dentro del rango.
- Verificación mediante el conteo directo de los rangos.
- Verificación mediante la medición directa la tensión de alimentación del sistema.
- Verificación en la interfaz del indicador del modelo, número de serie y presencia del chip.
- Verificación de las salidas ante el modo de trabajo y el modo de falla.
- Verificación mediante la medición directa la tensión de salida.
- Verificación mediante la medición directa de cada salida.
- Verificación en la interfaz del indicador de datos y estado del chip.
- Conteo directo del número de canales individuales.
- Verificación del uso de recursos en el myRio.
- La FPGA debe poder enviar y recibir datos mediante la API.
- Existencia de una tarjeta que contenga los elementos necesarios para el funcionamiento del sistema.
- Existe un encapsulado con la FPGA y la tarjeta electrónica que simulan el sistema.
- Se realizan varias mediciones en el sensor y se comparan las respuestas para extraer las variaciones por ruido.
- Conteo directo del número de rangos de aceleración que puede lograr el sistema de caracterización.
- Verificación de que se pueda acceder los datos luego de caracterizar el sensor.

	Factores de influencia	Importancia	Unidades	Valor / Rango	Justificación	Prueba a realizar
1	Grado de semejanza entre simulación y acelerómetro real ante cambios en temperatura y modificaciones en funciones	5	Documento / Prueba	N.A.	Se busca que el acelerómetro simulado tenga una respuesta lo más cercana posible a la de un acelerómetro real, por lo que se establece una comparación entre ambos elementos	Comparar la respuesta del acelerómetro real y le simulado bajo una temperatura específica
2	Sistema variará ante cambios temperatura y permite modificar funciones	5	Documento / Prueba	N.A.	Como parte de los requerimientos el sistema simulado debe responder ante cambios de temperatura y su respuesta varía ante modificaciones en el funcionamiento	
3	Rango de temperatura de trabajo	5I	Númérico	-40 < x < 125	Los rangos de trabajo de los acelerómetros son comúnmente estos, dependiendo el modelo y la aplicación específica estos rangos cambian. En este caso se busca tener temperaturas positivas y negativas para ejecutar la simulación	Se prueba la simulación en la temperatura máxima, mínima y 2 valores de temperatura con respuesta conocida dentro del rango
4	Cantidad de rangos de trabajo de la simulación	5	Númérico	2 < x < 3	La simulación debe poder ejecutarse sobre al menos 3 rangos de trabajo según los requerimientos del cliente	Verificación mediante el conteo directo de los rangos
5	Valor de tensión entrada del sistema	5	Volios	3.3V < x < 6V	La plataforma donde se simula el sistema debe ser alimentada con una fuente dentro de este rango de tensiones	Verificación mediante la medición directa la tensión de alimentación del sistema
6	Chip validado mediante secuencia: chip presente, modelo del chip y número de serie	5	Binario	Pasa	Validación del chip para que el usuario conozca los datos del mismo y su estado durante la ejecución	Verificación en la interfaz del indicador del modelo, número de serie y presencia del chip
7	Validar modo trabajo y modo falla para cada salida	5	Documento / Prueba	N.A.	Ya que los acelerómetros triaxiales poseen 3 salidas, se debe corroborar el funcionamiento de las mismas en los diversos modos	Verificación de las salidas ante el modo de trabajo y el modo de falla
8	Valor tensión de salida del sistema es compatible con TTL	5	Volios	4.75V < x < 5.25V	Las salidas deben acoplarse a otros sistemas, por lo que los rangos en las tensiones de salida deben ser compatibles con tecnología TTL, y así permitir el acople entre dispositivos	Verificación mediante la medición directa la tensión de salida
9	Salida es analógica y digital para cada salida	5	Binario	Cumple	Se debe tener una salida analógica y una salida digital para cada eje del sensor, con el fin de que el usuario pueda conectarlas a dispositivos externos y utilizar sus respuestas.	Verificación mediante la medición directa de cada salida
10	Datos y estado chip se visualizan en la interfaz	5	Binario	Cumple	El usuario debe conocer los datos del chip y su estado a través de la interfaz de usuario	Verificación en la interfaz del indicador de datos y estado del chip
11	Cantidad de canales individuales	5	Númérico	3 < x < 6	Se debe tener 1 canal para cada eje del acelerómetro, además se debe tener una salida analógica y una salida digital por eje	Conteo directo del número de canales individuales
12	FPGA del myRio simula el sistema	5	Binario	Cumple	El simulador corre dentro de la FPGA del myRio y utiliza una interfaz para comunicarse con el usuario	Verificación del uso de recursos en el myRio
13	API se comunica, mide y prueba el chip	5	Binario	Cumple	La API permite comunicarse con la FPGA para enviar y recibir información, además permite que el usuario vea la respuesta del sistema.	La FPGA debe poder enviar y recibir datos mediante la API
14	Tarjeta electrónica posee conexiones necesarias para su funcionamiento	5	Binario	Cumple	Se utiliza una tarjeta electrónica para poder regular las salidas y configurar las de ser necesario	Existencia de una tarjeta que contenga los elementos necesarios para el funcionamiento del sistema
15	Sistema simulado está dentro de un encapsulado que lo aisle y resguarde	5	Binario	Cumple	La FPGA está dentro de un encapsulado para evitar daños en la tarjeta electrónica	Existe un encapsulado con la FPGA y la tarjeta electrónica que simulan el sistema
16	Ruido y vibraciones son consideradas en el sistema de caracterización aislado y controlado del medio	4	Binario	Cumple	Se busca que la caracterización del acelerómetro se de bajo condiciones controladas con el fin de tener una función que lo describa de la manera más precisa posible	Se realizan varias mediciones en el sensor y se comparan las respuestas para extraer las variaciones por ruido.
17	Cantidad de rangos de aceleración que el sistema de caracterización logra	5	Númérico	2 < x < 4	El sistema debe poder acelerar el sensor el al menos 4 aceleraciones diferentes para poder trazar la curva de medición	Conteo directo del número de rangos de aceleración que puede lograr el sistema de caracterización
18	Sistema de medición guarda los datos de aceleración obtenidos	4	Binario	Cumple	Los datos deben visualizarse para poder revisarlos, compararlos y tabularlos	Verificación de que se pueda acceder los datos luego de caracterizar el sensor
19	Sistema de medición se puede reusar y cambiar acelerómetros	4	Binario	Cumple	El sistema puede usar utilizado reiteradamente además se pueden acoplar diferentes modelos de sensores	Conteo de la cantidad de veces que se utiliza el sistema y facilidad de acoplar u desacoplar sistema de sensores
20	Tiempo de adquisición es menor o igual a 2 semanas	4	Días	7 < x < 18	Se busca no tener retrasos con la entrega de los sensores a caracterizar, por lo que establece el periodo máximo de entrega en 15 días.	Conteo de días que dura en llegar los sensores
21	Costo de cada acelerómetro es menor o igual a 15 dólares	5I	\$USD	15 < x < 20	El rango está directamente asociado al costo de cada sensor y el máximo establecido por unidad.	Precios y cotización de los sensores

Figura 3.12: Factores de influencia y descripción de pruebas a realizar

- Conteo de la cantidad de veces que se utiliza el sistema y facilidad de acoplar u desacoplar sistema de sensores.

- Conteo de días que dura en llegar los sensores.
- Precios y cotización de los sensores.

Capítulo 4

Implementación del sistema de caracterización y del simulador

4.1 Sistema de caracterización

4.1.1 Calibración de acelerómetros

En el libro Mediciones Mecánicas (Mechanical Measurements) [44], se mencionan diversos métodos de calibración de acelerómetros, los cuales pueden ser estáticos o periódicos, dentro de los métodos estáticos está la centrifugación, en el cual se pueden determinar valores ilimitados de aceleración estática mediante una mesa rotativa. Este método utiliza la componente de la aceleración hacia el centro de la mesa para realizar la calibración. Se tiene la ecuación 4.1, donde a_n es la aceleración de la masa sísmica, r es el radio desde el centro de la mesa giratoria hasta el centro de la masa sísmica y f corresponde a la velocidad de rotación de la mesa en revoluciones por segundo (rev/s). En este método de calibración, la única desventaja recae en el alambrado de los elementos de medición, ya que se tiene una base rotativa.

$$a_n = r * (2\pi * f)^2 \left[\frac{m}{s^2} \right] \quad (4.1)$$

En esta aplicación la aceleración tangencial a_t posee una variación la aceleración en magnitudes muy pequeñas en comparación con la aceleración centrífuga. Por lo tanto este valor se desprecia durante la calibración de sensores. Se tiene la ecuación 4.2, donde a_t es la aceleración tangencial del sistema rotativo, α corresponde a la aceleración angular la cual se calcula con la ecuación 4.3, donde w corresponde a la velocidad angular en rad/s y t corresponde al tiempo.

$$a_t = \alpha * r \left[\frac{m}{s^2} \right] \quad (4.2)$$

$$\alpha = \frac{w2 - w1}{t2 - t1} = \frac{\Delta w}{\Delta t} \left[\frac{rad}{s^2} \right] \quad (4.3)$$

Por último, la aceleración de la gravedad ($1g = 9.81m/s^2$) trabaja de manera constante sobre el eje paralelo del sensor, por lo que se puede utilizar este valor como calibración también; pero se debe conocer con exactitud la fuerza que ejerce la gravedad en ese lugar.

4.1.2 Sistema de aceleración centrífuga

Con el fin de caracterizar el sensor MPU-6050 que se logró conseguir en el mercado nacional, se desarrolla el sistema de aceleración centrífuga, para esto se utilizó un controlador de motor que permitiera variar la velocidad de rotación de la mesa y que pueda ser controlado por un microcontrolador. Este debe de ser capaz de acelerar el instrumento de medición hasta $16g$ aproximadamente, el cual es el rango máximo de medición del sensor. Esto se puede lograr al variar el largo del radio del sistema y cambiando la velocidad del motor. En la figura 4.1 se muestra un ensamble con la propuesta del sistema de centrifugación. En el apéndice A.4 se tiene los planos de los diversos elementos del ensamble.

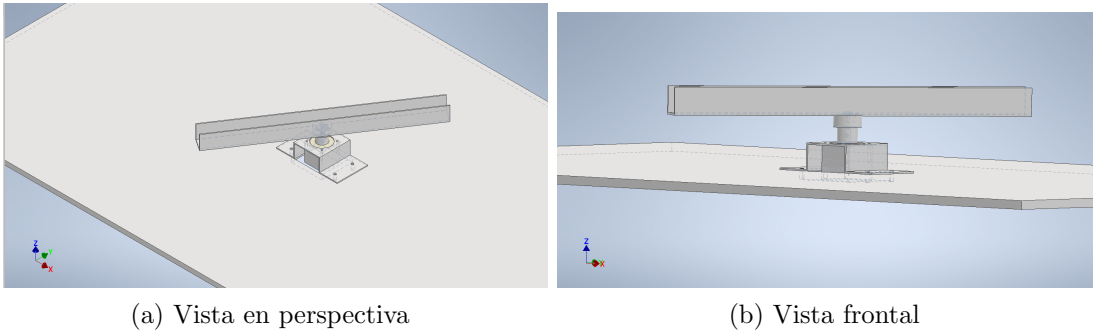


Figura 4.1: Diseño propuesto para el sistema de centrifugación.

Se utilizan elementos con los cuales se ha tenido experiencia previamente y se pueden encontrar en el mercado nacional [38].

- Motor a pasos **Stepper Motor with Cable [ROB-09238] (NEMA 17)**, el cual posee dos fases y un paso de 1.8° , requiere una fuente de alimentación de 12 V, consume 0.33 A y tiene un torque de $2.3kg \cdot cm$. Dado que la tensión de alimentación del motor es muy alta comparada con la tensiones de control de los microcontroladores.
- Controlador de motores a pasos **TB6600 Stepper Motor Drive Controller** el cual permite controlar el motor a pasos mediante señales de 5V, en este caso se utiliza un **Arduino Uno**, ya que sus salidas son de 5V. Con este controlador es posible variar la velocidad del sistema al modificar la cantidad de micro pasos del

motor, y controlar motores con alimentaciones desde los 9 hasta los 42VDC. El circuito de conexión se muestra la figura 4.2

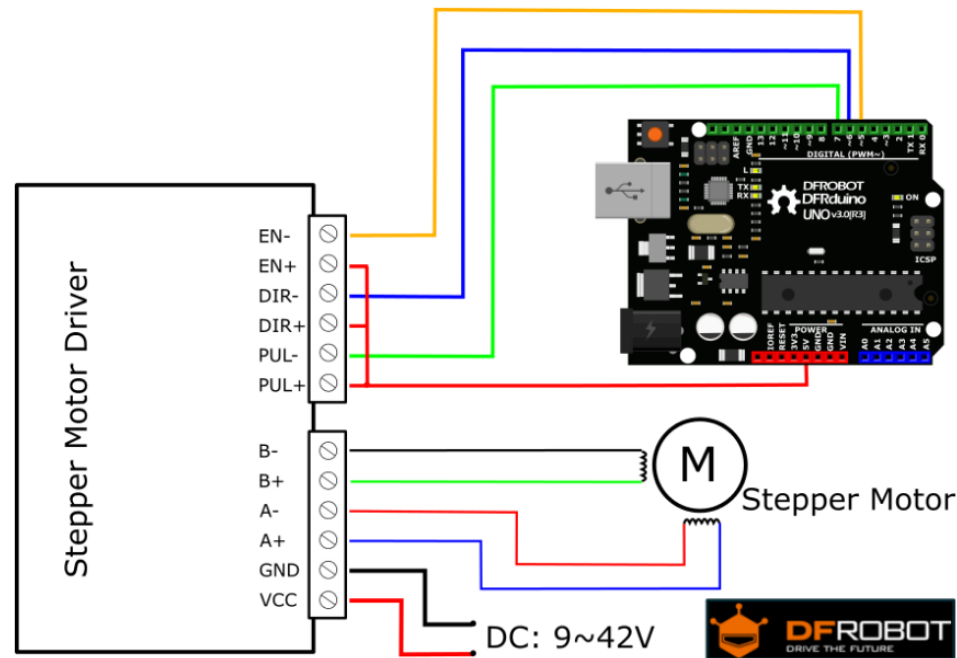


Figura 4.2: Diagrama de conexión entre el Arduino Uno, el TB6600 y el motor NEMA 17 [7]

- Fuente **S-240-24 Switch Power Supply 24VDC 10A** para alimentar el motor a pasos a través del controlador
- Un **LM2596 DC-DC Buck Converter Step-Down Power Module Output 1.25V-35V** como conversor de nivel para poder alimentar el Arduino Uno con 5V desde la fuente de 12V.

Todos estos componentes se lograron adquirir localmente en [38]. Con estos componentes y el código en el Arduino Uno, se construye una prueba de concepto de la centrifugadora. Sobre este se acopla el sistema de toma de datos del sensor **MPU-6050** mediante la **Raspberry Pi 3** con el fin de acelerar el sensor y recopilar datos durante el proceso. Se acopla un En la figura 4.3 se muestra el primer circuito de control implementado para la centrifugadora.

Se utiliza una base de madera de 120X75cm, en la cual se hace un agujero cuadrado en el centro para acoplar el motor a pasos mediante tornillos. Se utiliza una barra de aluminio para manufacturar un acople entre el motor y la barra en forma de U en la cual se colocará el sensor. Los planos de manufactura del acople se encuentran en el apéndice A.5. Se decide utilizar la viga en forma de U en vez de un plato giratorio, ya que el plato es pesado y el balanceo del plato es complejo comparado con el de una viga. Además, al tener forma de U, se pueden colocar componentes como las baterías dentro de la viga.

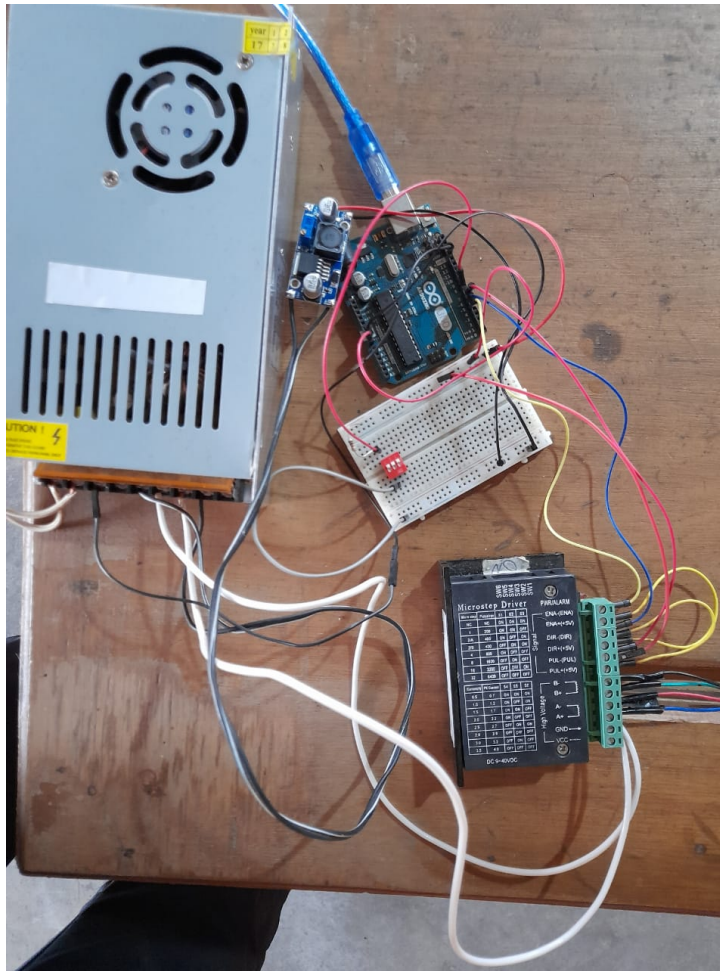


Figura 4.3: Primer modelo del sistema del circuito de la centrifugadora

4.1.3 Verificación de velocidades con Tacómetro

Dado que se debe conocer la velocidad a la cual está girando el sistema de caracterización, se realiza un código de prueba para verificar las velocidades máximas y mínimas de la centrifugadora. Esto permitirá tener valores de referencia de las velocidad antes de la toma de datos con el acelerómetro. Se utiliza un **Smart Sensor Digital Photo Tachometer modelo AR926** para realizar una medición inicial de las velocidades del sistema y asegurar que pueda llegar a 300RPM. Este tacómetro digital sin contacto utiliza un láser y cintas reflectivas para tomar las mediciones de las revoluciones por minuto. Posee una precisión de $\pm(0.05\% + 1d)$, con una resolución de 0.1 RPM para mediciones entre 0,5 y 999.9 RPM; y 1RPM para mediciones de más de 1000RPM. A través del código en el Arduino que se encuentra en el apéndice A.6, se varía la velocidad del motor al cambiar el valor de los microsegundos donde el sensor está encendido y apagado (Microsegundos ON/OFF), con esto se obtienen los resultados que se muestran en la tabla 4.1

Para calcular la precisión de la medición se utiliza la ecuación 4.4, en el cual se usa una resolución de 0.1 RPM para mediciones menores a 999.9 RPM

$$Presicion = \frac{0.05}{100} * RPM + 0.1 \quad (4.4)$$

Tabla 4.1: Velocidades del sistema de caracterización.

Microsegundos ON/OFF	Max RPM		Promedio RPM	Presición ($\pm$)
800	46.6	46.5	46.55	0.12
700	53.0	53.0	53.00	0.13
600	61.7	61.7	61.70	0.13
500	73.9	73.8	73.85	0.14
400	92.1	92.1	92.10	0.15
300	122.1	122.0	122.05	0.16
200	181.0	181.0	181.00	0.19
100	350.8	350.5	350.65	0.28

4.1.4 Sistema de toma de datos con Raspberry Pi

De la etapa de establecimientos y selección de candidatos a prototipo se propuso utilizar la Raspberry Pi como microcontrolador para el proceso de toma de datos. Este se programa mediante **Python** para poder recopilar mediciones de la aceleración en el sensores MPU-6050.

Se propone utilizar la librería de Python "mpu6050-raspberrypi" para realizar la lectura del acelerómetro, esta librería se obtiene de [4] y se instala a través de la terminal mediante los comandos:

```
sudo apt install python3-smbus
pip install mpu6050-raspberrypi
```

Se configura la Raspberry Pi para que guarde los datos de aceleración dentro de una USB con el fin de poder extraerlos con facilidad y no tener que conectar la Raspberry Pi a ningún monitor ni pantalla LCD. Los datos de la USB se pueden visualizar en la pantalla de una computadora a través de Excel en el cual se pueden procesar los datos obtenidos y determinar las ecuaciones que caracterizan al sensor. En la Raspberry Pi se debe habilitar la comunicación I^2C para poder realizar la lectura del sensor, para activar la comunicaión se escribe en la terminal:

1. **sudo raspi-config** + enter
2. Seleccionar "**Opciones de Interfaz**" + enter
3. Habilitar comunicación I^2C + enter + armar
4. Verificar la conexión del sensor mediante "**i2cdetect -y 1**", este comando indica las direcciones I^2C conectadas a la Raspberry Pi

En el apéndice A.7 se muestra el código de Python que permite recopilar datos del sensor y medir las RPM del sistema mientras se está acelerando. En este código final de la implementación no se utiliza la librería directamente, se utiliza otro código que contiene partes de la librería, ya que se debe reconfigurar el rango de trabajo del sensor, como se menciona en la sección 5.4 que se describe en el siguiente capítulo.

En la figura 4.4 se muestra el primer circuito para probar la toma de datos en la Raspberry Pi. Este circuito debe ser soldado en una placa perforada y acoplada al sistema de aceleración.

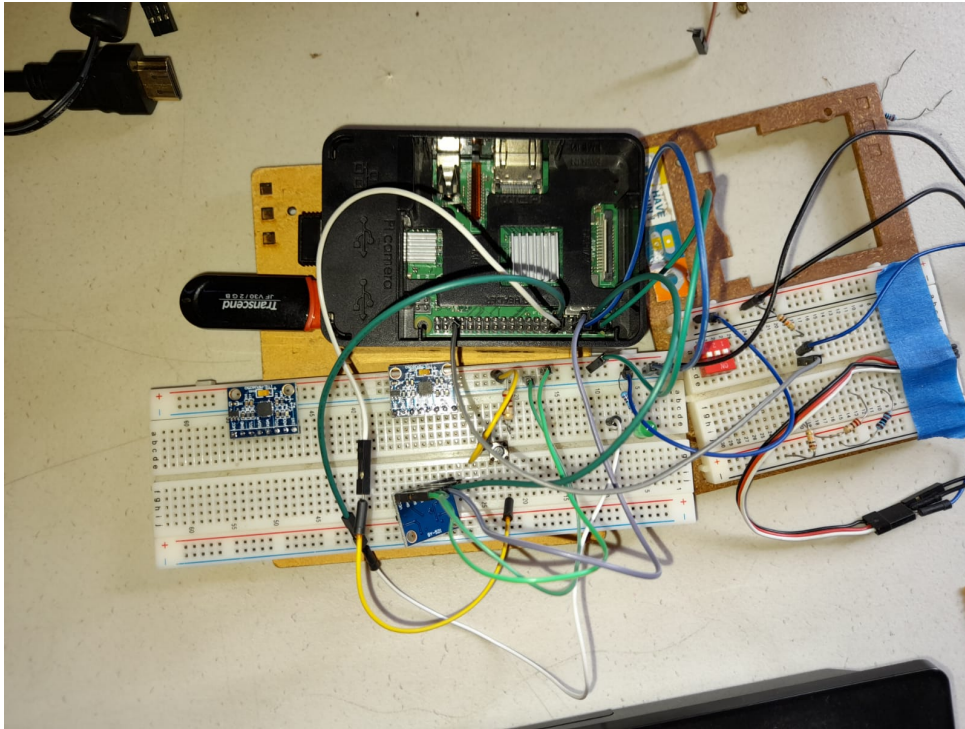


Figura 4.4: Circuito inicial para toma de datos en Raspberry Pi

El circuito de conexión entre la Raspberry Pi y el acelerómetro se suelda en una placa perforada y se colocan cables para conectar al controlador y al sensor. Se utiliza un LED (Light Emitting Diode, por sus siglas en inglés) verde para indicar si el programa se está ejecutando e indicar la presencia de una llave maya para guardar los datos. El código corre de manera automática una vez que se enciende la Raspberry Pi, el Led empieza a parpadear si no se cuenta con una llave maya, hasta que no se tenga una llave maya el código de adquisición de datos no correrá y el Led seguirá parpadeando. Si el Led no parpadea significa que se está en modo adquisición de datos y se guardan los datos en la llave maya. Para terminar la toma de datos se debe presionar el botón en la placa y una vez se apague el Led se puede extraer la llave maya para realizar la lectura de datos en un computador. En la figura 4.5 se muestra el diagrama de conexión del sistema de la toma de datos en la Raspberry Pi y en la figura 4.7 se muestra la placa soldada y conectada a para realizar la medición.

En la figura 4.8 se muestra el sistema de toma de datos acoplado a la viga del sistema

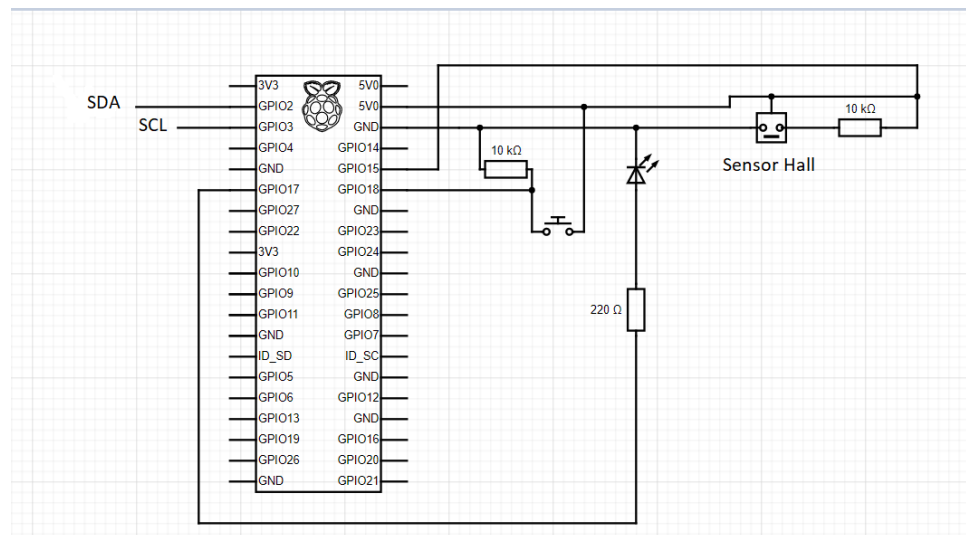


Figura 4.5: Diagrama de conexión del sistema de la toma de datos en la Raspberry Pi

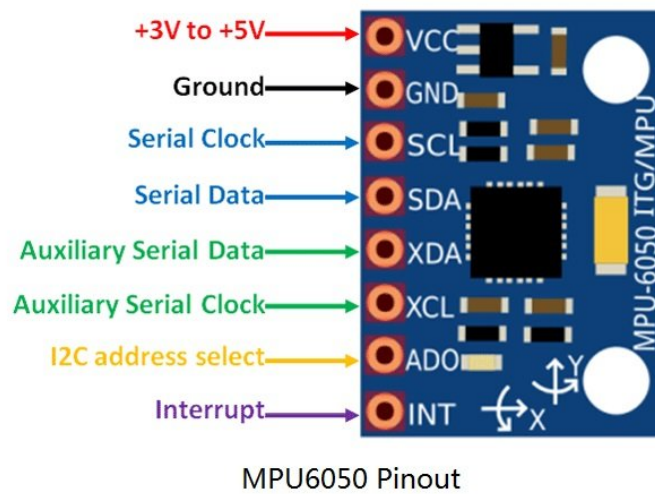


Figura 4.6: Diagrama de conexión del MPU6050 [8]

de aceleración. En la figura 4.9 se muestra todo el sistema de caracterización en la etapa inicial.

Se busca que la Raspberry Pi ejecute el código para la toma de datos de manera autónoma e independiente del sistema de aceleración. Por lo que se configura el código en modo “autorun” o de auto ejecución. Esto permite que una vez se encienda la Raspberry Pi el código de ejecuta automáticamente e inicia la recopilación de datos de aceleración. Para esto se debe configurar el archivo “autostart” en el directorio “/etc/xdg/Lxsession/LXDE-pi”. En este se coloca al final la siguiente linea de código:

```
@lxterminal -e python UnsoloSensor1.py
```

Donde “UnsoloSensor1.py” corresponde al nombre del código que se desea correr en modo de auto ejecución.

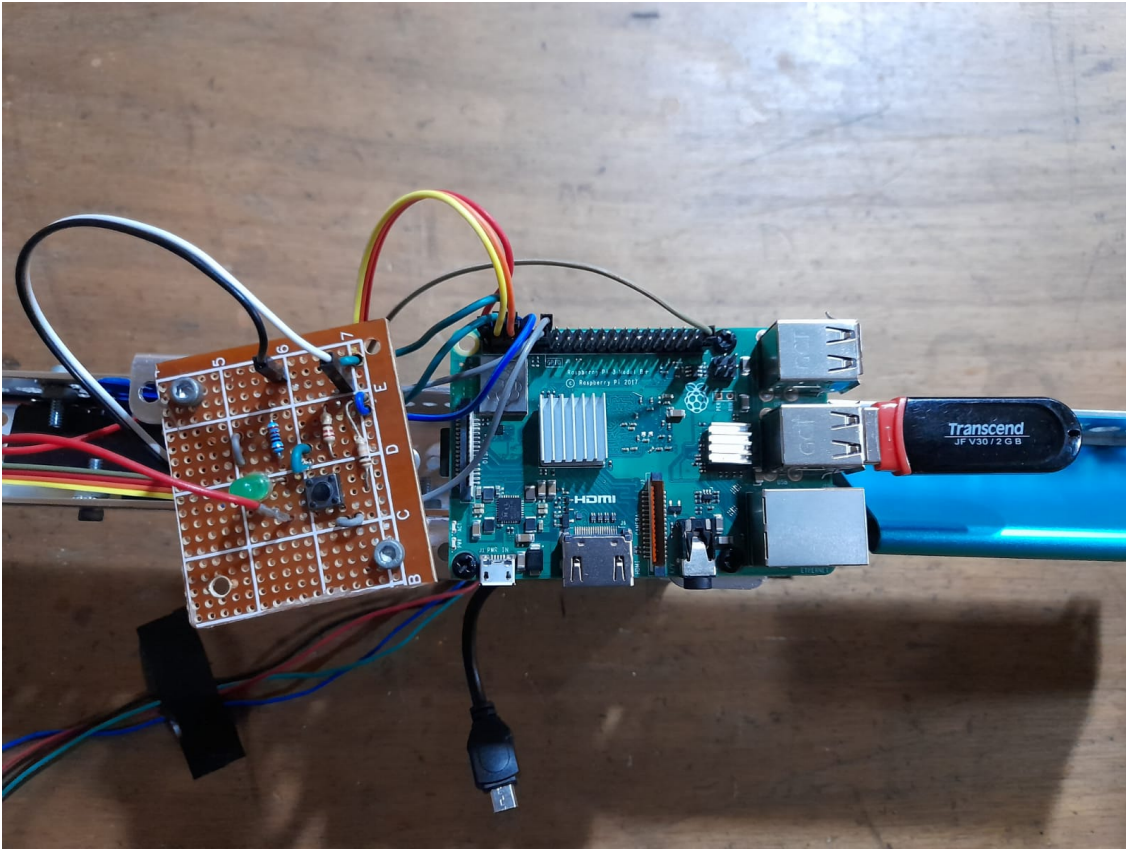


Figura 4.7: Placa soldada y conectada a la Raspberry Pi

4.1.5 Guardas de seguridad y cámara de aislamiento térmico

En la figura 4.10 se muestra el sistema de caracterización dentro de las guardas de protección. Estas mismas guardas sirven como cámara de aislamiento térmico para realizar las pruebas de aceleración con un cambio de temperatura, en la figura 4.11 se muestra por fuera el sistema completo con la cámara y en el apéndice A.4 se tiene los planos de la cámara de aislamiento.

Durante la caracterización en calentamiento se utiliza una secadora de cabello como fuente de aire caliente para calentar la cámara de aislamiento térmico. Se tiene un agujero en las guardas por donde entra el aire caliente al sistema y con la lectura del sensor TPM-10 se realiza la lectura de la temperatura interna de la cámara. El sensor TPM-10 tiene un rango de medición de -50°C a $+110^{\circ}\text{C}$ y una resolución de 0.1°C con un tiempo de actualización de 2 segundos, los datos de este sensor se encuentran en [23].

4.2 Implementación del simulador en la FPGA

Para implementar el simulador en la FPGA, se utiliza la plataforma de LabView para programar el sistema. Tomando como referencia al sensor ADXL377 de Analog Devices, se puede extraer la respuesta del sensor a partir de los datos suministrados en la hoja

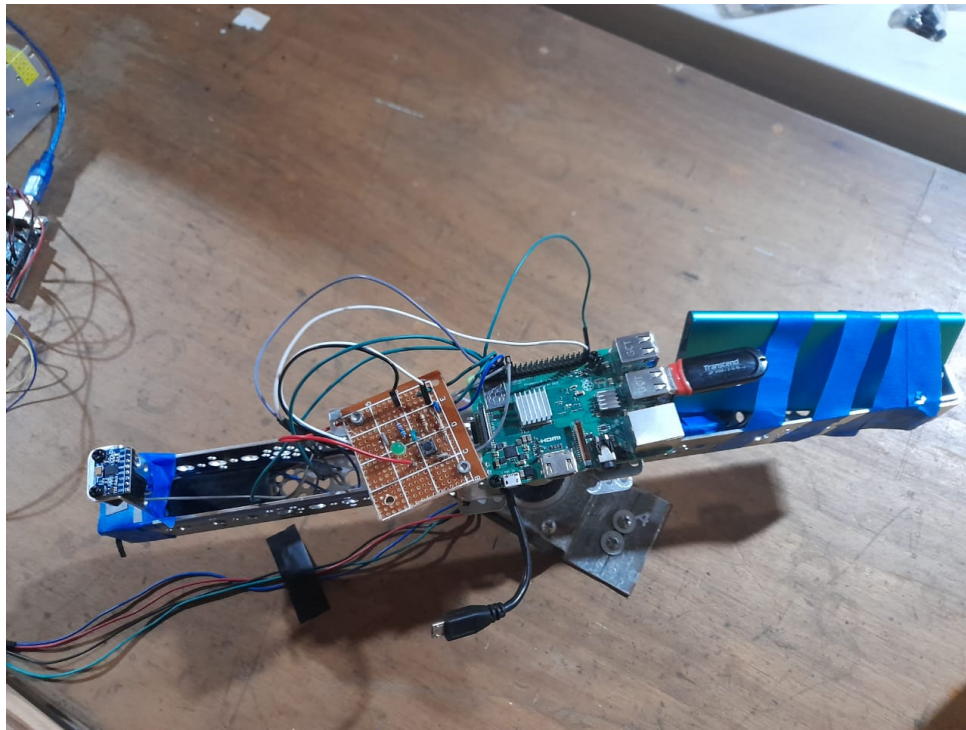


Figura 4.8: Primer modelo sistema de toma de datos sobre el sistema de aceleración

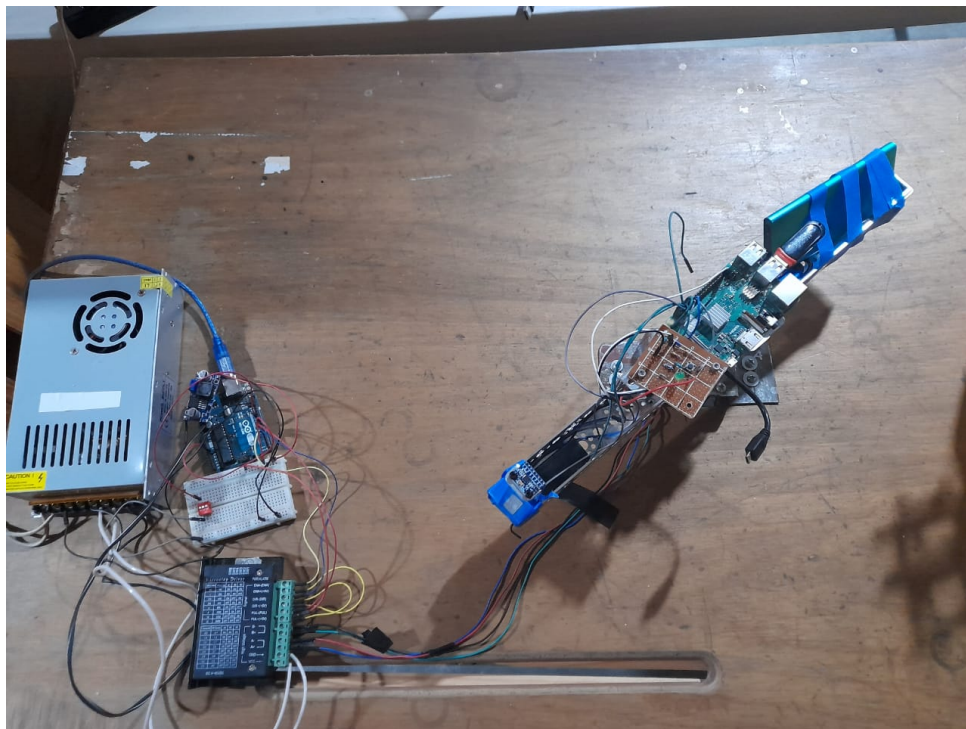


Figura 4.9: Primer modelo sistema de caracterización

de datos en [6]. Este acelerómetro permite medir valores de ± 200 g en sus tres ejes y tiene una tensión de salida de 3.3V. En figura 4.12 se muestra que la sensibilidad máxima de la medición es de 7.2 mV/g (milivoltios por cada g), una sensibilidad al cambio de



Figura 4.10: Primer modelo del sistema de guardas de protección



Figura 4.11: Primer modelo de la cámara de aislamiento térmico

temperatura de $\pm 0.02\%/^{\circ}\text{C}$ (por ciento por cada grado Celsius) y un valor a Cero g (Zero g BIAS - offset) de 1.6V obtenidos de la hoja de datos en [6]. Se puede obtener el valor de la aceleración en g a partir de los Voltios de salida del sensor, esto se hace mediante

4.5

SPECIFICATIONS

$T_A = 25^\circ\text{C}$, $V_S = 3\text{ V}$, $C_X = C_Y = C_Z = 0.1\text{ }\mu\text{F}$, acceleration = 0 g, unless otherwise noted. All minimum and maximum specifications are guaranteed. Typical specifications are not guaranteed.

Table 1.

Parameter	Test Conditions/Comments	Min	Typ	Max	Unit
SENSOR INPUT	Each axis				
Measurement Range			± 200		g
Nonlinearity	% of full scale up to 180 g		± 0.5		%
Cross-Axis Sensitivity ¹			± 1.4		%
SENSITIVITY, RATIOMETRIC ²	Each axis				
Sensitivity at X_{OUT} , Y_{OUT} , and Z_{OUT}	$V_S = 3\text{ V}$	5.8	6.5	7.2	mV/g
Sensitivity Change Due to Temperature ³	$V_S = 3\text{ V}$		± 0.02		%/ $^\circ\text{C}$
ZERO g BIAS LEVEL, RATIOMETRIC					
Zero g Voltage	$V_S = 3\text{ V}$, $T_A = 25^\circ\text{C}$	1.4	1.5	1.6	V
Zero g Offset vs. Temperature					
X-Axis and Y-Axis			± 12		mg/ $^\circ\text{C}$
Z-Axis			± 30		mg/ $^\circ\text{C}$

Figura 4.12: Hoja de datos ADXL377 [6]

$$a = \frac{(V - V_o)}{s} \quad (4.5)$$

Donde V corresponde a la tensión de salida del sensor, V_O es el Zero g BIAS (offset) y S corresponde a la sensibilidad del sensor. En el simulador el usuario debe ingresar el valor máximo de g en el que desea trabajar, por lo que se despeja la ecuación 4.5 para obtener la tensión de salida del sensor y generar este valor en las salidas analógicas y digitales de la simulación con lo que se obtiene la ecuación 4.6 y 4.7.

$$a * S + V_o = V \quad (4.6)$$

$$a * (7.2 * 10^{-3})V + 1.6V = V_{sensor} \quad (4.7)$$

Con esto se tiene una ecuación lineal que describe el comportamiento de la tensión de salida del sensor para cada valor de aceleración en g. Para obtener una salida compatible con tecnología TTL (salidas de 0-5V), se realiza un escalamiento de la tensión de salida del sensor, se tiene 3V máximo para la salida del sensor y se busca escalar este valor a 5V para generar un valor TTL. Donde el V_{sensor} se escala a V_{TTL} mediante la ecuación 4.8 y 4.9.

$$\frac{3}{5} = \frac{V_{sensor}}{V_{TTL}} \quad (4.8)$$

$$V_{TTL} = V_{sensor} * \frac{5}{3} = V_{sensor} * 1.66 \quad (4.9)$$

En figura 4.12 se muestra la relación existente entre la aceleración y la tensión escalada a TTL. Estos valores se obtienen a partir de las ecuaciones anteriores y el rango de aceleración de medición del sensor ADXL377.

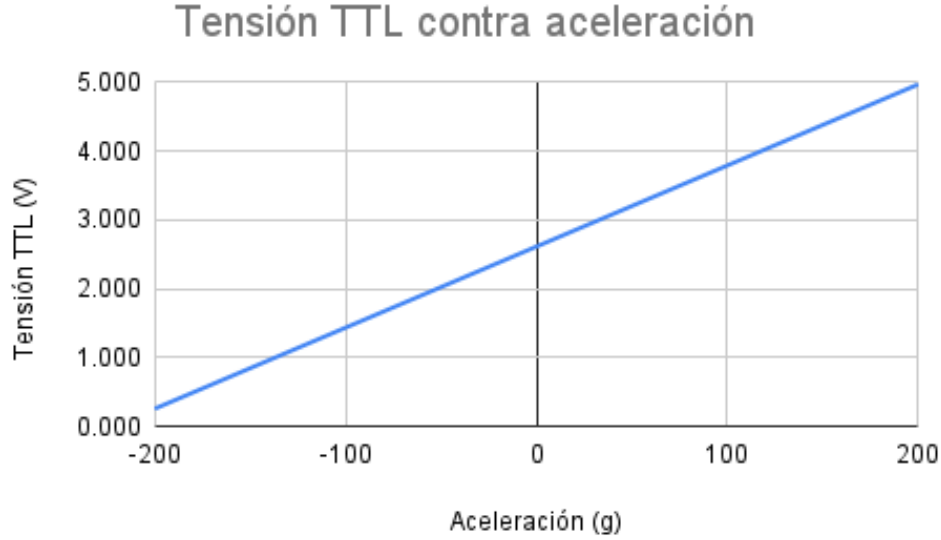


Figura 4.13: Relación entre tensión TTL del sensor y la aceleración

Finalmente, se busca que la respuesta del sensor simulado sea como la de un sensor real, se aplican las ecuaciones que describen el sistema de centrifugación en la sección 4.1.1, donde se coloca el sensor en un sistema de centrifugación con el fin de realizar la medición de la aceleración en la componente de la aceleración hacia el centro del sistema. Se asume que el motor de la centrifugadora puede girar RPM infinitas. Por lo que el simulador no se vería limitado por el motor y su respuesta es únicamente debido a las condiciones ambientales del sensor. Con la ecuación 4.10 se obtiene la aceleración centrífuga en m/s^2 , asumiendo que $r=0.2m$ y la velocidad del motor en RPM puede aumentar y disminuir y el motor puede cambiar de dirección.

$$a_n = 0.2 * \left(2\pi * \frac{RPM}{60} \right)^2 \left[\frac{m}{s^2} \right] \quad (4.10)$$

Este valor se divide entre $9.81 m/s^2$ para obtener el valor en unidades de g.

Asumiendo que para la ecuación 4.3 se inicia en $w_1=0$ y $t_2=0$, la velocidad angular w_2 en rad/s se obtiene de la velocidad del motor y $t_2=60s$ se obtiene al sustituir en la ecuación 4.2 el valor de aceleración tangencial en la ecuación 4.11:

$$a_t = \frac{\left(2\pi * \frac{RPM}{60} \right) - 0}{60 - 0} * r \left[\frac{m}{s^2} \right] \quad (4.11)$$

Este valor también se divide entre $9.81 m/s^2$ para obtener el valor en unidades de g.

En el eje paralelo a la fuerza de gravedad no se tiene ningún efecto asociado por el

movimiento rotatorio de la centrifugadora, por lo que el valor sobre este eje será muy cercano al de $1g$, si el sensor está calibrado.

Dado que los acelerómetros se ven influenciados por los cambios en temperatura se incluye el ruido dado por la temperatura en la simulación. En figura 4.12 se mostró el Offset debido a los cambios de temperatura. Estos valores se suman al valor de aceleración generado por la respuesta del simulador por cada grado Celsius a la cual se encuentre.

En la figura 4.14 se muestra el código en el cual se implementaron las ecuaciones que describen el comportamiento del sensor bajo una aplicación de centrifugación.

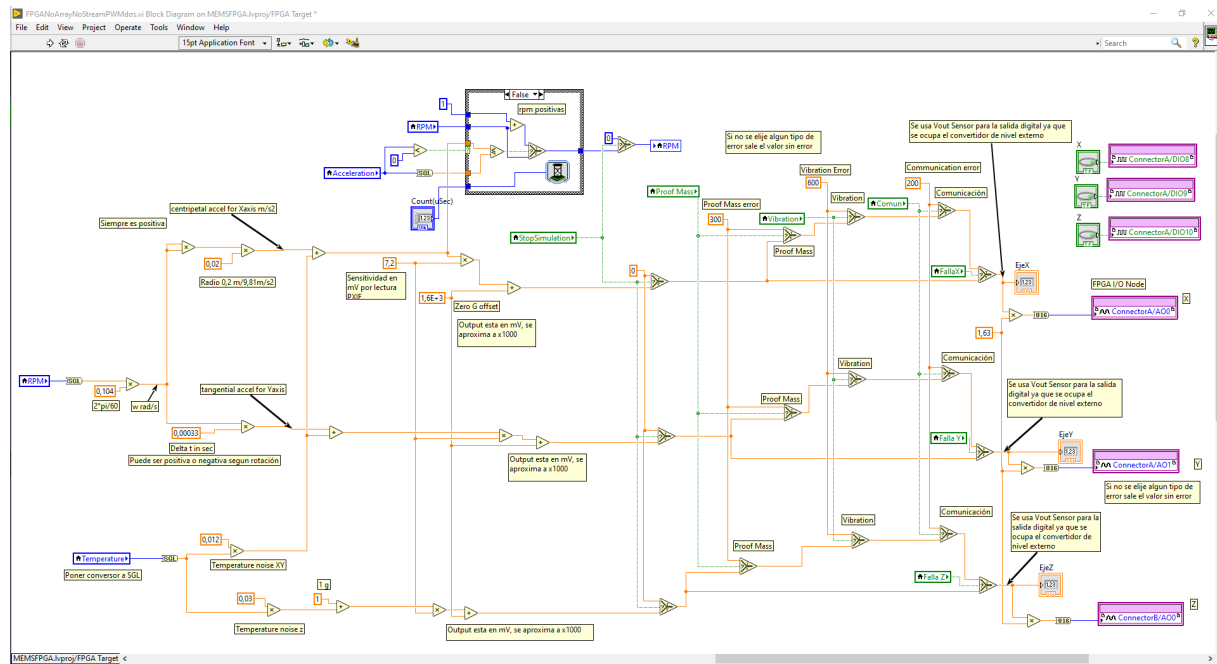


Figura 4.14: Código de la simulación en la FPGA

Seguidamente se mencionan los diversos *software* y plataformas utilizados para implementar las ecuaciones anteriores en el simulador.

4.2.1 Interfaz Simulador-Usuario

En esta sección se describe la implementación de la Interfaz de Usuario y la API con el fin de comunicar la simulación en la FPGA a las herramientas de verificación como TestStand.

Arquitectura PXI

Se utiliza un PXI como medio de desarrollo para la Interfaz de Usuario y sistema de validación mediante TestStand. Los PXI (PCI de extensiones para instrumentación) son una plataforma basada en computadoras personales (PC) para sistemas de medidas y

automatización. Proporcionan potencia y un bus de comunicación para soportar múltiples módulos de instrumentación dentro de un mismo chasis. Los PXI utilizan tecnología comercial de bus, son modulares y robustos [18]. En la figura 4.15 se muestra un PXI y sus componentes.

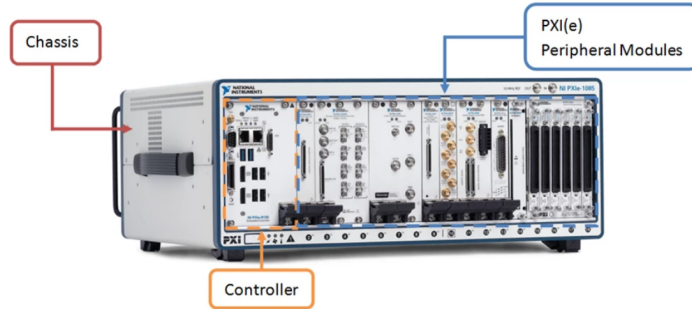


Figura 4.15: Componentes de un PXI [18].

El chasis del PXI corresponde a la cubierta de la PC de escritorio, el controlador PXI se puede ver como el CPU, este posee una memoria y puertos de entrada y salida. Finalmente, los módulos periféricos del PXI(e) permiten acoplar diversos instrumentos a utilizar [18].

Dentro de los elementos a utilizar en la implementación se tiene el **PXIe-1085** el cual corresponde a un chasis de 18 Ranuras. También se utiliza el **PXIe-6570**, este corresponde a un Instrumento de Patrón Digital, el cual está diseñado para caracterizar semiconductores y hacer pruebas de producción. Incluye el Digital Pattern Editor para configurar mapas de pines y patrones [19].

InstrumentStudio

La herramienta InstrumentStudio es un *software* que permite monitorizar y depurar sistemas de pruebas automatizadas, así como ver datos en pantallas unificadas y capturar pantallas de múltiples instrumentos y resultados de medidas. Permite guardar configuraciones para asegurar la repetibilidad de pruebas con dispositivos bajo prueba específicos o implementar pruebas de TestStand directamente. Algunos de los hardware que soporta son los controladores NI-SCOPE (Osciloscopio), NI-DCPOWER (Fuente variable) y NI-DMM (Multímetro) [20].

Editor de Patrones Digitales - Digital Pattern - DPE

El Digital Pattern Editor es una herramienta interactiva para importar, editar o crear patrones de prueba. El software integra hojas de edición para mapas de pin de dispositivo, especificaciones y patrones para desarrollar o editar vectores. El NI-Digital Pattern Driver permite desarrollar código de pruebas para interactuar con instrumentos de patrones

digitales PXI. El Digital Pattern Editor se utiliza para configurar los pines a utilizar en la comunicación. Este editor guarda los patrones y permite configurarlos para poder realizar las lecturas y escrituras en la FPGA [21].

Interfaz Software-Hardware

Para realizar la conexión entre el *software* y el *hardware* se utiliza la tarjeta CB-2162 y el cable NI SHC68-C68-D4 para conectarse con el PXIe-6570, y poder enviar y recibir información por el protocolo SPI. El NI CB-2162 (Tarjeta de color verde) es un bloque de conexión y prototipado para los instrumentos de patrones digitales, este posee: entradas y salidas digitales, permite realizar depuración, pruebas sobre prototipos y desarrollar interfaces para pruebas [13]. La conexión se realiza mediante un conector VHDCI, en la figura 4.16 se muestra la conexión entre el PXIe-6570 y la tarjeta CB-2162.

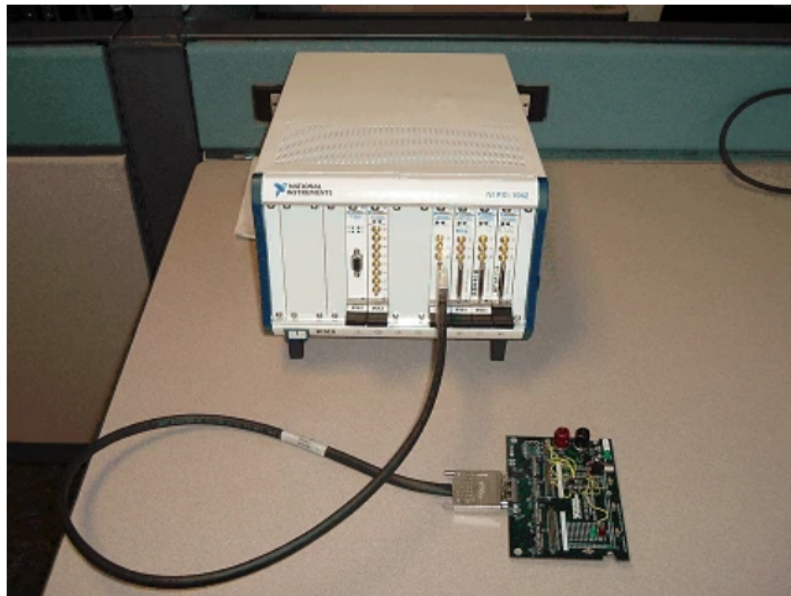


Figura 4.16: Conexión PXI y tarjeta CB-2162 [13].

Comunicación SPI

De los diversos métodos de comunicación mencionados (SPI, I2C, UART) se seleccionó el SPI, ya que se tenían diversos ejemplos de implementaciones de este tipo de comunicación en el PXIE lo que puede facilitar la implementación de la comunicación para esta aplicación. Para poder realizar la comunicación entre el PXI y la FPGA a través del CB-2162 se instalan los componentes de comunicación SPI para FPGA. Se crea un programa en el cual pueda leer y escribir información en los pines MOSI y MISO del myRio. En la figura 4.17 se muestra el código utilizado para leer y escribir información a través del protocolo SPI en la FPGA.

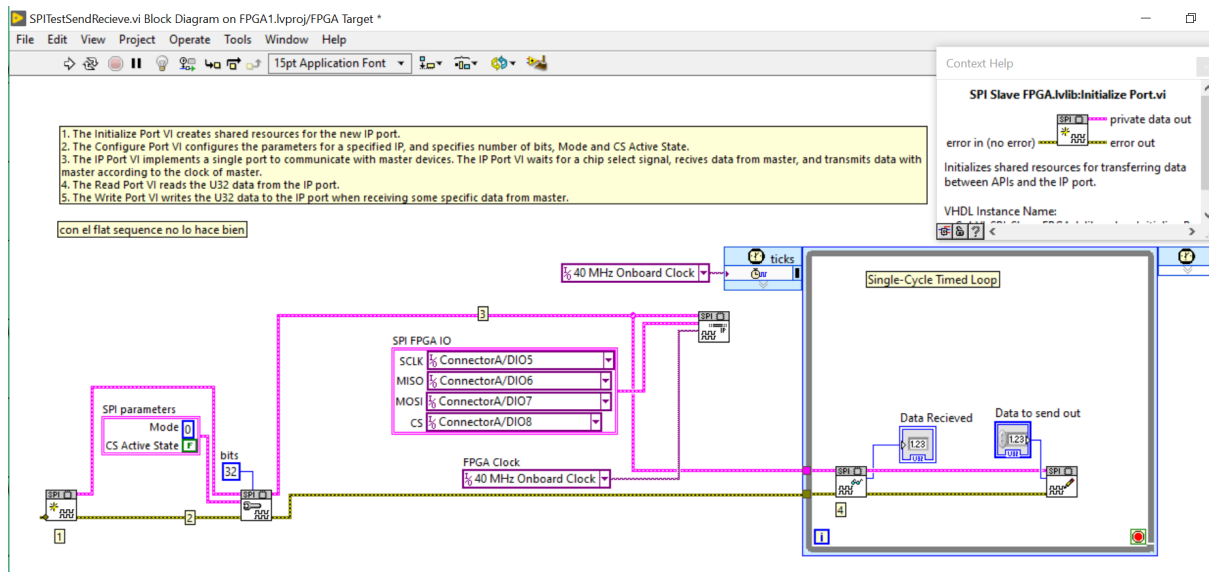


Figura 4.17: Código de lectura y escritura en el protocolo SPI

En la aplicación del DPE, se debe configurar los pines para la comunicación SPI, esto se hace mediante el mapa de pines de la tarjeta CB-2162. En este se definen los nombres de los pines como variables y se configura el canal a utilizar, en este caso de utilizan los canales 21, 23, 25 y 27 para los pines CS, MOSI, MISO y SCL respectivamente. En la figura 4.18 se muestra esta configuración.

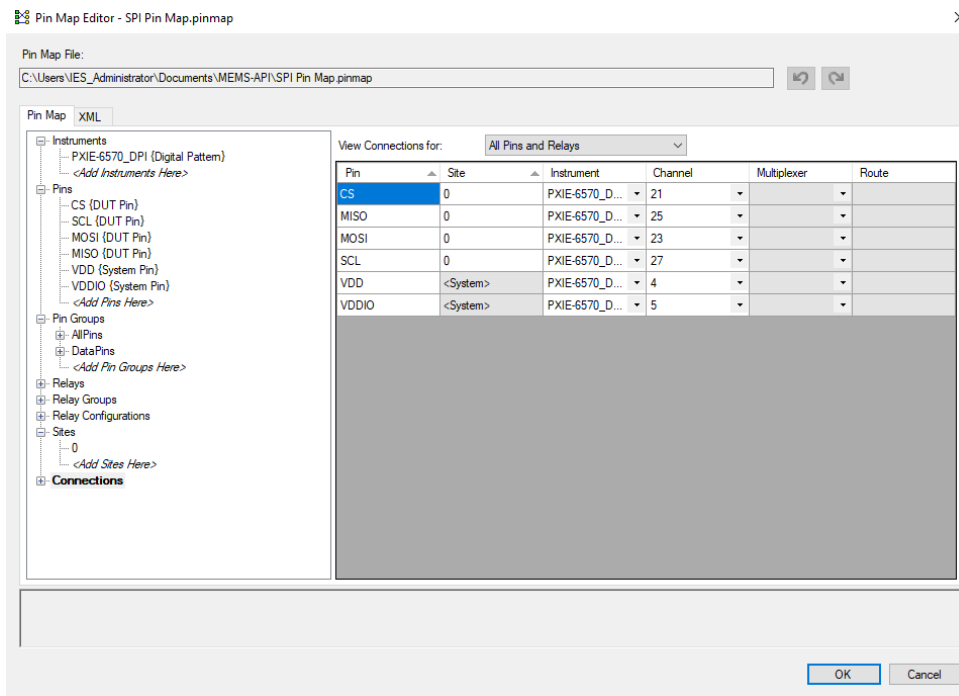


Figura 4.18: Mapa de pines utilizado para la comunicación SPI en el DPE

Una vez establecidos los pines para la comunicación se genera un patrón, este permite leer y escribir sobre los canales definidos. En la figura 4.19 se muestra el patrón para

realizar la lectura. Adicionalmente se configura un archivo tipo *waveform*, en el cual se establecen los parámetros como cantidad de bits (32) y formato (Hexadecimal) para realizar la lectura y escritura respectivamente.

Pattern Name: ■ Not Loaded

🔍 Comments, Exports, and Time Sets

Time Set	Label	Opcode	Vector	CS	SCL	MISO	MOSI	Comments
SPI		set_loop(1)	0	1	1	X	X	reg0 defines the number of registers to read
-		capture_start(capture_buffer)	1	1	1	X	X	Define the read buffer
-		source_start(source_buffer)	2	1	1	X	X	Define the register addresses buffer
-			3	1	1	X	X	
-	repeatRead		4	0	1	X	X	Start (CS = 0)
-			5	0	1	X	1	Read instruction (MOSI = 1)
-		repeat(32), source	6	1	1	X	0	Register address, reg1 defines the address bit-width
-		repeat(32), source	7	0	1	X	D	
-		repeat(32), source, capture	8	0	1	V	D	Register value, reg2 defines the value bit-width
-		end_loop(repeatRead)	9	1	1	X	X	Stop (CS = 1)
-		capture_stop	10	1	1	X	X	
-		halt	11	1	1	X	X	End Pattern

Figura 4.19: Configuración para la comunicación SPI en el DPE

Con el fin de enviar comandos a la FPGA, se establecen direcciones para cada acción que se le quiera configurar al simulador, en la tabla 4.2, se muestran los tipos de datos, la dirección, la función y el valor de salida que se envía de vuelta a través del SPI. Para los datos de temperatura y aceleración se colocan algunas de las direcciones, ya que el usuario puede enviar cualquier valor entre estos rangos para configurar la simulación.

Las direcciones de la aceleración se establecen a partir de 200 para que corresponda numéricamente y no se traslape con ningún otro tipo de datos. En el caso de la temperatura, se buscó que el rango no estuviera muy lejos del de aceleración, pero asegurando la independencia de cada uno para evitar errores.

4.2.2 Modo de Trabajo y selección de Modos de Falla

Los modos de trabajo y falla corresponden a secuencias o respuestas del sensor MEMS bajo condiciones que generan fallas o errores (modo de falla) y condiciones que permiten una respuesta sin fallas (modo trabajo). Estas respuestas dependen de la selección del usuario, ya sea que este requiera una respuesta sin fallas para realizar pruebas sin error o seleccione una respuesta con fallas para generar errores en sus pruebas.

El modo de trabajo propuesto busca obtener una tensión de salida sin fallas en los 3 ejes del acelerómetro al seleccionar una temperatura de simulación y el valor de la aceleración.

Para los modos de falla se realiza una búsqueda en diversas fuentes de información para poder tener diversos tipos de fallas diferentes. Los principales modos de falla encontrados para dispositivos MEMS son mecánicos, electrónicos y de comunicación.

Las dimensiones de los dispositivos MEMS varían desde un micrón a varios milímetros, por lo que una partícula de polvo o suciedad dentro del encapsulado puede generar errores en

Tabla 4.2: Direcciones para la comunicación SPI.

Dirección		Función	Salida SPI
Datos del Chip	0	Default	0
	1	Auto prueba - Self Test	0
	3	Estado Chip	1
	4	Numero de Serie	3216
	5	Modelo	377
Ejes de Falla	6	Falla Eje X	0
	7	Falla Eje Y	0
	8	Falla Eje Z	0
Modos Falla	9	Error Comunicación	0
	10	Error Masa Sísmica	0
	11	Error por Vibración	0
Aceleración	200	-200 g	0
	300	-100 g	0
	400	0 g	0
	500	100 g	0
	600	200 g	0
Temperatura	620	-40 °C	620
	660	0 °C	660
	710	50 °C	710
	745	85 °C	745
	785	125 °C	785
Pare	999	Para la Simulación	999

el sensor. Las estructuras son relativamente simples y pueden tener o no partes móviles, debido a esto en [47] se exponen diversos tipos de fallas mecánicas de los dispositivos MEMS y como estas afectan la fiabilidad de los sensores y actuadores.

Fractura mecánica

Una fractura se define como la ruptura de un material uniforme en dos secciones separadas estas pueden ser fracturas dúctiles, frágiles e intercristalinas. En los dispositivos MEMS las fracturas por materiales frágiles y policristalinos son las más comunes y estas son causa de falla catastrófica en el funcionamiento de los dispositivos. Normalmente los niveles de esfuerzo a los que están sometidos los acelerómetros MEMS están muy por debajo del esfuerzo de ruptura del material utilizado para construir la estructura mecánica [47].

Stiction o fricción estática

Corresponde a una de los problemas más importantes en los sistemas MEMS, los sistemas MEMS tienen estructuras internas tan pequeñas que las fuerzas superficiales hacen que las superficies se acoplen y entren en contacto. Algunas de estas fuerzas se dan debido a la condensación capilar, las fuerzas moleculares de van der Waals y las uniones químicas entre superficies [47].

Vibraciones y golpes

Las vibraciones son una de las variables más importantes respecto a la fiabilidad de los sistemas MEMS, debido a la sensibilidad y fragilidad de muchos de estos. Las vibraciones externas pueden causar adhesión superficial o fracturas en los soportes estructurales, lo que produce fallas. Los golpes corresponden a un único impacto mecánico en vez de un evento rítmico. Un golpe genera una transferencia de energía a través del dispositivo y de manera similar puede generar adhesión o fractura [47].

Descarga Electroestática y carga dieléctrica

La descarga electroestática o ESD, ocurre cuando un dispositivo no se maneja adecuadamente, el cuerpo humano crea un potencial eléctrico y al entrar en contacto con un dispositivo electrónico hace una descarga que genera un gran diferencia de potencial a lo largo de todo el circuito. Este efecto es catastrófico en circuitos y podría tener efectos similares en los MEMS [47].

Temperatura

Los esfuerzos dentro de los dispositivos MEMS son totalmente dependientes de la temperatura, el rango térmico de operación del sensor es determinado por el coeficiente de expansión linear. En muchos casos la expansión y contracción térmica se da en el metal del encapsulado y genera problemas ya que este coeficiente puede llegar a ser hasta 10 veces el del silicio por lo que el encapsulado se puede separar del sistema de masa sísmica y causar daños en los elementos internos [47].

Otros

En general se tiene fallas mecánicas como golpes, que pueden generar fracturas haciendo que el sensor no reacciona como es debido; fallas electrónicas donde el sensor se ve afectado por el ambiente como descargas electro estáticas o falla de cables de conducción interna y corto circuito; y también fallas por comunicación debido a problemas del chip durante el envío de datos.

Dado estos tipos de errores se propone el diagrama de flujo en la figura 4.20 en el cual se puede seleccionar un tipo de error para aplicar en la respuesta del simulador. Para esto se debe seleccionar el modo de falla y verificar cual tipo de error quiere probar el usuario. Al elegir algún tipo de error se busca generar una salida con un valor de error.

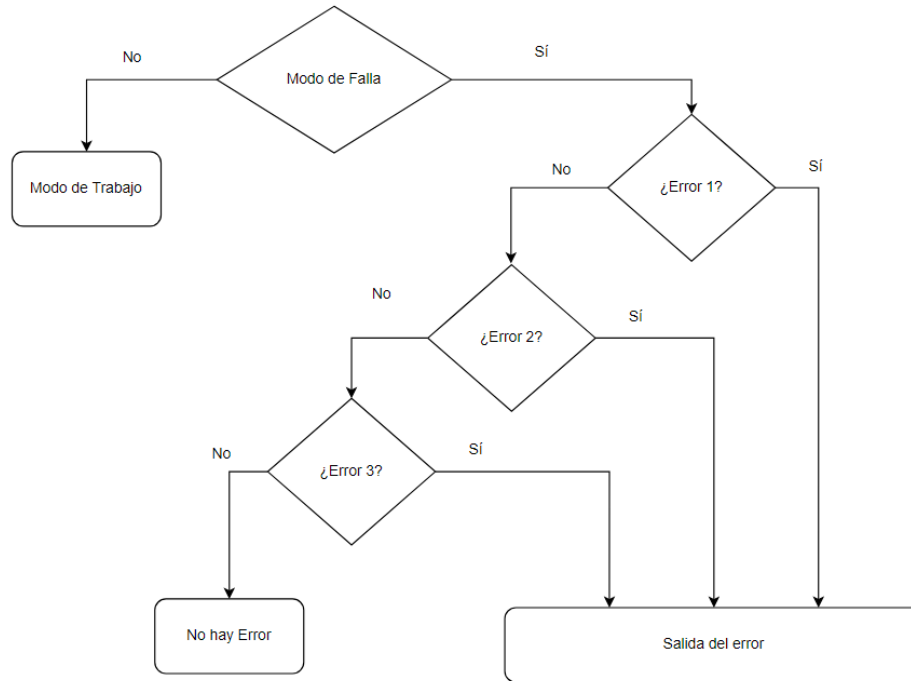


Figura 4.20: Diagrama de flujo para la selección de los modos

Selección de los ejes a fallar

De igual manera la simulación debe permitir hacer fallar uno, alguno o todos los ejes del sensor, para lograr esto se generó el diagrama de flujo de la figura 4.21 en el cual se realiza una evaluación sobre cuales ejes han sido seleccionados y se establecen casos para cada posibilidad.

Luego de implementar esta secuencia, se tienen problemas con la comunicación, ya que la implementación de este método hace que se retrase el envío de datos de vuelta a el PXI. Por lo tanto se implementa un método de establecimiento de variables por direcciones, lo que permite enviar datos en un solo ciclo de reloj y no hay atrasos en la lectura y escritura.

4.2.3 Salidas Digitales y Analógicas

Dado que se está simulando la respuesta de un sensor, es necesario tener canales de salida para realizar la lectura de los valores simulados. Se debe designar una salida por cada eje, y deben ser analógicas y digitales. Las salidas analógicas del myRio pueden funcionar en

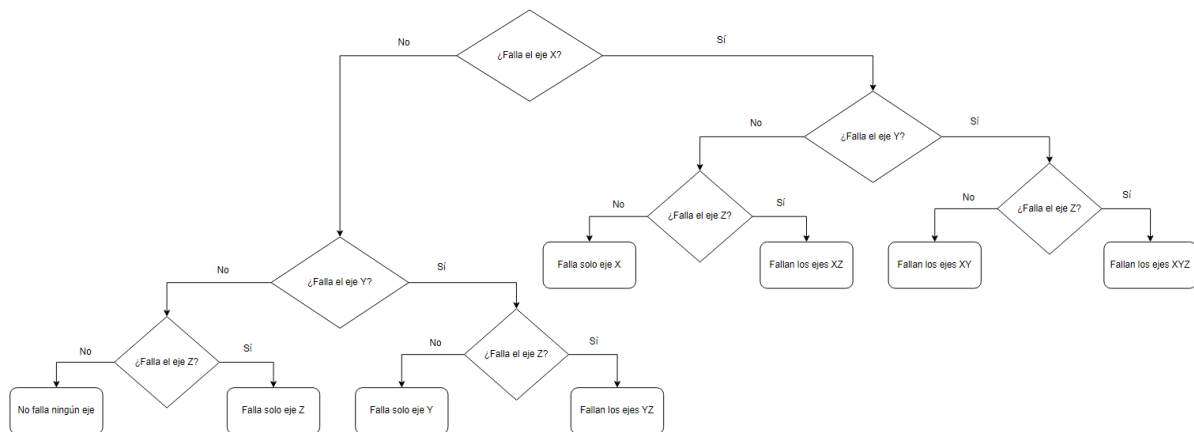


Figura 4.21: Diagrama de flujo para la selección de los ejes que fallan

valores TTL (de 0 a 5V), mientras que las salidas digitales requieren de un **convertor de nivel HW-221** para poder convertir de 3V a 5V las salidas digitales. El diagrama de conexión del convertidor de nivel se encuentra en la figura 4.22

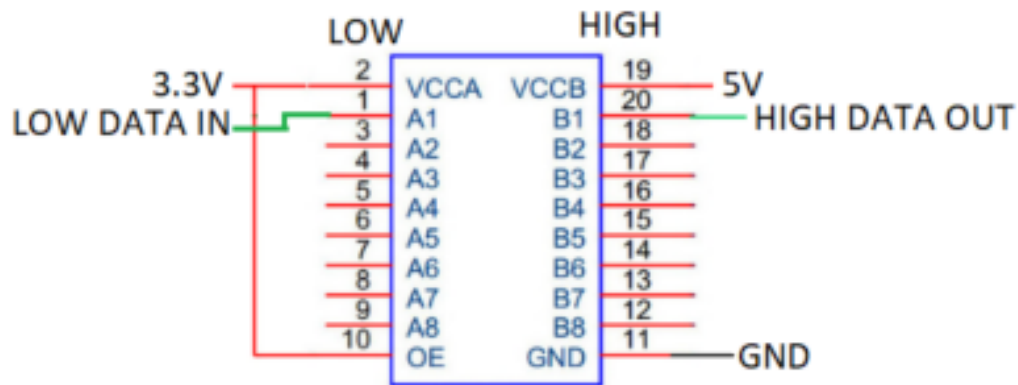


Figura 4.22: Diagrama conexión del convertidor de nivel HW-221.

EL valor de salida analógica del simulador se obtiene de manera directa a través de la FPGA, mientras que en las salidas digitales se busca convertir a PWM con el fin de representar el valor analógico de manera digital. Con respecto a la salida PWM, se implementa el código fuera de los recursos de la FPGA. Esto debido a que se requieren muchos recursos para implementarlo dentro de la FPGA y no permite compilar el programa. Se utiliza un método de comunicación entre los recursos de la FPGA y el myRio, lo que permite enviar y recibir información a la FPGA. En la figura 4.23 se muestra la sección del código que genera el PWM en los recursos del myRio y envía el valor a la FPGA para obtener las salidas digitales.

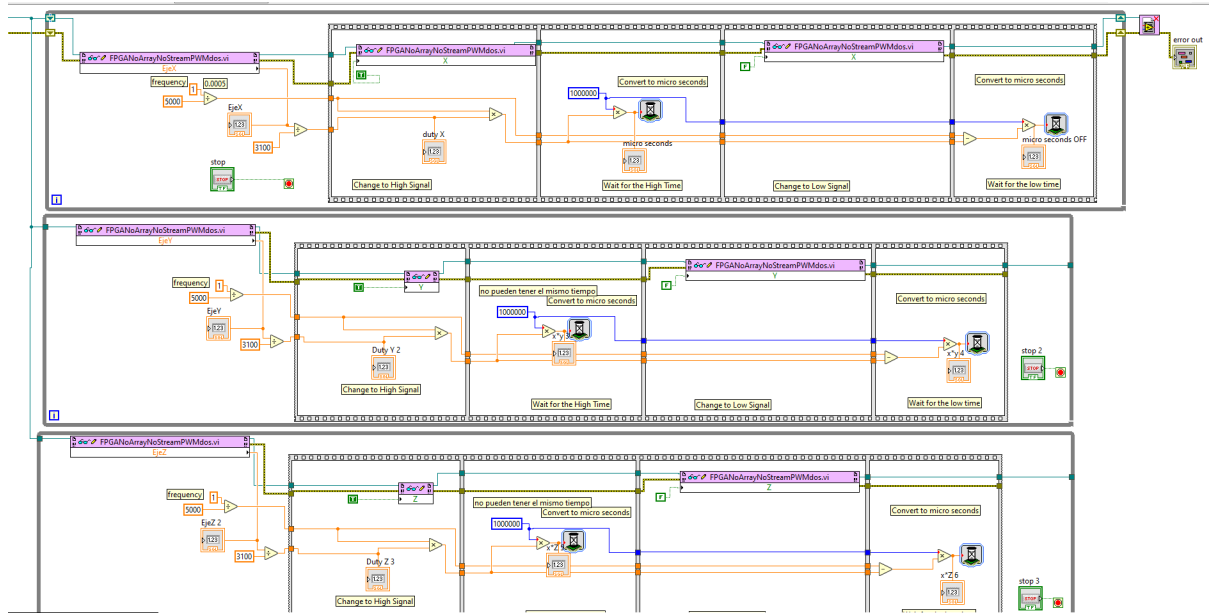


Figura 4.23: Sección del código de PWM implementado

4.3 Diseño del experimento y toma de datos

Con el fin de realizar la toma de datos con el sistema de caracterización se procede a diseñar un "experimento", en [28] el autor Montgomery establece los pasos para realizar el proceso de diseño de experimentos.

4.3.1 Reconocimiento del problema

Se tiene como problema que se busca caracterizar el acelerómetro ante variaciones de temperatura.

4.3.2 Selección de factores, niveles y variable de respuesta

Los factores corresponden a los valores que van a ser variados durante el experimento y los niveles indican el nivel a los que se van a correr. La variable de respuesta corresponde a aquel elemento en el experimento que proporcione información importante del proceso bajo estudio [28].

En este caso, según los objetivos del proyecto se debe desarrollar el proceso de toma de datos para un acelerómetro real de 3 ejes en al menos 3 rangos establecidos de aceleración y determinar las variables de influencia.

Las variables que se pueden medir en el sistema de caracterización son: aceleración, revoluciones por minuto (RPM) y temperatura ambiente. Estas tres variables se pueden medir de manera independiente. La aceleración se puede medir en un rango de $\pm 16g$,

la temperatura se puede medir en un rango de -50°C a $+100^{\circ}\text{C}$ y las RPM se pueden medir desde 0 hasta 500 RPM. Estos rangos se obtienen de las hojas de datos de los diversos dispositivos y de la medición previa de las RPM a las que puede llegar el sistema de aceleración. Los rangos en los cuales las variables van a ser variados corresponden a: aceleración, de 0g a 16g; las RPM se variaron de 0 a 300 RPM para lograr llegar a los 16g; y la temperatura se busca variar desde la temperatura ambiente (25°C) hasta el doble (50°C). Pero se utiliza una temperatura de 46°C para las mediciones ya que el motor a pasos tiene un rango de temperatura de operación de -10°C a 50°C . Por lo que al sobrepasar los 50°C el motor se des sincroniza y se apaga, lo que puede generar daños en el motor y contratiempos al realizar las mediciones.

Los valores a ser variados durante el experimento (factores) corresponden a la temperatura y las RPM, de la variación de estos se obtiene como respuesta una aceleración.

4.3.3 Selección del diseño de experimento

Se tienen 2 factores (temperatura y RPM) en 2 niveles cada uno (0g y 16g para la aceleración; para la temperatura: valor ambiente y 46°C), por lo que se puede aplicar el diseño de 2^k , donde k representa el número de factores. Este proceso se debe hacer por cada eje, por lo que se obtiene $2^2 * 3 = 12$ mediciones por cada eje, al variar las RPM y la temperatura del proceso.

Pero con el fin de obtener una confianza del 95% se deben realizar 16 mediciones por cada condición. Esto se obtiene de la ecuación 2.24 de [34]. por lo que por cada eje se deben realizar 16 mediciones y por cada temperatura también. Donde n corresponde a la cantidad de medidas a realizar por situación, $\sigma = 0.1$ corresponde a la desviación estándar y $e = 0.05$ corresponde al error. Además, $z_{\alpha/2} = 1.96$ es el valor z que deja una área de $\alpha/2$ a la derecha de la curva normal.

$$n = \left(\frac{1.96 * 0.1}{0.05} \right)^2 = 15.3 \quad (4.12)$$

Este valor se redondea al entero siguiente, el cual corresponde a 16 mediciones por cada condición. Dado que se busca caracterizar la respuesta del acelerómetro ante una aceleración, se hace un barrido por las RPM de 0 a 300 con el fin de obtener la curva de respuesta del sensor y de esta se extraen las muestras del experimento.

Capítulo 5

Resultados y análisis

5.1 Sistema de toma de datos

Siguiendo los resultados del diseño de experimento se inicia el proceso de toma de datos del sensor para realizar la caracterización.

5.1.1 Caracterización del acelerómetro

Del sistema de caracterización se obtuvieron datos de aceleración con respecto a la variación de las rotaciones del sistema y al cambio en temperatura de la cámara. En la figura 5.1 se muestran los datos de una medición realizada en sistema con el eje X del sensor apuntando hacia el centro del sistema. El sensor se coloca de manera que el eje X quede paralelo a la fuerza centrípeta, en la figura 5.2 se muestran las direcciones positivas de los ejes.

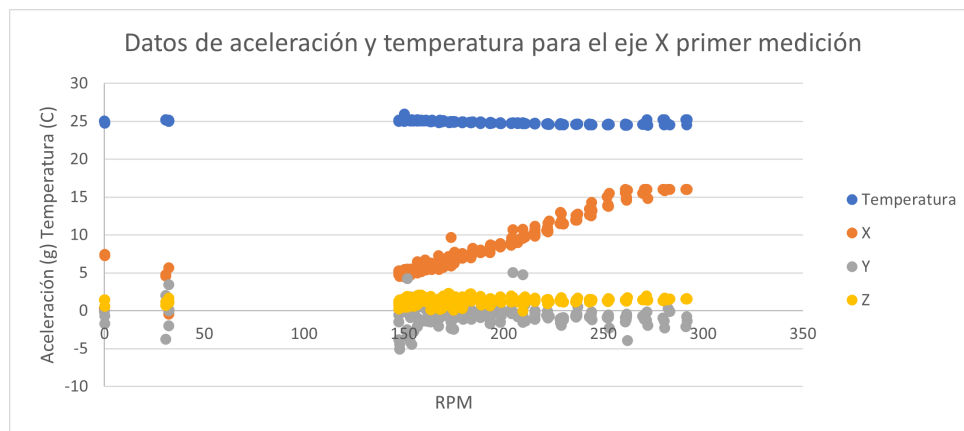


Figura 5.1: Primer medición para el eje X en el sistema de centrifugación.

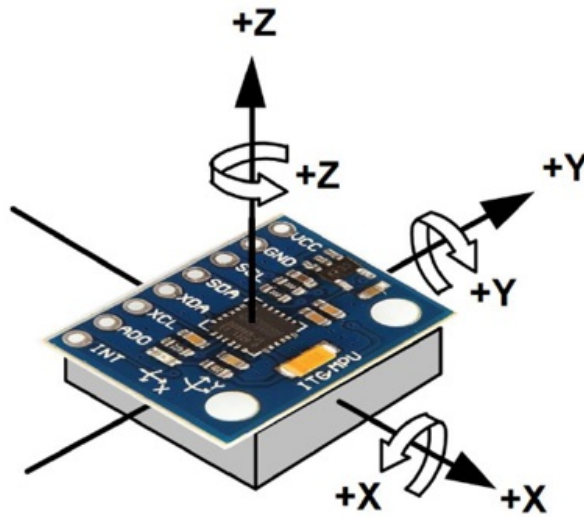


Figura 5.2: Orientación de los ejes en el MPU6050.

Para este se utilizó un código en el cual se aumentaba la velocidad de sistema hasta 270RPM, luego se disminuye la velocidad hasta que se detenga la rotación. Estos datos se visualizan a través de Excel, con el fin de poder graficar y analizar los datos, en la figura 5.3 se muestran algunos de los datos de la primer medición para el eje X del sensor. En este se tiene una columna para las revoluciones por minuto (RPM), la temperatura medida por el sensor (Temperatura), y la aceleración en cada eje (X,Y,Z).

	A	B	C	D	E
1	rpm	Temperatura	X	Y	Z
2	0	25.048	0.301	-0.108	1.44
3	32.1	25.095	0.112	0.124	1.202
4	32.1	25.048	0.024	0.029	1.135
5	32.1	25.001	-0.017	0.029	1.059
6	32.1	25.048	-0.017	0.028	1.046
7	32.1	25.048	-0.124	0.033	1.094
8	32.1	25.048	0.025	0.041	1.139
9	32.1	25.048	0.018	0.021	1.132
10	32.1	25.142	-0.429	3.413	1.658
11	32.1	25.095	5.647	-2.012	1.389
12	30.71	25.189	4.587	-3.743	1.18
13	30.71	25.142	4.732	1.985	0.683
14	147.99	25.095	4.548	-5.088	0.371
15	147.99	25.142	4.612	-4.457	1.453
16	147.38	25.095	5.171	-0.587	1.04
17	147.38	25.048	4.752	-3.803	1.272
18	151.72	25.142	4.871	4.212	1.188
19	151.72	25.189	4.568	-3.181	0.477
20	147.39	25.142	5.255	-1.448	0.261
21	147.39	25.001	5.051	0.982	0.754

Figura 5.3: Datos en obtenidos del sensor en Excel.

Los datos del Excel se grafican con el fin de obtener la curva del sensor, en la figura 5.1

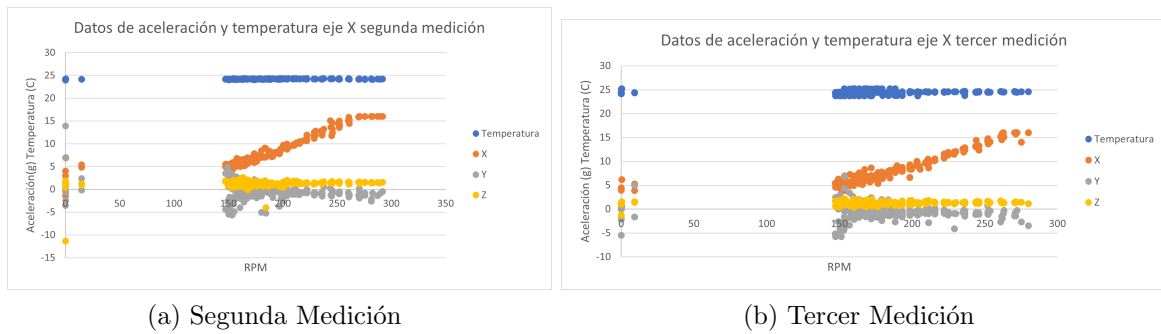


Figura 5.4: Segunda y tercer medición para el eje X en el sistema de centrifugación

se muestra como al aumentar las RPM del sistema hasta aproximadamente 270 RPM en el eje X se miden 16g únicamente, esto se da por el rango de medición del sensor. Aceleraciones que sean mayores de 16g dan como resultado un valor fijo de 16g. En esta dirección, el eje Z está siendo afectado únicamente por la aceleración de la gravedad y las vibraciones del sistema de medición. Debido a esto es que se tiene una distribución horizontal entre 0 y 2g de aceleración para el eje Z. El eje Y, se ve influenciado por la aceleración tangencial del sistema de rotación. Todos estos datos se guardan en la llave maya con el fin de tener acceso a la información generada. Durante la medición de la aceleración se realizó una medición de la temperatura ambiente a través del sensor de temperatura del MPU6050, del sensor se obtuvo una temperatura de 25.198°C promedio. El valor se valida con el sensor de temperatura TPM-10 que está dentro de la cámara térmica, para esta medición se determina un valor de 23.1°C , por lo que se obtuvo un error en la medición de 2.098°C para esta distribución.

Aunque el MPU6050 tiene un rango de medición de máximo 16g, se sobrepasa este nivel para asegurar que el sensor se está acelerando hasta su máximo y además verificar su comportamiento. En este caso se tiene una respuesta de valor fijo de 16g para valores de velocidad de más de 270 RPM. En la teoría, se menciona que los acelerómetros se diseñan para soportar más de la aceleración de su rango indicado en las hojas de datos, por lo que el sensor no se va a dañar si se sobrepasa el rango de aceleración indicado en este.

En la figura 5.4 se muestran los resultados de la segunda y tercer medición realizadas con el sistema de aceleración para el eje X del sensor. A partir de los 5g ambas curvas tienen la misma tendencia y comportamiento. Se toman 16 mediciones para cada eje con el fin de verificar la repetibilidad de la medición como se mencionó en la sección 4.3.

La temperatura ambiente de la figura 5.4 medida con el MPU6050 es de 24.059°C y 24.577°C promedio respectivamente. Mientras que con el sensor de temperatura TPM-10, para esta medición se obtuvo un valor de 23.3°C y 23.5°C , con lo que se tiene un error en la medición de 0.759°C y 1.077°C .

El resultado de las 16 mediciones para cada caso del eje X se colocan en la figura 5.5. En esta se tiene la medición de temperatura a través del TPM-10, el número de medición, el valor mínimo de RPM detectadas por el sistema y la aceleración en el eje X en ese

instante, así como el valor de la aceleración más cercano a 16g y el valor de RPM en ese instante. Con esta tabla se logra comparar las 16 mediciones y se observa como todas tienen un comportamiento similar al realizar un aumento en la aceleración y un cambio de temperatura en el sensor. La respuesta del sensor ante un cambio de temperatura permite obtener la varianza que existe en el sensor debido a factores térmicos.

		Medición	Temperatura (+/- 0.1)	RPM 0 (+/- 0.01)	Aceleración 0 (+/- 0.001)	RPM 1 (+/-0.0 1)	Aceleración 1 (+/- 0.001)
Eje X	Temperatura ambiente	1	23.1	0.00	0.301	260.90	15.530
		2	23.3	0.00	0.054	270.72	15.968
		3	23.5	0.00	0.019	262.03	15.896
		4	23.8	0.00	0.043	260.35	14.713
		5	24.1	0.00	0.028	261.30	15.672
		6	24.3	0.00	0.022	262.27	15.914
		7	24.2	0.00	0.023	270.18	15.837
		8	24.7	0.00	0.021	261.29	14.729
		9	24.6	0.00	0.022	270.87	15.967
		10	25.2	0.00	0.017	269.41	15.844
		11	25.3	0.00	0.026	260.95	15.207
		12	25.5	0.00	0.034	270.31	15.898
		13	24.5	0.00	0.024	265.59	15.534
		14	24.1	0.00	0.032	265.67	15.875
		15	24.1	0.00	0.073	265.61	15.940
		16	23.9	0.00	0.019	265.66	15.007
		Promedio	24.3	0.00	0.047	265.19	15.596
		Desviación estandar	0.7	0	0.069	4.01	0.441
	Temperatura Alta	1	46.3	0.00	0.039	265.68	15.065
		2	47.3	0.00	0.021	265.71	15.982
		3	47.5	0.00	0.040	271.54	15.256
		4	46.7	0.00	0.065	265.68	15.767
		5	46.0	0.00	0.045	265.71	15.935
		6	45.9	0.00	0.036	265.71	15.497
		7	46.5	0.00	0.070	265.62	15.521
		8	46.9	0.00	0.023	265.30	15.646
		9	45.8	0.00	0.032	265.66	15.498
		10	45.6	0.00	0.032	265.80	14.866
		11	45.9	0.00	0.042	266.85	15.908
		12	45.9	0.00	0.038	265.69	15.981
		13	45.6	0.00	0.022	265.71	15.464
		14	45.1	0.00	0.063	265.65	15.989
		15	46.1	0.00	0.034	265.75	15.902
		16	46.2	0.00	0.074	265.70	15.875
		Promedio	46.2	0.00	0.042	266.11	15.635
		Desviación estandar	0.6	0.0	0.0	1.48	0.348

Figura 5.5: Resultados obtenidos de las mediciones realizadas para el eje X.

En esta tabla se muestran los valores promedios de las mediciones realizadas y las des-

viaciones estándar en cada edición. El cambio de temperatura realizado es de 22°C , al comparar los valores promedios de las aceleraciones ante el cambio de temperatura se observa que la respuesta del eje X varía en 0.04g aunque la temperatura haya aumentado.

Para poder ratificar que la temperatura no cambia la respuesta del sensor sería necesario realizar más mediciones en otros rangos de temperatura con el fin de ampliar la estadística y obtener la respuesta en un rango más amplio.

Este proceso se repite para los ejes Y y Z del sensor, mediante el cambio de la orientación del sensor y al colocar el eje Y del sensor apuntando hacia el centro del sistema, se obtiene una respuesta como se muestra la figura 5.6

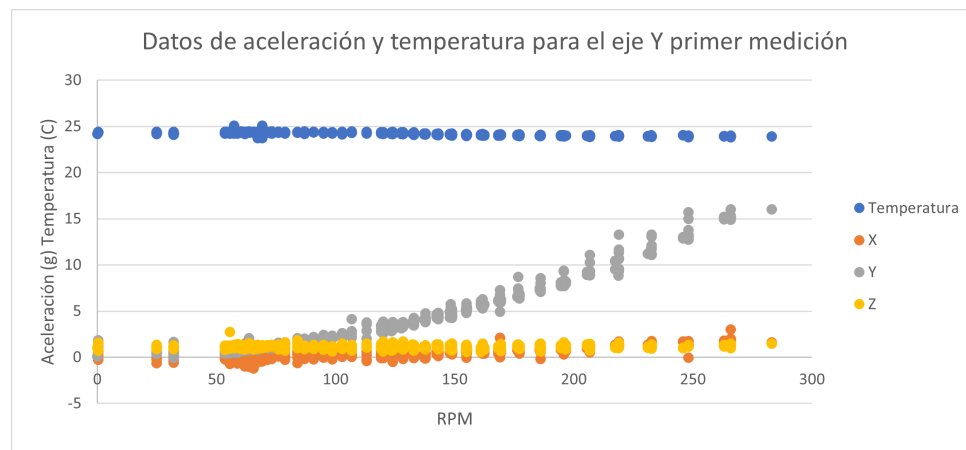


Figura 5.6: Primer medición para el eje Y en el sistema de centrifugación.

El resultado de las 16 mediciones para cada caso del eje Y se colocan en el apéndice A.8, en la figura A.1. En esta se tiene la medición de temperatura a través del TPM-10, el número de medición, el valor mínimo de RPM detectadas por el sistema y la aceleración en el eje Y en ese instante, así como el valor de la aceleración más cercano a 16g y el valor de RPM en ese instante. Con esta tabla se logra comparar las 16 mediciones y se observa como el resultado en la aceleración 1 tienen un comportamiento similar al realizar un aumento en la aceleración y un cambio de temperatura en el sensor. La respuesta del sensor ante un cambio de temperatura permite obtener la varianza que existe en el sensor debido a factores térmicos.

En esta tabla del apéndice se muestran los valores promedios de las mediciones realizadas y las desviaciones estándar en cada edición. El cambio de temperatura realizado es de 22.3°C , al compara los valores promedios de las aceleraciones ante el cambio de temperatura se observa que la respuesta del eje Y varía en 0.08g aunque la temperatura haya aumentado.

Al colocar el eje Z del sensor apuntando hacia el centro del sistema y se obtiene una respuesta como la de la figura 5.7

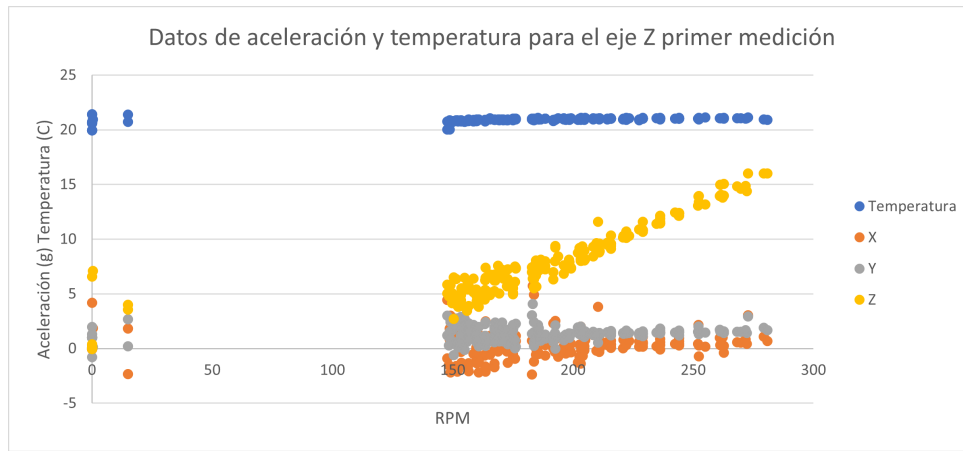


Figura 5.7: Primer medición para el eje Z en el sistema de centrifugación.

El resultado de las 16 mediciones para cada caso del eje Z se colocan en el apéndice A.8, en la figura A.2. En esta se tiene la medición de temperatura a través del TPM-10, el número de medición, el valor mínimo de RPM detectadas por el sistema y la aceleración en el eje Z en ese instante, así como el valor de la aceleración más cercano a 16g y el valor de RPM. Con esta tabla se logra comparar las 16 mediciones y se observa que el cambio de temperatura realizado es de 24.6°C , al comparar los valores promedios de las aceleraciones ante el cambio de temperatura se observa que la respuesta del eje Z varía en 0.291g ante un aumento en la temperatura.

Al comparar los cambios de aceleración entre los 3 ejes a temperatura ambiente se tiene que el eje X tiene un promedio de 15.596g , el eje Y de 15.277g y el eje Z de 15.241g . Estos valores muestran que el sensor tiene una respuesta estable para los 3 ejes a temperatura ambiente.

Mientras que al comparar los 3 ejes a 46°C se tiene en el eje X un promedio de 15.635g , el eje Y de 15.197g y el eje Z de 14.950g . En este caso al aumentar la temperatura la respuesta del eje Z es la que varía su respuesta en comparación con los ejes X y Y.

5.1.2 Respuesta ante cambio de temperatura

Con respecto al cambio de temperatura se toma como referencia el valor del TPM-10, pero cabe recalcar que este sensor no se encuentra junto a el acelerómetro, sino que está acoplado a la base del motor, por lo que existe una incertidumbres de medición debido a la distancia entre este y el acelerómetro. Con el fin de poder obtener una temperatura constante dentro de la cámara de aislamiento térmico se hace una etapa de precalentamiento, en la cual se hace llegar la cámara a 47°C con la secadora de cabello en velocidad alta. Una vez se llegue a esta temperatura se coloca la velocidad de la secadora en bajo y se inicia el proceso de toma de datos y aceleración del sensor. Esto con el fin de que las mediciones se hagan en un rango de temperatura cercano a los 46°C . Cabe destacar que, se utiliza una temperatura de 46°C para las mediciones ya que el motor a pasos tiene

un rango de temperatura de operación de -10°C a 50°C , por lo que al sobrepasar los 50°C el motor se des sincroniza y se apaga.

En la figura 5.8 se tienen los datos del Offset del sensor debido al cambio de temperatura. En este caso se tiene una diferencia de $\pm 25\text{mg}$ entre los eje X,Y y el eje Z. El eje Z evidencia una diferencia de $\pm 60\text{mg}$ al variar la temperatura, mientras que los eje X y Y cambian en $\pm 35\text{mg}$.

ZERO-G OUTPUT						
Initial Calibration Tolerance	X and Y axes		± 50		mg	1
	Z axis		± 80		mg	
Zero-G Level Change vs. Temperature	X and Y axes, 0°C to $+70^{\circ}\text{C}$		± 35			
	Z axis, 0°C to $+70^{\circ}\text{C}$		± 60		mg	

Figura 5.8: Offset en el MPU6050 debido al cambio en al temperatura [22]

Al verificar los cambios en las aceleraciones cuando se tienen 0 RPM se tiene para el eje X una variación de 0.002g , para el eje Y una disminución de 0.017g y para el eje Z una disminución de 0.951g . De esta respuesta se puede ver el cambio que tiene el sensor cuando no está siendo acelerado y hay una variación en la temperatura. El eje Z es el que disminuye su respuesta en comparación con los ejes X y Y.

En la figura 5.9 se tienen las gráficas de la respuesta del sensor ante un aumento en la temperatura, en esta se muestra la respuesta de cada eje al ser caracterizados en el sistema de centrifugación. Se observa que las curvas de los 3 ejes tienen la misma tendencia al ser acelerados hasta 16g . Con esto se demuestra que la respuesta del sensor en sus 3 ejes es simétrica y los cambios existentes son mínimos al variar la temperatura aproximadamente 22°C . Con esto se comprueba que el 100% de los datos recopilados del acelerómetro real de 3 ejes se mantienen dentro de los límites establecidos para la variable de temperatura.

5.1.3 Respuestas obtenidas

En la figura 5.10 se tienen la gráfica de la aceleración en función de las RPM para la primer medición del eje X, en este se extrae la ecuación que describe el comportamiento de la aceleración en función de las revoluciones del sistema de caracterización. La ecuación lineal de tendencia no calza totalmente con los datos obtenidos por el sensor. La ecuación obtenida corresponde a una aproximación de la tendencia del sensor, tomando los puntos máximos y mínimos de la aceleración. Además, para cada medición realizada se obtiene una ecuación diferente según el comportamiento que tuvo el sensor en cada proceso de aceleración.

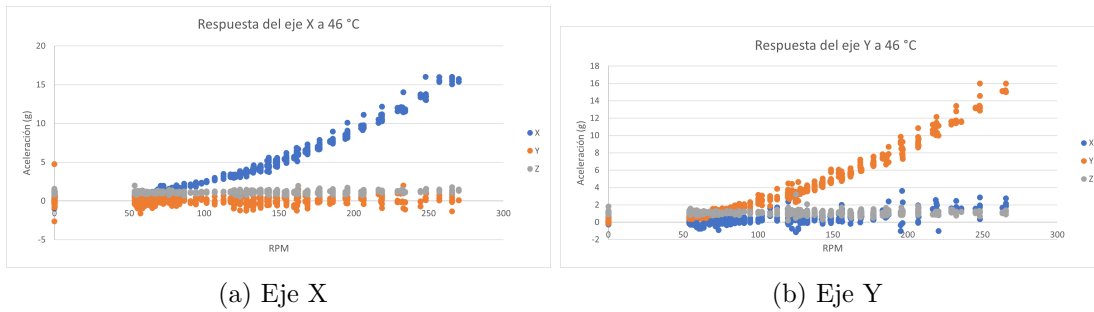
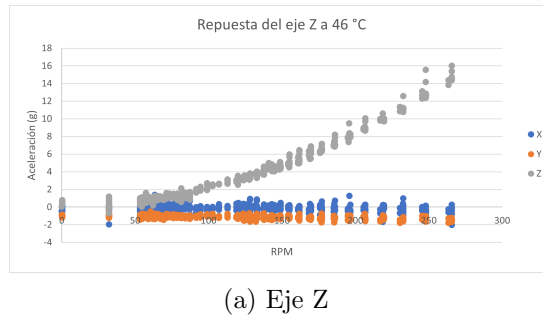


Figura 5.9: Respuesta del eje X, Y y Z ante un aumento de temperatura.



(a) Eje Z

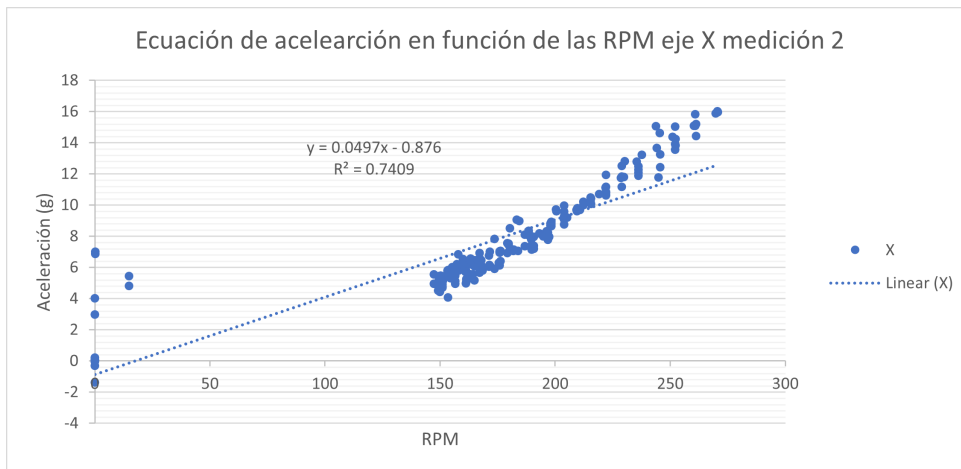


Figura 5.10: Gráfica con la ecuación de la aceleración en función de las RPM para medición 2 del eje X.

En la figura 5.11 se tienen las 16 respuestas del sensor, en este se puede ver la tendencia de la respuesta del sensor ante un aumento de temperatura y como se comportó el sensor durante las 16 mediciones. En esta se observa como las curvas tienen una tendencia creciente y que ninguna sobrepasa el valor de 16g, ya que como se mencionó el sensor está configurado para utilizar un valor de 16g cuando se esté fuera del rango de medición. Además, se tienen puntos aislados fuera de la curva para RPM menores a 50. Esto se da por ruido y saltos en el sensor cuando el motor está rompiendo la inercia de arranque.

En la figura 5.12 se tienen la gráfica de la aceleración en función de las RPM para la segunda medición del eje Y.

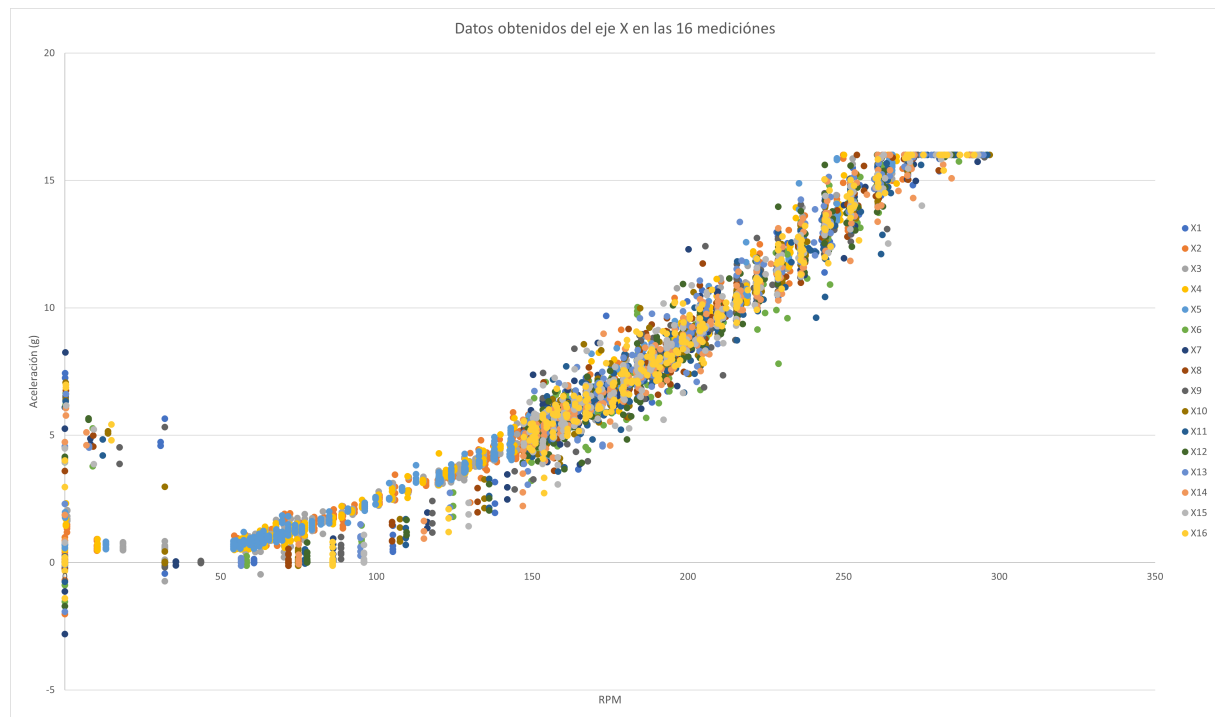


Figura 5.11: Gráfica con la respuesta de las 16 mediciones para el eje X.

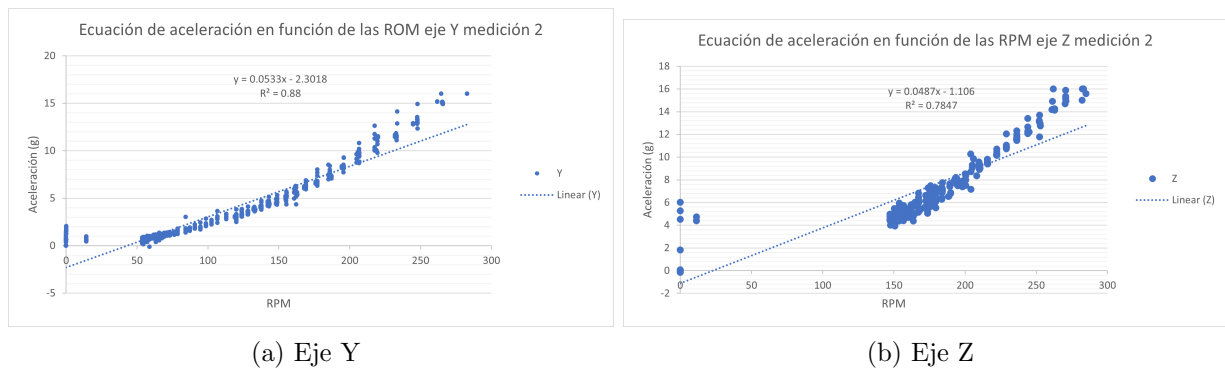


Figura 5.12: Gráfica con la ecuación de la aceleración en función de las RPM para medición 2 del eje Y Z.

En la figura A.3 y A.4 del apéndice A.9 se tiene las 16 respuestas del sensor para los ejes X y Y, en este se puede ver la tendencia de la respuesta ante un aumento de aceleración y su comportamiento durante las 16 mediciones.

5.2 Simulación

El sistema simulado se implementó en la FPGA y se obtiene como salida una aceleración relativa a la temperatura seleccionada por el usuario y el rango de aceleración establecida. La implementación final corresponde a la aplicación de las ecuaciones del sistema de aceleración centrífuga sobre el sensor ADXL377. Se obtienen salidas analógicas y digitales

para cada eje, las cuales que representan un valor de aceleración en voltios como se mencionó en la sección 4.2.

Los modos de trabajo implementados en el simulador corresponden a los principales modos de falla encontrados para dispositivos MEMS: error mecánico, electrónico y de comunicación, los cuales se mencionan en la sección 4.2.2. Estos se configuran como respuestas constantes, ya que no se tiene una respuesta específica para estos tipos de errores y debido a las limitaciones de memoria de la FPGA del myRIO esta implementación permite obtener una respuesta como error bajo las condiciones del sistema. Se configura un error fuera del rango posible para la aplicación.

Para el error por comunicación se coloca una respuesta fija de 0.5mv con el fin de interferir la respuesta normal de la aplicación, para el error mecánico por falla en la masa de prueba se configura una respuesta fija de 0.7mV, ya que este valor asegura que durante la falla siempre va a estar fuera del rango del sensor; el error por vibración del sensor se configura principalmente para obtener una respuesta en el eje Z del sensor, ya que bajo ninguna condición de aceleración este tienen una varianza al menos que sea por un cambio en la temperatura. El error por vibración da una respuesta de 1V. En la figura 5.13 se muestra la interfaz de usuario del simulador a través de API. En esta se configura la temperatura, aceleración y modos de sistema. Además, este posee un botón de paro de la simulación que pone todas las salidas en cero, como si el sistema estuviera apagado.

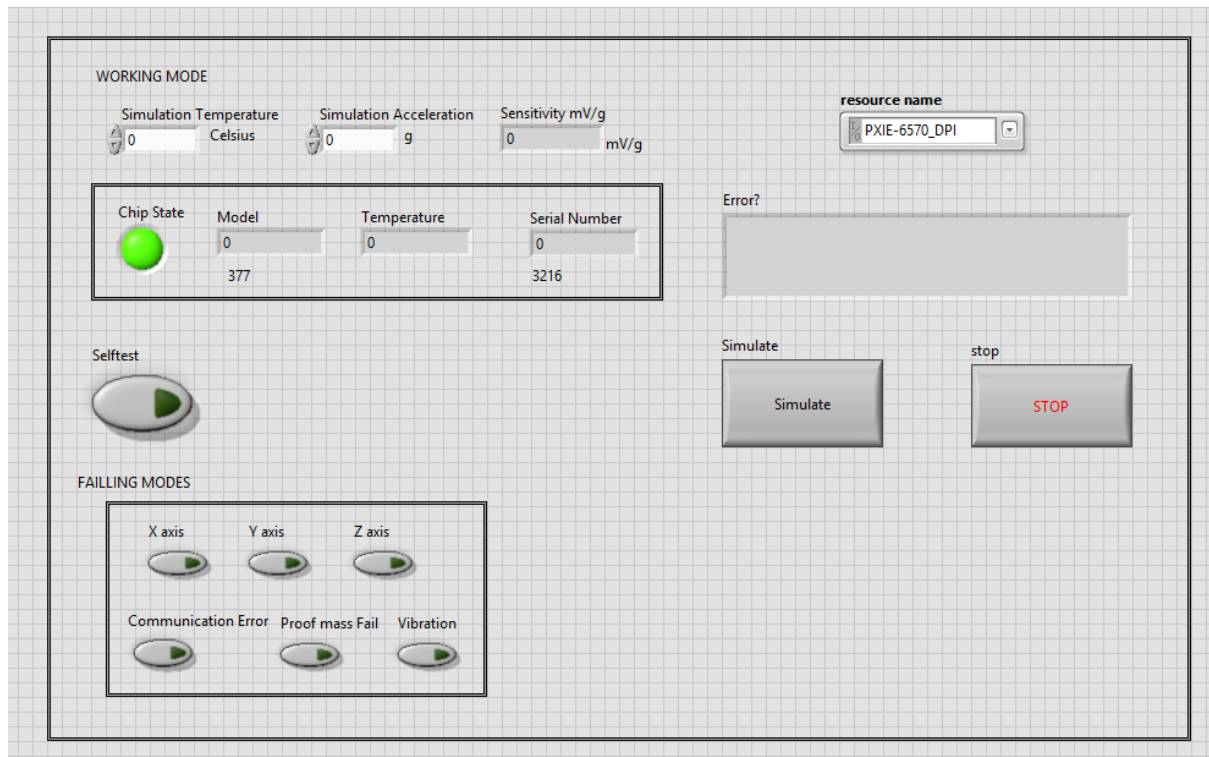


Figura 5.13: Interfase de usuario del simulador a través de API.

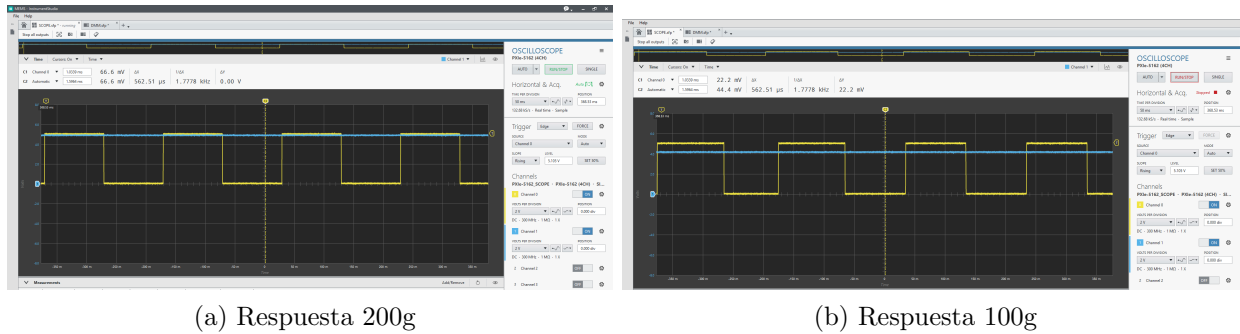


Figura 5.14: Respuesta del simulador para una aceleración a 20°C.

5.2.1 Respuesta sistema simulado en la FPGA

En la figura 5.14 se muestra la respuesta del simulador para una aceleración de 0g a una temperatura de 20°C. Se obtiene una respuesta analógica y una respuesta digital del valor de la aceleración que se simula en la FPGA.

Con ayuda del InstrumentStudio se extraen los valores de tensión para las salidas analógicas y se tabulan para verificar su respuesta. En la figura 5.15 se tiene las mediciones realizadas para el simulador en un rango de 0g a 16g y su respuesta ante un cambio de temperatura.

Condición		Datos simulados		
Temperatura (C)	Aceleración (g)	X (+-0.001V)	Y (+-0.001V)	Z (+-0.001V)
10	0	2.930	2.930	2.947
	1	2.942	2.930	2.947
	2	2.955	2.930	2.947
	3	2.969	2.930	2.947
	4	2.981	2.930	2.947
	5	2.996	2.930	2.947
	6	3.008	2.930	2.947
	7	3.021	2.930	2.947
	8	3.035	2.930	2.947
	9	3.048	2.930	2.947
	10	3.060	2.930	2.947
	11	3.075	2.930	2.947
	12	3.087	2.930	2.947
	13	3.102	2.930	2.947
	14	3.114	2.930	2.947
	15	3.127	2.930	2.947
20	0	2.933	2.932	2.950
	1	2.942	2.932	2.950
	2	2.955	2.932	2.950
	3	2.968	2.932	2.950
	4	2.981	2.932	2.950
	5	2.994	2.932	2.950
	6	3.008	2.932	2.950
	7	3.021	2.932	2.950
	8	3.035	2.932	2.950
	9	3.048	2.932	2.950
	10	3.061	2.932	2.950
	11	3.075	2.932	2.950
	12	3.088	2.932	2.950
	13	3.101	2.932	2.950
	14	3.114	2.932	2.950
	15	3.127	2.932	2.950
46	0	2.936	2.936	2.961
	1	2.942	2.936	2.961
	2	2.955	2.936	2.961
	3	2.969	2.936	2.961
	4	3	2.936	2.961
	5	2.994	2.936	2.961
	6	3.008	2.936	2.961
	7	3.021	2.936	2.961
	8	3.035	2.936	2.961
	9	3.048	2.936	2.961
	10	3.06	2.936	2.961
	11	3.074	2.936	2.961
	12	3.088	2.936	2.961
	13	3.100	2.936	2.961
	14	3.114	2.936	2.961
	15	3.127	2.936	2.961
80	0	2.942	2.940	2.975
	1	2.942	2.940	2.975
	2	2.955	2.940	2.975
	3	2.969	2.940	2.975
	4	2.981	2.940	2.975
	5	2.994	2.940	2.975
	6	3.008	2.940	2.975
	7	3.021	2.940	2.975
	8	3.035	2.940	2.975
	9	3.047	2.940	2.975
	10	3.061	2.940	2.975
	11	3.074	2.940	2.975
	12	3.087	2.940	2.975
	13	3.099	2.940	2.975
	14	3.114	2.940	2.975
	15	3.126	2.940	2.975
	16	3.140	2.940	2.975

Figura 5.15: Respuesta del simulador en un rango de 0g a 16g ante un cambio de temperatura.

En la figura A.5 y A.6 del apéndice A.10 se tienen las tablas con todas las mediciones realizadas para el simulador. Se puede observar como la respuesta ante el cambio de temperatura del sistema simulador no sobrepasa los 0.001g.

En la figura 5.16 se tiene la respuesta del simulador en los 3 ejes ante una aceleración en la centrifugadora. Se observa como esta respuesta tienen la misma tendencia que obtenida mediante el sistema de caracterización.

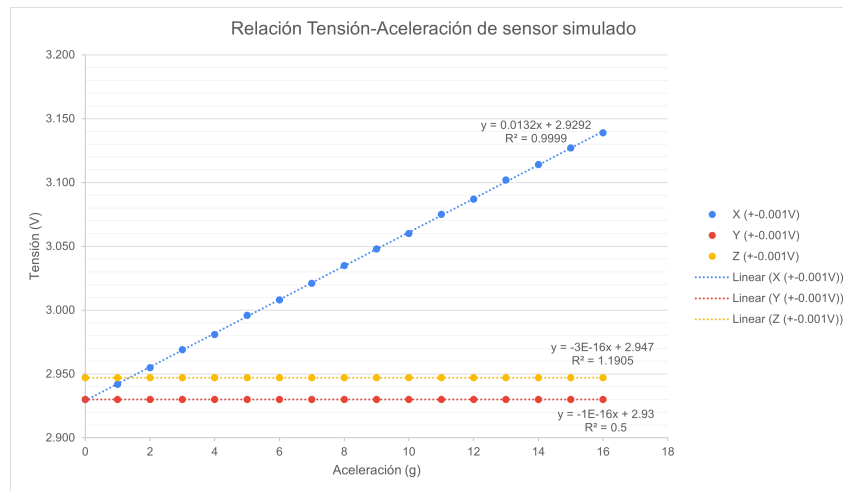
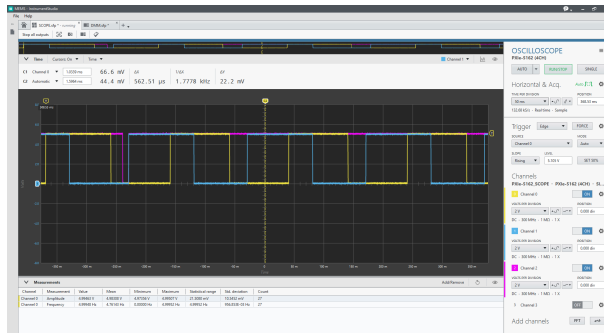
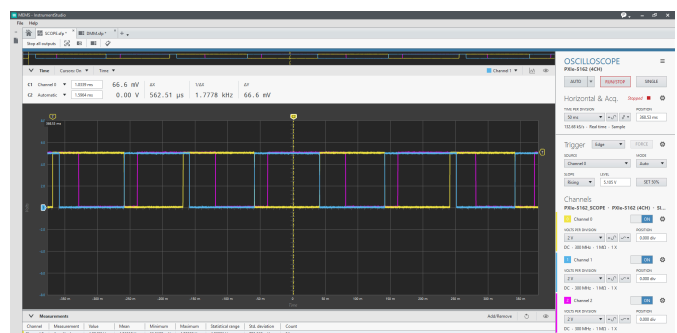


Figura 5.16: Respuesta del simulador ante una aceleración en la centrifugadora.

En la figura 5.17 se tiene la respuesta del simulador en los 3 ejes ante una aceleración en la centrifugadora. Se observa como esta respuesta tienen la misma tendencia que la obtenida mediante el sistema de caracterización.



(a) Respuesta 200g



(b) Respuesta 20g

Figura 5.18: Salida PWM del simulador para una aceleración a 20°C.

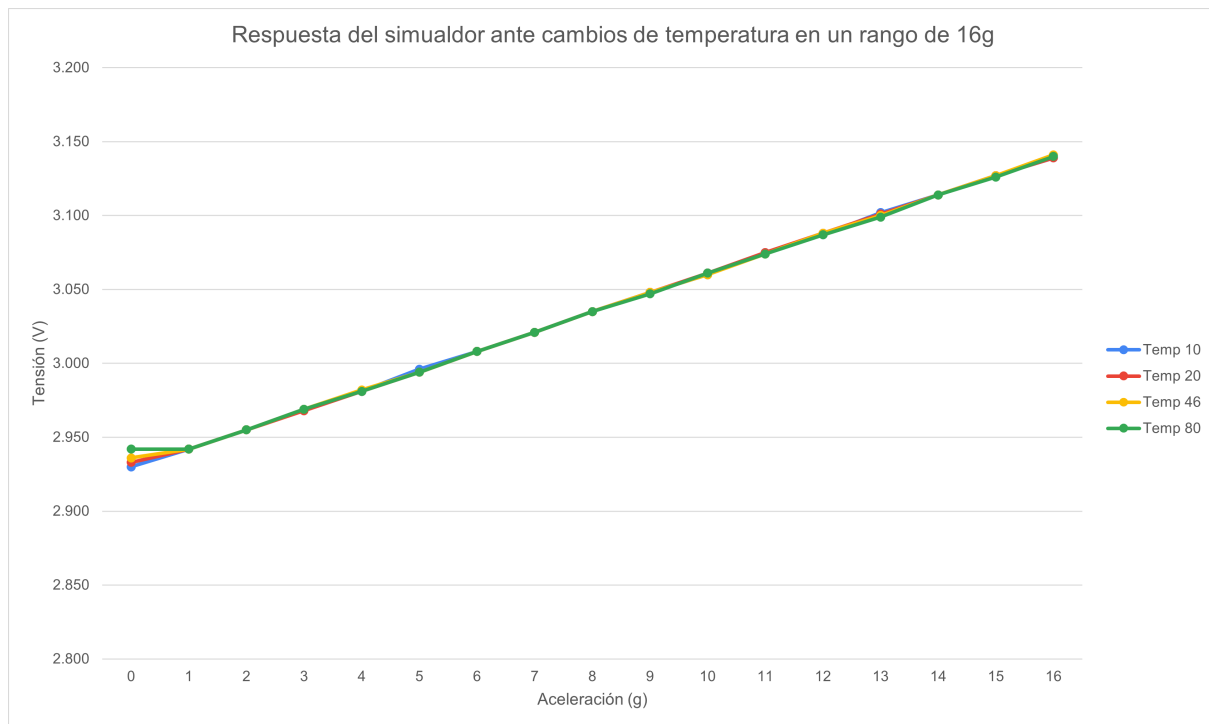


Figura 5.17: Respuesta del simulador ante cambios de temperatura.

En las salidas digitales se tiene al respuesta del PWM, en este el cambio se ve reflejado en el periodo de trabajo según sea la aceleración del simulador. En la figura 5.18 se muestra la respuesta del simulador para un cambio de aceleración de 200g y 20g. En este se observa como el periodo de la respuesta morada disminuyó al pasar a 20g.

Al verificar la respuesta del simulador se obtiene que el sistema posee una precisión de al menos 90% en los datos obtenidos de la temperatura ante una confirmación del estado chip, el número de serie y solicitud de modo de trabajo o modo de fallo.

5.2.2 Sistema real y simulado

Para comparar el sensor simulado y el real se debería tener el mismo modelo de sensor, pero en este caso debido a la dificultad de conseguir sensores en el mercado nacional, se utiliza el MPU6050 que permite alcanzar hasta 16g de aceleración como sensor a simular. Se utiliza el sensor ADXL377 para establecer la respuesta del simulador, ya que como requerimientos de la empresa el simulador debe variar en un rango amplio, de por ejemplo $\pm 200g$ en comparación con el rango del MPU6050 de $\pm 16g$.

Con el fin de comparar la respuesta del simulador y el sensor real se configura el simulador en valores de aceleración dentro del rango de 0g a 16g y con temperaturas entre 20°C y 46°C. Estos datos se grafican y se comparan las respuestas del sensor real y el sensor simulado.

Dado que al desarrollar esta prueba de concepto se debe ampliar la estadística y cambiar la respuesta del simulador de ser necesario. Para esto se debe realizar la toma de datos en rangos más amplios de temperatura, comparar la respuesta y modificar el simulador de ser necesario.

En la figura 5.19 se muestra una tabla con los datos simulados para un rango de 0 a 16g, a 20°C y 46°C. En la última columna se observa el error que hay entre el valor condicional del simulador (segunda columna) y la respuesta obtenida en las salidas del simulador al convertirlas a aceleración (penúltima).

Condición		Datos simulados				
Temperatura (C)	Aceleración (g)	X (+0.001V)	Y (+0.001V)	Z (+0.001V)	Aceleración Simulada (g)	Error del simulador (g)
20	0	2.933	2.932	2.950	-0.157	0.157
	1	2.942	2.932	2.950	0.596	0.404
	2	2.955	2.932	2.950	1.683	0.317
	3	2.968	2.932	2.950	2.771	0.229
	4	2.981	2.932	2.950	3.859	0.141
	5	2.994	2.932	2.950	4.946	0.054
	6	3.008	2.932	2.950	6.118	-0.118
	7	3.021	2.932	2.950	7.205	-0.205
	8	3.035	2.932	2.950	8.377	-0.377
	9	3.048	2.932	2.950	9.465	-0.465
	10	3.061	2.932	2.950	10.552	-0.552
	11	3.075	2.932	2.950	11.724	-0.724
	12	3.088	2.932	2.950	12.811	-0.811
	13	3.101	2.932	2.950	13.899	-0.899
	14	3.114	2.932	2.950	14.987	-0.987
	15	3.127	2.932	2.950	16.074	-1.074
	16	3.139	2.932	2.950	17.078	-1.078
46	0	2.936	2.936	2.961	0.094	-0.094
	1	2.942	2.936	2.961	0.596	0.404
	2	2.955	2.936	2.961	1.683	0.317
	3	2.969	2.936	2.961	2.855	0.145
	4	3	2.936	2.961	3.942	0.058
	5	2.994	2.936	2.961	4.946	0.054
	6	3.008	2.936	2.961	6.118	-0.118
	7	3.021	2.936	2.961	7.205	-0.205
	8	3.035	2.936	2.961	8.377	-0.377
	9	3.048	2.936	2.961	9.465	-0.465
	10	3.06	2.936	2.961	10.469	-0.469
	11	3.074	2.936	2.961	11.640	-0.640
	12	3.088	2.936	2.961	12.811	-0.811
	13	3.100	2.936	2.961	13.815	-0.815
	14	3.114	2.936	2.961	14.987	-0.987
	15	3.127	2.936	2.961	16.074	-1.074
	16	3.141	2.936	2.961	17.246	-1.246

Figura 5.19: Respuesta del simulador ante una aceleración en la centrifugadora.

Al tomar 5 de las mediciones del eje X del sensor MPU6050 al azar a una temperatura de 46°C y comparar estas respuestas con las del simulador se obtiene la figura 5.20. En esta se utiliza la respuesta más cercana del simulador para comparar con el sensor real. Además, se calcula el error en la medición y el porcentaje de error entre ambos valores.

Temperatura (C)	Sensor Real	Sensor simulado		Error (g)	% Error
	Valor medido (+/- 0.001 g)	Valor simulado (+/- 0.001 g)	Valor solicitado (+/- 1 g)		
46.000	16.000	16.074	15	-0.074	0.46
	13.519	13.815	13	-0.296	2.14
	6.026	6.118	6	-0.092	1.50
	5.052	4.946	5	0.106	2.14
	1.521	1.683	2	-0.162	9.63

Figura 5.20: Respuesta del simulador ante una aceleración en la centrifugadora.

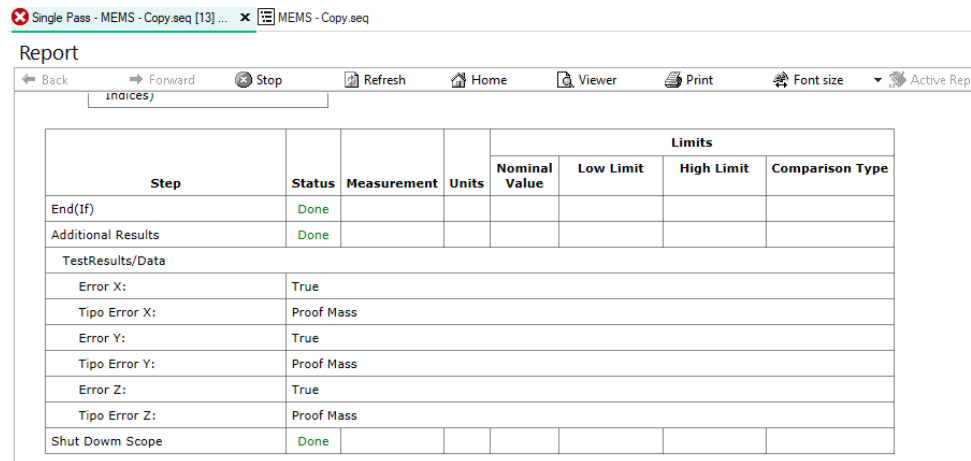
5.3 Verificación modos en TestStand a través de API

Con el fin de tener un DUT del acelerómetro se programa en TestStand una secuencia de validación. En este se prueba que el "Selftest" funcione y que los modos de fallo fallen adecuadamente. En la figura 5.21 se muestra la secuencia implementada. Se utiliza el osciloscopio como medio de medición, las mediciones realizadas se verifican automáticamente en TestStand con el fin de indicar si la respuesta de generada por la simulación posee una falla o corresponde a una respuesta del modo de trabajo del sensor.

MEMS - TestSequence.seq		
Steps: MainSequence		
STEP	DESCRIPTION	SETTINGS
- Setup (1)		
Initialize Scope	Action, InitializeScope.vi	Result Recording: Disabled
<End Group>		
- Main (22)		
ChipState	Numeric Limit Test, x == 1, DataCaptureTest.vi	Post Action
SerialNumber	Numeric Limit Test, x == 3216, DataCaptureTest.vi	
Model	Numeric Limit Test, x == 377, DataCaptureTest.vi	
VoltageLevel X	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Post Action, Additional Results
VoltageLevel Y	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Additional Results
VoltageLevel Z	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Additional Results
if	Locals.ErrorX==True	
VoltageLevel X Communication	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Post Action, Additional Results
VoltageLevel X ProofMass	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results
VoltageLevel X Vibration	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results
End		
if	Locals.ErrorY==True	
VoltageLevel Y Communication	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Post Action, Additional Results
VoltageLevel Y ProofMass	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results
VoltageLevel Y Vibration	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results
End		
if	Locals.ErrorZ==True	
VoltageLevel Z Communication	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Post Action, Additional Results
VoltageLevel Z ProofMass	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results
VoltageLevel Z Vibration	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results
End		
Additional Results	Records 6 Additional Result(s)	
<End Group>		
- Cleanup (1)		
Shut Down Scope	Action, CloseScope.vi	
<End Group>		

Figura 5.21: Secuencia de prueba en TestStand.

Como respuesta de la secuencia se tiene un reporte, en este se muestra la condición de los 3 ejes, en la figura 5.22 se extrae la sección con los resultados. En el caso que exista un error en algún eje se tendrá como respuesta "True" en "Error #(X,Y,Y)", esta respuesta es para cada eje (X,Y,Y). Seguidamente se tiene el tipo de error para cada eje: "Tipo Error #(X,Y,Y)". El resultado de tipo de error puede ser: *Communication*, *Proof Mass* o *Vibration*.



Step	Status	Measurement	Units	Limits			
				Nominal Value	Low Limit	High Limit	Comparison Type
End(If)	Done						
Additional Results	Done						
TestResults/Data							
Error X:	True						
Tipo Error X:	Proof Mass						
Error Y:	True						
Tipo Error Y:	Proof Mass						
Error Z:	True						
Tipo Error Z:	Proof Mass						
Shut Down Scope	Done						

Figura 5.22: Ejemplo de errores obtenidos en el reporte

Al implementar esta secuencia de prueba se logra que la verificación de los modos de prueba del sistema cumpla al 100% los requerimientos de validación del chip.

5.4 Problemas encontrados e imprevistos

Con respecto a la implementación del simulador, inicialmente se tuvieron problemas con el protocolo de comunicación **SPI**, ya que no se tenían conocimientos previos de los programas ni componentes utilizados para realizar la comunicación. Se tomaron partes de código de programas de prueba de LabView y se solicitó ayuda a los mentores de la empresa cuando se tuvieron problemas en la implementación, con esto se obtuvo una comunicación mediante protocolo SPI funcional.

Con respecto a la **memoria de la FPGA**, se tuvieron problemas una vez que se insertaron los modos de falla. Inicialmente el código lograba compilar con un 95% de su memoria. Pero debido al uso de recursos que tenía la implementación de los modos de falla y las salidas PWM el espacio disponible para realizar la compilación bajó a un 90%. Esto generó retrasos ya que al realizar algunos cambios se sobrepasaba este límite y la compilación fallaba, por lo que muchos elementos dentro de las ecuaciones y los métodos de comunicación tuvieron que ser reescritos para no sobrepasar este límite. Un ejemplo corresponde a las dirección de los eje que fallan y los modos, inicialmente se utilizaba un arreglo de datos binarios para indicar cuales ejes y modos estaban seleccionados. El valor se enviaba como un número decimal y en la FPGA se convertía nuevamente a un

número binario que se extraía en un arreglo para verificar los elementos seleccionados. Pero debido a el consumo de recursos que poseen los arreglos se tuvo que cambiar esta implementación, que era la más sencilla, a una que requería de la implementación de más elementos de control en la simulación. En la figura 5.23 se muestra el porcentaje de LUT (LookUp Table, por sus siglas en inglés) utilizadas en la FPGA para el simulador, si este valor de LUT sobrepasa un 90%, el código no compila, por lo que no se puede ejecutar en la FPGA.

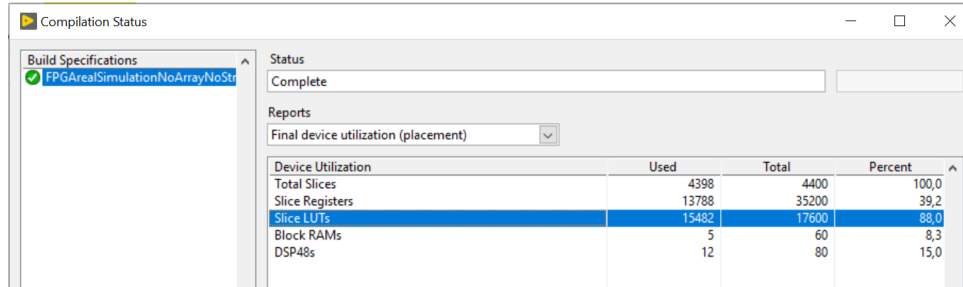


Figura 5.23: Porcentaje de LUT utilizadas en la FPGA para el simulador.

En el caso de los pines digitales de salida del **myRIO**, se tuvieron problemas ya que los valores de salida no son compatibles con tecnología TTL, por lo que se debió buscar una forma fácil de poder convertir el nivel de tensión de las salidas digitales a valores TTL. Para esto se utiliza la tarjeta convertidora de nivel HW-221 con el fin de aumentar la tensión de salida.

Al implementar el PWM que sale por los pines digitales, se tuvo como problema que el periodo en alto del PWM no cambiaba visualmente pero su tiempo programático si mostraba el cambio. Se propuso verificar las tierras del myRIO para descartar un error de conexión y se realizaron pruebas para verificar este error. La conexión a tierra no era el problema, era el algoritmo implementado, ya que todas las salidas estaban en el mismo ciclo *while loop*. Al colocar cada salida en un ciclo independiente se logra conseguir la salida correcta del PWM.

La salidas analógicas del myRIO también se tuvieron que escalar debido a que las salidas del sensor simulado no eran compatibles con tecnología TTL, en este caso se hizo el escalado de manera programática. Además, los valores de tensión de salida generados por el código no estaban escalados para los valores del myRIO, por lo que se multiplicaron por 1000 para nivelar la tensión de salida.

En el **sistema de adquisición de datos** se tuvieron problemas al implementar el código en modo auto ejecución, ya que el archivo que se debía editar tenia permisos, por lo que se tuvo que reiniciar el sistema operativo de la **Raspberry Pi** para poder editar el archivo. También con respecto a la toma de datos, se tuvieron problemas para implementar el **address switching**, debido a los protocolos de comunicación, por lo que en vez de tomar los datos de 3 sensores a la vez se tomaron datos de un sensor a la vez. Además, se tuvieron problemas con el sensor MPU6050, inicialmente el sensor se encontraba configurado para

trabajar en el rango de 2g, por lo que aunque el sensor estuviera siendo acelerado a más de este valor no podía reportar el aumento de la aceleración. En la figura 5.24 se muestra una de las pruebas realizadas para la medición con el sensor, donde se visualiza que aunque las rotaciones aumentaran la aceleración reportaba en su máximo solo 2g. También se tuvieron problemas con los cables que comunicaban al sensor y la Raspberry Pi, debido a que uno de los cables no estaba haciendo bien contacto y se tuvo que medir nuevamente los datos obtenidos para el eje Z del sensor durante una de las pruebas de medición.



Figura 5.24: Primeros datos obtenidos para el eje X en el sistema de centrifugación.

Al momento de realizar la caracterización con cambio en la temperatura, se utiliza una temperatura de 46°C ya que el motor a pasos tiene un rango de temperatura de operación de -10°C a 50°C. Por lo que al sobrepasar los 50°C el motor se des sincroniza y se apaga. Esto genera limitaciones al momento de evaluar la respuesta del sensor ante un cambio de temperatura, ya que no se puede llegar hasta el máximo de temperatura de trabajo del sensor.

5.5 Cumplimiento de las métricas

Con el fin de validar el cumplimiento de las métricas del proyecto se verifica cada una y se compara con su valor ideal. En la figura 5.25 se muestran las métricas y los valores reales obtenidos, así como su método de validación y justificación. La lista indica los resultados de las pruebas realizadas y el cumplimiento de las métricas.

	Factores de Influencia	Importancia	Unidades	Valor / Rango	Prueba realizada	Resultado
1	Grado de semejanza entre simulación y acelerómetro real ante cambios en temperatura y modificaciones en funciones	5	Documento / Prueba	N.A.	Comparar la respuesta del acelerómetro real y la simulado bajo una temperatura específica	N.A
2	Sistema varía ante cambios temperatura y permite modificar funciones	5	Documento / Prueba	N.A.		N.A
3	Rango de temperatura de trabajo	5	Númérico	$-40 < x < 125$	Se prueba la simulación en la temperatura máxima, mínima y 2 valores de temperatura con respuesta conocida dentro del rango	$-40 < x < 85$
4	Cantidad de rangos de trabajo de la simulación	5	Númérico	$2 < x < 3$	Verificación mediante el conteo directo de los rangos, el usuario puede elegir cualquier valor de aceleración entre -200g y +200g	-200g y +200g
5	Valor de tensión entrada del sistema	5	Voltios	$3.3V < x < 5V$	Verificación mediante la medición directa la tensión de alimentación del sistema, se mide mediante el PXI la tensión de entrada del sistema y es de 14.70V; el valor de la señal SPI es de 4.88V	4.88V
6	Chip validado mediante secuencia: chip presente, modelo del chip y número de serie	5	Binario	Pasa	Verificación en la interfaz del indicador del modelo, número de serie y presencia del chip	Pasa
7	Validar modo trabajo y modo falla para cada salida	5	Documento / Prueba	N.A.	Verificación de las salidas ante el modo de trabajo y el modo de falla mediante TestStand	N.A
8	Valor tensión de salida del sistema es compatible con TTL	5	Voltios	$4.75V < x < 5.25V$	Verificación mediante la medición directa la tensión de salida, la tensión máxima de salida es de 4.996 V	4.996 V
9	Salida es analógica y digital para cada salida	5	Binario	Cumple	Verificación mediante la medición directa de cada salida	Cumple
10	Datos y estado chip se visualizan en la interfaz	5	Binario	Cumple	Verificación en la interfaz del indicador de datos y estado del chip	Cumple
11	Cantidad de canales individuales	5	Númérico	$3 < x < 6$	Conteo directo del número de canales individuales	Cumple
12	FPGA del myRio simula el sistema	5	Binario	Cumple	Verificación del uso de recursos en el myRio al compilar el código, se obtienen que un 88% de los recursos está siendo utilizado en la implementación del simulador	Cumple
13	API se comunica, mide y prueba el chip	5	Binario	Cumple	La FPGA envía y recibe datos mediante la API	Cumple
14	Tarjeta electrónica posee conexiones necesarias para su funcionamiento	5	Binario	Cumple	Existencia de una tarjeta que contenga los elementos necesarios para el funcionamiento del sistema	Cumple
15	Sistema simulado está dentro de un encapsulado que lo aisle y resguarde	5	Binario	Cumple	Existe un encapsulado con la FPGA y la tarjeta electrónica que simulan el sistema	Cumple
16	Ruido y vibraciones son consideradas en el sistema de caracterización aislado y controlado del medio	4	Binario	Cumple	Se realizan varias mediciones en el sensor y se comparan las respuestas para extraer las variaciones por ruido.	Cumple
17	Cantidad de rangos de aceleración que el sistema de caracterización logra	5	Númérico	$2 < x < 4$	Conteo directo del número de rangos de aceleración que puede lograr el sistema de caracterización, el sistema puede variar su velocidad desde aproximadamente 0g y hasta más de 16g.	Desde 0g hasta 16g
18	Sistema de medición guarda los datos de aceleración obtenidos	4	Binario	Cumple	Los datos se pueden acceder a través de la llave maña luego de caracterizar el sensor	Cumple
19	Sistema de medición se puede reusar y cambiar acelerómetros	4	Binario	Cumple	Conteo de la cantidad de veces que se utiliza el sistema y facilidad de acoplar u desacoplar sistema de sensores	Cumple
20	Tiempo de adquisición es menor o igual a 2 semanas	4	Días	$7 < x < 18$	El sensor MPU6050 se consiguió inmediatamente el día de la compra. No se tuvo que esperar a ningún envío o retraso del mismo. Ya lo tenían en la tienda.	0 días
21	Costo de cada acelerómetro es menor o igual a 15 dólares	5	\$USD	$15 < x < 20$	Precios de compra de uno de los sensores.	\$8.96 unidad

Figura 5.25: Lista con los resultados de las pruebas realizadas.

En el apéndice A.11 se tiene el manual de Usuario para el simulador. Este documento indica los pasos para compilar el código, ejecutarlo y realizar pruebas sobre el sistema simulado.

5.6 Síntesis del análisis

El sistema simulador de un acelerómetro MEMS de 3 ejes corre en una FPGA y trabaja con rangos de temperatura de -40 a 85 °C. Se desarrolló un software que simula el comportamiento de un acelerómetro MEMS de tres (3) ejes y muestre como resultado un valor confiable de temperatura y el estado del chip ante una solicitud del usuario. La comunicación se realizó mediante el protocolo SPI, el cual envía y recibe datos de la FPGA donde se encuentra el sistema simulado. Para el error por comunicación se obtiene una respuesta fija de 0.5mv, al respuesta del error de mecánico de falla en la masa de prueba es de 0.7mV, finalmente el error por vibración da una respuesta de 1V.

A través de un proceso de toma de datos para un acelerómetro real de 3 ejes se obtienen valores de aceleración desde 0.010g hasta 16g en el sensor MPU6050. El cambio de aceleración en el eje X tiene un promedio de 15.596g, en el eje Y de 15.277g y el eje Z de 15.241g. Estos valores muestran que el sensor tiene una respuesta estable para los 3 ejes a temperatura ambiente.

En 0 RPM se tiene para el eje X un cambio de 0.002g, para el eje Y una disminución de 0.017g y para el eje Z una disminución de 0.951g. De esta respuesta se puede ver el cambio que tiene el sensor cuando no está siendo acelerado y hay un cambio de temperatura. El eje Z es el que disminuye su respuesta en comparación con los ejes X Y.

Se diseña un sistema simulado de un acelerómetro MEMS de 3 ejes en una FPGA, el sensor es simulado bajo las condiciones de una aplicación de centrifugación. La FPGA del myRIO no permite compilar códigos si se sobrepasa el 90% de sus LUT, por lo que se tuvo que simplificar muchas secciones de código con el fin de utilizar una menor cantidad de recursos y que el código pudiera ser ejecutado en la FPGA.

Los tres modos de falla del sistema simulado se asocian a fallas reales de los dispositivos MEMS y poseen una respuesta fija para cada estado, dado la falta de memoria de la FPGA. Se logra obtener la respuesta del sensor en modo de trabajo, ante una aplicación de un sistema de aceleración centrífuga.

La respuesta ante el cambio de temperatura del sistema simulador no sobrepasa los 0.001V con lo que se logra obtener una respuesta estable para cambios de temperatura en el rango de operación del sensor.

Se valida el simulador para modo de trabajo y modo de falla, mediante API (Application Programming Interface) y el *software* de validación de DUT, TestStand. Según sea la configuración del usuario la secuencia de prueba puede pasar o fallar al realizar las lecturas de las salidas del sistema simulado.

La validación de datos se da mediante el sistema de caracterización del sensor, el cual permite variar la aceleración del sensor y guardar estos datos en una llave maya con el fin de visualizarlos. El proceso de toma de datos y variables de influencia para el acelerómetro real de 3 ejes se acopla en una Raspberry Pi, ya que esta incluye los puestos USB para conectar la llave maya y no requerir de otros circuitos para almacenar la información; los puertos USB están disponibles en la computadora lo que facilita la visualización de los datos obtenidos por el sistema.

Adicionalmente, se presenta en el apéndice A.12 el manual de uso del sistema de caracterización.

Capítulo 6

Análisis Económico

6.1 Costo Material

En este apartado se muestran los elementos utilizados en la implementación del proyecto y sus precios. Dado que algunos componentes ya se tenían a la mano se colocan los precios a los que se conseguirían actualmente.

6.1.1 Costo Implementación simulador

En la figura 6.1 se muestran los costos realizados como parte de la implementación del sistema simulado. En este se colocan los precios de todos los componentes, equipos y *software* utilizados para poder desarrollar el sistema simulado.

Distribuidor	Material	Divisa	Precio unitario	Unidades	Total Colones
NI	Dell latitude 7420 Laptop	\$	1749	1	1147011.69
	myRio	\$	1200	1	786972.00
	labView Software	\$	1100	1	721391.00
	TestStand Software	\$	878.78	1	576312.71
MicroJPM	8-Channel Logic Level Converter - Bi-Directional [Generic]	\$	5.95	1	3902.07
Tercero	Encapsulado del simulador	₡	192402	1	192402.00
				Total	3427991.47

Figura 6.1: Costos de la implementación del sistema simulado

Se obtienen los *software*, computadora y myRIO de la empresa. La carcasa que encapsula al simulador y todos los componentes internos de este se obtienen de un tercero que lo desarrolló en la empresa, se coloca el precio que el tercero estimó. La tarjeta convertidora de nivel se consigue por aparte para poder obtener salidas digitales en formato TTL. Con esto se obtiene un costo del simulador de 3 427 991.47 colones para su implementación.

6.1.2 Costo Implementación sistema caracterización

En la figura 6.2 se muestran los precios de los componentes que ya se tenían a disposición para la implementación. En este caso los *software* utilizados para realizar los códigos son de acceso gratuito, por lo que no se incluyen en los costos.

Distribuidor	Material	Divisa	Precio unitario	Unidades	Total Colones
Propio	Fuente 12V S-240-24 Switch Power Supply 24VDC @ 10A	\$	29.95	1	19641.51
	Arduino	\$	35.95	1	23576.37
	Raspberry Pi 3	\$	98.95	1	64892.40
	Sensor Hall	\$	2.2	1	1442.78
	Placas de soldar	\$	1.8	1	1180.46
	Stepper Stepper Motor with Cable [ROB-09238] (NEMA 17)	\$	24.5	1	16067.35
	TB6600 Stepper Motor Driver Controller 9~40V @ 4A	\$	19.95	1	13083.41
	LM2596 DC-DC Buck Converter Step-Down Power Module Output 1.25V-35V	\$	4.95	1	3246.26
	Tetrix kit Piezas de aluminio	\$	406.25	1	266422.81
	Base de madera	₡	23250	1	23250.00
	Temometro	\$	7.95	1	5213.69
	Secadora de pelo	\$	19.93	1	13070.29
	Piezas de madera para cámara	₡	9995	1	9995.00
	Tacómetro Smart Sensor Digital Photo Tachometer modelo AR926	\$	25.95	1	17018.27
	Tapa PVC	₡	920	1	920.00
				Total	479020.60

Figura 6.2: Costos de los materiales y componentes previamente disponibles

En la figura 6.3 se muestran los precios de las partes y piezas que se compraron durante el desarrollo del proyecto, con el fin de poder implementar el sistema de caracterización.

Distribuidor	Material	Divisa	Precio unitario	Unidades	Total Colones
MicroJPM	Holder Baterias	₡	1268.75	2	2537.50
	Cable micro USB	₡	2220.34	1	2220.34
	Jumpers Macho-Macho	₡	1153.4	2	2306.80
	Socket Pin	₡	720.87	1	720.87
	Pin Hembra	₡	403.69	1	403.69
Fausto Jara LTDA	Holder Baterias	₡	619	2	1400.00
Aceros Cartago	Barra de aluminio	₡	565.48	2	1278.00
Tornillos Toyman	Prisioneros 1/4	₡	34.67	4	138.68
	Prisioneros 1/4	₡	39.43	4	157.72
	Prisioneros 1/4	₡	34.3	4	137.20
				Total	11300.80

Figura 6.3: Costos de la implementación del simulado

6.1.3 Costo total del material

El costo de la implementación del simulador se suma al costo de la implementación del sistema de caracterización para obtener el costo total del material invertido en el proyecto.

$$Total = 3427991.47 + 479020.60 + 11300.80 = 3915312.87 \text{ colones} \quad (6.1)$$

6.2 Costo del recurso Humano

El costo del recurso humano corresponde al precio por hora laborada del desarrollador del proyecto y al inversión en viáticos, como transporte y alimentación. En este caso se tiene un total de 662 horas trabajadas y un subsidio de 2 500 000 colones, por lo que el costo por hora corresponde a :

$$\frac{2500000 \text{ Colones}}{662 \text{ Horas}} = 3776.43 \frac{\text{Colones}}{\text{Hora}} \quad (6.2)$$

En la figura 6.4 se muestra el costo de los viáticos incurridos durante el desarrollo del proyecto.

Total Transporte(₡)	420215
Alimentación (₡)	500000
Total Viáticos (₡)	920215

Figura 6.4: Costo de los viáticos

Para obtener el costo de fabricación del proyecto se suma el costo material y el costo del recurso humano invertido en el mismo. Con lo que se obtiene un costo de 7 335 527.87 colones, al cual se le suma el 30% para obtener el precio de venta o precio total del proyecto. El 30% corresponde a 2 200 658.36 colones. El precio de venta del proyecto hacia la empresa sería de 9 536 186.23 colones.

La empresa no va a tener una ganancia monetaria directa del proyecto ya que este sistema es un dispositivo que no se va a vender, sino que va a ser utilizado por la empresa únicamente. Con el desarrollo de este proyecto la empresa tiene un ahorro en instrumentación y con la a implementación del mismo la empresa le puede demostrar a la casa matriz la capacidad que tienen de hacer desarrollos propios con el fin de mejorar el uso de recursos.

Como ganancia la empresa obtiene una herramienta desarrollada propiamente para poder atender las preguntas de los clientes ante problemas que tengan en aplicaciones de sus dispositivos, también permite realizar pruebas de concepto fuera del sitio de implementación y validar resultados de sensores de aceleración genéricos. Además, permite reducir los costos al no tener que comprar acelerómetros ni tener que invertir tiempo y dinero en realizar los circuitos de conexión de los mismos para obtener datos con el fin de probar los sistemas.

Capítulo 7

Conclusiones, Recomendaciones y Trabajos Futuros

7.1 Conclusiones

Al desarrollar el proyecto se logró implementar un software que simule el comportamiento de un acelerómetro MEMS de tres ejes que muestre como resultado un valor confiable de temperatura y el estado del chip ante una solicitud del usuario en la aplicación de centrifugación. Además, se implementó una API para comunicar al usuario con el simulador de acelerómetro MEMS, el cual puede trabajar en un rango de aceleraciones entre 0g y 200g.

1. Se desarrolló el sistema simulador de un acelerómetro MEMS de 3 ejes que corre en una FPGA y trabaja con rangos de temperatura de -40 a 85 °C. Este corresponde a un software que simula el comportamiento de un acelerómetro MEMS de tres (3) ejes y muestre como resultado un valor confiable de temperatura y el estado del chip ante una solicitud del usuario.
2. La comunicación se realizó mediante el protocolo SPI, el cual envía y recibe datos de la FPGA donde se encuentra el sistema simulado.
3. Para el error por comunicación se obtiene una respuesta fija de 0.5mv, al respuesta del error de mecánico de falla en la masa de prueba es de 0.7mV, finalmente el error por vibración da una respuesta de 1V.
4. A través de un proceso de toma de datos para un acelerómetro real de 3 ejes se obtienen valores de aceleración desde 0.010g hasta 16g en el sensor MPU6050. El cambio de aceleración en el eje X tiene un promedio de 15.596g, en el eje Y de 15.277g y el eje Z de 15.241g. Estos valores muestran que el sensor tiene una respuesta estable para los 3 ejes a temperatura ambiente.

5. En 0 RPM se tiene para el eje X un cambio de 0.002g, para el eje Y una disminución de 0.017g y para el eje Z una disminución de 0.951g. De esta respuesta se puede ver el cambio que tiene el sensor cuando no está siendo acelerado y hay un cambio de temperatura. El eje Z es el que disminuye su respuesta en comparación con los ejes X y Y.
6. El 100% de los datos recopilados del acelerómetro real de 3 ejes se mantienen dentro de los límites establecidos para la variable de temperatura.
7. Los tres modos de falla del sistema simulado se asocian a fallas reales de los dispositivos MEMS y poseen una respuesta fija para cada estado, dado la falta de memoria de la FPGA.
8. Se logra obtener la respuesta del sensor en modo de trabajo, ante una aplicación de un sistema de aceleración centrífuga.
9. Al verificar la respuesta del simulador se obtiene que el sistema posee una precisión de al menos 90% en los datos obtenidos de la temperatura ante una confirmación del estado chip, el número de serie y solicitud de modo de trabajo o modo de fallo.
10. Al implementar la secuencia de prueba se logra que la verificación de los modos de prueba del sistema cumpla al 100% los requerimientos de validación del chip.
11. La respuesta ante el cambio de temperatura del sistema simulador no sobrepasa los 0.001V con lo que se logra obtener una respuesta estable para cambios de temperatura.
12. Se concluye que en el caso de la variable de temperatura, en el rango de valores estudiado, no es determinante.
13. Se valida el simulador para modo de trabajo y modo de falla, mediante API (Application Programming Interface) y el *software* de validación de DUT, TestStand.

7.2 Recomendaciones y Trabajos Futuros

Como ampliación de este trabajo se propone implementar el convertidor PWM dentro de los recursos de la FPGA, ya que de esta manera se asegura que el sistema simulado en su totalidad este siendo ejecutado por la FPGA y no se tengan recursos compartidos del myRIO.

Con respecto al sistema de toma de datos, se propone retomar los datos de aceleración con un sensor que tenga un rango de medición mayor a 16g, así como cambiar el motor a uno que tenga una temperatura de funcionamiento mayo de 50°C. Esto con el fin de obtener una respuesta más amplia ante diversos cambios de temperatura y aceleraciones en la magnitudes de los 100g.

7.3 Valoración personal

Al realizar este proyecto, los conocimientos obtenidos durante el transcurso de la implementación abarcan áreas como el diseño, la programación y manufactura; así como aprendizaje de implementar código en diversos lenguajes de programación y el uso de controladores y sensores para la toma de datos.

Asimismo, el poder estar en la empresa permitió vivir en primera persona algunas de las exigencias del mundo profesional, como lo fueron la preparación de demostraciones, reuniones para la verificación de avances y entrevistas con diversos miembros de equipos para aclarar dudas.

Finalmente, el desarrollo de este proyecto permitió implementar conocimientos adquiridos y no adquiridos durante el tiempo de estudio en la academia, por lo que este trabajo compila diversos temas de cursos que se han llevado durante la carrera.

Bibliografía

- [1] Adafruit. Adafruit feather huzzah esp8266 overview [online]. 2022 [visitado el 17 de junio de 2022]. URL <https://learn.adafruit.com/adafruit-feather-huzzah-esp8266>.
- [2] National Aeronautics and Space Administration. Learning launchers: A spin on things – centrifuge [online]. 2019 [visitado el 17 de junio de 2022]. URL https://www.nasa.gov/audience/foreducators/stem-on-station/learning-launchers_centrifuge.
- [3] National Aeronautics and Space Administration. 20-g centrifuge (8.84-meter radius centrifuge) [online]. 2020 [visitado el 17 de junio de 2022]. URL <https://www.nasa.gov/ames/research/space-biosciences/20-g-centrifuge>.
- [4] Arijit1080. Mpu6050-with-raspberry-pi [online]. 2020 [visitado el 17 de junio de 2022]. URL <https://github.com/Arijit1080/mpu6050-with-Raspberry-Pi>.
- [5] C. de la Fuente. *Guía práctica para la publicación de Datos Abiertos usando API's*. Secretaría de Estado de Digitalización e Inteligencia Artificial del Ministerio de Asuntos Económicos y Transformación Digital.
- [6] Analog Devices. Adxl377 [online]. 2022 [visitado el 17 de junio de 2022]. URL <https://www.analog.com/en/products/adxl377.html#product-overview>.
- [7] DFROBOT. Introduction tb6600 arduino stepper motor driver [online, visitado el 17 de junio de 2022]. URL https://wiki.dfrobot.com/TB6600_Stepper_Motor_Driver_SKU_DRI0043.
- [8] Google. Google imagenes [online]. 2022 [visitado el 17 de junio de 2022]. URL <https://images.google.com/>.
- [9] R.C. Hibbeler. *Ingeniería mecánica - Estática Decimosegunda edición*. PEARSON EDUCACIÓN, México, 2010.
- [10] T. Hsu. *MEMS MICROSYSTEMS Design and Manufacture*. Tata McGraw-Hill, 2002.

- [11] Direct Industry. Simulador de señales de salida para acelerómetro 4830b [online]. 2022 [visitado el 17 de junio de 2022]. URL <https://www.directindustry.es/pr od/endevco/product-5413-1743226.html>.
- [12] National Instruments. "user guide and specification ni myrio-1900". myRioSpec-376047c Guía de Usuario, [revisado el 03 de marzo de 2022].
- [13] National Instruments. Interfacing to the digital pattern instrument or digital waveform instrument [online]. 2020 [visitado el 17 de junio de 2022]. URL <https://www.ni.com/es-cr/support/documentation/supplemental/06/interfacing-to-the-digital-pattern-instrument-or-digital-wavefor.html>.
- [14] National Instruments. Understanding the i2c two wire bus interface with ni labview [online]. 2020 [visitado el 17 de junio de 2022]. URL <https://www.ni.com/es-cr/innovations/white-papers/13/understanding-the-i2c-two-wire-bus-interface-with-ni-labview.html>.
- [15] National Instruments. Understanding the spi bus with ni labview [online]. 2020 [visitado el 17 de junio de 2022]. URL <https://www.ni.com/es-cr/innovations/white-papers/09/understanding-the-spi-bus-with-ni-labview.html>.
- [16] National Instruments. National instruments [online]. 2021 [visitado el 17 de junio de 2022]. URL <https://www.ni.com/en-ca.html>.
- [17] National Instruments. National instruments [online]. 2021 [visitado el 17 de junio de 2022]. URL <https://www.ni.com/es-cr/shop/labview.html>.
- [18] National Instruments. Introducción a la arquitectura pxi [online]. 2022 [visitado el 17 de junio de 2022]. URL <https://www.ni.com/es-cr/innovations/white-papers/14/introduction-to-the-pxi-architecture.html>.
- [19] National Instruments. Pxie-6570 instrumento de patrón digital pxi [online]. 2022 [visitado el 17 de junio de 2022]. URL <https://www.ni.com/es-cr/support/model.pxie-6570.html>.
- [20] National Instruments. ¿qué es el software instrumentstudio? [online]. 2022 [visitado el 17 de junio de 2022]. URL <https://www.ni.com/es-cr/shop/electronic-test-instrumentation/application-software-for-electronic-test-and-instrumentation-category/instrumentstudio.html>.
- [21] National Instruments. ¿qué son los instrumentos de patrones digitales pxi? [online]. 2022 [visitado el 17 de junio de 2022]. URL <https://www.ni.com/es-cr/shop/electronic-test-instrumentation/digital-instruments/what-are-digital-pattern-instruments.html>.
- [22] InvenSense. "mpu-6000 and mpu-6050 product specification revision 3.3". PS-MPU-6000A-00 Hoja de datos, Mayo, 2012 [revisado el 03 de marzo de 2022].

- [23] Micro JPM. Tpm-10 thermometer - temperature gauge lcd digital [online]. 2022 [visitado el 17 de junio de 2022]. URL <https://www.microjpm.com/products/thermometer-temperature-gauge-lcd-digital/>.
- [24] V. Kaajakari. *Practical MEMS*. Small Gear Publishing, 2009.
- [25] L. Kelion. Escasez de microchips: por qué hay una crisis de semiconductores y cómo puede afectar [online]. 2021 [visitado el 17 de junio de 2022]. URL <https://www.bbc.com/mundo/noticias-55955119>.
- [26] C. Liu. *Foundations of MEMS — 2nd ed.* Pearson Educación Prentice Hall, Upper saddle river, 2012.
- [27] J. Lopez. Hay un plan para acabar con la crisis de los semiconductores: en qué consiste y cuándo llegará el final según los fabricantes de chips [online]. 2021 [visitado el 17 de junio de 2022]. URL <https://www.xataka.com/componentes/hay-plan-para-acabar-tesis-semiconductores-que-consiste-cuando-llegara-final-fabricantes-chips#:~:text=El%20problema%20es%20que%20las, posible%20 someterlas%20a%20m%C3%A1s%20estr%C3%A9s>.
- [28] D. Montgomery. *Introduction to Statistical Quality Control Sixth Edition*. Wiley, 2009.
- [29] BBC News Mundo. URL https://www.youtube.com/watch?v=Rm8ulgsEaPk&ab_channel=BBCNewsMundo. Qué es la crisis de los contenedores y cómo afecta al comercio mundial y a tu bolsillo BBC Mundo, 15 septiembre 2021. [Video].
- [30] M. Méndez. La escasez de chips ya tiene un gran perjudicado: por qué acabarás pagándola tú [online]. 2021 [visitado el 17 de junio de 2022]. URL https://www.elconfidencial.com/tecnologia/2021-07-29/procesadores-chips-intel-qualcomm-tsmc_3203696/.
- [31] Efecto Naim. URL https://www.youtube.com/watch?v=PfYtLRrvPEc&ab_channel=EfectoNaim. Crisis de contenedores: cómo la pandemia agravó años de negligencia, 6 diciembre 2021. [Video].
- [32] Insider News. URL https://www.youtube.com/watch?v=LiH-SuDh0yU&ab_channel=InsiderNews. Decoding The Supply Chain Crisis And The Everything Shortage Decoded, 1 Diciembre 2021. [Video].
- [33] Raspberry Pi. Raspberry pi 3 model b [online, visitado el 17 de junio de 2022]. URL <https://www.raspberrypi.com/products/raspberry-pi-3-model-b/>.
- [34] S. Myers y K. Ye R. Walpole, R. Myers. *Probabilidad y estadística para ingeniería y ciencias Novena Edición*,. PEARSON EDUCACIÓN, México, 2012.
- [35] S.S. Rao. *Vibraciones mecánicas Quinta edición*. PEARSON EDUCACIÓN, México, 2012.

- [36] E. Rucci. *FPGA: ¿los procesadores del futuro?* Facultad de Informática, UNLP. Año 3, Número 6, página 46. URL http://sedici.unlp.edu.ar/bitstream/handle/10915/64231/Documento_completo.pdf?sequence=1.
- [37] N. Ramirez y C. Acosta S. Plaza. *API de servicios web orientados a accesibilidad*. Universidad Complutense de Madrid, 2015.
- [38] MicroJPM S.A. Microjpm página principal [online]. 2021 [visitado el 17 de junio de 2022]. URL <https://www.microjpm.com/>.
- [39] Rohde Schwarz. Entendiendo el uart [online]. 2022 [visitado el 17 de junio de 2022]. URL https://www.rohde-schwarz.com/lat/productos/prueba-y-medicion/osciloscopios/educational-content/entendiendo-el-uart_254524.html.
- [40] NXP Semiconductor. "mma8452q, 3-axis, 12-bit/8-bit digital accelerometer". MMA8452Q Hoja de datos, Abril, 2016 [revisado el 03 de marzo de 2022].
- [41] Hansford Sensor. Simulador de acelerómetro [online]. 2022 [visitado el 17 de junio de 2022]. URL <https://www.hansfordsensors.com/es/products/installation-products/accelerometer-simulator/>.
- [42] J. Serrano. *Introduction to FPGA design*. CERN, Geneva, Switzerland. URL <https://cds.cern.ch/record/1100537/files/p231.pdf>.
- [43] C. Sisterna. *Field Programmable Gate Arrays*. DEA. URL http://dea.unsj.edu.ar/sisdig2/Field%20Programmable%20Gate%20Arrays_A.pdf.
- [44] R. Marangoni y J. Lienhard T. Beckwith. *Mechanical Measurements Sixth Edition*. Pearson Learning Solutions, Boston, 2011.
- [45] B. Balachandra y E. Magrab. *Vibrations Second Edition*. CENGEL Learning, 2009.
- [46] R. Serway y J. Jewett. *Física para ciencias e ingeniería Volumen 1. Séptima Edición*. Cengage Learning Editores, 2008.
- [47] D. Fonseca y M. Sequeira. "on mems reliability and failure mechanisms", 2011. Department of Mechanical Engineering, The University of Alabama, Tuscaloosa. International Journal of Quality, Statistics, and Reliability; Volume 2011, Article ID 820243; doi:10.1155/2011/820243.
- [48] I. López y P. López. "sistema automatizado de calibración de sensores de desplazamiento, fuerza y aceleración", 2006. Tesis, Universidad Nacional Autónoma de México, Ciudad Universitaria, México, D.F.

Apéndice A

Apéndice

A.1 Tabla completa con los enunciados del cliente

Tipo	Original/Suplemento	Enunciado del Cliente	Necesidad Integrada	Enunciado correcto Necesidad	Escala	Comentario
Requisitos del Sistema	General	Crear y controlar un simulador programable de un acelerómetro MEMS capaz de dar salidas confiables	El sistema simulado debe responder de manera confiable como si fuera un acelerómetro MEMS	El sistema simulado de manera confiable un acelerómetro MEMS	5	
		Debe ser capaz de dar salidas de diferentes resoluciones de bits	El sistema simulado variará su respuesta según la temperatura registrada y pueden agregarse nuevas funcionalidades	El sistema simulado variará su respuesta según la temperatura registrada y pueden agregarse nuevas funcionalidades	5	No necesariamente tiene que dar salidas de diferentes resoluciones de bits (que se complican más)
		La temperatura simulada de operación debe ser entre -40 y +125 celcius	El sistema simulado trabaja en un rango de temperatura entre -40 y +125 celcius	El sistema simulado trabaja en un rango de temperatura entre -40 y +125 celcius	5	
		Al menos 3 rangos de medición, con su respectiva sensibilidad en mV/g	El sistema simulado se configura en al menos 3 rangos diferentes e indica la sensibilidad en mV/g	El sistema simulado indica al menos 3 rangos de diferentes	5	
		Alimentación del sistema entre 3.3V y 5V, compatible con fuentes externas a 3.3V y 5V	El sistema simulado es alimentado con una fuente de entre 3.3V y 5V	El sistema simulado indica e indica está presente, el modo del chip y el número de serie del mismo	5	
Modo Trabajo	Modo Trabajo	Se debe tener un registro a memoria de la temperatura y el modo de funcionamiento del chip y al número de serie	El sistema simulado indica e indica está presente, el modo del chip y el número de serie del mismo	El sistema simulado indica e indica está presente, el modo del chip y el número de serie del mismo	5	
		Debe tener modo de trabajo y modo de configuración y gestión a través de un puerto de comunicación	El sistema simulado se puede configurar en modo de trabajo o modo de configuración a través de un puerto de comunicación	El sistema simulado se puede configurar en modo de trabajo o modo de configuración a través de un puerto de comunicación	5	
		Se debe tener un registro a memoria de la temperatura y el modo de funcionamiento del chip y al número de serie	El sistema simulado indica e indica está presente, el modo del chip y el número de serie del mismo	El sistema simulado indica e indica está presente, el modo del chip y el número de serie del mismo	5	
		Las salidas deben ser mediante protocolo I2C, SPI, UART (CAN), tanto analógico como digital	El sistema simulado tiene salidas tanto analógicas como digitales, mediante algún protocolo I2C	El sistema simulado tiene salidas analógicas y digitales para cada uno de sus pines, mediante algún protocolo I2C	5	
		Debe registrar la temperatura y el modo de funcionamiento del chip	El sistema simulado registra la temperatura y el modo de funcionamiento del chip	El sistema simulado registra la temperatura y el modo de funcionamiento del chip	5	
Empaques	Software	Cada cosa debe tener un canal necesario	El sistema simulado posee un canal necesario para cada cosa	El sistema simulado posee un canal necesario para cada cosa	5	
		Asociaciones simuladas en la PPGA del micro	El sistema simulado utiliza API para comunicarse, medir y probar el chip	El sistema simulado utiliza API para comunicarse, medir y probar el chip	5	
		Tarjetas seleccionadas con las condiciones necesarias	El sistema simulado posee una tarjeta electrónica con las conexiones necesarias para su funcionamiento	El sistema simulado posee una tarjeta electrónica con las conexiones necesarias para su funcionamiento	5	
		Exposición de cada cosa y gestión al sistema	El sistema simulado está dentro de un encapsulado que lo protege y respalda	El sistema simulado está dentro de un encapsulado que lo protege y respalda	5	
		Manual de conexiones con los modos de operación y especificaciones	El sistema simulado posee una manual de operación con los modos de trabajo y las especificaciones del sistema	El sistema simulado valida el chip mediante una manual de operación con los modos de trabajo y las especificaciones del sistema	5	
Sistema Caracterización	Ultrasonidos y Ruido	Se requiere para validar el chip	El sistema simulado posee una biblioteca para validar el chip	El sistema simulado valida el chip mediante una biblioteca	3	Integrado e interesante
		El ruido y las vibraciones no se debe al procesamiento	El sistema de caracterización compensa los errores de ruido y vibraciones en el sistema	El sistema de caracterización compensa los errores de ruido y vibraciones en el sistema	4	
		La estructura está basada del medio ambiente	El sistema de caracterización está basado del medio ambiente y en un medio controlado para la medición	El sistema de caracterización está basado del medio ambiente y en un medio controlado para la medición	5	
		Se logra obtener el MEMS hasta los rangos de trabajo deseados	El sistema de caracterización logra validar la aceleración hasta los rangos de trabajo deseados	El sistema de caracterización logra validar la aceleración hasta los rangos de trabajo deseados	4	
		Los datos del MEMS se pueden conocer algo de la prueba	El sistema de caracterización guarda los datos obtenidos y permite accesarlos en cualquier momento	El sistema de caracterización se puede utilizar independientemente y se puede cambiar los sensores a utilizar	4	
Pruebas	Pruebas	Se puede usar el sistema con varios sensores y en muchas ocasiones	El sistema de caracterización se puede utilizar independientemente y se puede cambiar los sensores a utilizar	El sistema de caracterización se puede utilizar independientemente y se puede cambiar los sensores a utilizar	4	
		Los acelerómetros a utilizar están disponibles en Costa Rica y el tiempo de adquisición es menor o igual a 2 semanas	Los acelerómetros a utilizar están disponibles en Costa Rica y el tiempo de adquisición es menor o igual a 2 semanas	Los acelerómetros a utilizar están disponibles en Costa Rica y el tiempo de adquisición es menor o igual a 2 semanas	4	
		El precio de los acelerómetros no sobrepasa los 15 dólares la unidad	El precio de los acelerómetros a utilizar no sobrepasa los 15 dólares la unidad	El precio de los acelerómetros a utilizar no sobrepasa los 15 dólares la unidad	5	Desearía al final, se me hiciera necesario simular un chip tan caro

A.2 Encuesta al cliente

Encuesta: Desarrollo de un simulador de un acelerómetro de tres (3) ejes microelectromecánicos (MEMS)

Para cada una de las siguientes funciones del sistema, por favor indique en una escala de 1 a 5 qué tan importante es esa función para usted.

Por favor use la siguiente escala:

1. La función es indeseable. No consideraría un sistema con esta función.
2. La función no es importante, pero no me importaría tenerla.
3. Sería bueno tener esa función, pero no es necesaria.
4. La función es altamente deseable, pero consideraría un sistema sin ella.
5. La función es de importancia crítica. No consideraría un producto sin esta función.

También indique, con una marca en la caja de la derecha, si piensa usted que la función es única, interesante y/o inesperado

Con respecto al sistema simulado de un MEMS

Escala	Función	Marque una X si la función en escala de 1 a 5 es única, interesante y/o inesperada.
5	El sistema simula de manera confiable un acelerómetro MEMS	
5	El sistema simulado varía su respuesta según la temperatura ingresada y pueden agregarse nuevas funcionalidades	
5	El sistema simulado trabaja en un rango de temperatura entre -40 y +125 celsius	No necesariamente tiene que ser un rango tan amplio (en caso de que se complicara más)
5	El sistema simulado se configura en al menos 3 rangos de diferentes aceleraciones e indica la sensibilidad en mv/g	
5	El sistema simulado es alimentado con una fuente de entre 3.3V y 5V	
5	El sistema simulado indica si el chip está presente, el modelo del chip y el número de serie del mismo	
5	El sistema simulado se configurar en modo de trabajo o modo de falla a través de la comunicación con el chip	
5	El sistema simulado puede dar respuesta en el modo de prueba o el modo de falla en una, alguna o todas las salidas	
5	El sistema simulado es compatible con tecnología TTL en sus salidas	
5	El sistema simulado tiene salidas analógicas y digitales	

	para cada uno de sus ejes, mediante algún protocolo IC	
5	El sistema simulado muestra al usuario la temperatura actual y el estado del chip	
5	El sistema simulado posee un canal individual para cada salida.	
5	El sistema simula un acelerómetro MEMS en la FPGA del myRio	
5	El sistema simulado utiliza API para comunicarse, medir y probar el chip	
5	El sistema simulado posee una tarjeta electrónica con las conexiones necesarias para su funcionamiento	
5	El sistema simulado está dentro de un encapsulado que lo aisle y resguarde	
5	El sistema simulado posee un manual de operación con los modo de trabajo y las especificaciones del sistema	
5	El sistema simulado valida el chip mediante una una secuencia	

Con respecto al sistema de caracterización de un MEMS

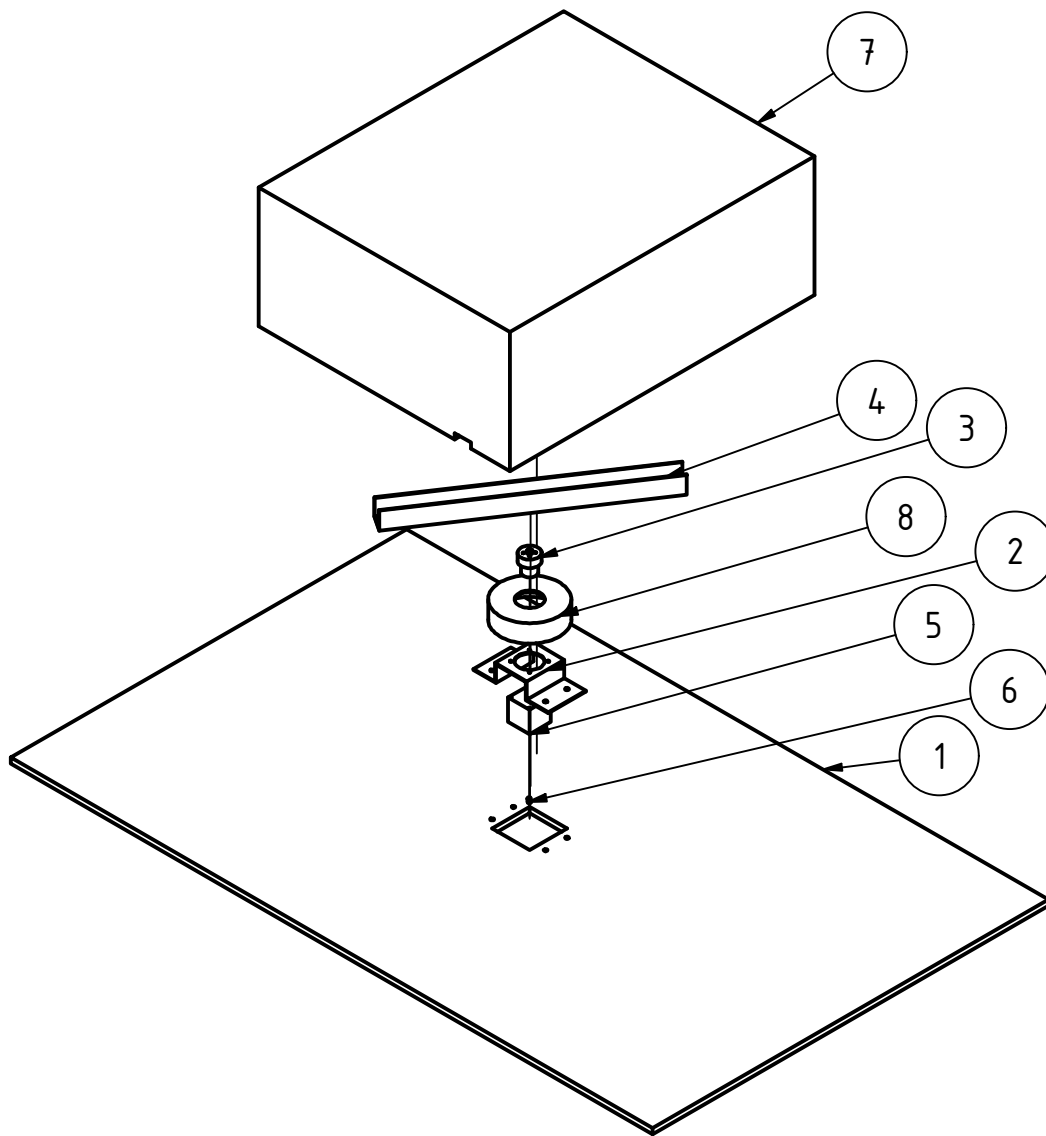
Escala	Función	Marque una X si la función en escala de 1 a 5 es única, interesante y/o inesperada.
3	El sistema de caracterización compensa los errores de ruido y vibraciones en el sistema	Inesperada e interesante
4	El sistema de caracterización está aislado del medio ambiente y en un medio controlado para la medición	
5	El sistema de caracterización varía la aceleración hasta los rangos de trabajo establecidos.	
4	El sistema de caracterización guarda los datos obtenidos y permite accesarlos	
4	El sistema de caracterización se puede utilizar repetidamente y se puede cambiar los sensores a utilizar	
4	Los acelerómetros a utilizar están disponibles en Costa Rica y el tiempo de adquisición es menor o igual a 2 semanas	
5	El precio de los acelerómetros a utilizar no sobrepasa los 15 dólares la unidad.	Deseable: al final, no es tan necesario simular un chip tan caro

Gracias por completar la encuesta

A.3 Lista de métricas

[illegible]

A.4 Planos ensamble sistema de caracterización



PARTS LIST

ITEM	QTY	PART NUMBER	DESCRIPTION
1	1	Base Madera	Lamina de MDF de 9mm
2	1	Base Motor	Lamina de Hierro de 2mm
3	1	Acople Eje	Pieza de aluminio, manufactura Propia
4	1	Viga U	Del kit de piezas Tetrix, 40cm de longitud
5	1	Motor	Stepper Stepper Motor with Cable [ROB-09238] (NEMA 17)
6	1	Prisionero Eje	Prisioneros de 1/4, rosca corriente
7	1	Camara Aislamiento Termico	MDF de 7mm
8	1	Aislamiento del motor	Tapa PVC 3 pulgadas

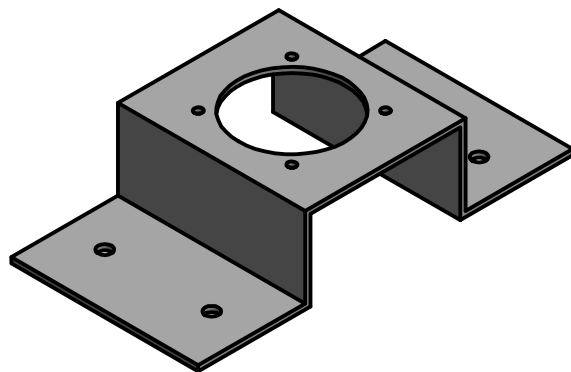
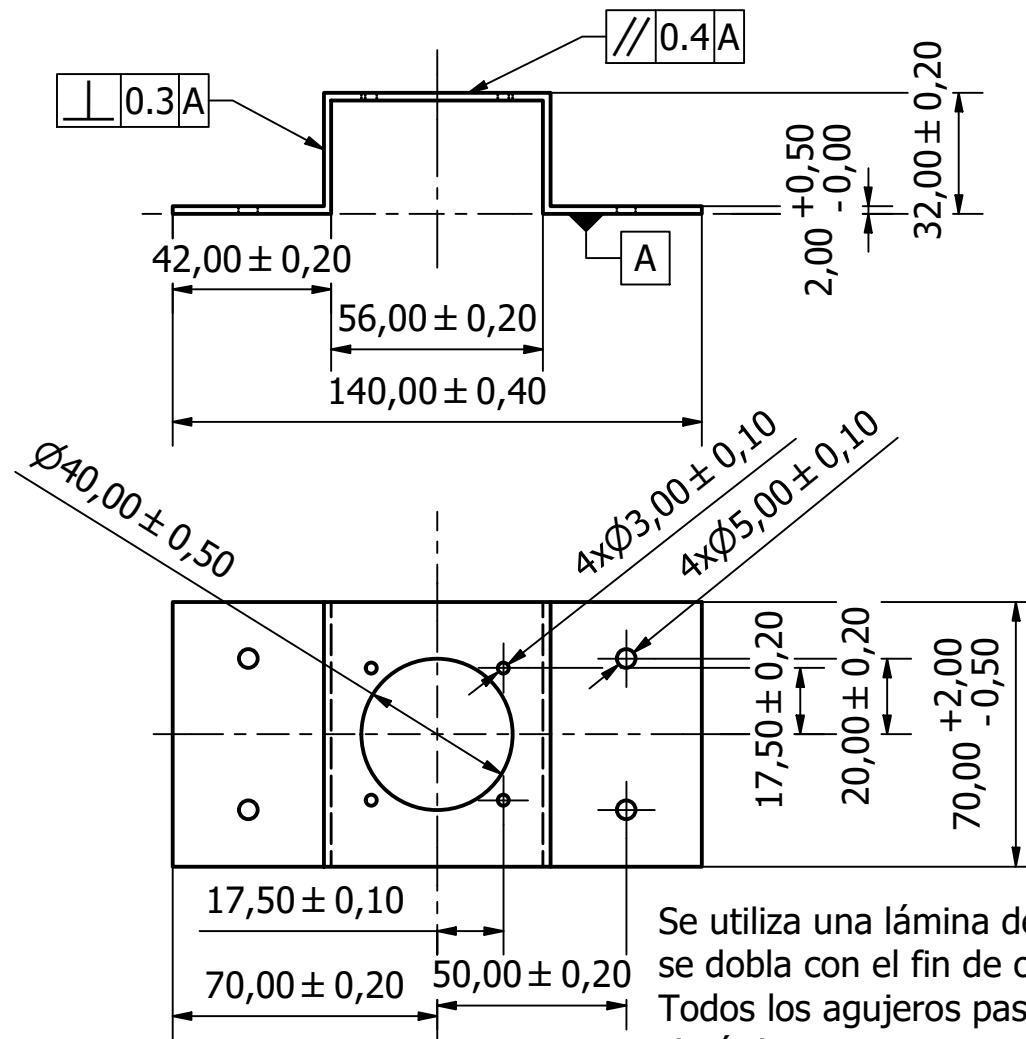
	Date	Name
Drawn	5/28/2022	C.Gómez
Checked		
Standard		

1:10

Ensamble del sistema

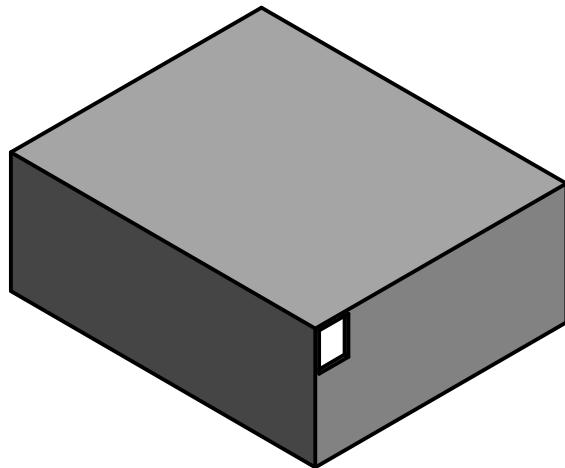
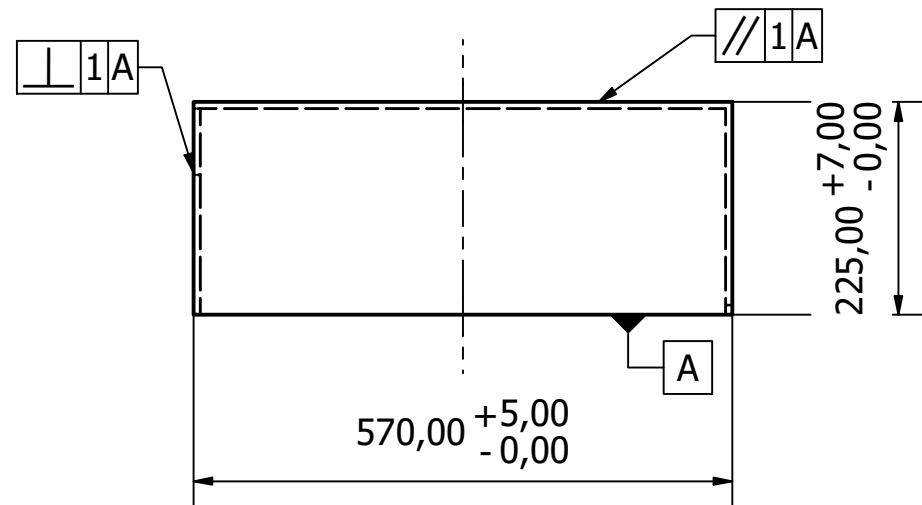
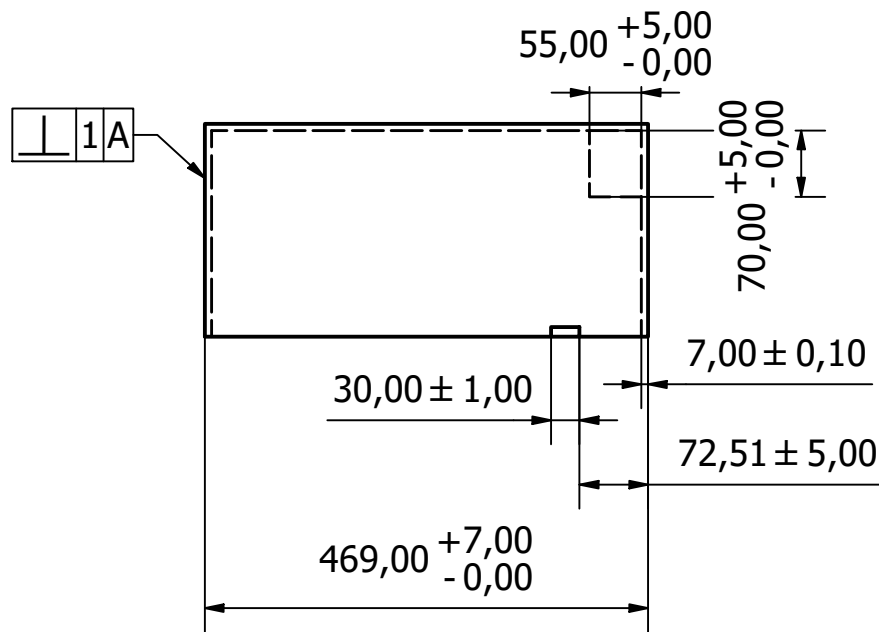
1

A4



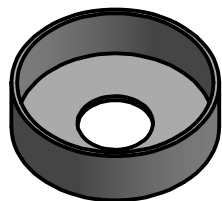
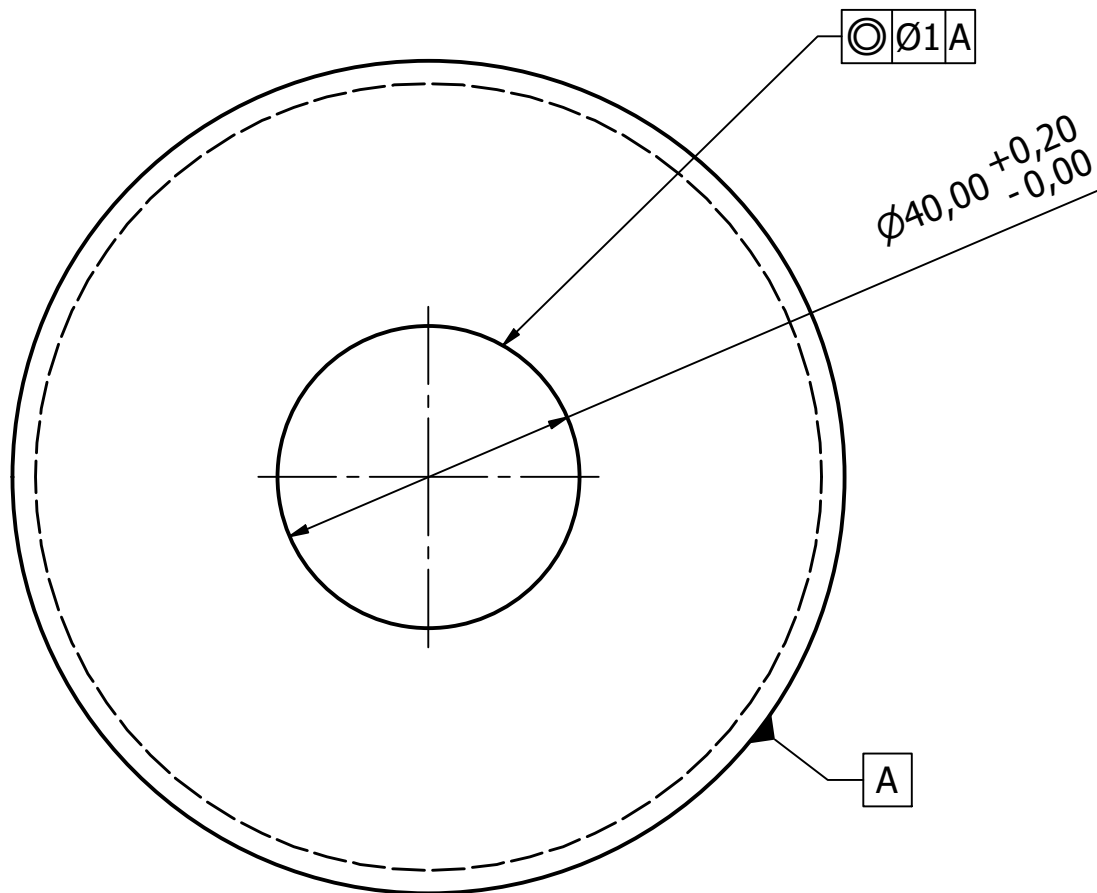
Se utiliza una lámina de hierro de 2mm de espesor, la cual se dobla con el fin de obtener la forma correspondiente
 Todos los agujeros pasan a través de la pieza y son simétricos.
 Todas las unidades en mm.

Designed by C. Gómez	Checked by	Approved by	Date A4	Date 4/2/2022	1:2
Instituto Tecnológico de Costa Rica			Plano Base Motor		
			Base del Motor	Edition	Sheet 1 / 1



Se utilizan laminas de MDF de 7mm de espesor para ensamblar la cámara de aislamiento.
Todas las unidades en mm.

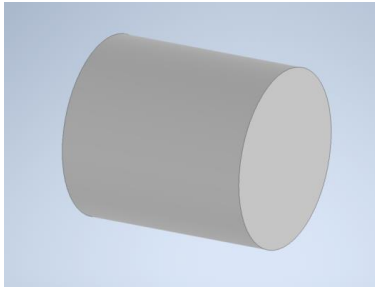
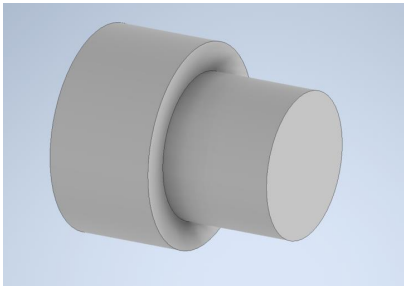
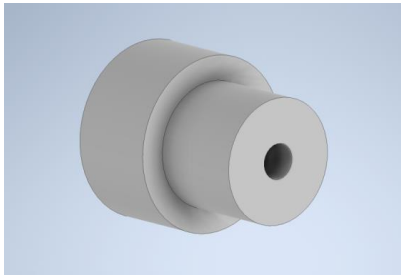
Designed by C. Gómez	Checked by	Approved by	Date A4	Date 4/2/2022	1:8
Instituto Tecnológico de Costa Rica			Plano cámara aislamiento térmico		
			Cámara térmica	Edition	Sheet 1 / 1

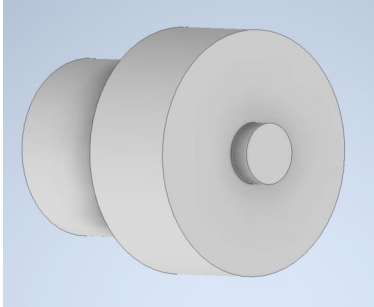
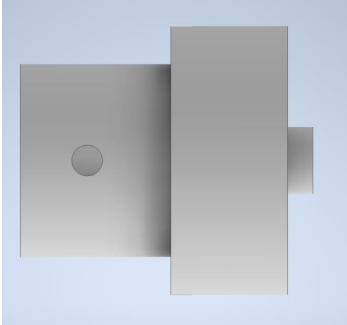
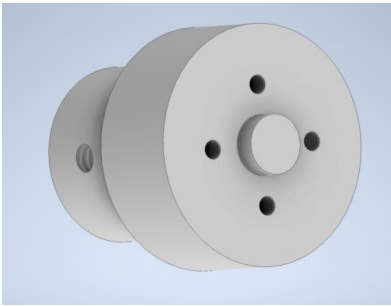


Se utiliza una tapa comercial estandar para tuberias de PVC
con 3 pulgadas de diámetro.
Todas las unidades en mm.

Designed by C. Gómez	Checked by	Approved by	Date A4	Date 4/2/2022	1:1
Instituto Tecnológico de Costa Rica			Plano aislamiento térmico		
			Aislamiento Térmico	Edition	Sheet 1 / 1

A.5 Planos proceso de manufactura para acople

Fases de Maquinado					Cliente	Tecnológico de Costa Rica
Nombre Parte:		Acople Eje			No. Parte	Acople-Eje
No. Op:	Máquina	Tipo soporte	Herramienta	Detalles Operación	Croquis	
1	Sierra de Cinta	N/A	N/A	Se obtiene una barra de 35mm de diámetro por 35mm		
2	Torno	Husillo de 3 muelas autocentrante	Buril de desbaste	Se quita el material extra mediante desbaste y se da el diámetro final del acople. Desbaste y acabado en ítem: 1 y 2		
3	Torno	Husillo de 3 muelas autocentrante	Broca de 6.5mm de diámetro en cabezal móvil	Se realiza un agujero de 18mm de profundidad. Se utiliza primero una broca de centro para hacer camino para la broca. Acabado en ítem: 3 y 4		

4	Torno	Husillo de 3 muelas autocentrante	Buril de desbaste	Se quita el material extra mediante desbaste y se da el diámetro final del acople. Desbaste y acabado en ítem: 5 y 6	
5	Taladro de Banco	Prensa de banco	Broca para metal de 5mm	Se realiza un agujero de 5mm de diámetro centrado en el acople del eje. Se realiza un roscado con la herramienta macho para generar los hilos en el acople roscable. Acabado del ítem: 7 y 8	
6	Taladro de Banco	Prensa de banco	Broca para metal de 5mm	Se realizan 4 agujeros de 3mm de diámetro equidistantes. Se realiza un roscado con la herramienta macho para generar los hilos en el acople roscable. Acabado del ítem: 9 y 10	

Hoja de Procesos				Cliente	Tecnológico de Costa Rica
Nombre Parte:		Acople Eje		No. Parte	Acople-Eje
No. Plano		Código G			
Material	Aluminio AA 6061 T6511 1-1/4				
Detalles:		Fase 1			
Op	Herramienta	No. Herramienta	Parametros		Detalles y Observaciones
11	Sierra de Cinta		n [rpm]	-	Cortar la barra de 35mm de diámetro por 35mm de largo $\pm 1\text{mm}$
			Vc [m/min]	-	
			Avance [mm/min]	-	

Hoja de Procesos				Cliente	Tecnológico de Costa Rica
Nombre Parte:		Acople Eje		No. Parte	Acople- Eje
No. Plano	Plano de Fase 2	Código G			
Material	Aluminio AA 6061 T6511 1-1/4				
Detalles del Montaje:		Husillo de 3 muelas autocentrante , de un extremo			
Op	Herramienta	No. Herramienta	Parametros		Detalles y Observaciones
21	Burin de desbaste		n [rpm]	600	Desbaste
			Vc [m/min]	-	
			Avance [mm/min]	-	

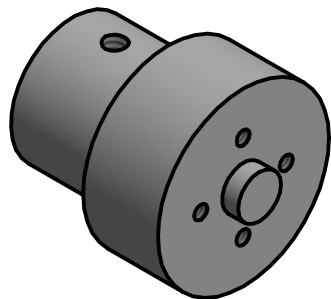
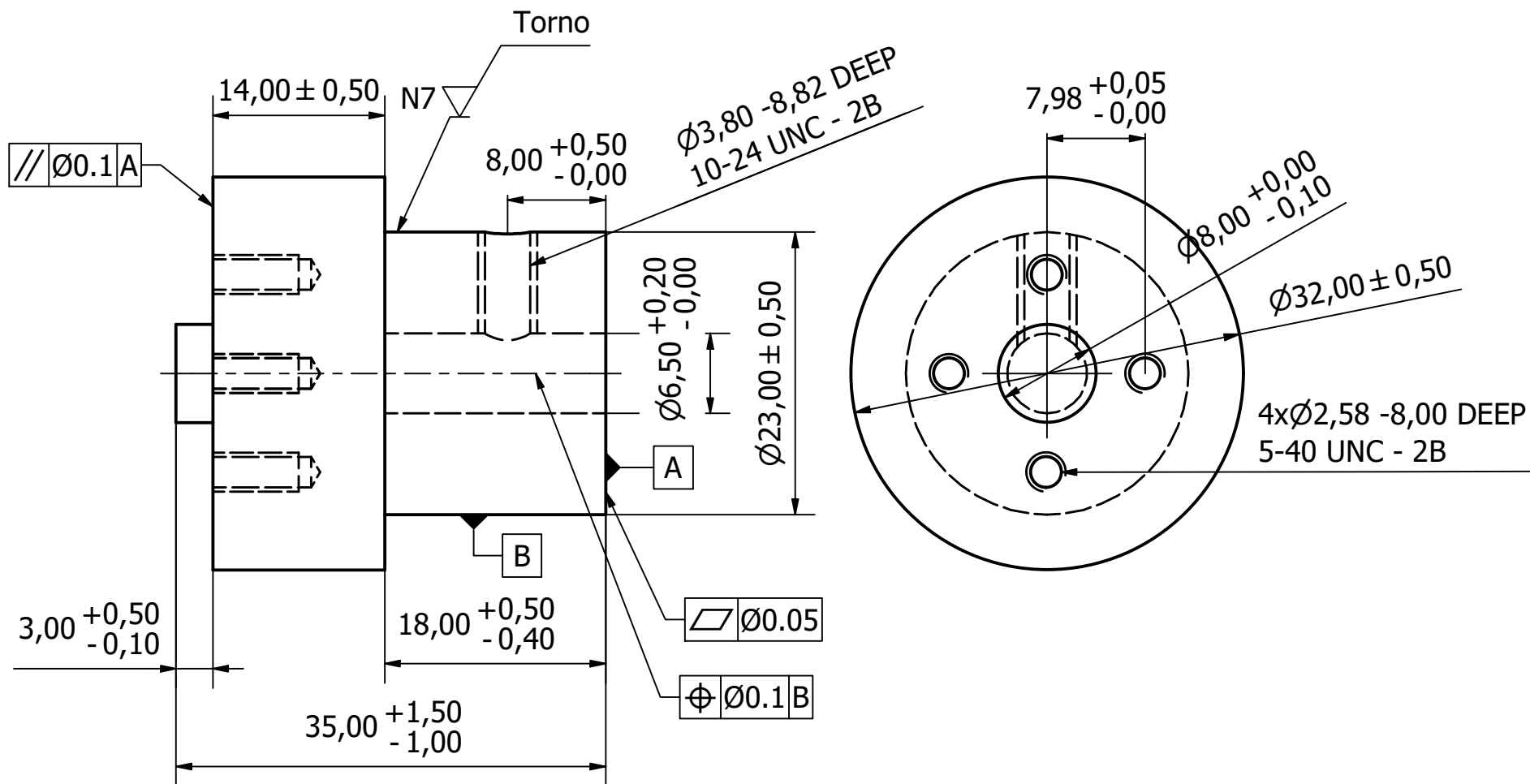
Hoja de Procesos				Cliente	Tecnológico de Costa Rica
Nombre Parte:		Acople Eje		No. Parte	Acople- Eje
No. Plano	Plano de Fase 3	Código G			
Material	Aluminio AA 6061 T6511 1-1/4				
Detalles del Montaje:		Husillo de 3 muelas autocentrante de un extremo y cabezal móvil			
Op	Herramienta	No. Herramienta	Parametros		Detalles y Observaciones
31	Broca de centro		n [rpm]	650	Broca de centro para hacer camino para la broca
			Vc [m/min]		
			Avance [mm/min]	-	
32	Broca de 6.5mm de diámetro		n [rpm]		Se realiza un agujero de 18mm de profundidad concéntrico.
			Vc [m/min]	-	
			Avance [mm/min]		

Hoja de Procesos				Cliente	Tecnológico de Costa Rica
Nombre Parte:		Acople Eje		No. Parte	Acople-Eje
No. Plano	Plano de Fase 4		Código G		
Material	Aluminio AA 6061 T6511 1-1/4				
Detalles del Montaje:		Husillo de 3 muelas autocentrante de un extremo			
Op	Herramienta	No. Herramienta	Parametros		Detalles y Observaciones
41	Burin de desbaste		n [rpm]	600	Torneado del contorno del soporte de 3mm
			Vc [m/min]		
			Avance [mm/min]		

Hoja de Procesos				Cliente	Tecnológico de Costa Rica
Nombre Parte:		Acople Eje		No. Parte	Acople-Eje
No. Plano	Plano de Fase 5		Código G		
Material	Aluminio AA 6061 T6511 1-1/4				
Detalles del Montaje:		Prensa de banco			
Op	Herramienta	No. Herramienta	Parametros		Detalles y Observaciones
51	Broca para metal de 5mm		n [rpm]	650	Se realiza un agujero de 5mm de diámetro centrado en el acople del eje.
			Vc [m/min]		
			Avance [mm/min]		
52	Macho para rosca		n [rpm]	-	Se realiza un roscado con la herramienta macho para generar la rosca interna. Rosca de 2 BSPP, 11 hilos por pulgada en todo el agujero
			Vc [m/min]	-	
			Avance [mm/min]	-	

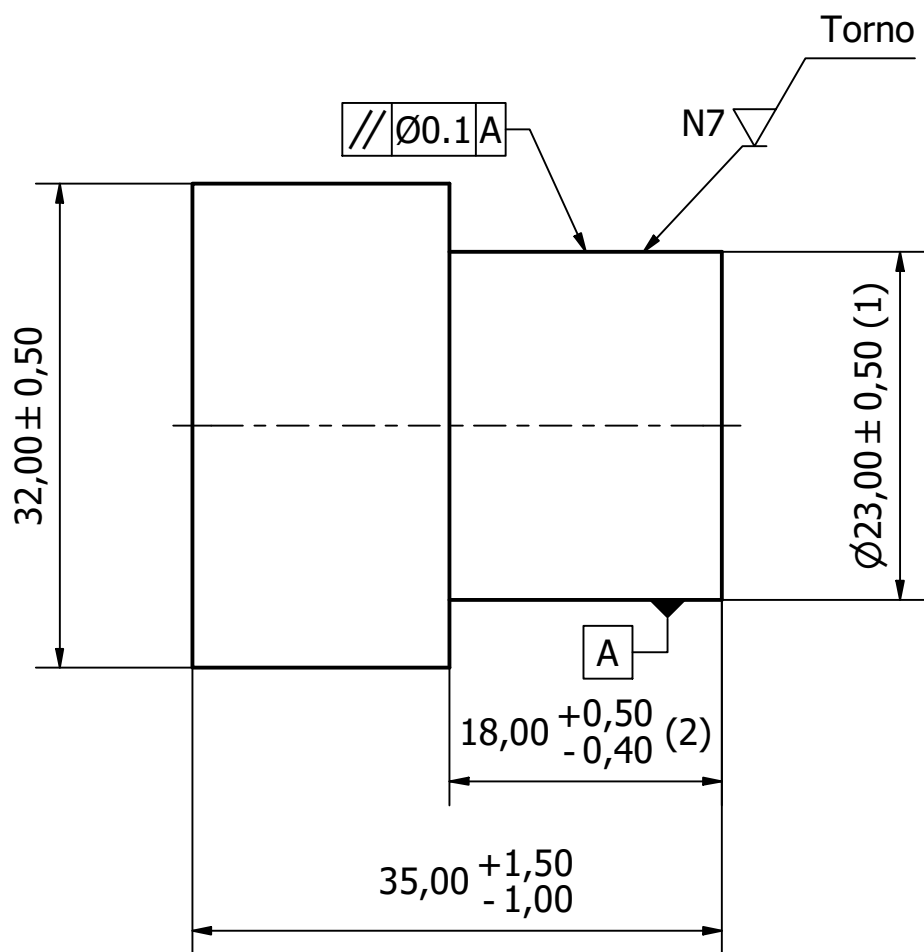
Hoja de Procesos				Cliente	Tecnológico de Costa Rica
Nombre Parte:		Acople Eje		No. Parte	Acople-Eje
No. Plano	Plano de Fase 5		Código G		
Material	Aluminio AA 6061 T6511 1-1/4				
Detalles del Montaje:		Prensa de banco			
Op	Herramienta	No. Herramienta	Parametros		Detalles y Observaciones
61	Broca para metal de 5mm		n [rpm]	650	Se realizan 4 agujeros de 3mm de diámetro equidistantes.
			Vc [m/min]		
			Avance [mm/min]		
62	Macho para rosca		n [rpm]	-	Se realiza un roscado con la herramienta macho para generar la rosca interna. Rosca de 2 BSPP, 11 hilos por pulgada en todo el agujero.
			Vc [m/min]	-	
			Avance [mm/min]	-	

[illegible]

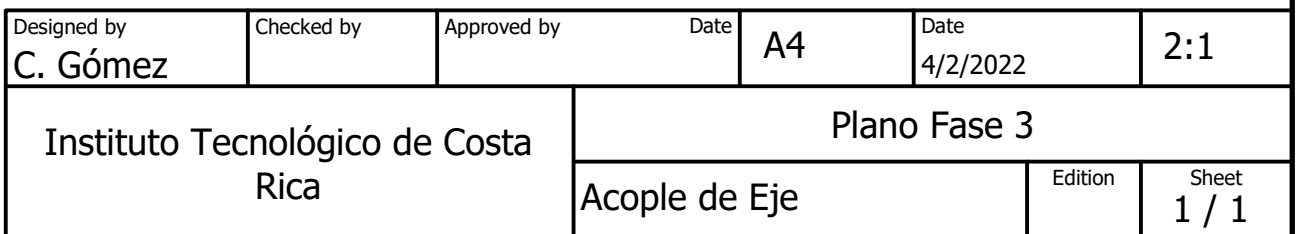


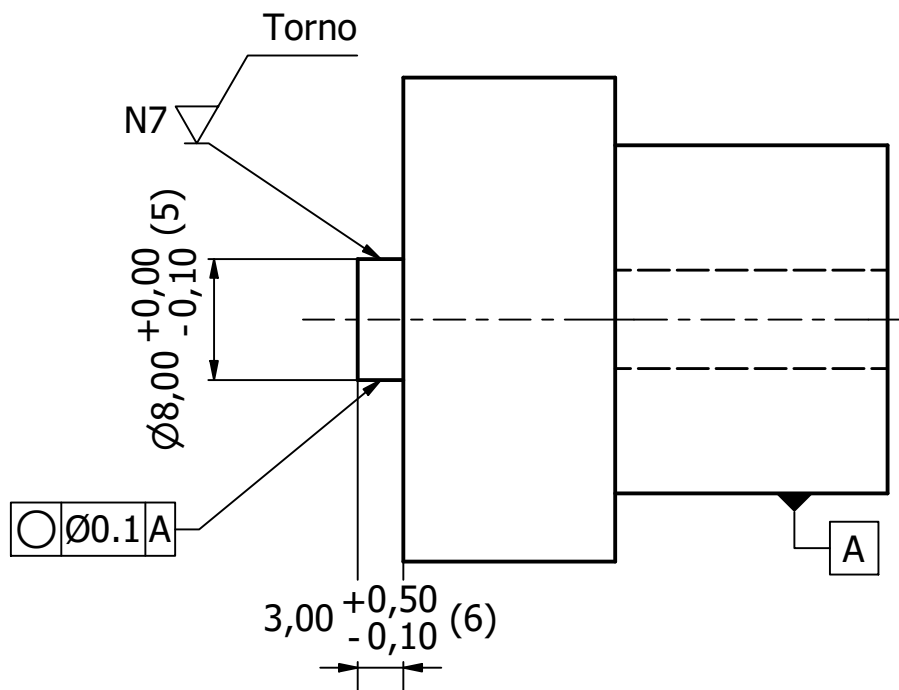
Nota: Todas las medidas en mm
Acabado de superficies no indicadas con rugosidad N7 Torneado

Designed by C. Gómez	Checked by	Approved by	Date A4	Date 4/2/2022	2:1
Instituto Tecnológico de Costa Rica			Plano de Cliente		
			Acople de Eje	Edition	Sheet 1 / 1



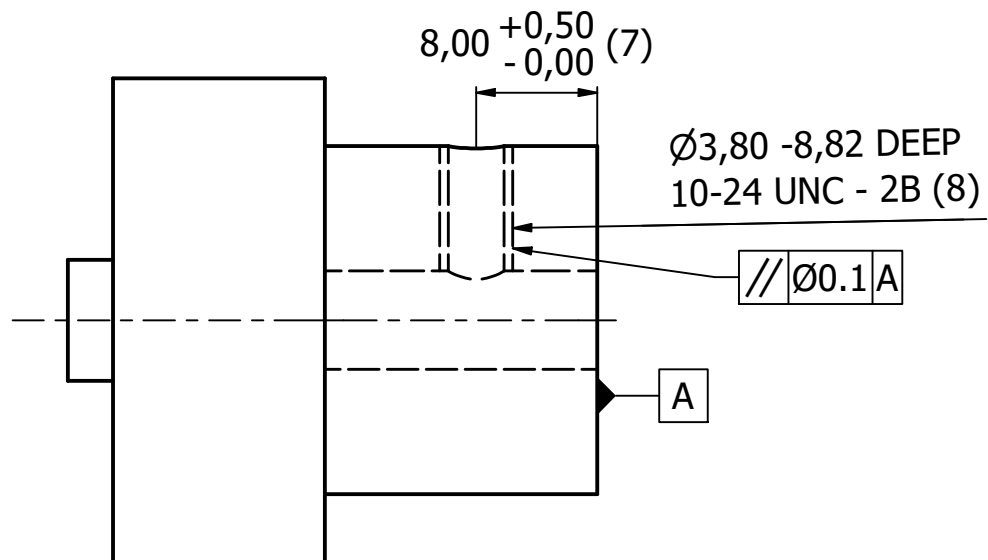
Designed by C. Gómez	Checked by	Approved by	Date A4	Date 4/2/2022	2:1
Instituto Tecnológico de Costa Rica			Plano Fase 2		
			Acople de Eje	Edition	Sheet 1 / 1



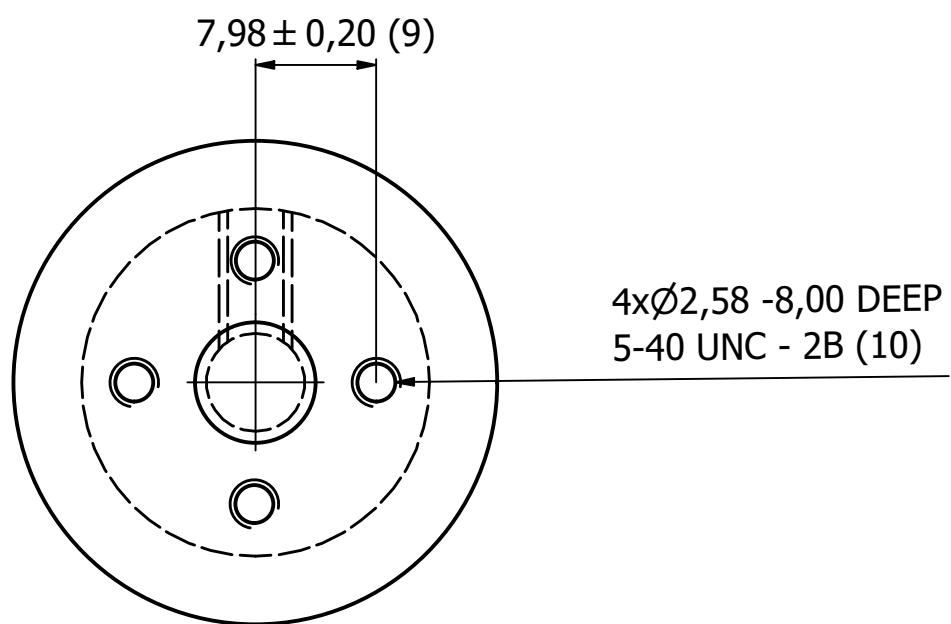


Designed by C. Gómez	Checked by	Approved by	Date A4	Date 4/2/2022	2:1
Instituto Tecnológico de Costa Rica			Plano Fase 4		
			Acople de Eje	Edition	Sheet 1 / 1

Agujero pasa hasta siguiente superficie



Designed by C. Gómez	Checked by	Approved by	Date A4	Date 4/2/2022	2:1
Instituto Tecnológico de Costa Rica			Plano Fase 5		
			Acople de Eje	Edition	Sheet 1 / 1



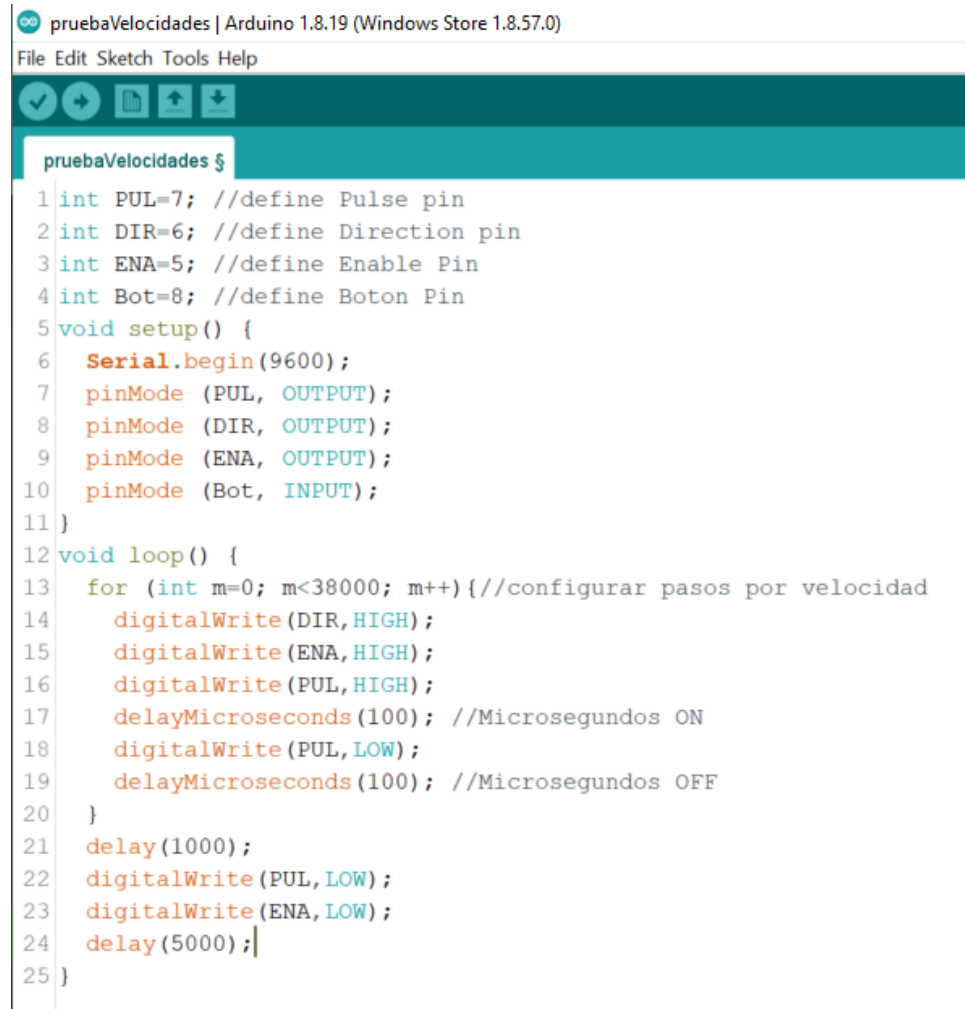
4 agujeros equidistantes a 90°

Agujero no pasa hasta la otra cara, profundidad de 8mm

Designed by C. Gómez	Checked by	Approved by	Date A4	Date 4/2/2022	2:1
Instituto Tecnológico de Costa Rica			Plano Fase 6		
			Acople de Eje	Edition	Sheet 1 / 1

A.6 Código de Arduino

Código en Arduino para validar velocidades del sistema

The image shows a screenshot of the Arduino IDE interface. The title bar at the top reads 'pruebaVelocidades | Arduino 1.8.19 (Windows Store 1.8.57.0)'. Below the title bar is a menu bar with 'File', 'Edit', 'Sketch', 'Tools', and 'Help'. Underneath the menu bar is a toolbar with icons for checking, running, saving, and uploading. The main text area contains the following C++ code:

```
pruebaVelocidades $
1 int PUL=7; //define Pulse pin
2 int DIR=6; //define Direction pin
3 int ENA=5; //define Enable Pin
4 int Bot=8; //define Boton Pin
5 void setup() {
6   Serial.begin(9600);
7   pinMode (PUL, OUTPUT);
8   pinMode (DIR, OUTPUT);
9   pinMode (ENA, OUTPUT);
10  pinMode (Bot, INPUT);
11 }
12 void loop() {
13   for (int m=0; m<38000; m++){//configurar pasos por velocidad
14     digitalWrite(DIR,HIGH);
15     digitalWrite(ENA,HIGH);
16     digitalWrite(PUL,HIGH);
17     delayMicroseconds(100); //Microsegundos ON
18     digitalWrite(PUL,LOW);
19     delayMicroseconds(100); //Microsegundos OFF
20   }
21   delay(1000);
22   digitalWrite(PUL,LOW);
23   digitalWrite(ENA,LOW);
24   delay(5000);
25 }
```

A.7 Código de Python para la Raspberry Pi

```

"""
This program handles the communication over I2C
between a Raspberry Pi and a MPU-6050 Gyroscope / Accelerometer combo.
Released under the MIT License
Copyright (c) 2015, 2016, 2017, 2021 Martijn (martijn@mrtijn.nl) and contributors
https://github.com/m-rtij/mpu6050
"""

import smbus
import RPi.GPIO as GPIO

import time
import datetime, sys
import shutil #Mover y copiar archivos de texto
import os #Accesar el sistema para poder apagar la raspberry

class mpu6050:

    # Global Variables
    GRAVITY_MS2 = 9.80665
    address = None
    bus = None

    # Scale Modifiers
    ACCEL_SCALE_MODIFIER_2G = 16384.0
    ACCEL_SCALE_MODIFIER_4G = 8192.0
    ACCEL_SCALE_MODIFIER_8G = 4096.0
    ACCEL_SCALE_MODIFIER_16G = 2048.0

    GYRO_SCALE_MODIFIER_250DEG = 131.0
    GYRO_SCALE_MODIFIER_500DEG = 65.5
    GYRO_SCALE_MODIFIER_1000DEG = 32.8
    GYRO_SCALE_MODIFIER_2000DEG = 16.4

    # Pre-defined ranges
    ACCEL_RANGE_2G = 0x00
    ACCEL_RANGE_4G = 0x08
    ACCEL_RANGE_8G = 0x10
    ACCEL_RANGE_16G = 0x18

    GYRO_RANGE_250DEG = 0x00
    GYRO_RANGE_500DEG = 0x08
    GYRO_RANGE_1000DEG = 0x10
    GYRO_RANGE_2000DEG = 0x18

    FILTER_BW_256=0x00
    FILTER_BW_188=0x01
    FILTER_BW_98=0x02
    FILTER_BW_42=0x03
    FILTER_BW_20=0x04

```

```
FILTER_BW_10=0x05
FILTER_BW_5=0x06

# MPU-6050 Registers
PWR_MGMT_1 = 0x6B
PWR_MGMT_2 = 0x6C

ACCEL_XOUT0 = 0x3B
ACCEL_YOUT0 = 0x3D
ACCEL_ZOUT0 = 0x3F

TEMP_OUT0 = 0x41

GYRO_XOUT0 = 0x43
GYRO_YOUT0 = 0x45
GYRO_ZOUT0 = 0x47

ACCEL_CONFIG = 0x1C
GYRO_CONFIG = 0x1B
MPU_CONFIG = 0x1A

def __init__(self, address, bus=1):
    self.address = address
    self.bus = smbus.SMBus(bus)
    # Wake up the MPU-6050 since it starts in sleep mode
    self.bus.write_byte_data(self.address, self.PWR_MGMT_1, 0x00)

# I2C communication methods

def read_i2c_word(self, register):
    """Read two i2c registers and combine them.
    register -- the first register to read from.
    Returns the combined read results.
    """
    # Read the data from the registers
    high = self.bus.read_byte_data(self.address, register)
    low = self.bus.read_byte_data(self.address, register + 1)

    value = (high << 8) + low

    if (value >= 0x8000):
        return -((65535 - value) + 1)
    else:
        return value

# MPU-6050 Methods

def get_temp(self):
    """Reads the temperature from the onboard temperature sensor of the
    MPU-6050.
```

```

Returns the temperature in degrees Celcius.
"""
raw_temp = self.read_i2c_word(self.TEMP_OUT0)

# Get the actual temperature using the formule given in the
# MPU-6050 Register Map and Descriptions revision 4.2, page 30
actual_temp = (raw_temp / 340.0) + 36.53

return actual_temp

def set_accel_range(self, accel_range):
    """Sets the range of the accelerometer to range.
    accel_range -- the range to set the accelerometer to. Using a
    pre-defined range is advised.
    """
    # First change it to 0x00 to make sure we write the correct value later
    self.bus.write_byte_data(self.address, self.ACCEL_CONFIG, 0x00)

    # Write the new range to the ACCEL_CONFIG register
    #self.bus.write_byte_data(self.address, self.ACCEL_CONFIG, accel_range)
    self.bus.write_byte_data(self.address, self.ACCEL_CONFIG, accel_range)

def read_accel_range(self, raw = True):#False
    """Reads the range the accelerometer is set to.
    If raw is True, it will return the raw value from the ACCEL_CONFIG
    register
    If raw is False, it will return an integer: -1, 2, 4, 8 or 16. When it
    returns -1 something went wrong.
    """
    raw_data = self.bus.read_byte_data(self.address, self.ACCEL_CONFIG)

    if raw is True:
        return raw_data
    elif raw is False:
        if raw_data == self.ACCEL_RANGE_2G:
            return 2
        elif raw_data == self.ACCEL_RANGE_4G:
            return 4
        elif raw_data == self.ACCEL_RANGE_8G:
            return 8
        elif raw_data == self.ACCEL_RANGE_16G:
            return 16
        else:
            return -1

def get_accel_data(self, g = True):#False
    """Gets and returns the X, Y and Z values from the accelerometer.
    If g is True, it will return the data in g
    If g is False, it will return the data in m/s^2
    Returns a dictionary with the measurement results.

```

```

"""
x = self.read_i2c_word(self.ACCEL_XOUT0)
y = self.read_i2c_word(self.ACCEL_YOUT0)
z = self.read_i2c_word(self.ACCEL_ZOUT0)

accel_scale_modifier = None
accel_range = self.read_accel_range(True)

if accel_range == self.ACCEL_RANGE_2G:
    accel_scale_modifier = self.ACCEL_SCALE_MODIFIER_2G
elif accel_range == self.ACCEL_RANGE_4G:
    accel_scale_modifier = self.ACCEL_SCALE_MODIFIER_4G
elif accel_range == self.ACCEL_RANGE_8G:
    accel_scale_modifier = self.ACCEL_SCALE_MODIFIER_8G
elif accel_range == self.ACCEL_RANGE_16G:
    accel_scale_modifier = self.ACCEL_SCALE_MODIFIER_16G
else:
    print("Unkown range - accel_scale_modifier set to
self.ACCEL_SCALE_MODIFIER_2G")
    accel_scale_modifier = self.ACCEL_SCALE_MODIFIER_2G

x = x / accel_scale_modifier
y = y / accel_scale_modifier
z = z / accel_scale_modifier

if g is True:
    return {'x': x, 'y': y, 'z': z}
elif g is False:
    x = x * self.GRAVITIY_MS2
    y = y * self.GRAVITIY_MS2
    z = z * self.GRAVITIY_MS2
    return {'x': x, 'y': y, 'z': z}

contaA = 0
boton1=18
ledAzul=17
RPM=0
last_time=time.time()
this_time=time.time()

#Setup
GPIO.setmode(GPIO.BCM)
GPIO.setup(04,GPIO.IN)
GPIO.setup(boton1,GPIO.IN)
BotonFin= GPIO.input(boton1)
GPIO.setup(ledAzul,GPIO.OUT)
GPIO.output(ledAzul,GPIO.HIGH)

GPIO.setwarnings(False)

```

```

#sensor address
mpu1 = mpu6050(0x68) #ADO is LOW
# Write the new range to the ACCEL_CONFIG register
#self.bus.write_byte_data(self.address, self.ACCEL_CONFIG, accel_range)
#self.bus.write_byte_data(self.address, self.ACCEL_CONFIG, 0x18)
mpu1.set_accel_range(0x18)
time.sleep(0.4)

while True:
    try:
        # Si la llave maya esta conectada abre el archivo Notocar.txt
        f = open("/media/pi/PROYECTOLOT/Notocar.txt")
        print("SI hay llave")
        time.sleep(1)
        break
    except:
        # Si no hay llav maya hace parpadear el led
        print("NO hay llave")
        GPIO.output(ledAzul,GPIO.HIGH)
        time.sleep(0.5)
        GPIO.output(ledAzul,GPIO.LOW)
        time.sleep(0.5)

time.sleep(0.4)

#Callbacks
def CuentaA(channel):
    global RPM,this_time,last_time
    this_time=time.time()
    RPM=(1/(this_time-last_time))*60 #RPM instantaneas
    last_time=this_time
    print RPM

#Interrupciones
GPIO.add_event_detect(04, GPIO.RISING, callback = CuentaA)

#Abre el documento para guardar los datos
archivoactual=time.strftime("%Y-%m-%d-%H-%M")
print(archivoactual)
archivo=open(archivoactual,'w')
archivo.close()
archivo=open(archivoactual,'a') #'a' is used to append data
archivo.write('rpm'+',')
archivo.write('temp' +',')
#round(mpu1.get_temp(),2)to round to 2 decimal the data en the txt file
archivo.write('X' +',')
archivo.write('Y' +',')
archivo.write('Z' +'\n')

```

```
archivo.close()

#Bucle principal
while BotonFin == False:
#for x in range(380):
#    print("Temp 1: "+str(mpu1.get_temp())) #getting temperature data from sensor 1
#    accel_data1 = mpu1.get_accel_data() #getting accel data from sensor 1
#    print("Acc X : "+str(accel_data1['x']))
#    print("Acc Y : "+str(accel_data1['y']))
#    print("Acc Z : "+str(accel_data1['z']))
#    print("-----")
#    BotonFin = GPIO.input(boton1)
#    archivo=open(archivoactual,'a') #'a' is used to append data
#    archivo.write('% s'%round(RPM,2) +',')
#    archivo.write('% s'%round(mpu1.get_temp(),3) +',')
#    #round(mpu1.get_temp(),2)to round to 2 decimal the data en the txt file
#    archivo.write('% s'%round(accel_data1['x'],3) +',')
#    archivo.write('% s'%round(accel_data1['y'],3) +',')
#    archivo.write('% s'%round(accel_data1['z'],3) +'\n')
#    archivo.close()
#    time.sleep(0.2)
#    GPIO.output(ledAzul,GPIO.LOW)
#    time.sleep(0.2)
#    GPIO.output(ledAzul,GPIO.HIGH)

print("fin")
shutil.copy(archivoactual,"copia.txt")
shutil.move(archivoactual,"/media/pi/PROYECTOLOT")
GPIO.output(ledAzul,GPIO.LOW)
GPIO.cleanup()
os.system('sudo shutdown now') #apaga la raspbe
```

A.8 Gráficas de las otras mediciones

		Medición	Temperatura (+/- 0.1)	RPM 0 (+/- 0.01)	Aceleración 0 (+/- 0.001)	RPM 1 (+/-0.0 1)	Aceleración 1 (+/- 0.001)
Eje Y	Temperatura ambiente	1	23.9	0.00	0.091	265.85	15.130
		2	23.5	0.00	0.540	265.69	15.051
		3	23.3	0.00	0.018	262.89	15.240
		4	23.2	0.00	0.025	263.86	15.597
		5	23.1	0.00	0.034	265.73	15.570
		6	22.9	0.00	0.037	265.63	15.118
		7	22.9	0.00	0.021	265.64	15.428
		8	22.9	0.00	0.021	265.75	15.174
		9	22.8	0.00	0.022	265.62	15.044
		10	22.7	0.00	0.021	265.57	15.972
		11	22.6	0.00	0.029	265.98	15.087
		12	22.6	0.00	0.022	265.72	15.049
		13	27.6	0.00	0.014	265.74	15.538
		14	26.1	0.00	0.040	265.25	15.274
		15	26.1	0.00	0.030	266.39	15.042
		16	25.9	0.00	0.027	265.75	15.123
		Promedio	23.9	0.00	0.062	265.44	15.277
		Desviación estandar	1.6	0	0.129	0.86	0.270
	Temperatura Alta	1	45.9	0.00	0.058	265.72	15.089
		2	46.3	0.00	0.042	265.63	15.042
		3	46.6	0.00	0.051	265.7	15.055
		4	46.7	0.00	0.027	265.76	15.079
		5	46.4	0.00	0.015	265.65	15.129
		6	46.1	0.00	0.028	265.71	15.090
		7	45.6	0.00	0.042	265.65	15.187
		8	44.6	0.00	0.066	266.13	15.176
		9	45.8	0.00	0.079	265.63	15.105
		10	46.9	0.00	0.035	265.71	15.370
		11	46.1	0.00	0.066	265.69	15.001
		12	46.1	0.00	0.079	265.72	15.611
		13	46.5	0.00	0.019	265.51	15.235
		14	46.5	0.00	0.023	265.68	15.377
		15	46.9	0.00	0.057	265.66	15.329
		16	46.8	0.00	0.035	271.49	15.279
		Promedio	46.2	0.00	0.045	266.07	15.197
		Desviación estandar	0.6	0.0	0.0	1.45	0.162

Figura A.1: Resultados obtenidos de las mediciones realizadas para el eje Y.

		Medición	Temperatura (+/- 0.1)	RPM 0 (+/- 0.01)	Aceleración 0 (+/- 0.001)	RPM 1 (+/-0.0 1)	Aceleración 1 (+/- 0.001)
Eje Z	Temperatura ambiente	1	19.9	0.00	0.045	284.97	15.586
		2	20.0	0.00	0.042	271.75	14.85
		3	20.5	0.00	0.069	272.54	15.864
		4	20.6	0.00	0.043	274.68	15.811
		5	20.9	0.00	0.048	272.21	14.868
		6	20.9	0.00	0.033	270.60	15.195
		7	21.3	0.00	0.04	270.54	15.854
		8	21.5	0.00	0.047	270.70	15.321
		9	21.8	0.00	0.036	282.35	15.401
		10	22.0	0.00	0.045	280.48	15.943
		11	22.8	0.00	0.038	280.21	15.822
		12	22.8	0.00	0.028	271.29	14.965
		13	22.8	0.00	0.105	265.57	14.549
		14	22.3	0.00	0.102	267.39	14.712
		15	22.3	0.00	0.109	265.63	14.576
		16	22.3	0.00	0.108	264.82	14.538
		Promedio	21.5	0.0	0.1	272.86	15.241
		Desviación estandar	1.0	0	0.030	6.17	0.525
	Temperatura Alta	1	46.5	0.00	0.044	265.72	14.705
		2	46.6	0.00	0.024	265.71	14.358
		3	45.4	0.00	0.039	265.69	15.395
		4	45.6	0.00	0.072	265.65	14.624
		5	45.3	0.00	0.072	265.71	15.853
		6	45.7	0.00	0.049	265.67	15.750
		7	46.1	0.00	0.066	265.77	14.617
		8	45.8	0.00	0.025	265.76	14.258
		9	45.8	0.00	0.039	265.74	15.729
		10	45.9	0.00	0.086	265.65	15.970
		11	46.1	0.00	0.078	265.64	14.567
		12	46.1	0.00	0.024	265.68	14.396
		13	47.5	0.00	0.025	265.69	15.045
		14	46.7	0.00	0.032	265.71	14.573
		15	46.5	0.00	0.035	263.91	14.752
		16	46.3	0.00	0.078	265.69	14.615
		Promedio	46.1	0.00	0.049	265.59	14.950
		Desviación estandar	0.6	0.0	0.022	0.45	0.587

Figura A.2: Resultados obtenidos de las mediciones realizadas para el eje Z.

A.9 Gráfica de todas las mediciones



Figura A.3: Gráfica con la respuesta de las 16 mediciones para el eje Y.

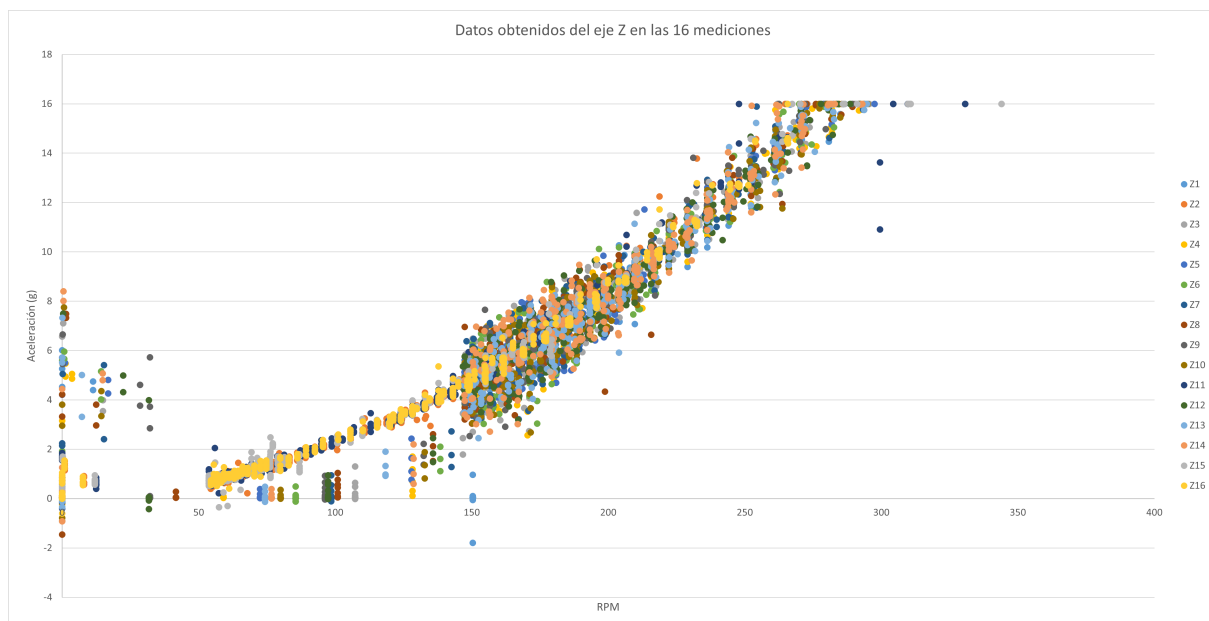


Figura A.4: Gráfica con la respuesta de las 16 mediciones para el eje Z.

A.10 Gráfica de todas las mediciones

Condición		Datos simulados		
Temperatura (C)	Aceleración (g)	X ($\pm 0.001V$)	Y ($\pm 0.001V$)	Z ($\pm 0.001V$)
-10	10	3.061	2.927	2.939
	20	3.192	2.927	2.939
	30	3.325	2.927	2.939
	40	3.457	2.927	2.939
	50	3.589	2.927	2.939
	60	3.721	2.927	2.939
	70	3.853	2.927	2.939
	80	3.984	2.927	2.939
	90	4.117	2.927	2.939
	100	4.248	2.927	2.939
	110	4.380	2.927	2.939
	120	4.513	2.927	2.939
	130	4.644	2.927	2.939
	140	4.774	2.927	2.939
	150	4.910	2.927	2.939
	160	4.996	2.927	2.939
	170	4.996	2.927	2.939
	180	4.996	2.927	2.939
	190	4.996	2.927	2.939
	200	4.996	2.927	2.939
10	10	3.060	2.930	2.947
	20	3.193	2.930	2.947
	30	3.325	2.930	2.947
	40	3.457	2.930	2.947
	50	3.590	2.930	2.947
	60	3.721	2.930	2.947
	70	3.853	2.930	2.947
	80	3.984	2.930	2.947
	90	4.115	2.930	2.947
	100	4.248	2.930	2.947
	110	4.379	2.930	2.947
	120	4.512	2.930	2.947
	130	4.642	2.930	2.947
	140	4.776	2.930	2.947
	150	4.907	2.930	2.947
	160	4.996	2.930	2.947
	170	4.996	2.930	2.947
	180	4.996	2.930	2.947
	190	4.996	2.930	2.947
	200	4.996	2.930	2.947
20	10	3.061	2.932	2.950
	20	3.193	2.932	2.950
	30	3.324	2.932	2.950
	40	3.457	2.932	2.950
	50	3.589	2.932	2.950
	60	3.720	2.932	2.950
	70	3.851	2.932	2.950
	80	3.985	2.932	2.950
	90	4.118	2.932	2.950
	100	4.250	2.932	2.950
	110	4.380	2.932	2.950
	120	4.513	2.932	2.950
	130	4.644	2.932	2.950
	140	4.774	2.932	2.950
	150	4.900	2.932	2.950
	160	4.996	2.932	2.950
	170	4.996	2.932	2.950
	180	4.996	2.932	2.950
	190	4.996	2.932	2.950
	200	4.996	2.932	2.950

Figura A.5: Mediciones del simulador.

Condición		Datos simulados		
Temperatura (C)	Aceleración (g)	X (+0.001V)	Y (+0.001V)	Z (+0.001V)
46	10	3.060	2.936	2.961
	20	3.192	2.936	2.961
	30	3.324	2.936	2.961
	40	3.458	2.936	2.961
	50	3.590	2.936	2.961
	60	3.721	2.936	2.961
	70	3.852	2.936	2.961
	80	3.986	2.936	2.961
	90	4.118	2.936	2.961
	100	4.250	2.936	2.961
	110	4.380	2.936	2.961
	120	4.513	2.936	2.961
	130	4.644	2.936	2.961
	140	4.773	2.936	2.961
	150	4.908	2.936	2.961
	160	4.996	2.936	2.961
	170	4.996	2.936	2.961
	180	4.996	2.936	2.961
	190	4.996	2.936	2.961
	200	4.996	2.936	2.961
80	10	3.061	2.940	2.975
	20	3.192	2.940	2.975
	30	3.325	2.940	2.975
	40	3.456	2.940	2.975
	50	3.590	2.940	2.975
	60	3.720	2.940	2.975
	70	3.851	2.940	2.975
	80	3.984	2.940	2.975
	90	4.115	2.940	2.975
	100	4.247	2.940	2.975
	110	4.378	2.940	2.975
	120	4.511	2.940	2.975
	130	4.644	2.940	2.975
	140	4.774	2.940	2.975
	150	4.910	2.940	2.975
	160	4.996	2.940	2.975
	170	4.996	2.940	2.975
	180	4.996	2.940	2.975
	190	4.996	2.940	2.975
	200	4.996	2.940	2.975
-40	10	3.061	2.922	2.927
	20	3.192	2.922	2.927
	30	3.325	2.922	2.927
	40	3.457	2.922	2.927
	50	3.590	2.922	2.927
	60	3.720	2.922	2.927
	70	3.852	2.922	2.927
	80	3.982	2.922	2.927
	90	4.115	2.922	2.927
	100	4.247	2.922	2.927
	110	4.379	2.922	2.927
	120	4.512	2.922	2.927
	130	4.642	2.922	2.927
	140	4.774	2.922	2.927
	150	4.910	2.922	2.927
	160	4.996	2.922	2.927
	170	4.996	2.922	2.927
	180	4.996	2.922	2.927
	190	4.996	2.922	2.927
	200	4.996	2.922	2.927

Figura A.6: Mediciones del simulador (Continuación).

A.11 Manual de usuario para el simulador

Seguidamente se adjunta el manual de usuario para el simulador, este se redacta en inglés debido a los requerimientos de la empresa.

Manual of Operation

MEMS accelerometer Simulator

-Carla Gómez Carrasquilla

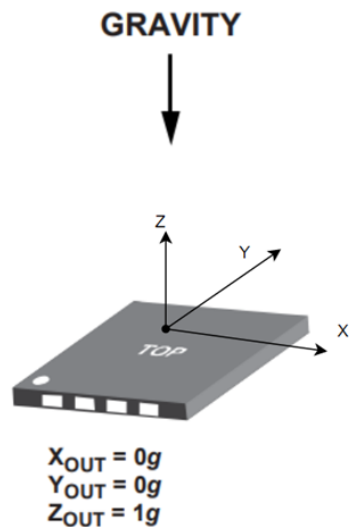
How the system works	2
Code Setup	3
For the PXIE configuration	3
myRio and FPGA configuration	4
For the user interface in PXIe	6
For running the Application	7
Running in Self-Test Mode	7
Running in Working Mode	8
Running in Failing Mode	9
TestStand DUT Sequence	11

How the system works

An accelerometer is a sensor that represents an acceleration through an output voltage; the output for the simulation should be a voltage too.

In this application, the sensor is settled in a centrifuge, a machine that uses rotational force to move its contents away from its center. In this case the machine will be changing its RPM (rotations per minute) in order to generate an acceleration that will be detected through the sensor.

The centrifuge has a 0,2m radius and can rotate clockwise or counterclockwise. The sensor is supposed to be set 0,2m away from the center of the centrifuge and its axis are pointing as the figure below:



Where X axis is pointing outside the center of the centrifuge, and Y axis is pointing counterclockwise. In a static state, 0 RPM, the output values in g for the sensor are 0g, 0g and 1g, as shown in the image.

The acceleration on the x axis will always be positive given that changing the rotation direction will not change the centrifuge force acting on the X axis. The acceleration on the Y axis can be positive or negative depending on the rotation direction, but the voltage generated will always be positive. On the Z axis, a 1g force will be always shown, given that gravity is working over this axis and the centrifuge force has no effect over this axis.

Code Setup

From the repository download and unzip the next files:

 MEMS APP	Just now	15eac53b Demo2 Carl...
 MEMS FPGA	Just now	15eac53b Demo2 Carl...

MEMS APP should be run in the PXIE and MEMS FPGA should be run in the myRio

For the FPGA configuration

1. In labView 2019, open the project “MEMSFPGA”, make sure the code “FPGANoArrayNoStreamPWMDos.vi” has already been compiled and is ready to deploy. Also the code “HostSimulationOutput5” should be ready to run.
2. Set both codes to autoRun, in the FPGA and in the realTime resources

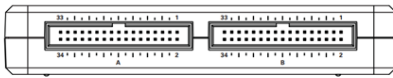
For the PXIE configuration

1. In the PXIE, open a project in labView2020 and open MEMS APP>MEMS API Proj>MEMS API PROJ.
2. Open NI TestStand 2021 and open MEMS APP>TestStand> “MEMS-TestSequence” sequence.

myRio and FPGA configuration

1. Make sure the connections between the FPGA and the CB-2162 are as follows. Use port A of the myRio.

DIO5/SPI.CLK should be connected to pin 27. Pin DIO6/SPI.MISO should be connected to pin 25. Pin DIO7/SPI.MOSI should be connected to pin 23. Pin DIO8/PWM should be connected to pin 21.

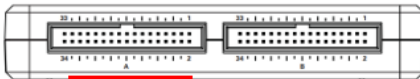


+3.3 V	DIO10 / PWM2	DIO9 / PWM1	DIO8 / PWM0	DIO7 / SPI.MOSI	DIO6 / SPI.MISO	DIO5 / SPI.CLK	DIO4	DIO3	DIO2	DIO1	DIO0	AI3	AI2	AI1	AI0	+5V
33	31	29	27	25	23	21	19	17	15	13	11	9	7	5	3	1
34	32	30	28	26	24	22	20	18	16	14	12	10	8	6	4	2
DIO15 / I2C.SDA	DIO14 / I2C.SCL	DGND	DGND	DIO13	DGND	DIO12 / ENC.B	DGND	DIO11 / ENC.A	DGND	UART.TX	DGND	UART.RX	DGND	AO1	AO0	

SPI Pin Map pinmap

View Connections for: All Pins and Relays

Pin	Site	Instrument	Channel	Multiplexer	Route
CS	0	PXIE-6570_D...	21		
MISO	0	PXIE-6570_D...	25		
MOSI	0	PXIE-6570_D...	23		
SCL	0	PXIE-6570_D...	27		
VDD	<System>	PXIE-6570_D...	4		
VDDIO	<System>	PXIE-6570_D...	5		

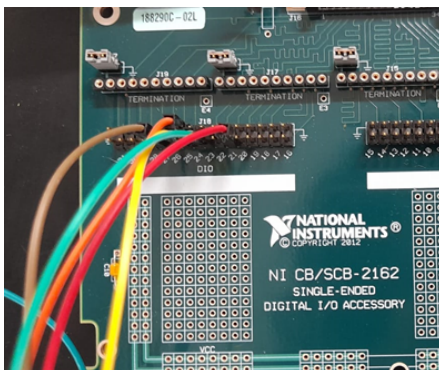


+3.3 V	DIO10 / PWM2	DIO9 / PWM1	DIO8 / PWM0	DIO7 / SPI.MOSI	DIO6 / SPI.MISO	DIO5 / SPI.CLK	DIO4	DIO3	DIO2	DIO1	DIO0	AI3	AI2	AI1	AI0	+5V
33	31	29	27	25	23	21	19	17	15	13	11	9	7	5	3	1
34	32	30	28	26	24	22	20	18	16	14	12	10	8	6	4	2
DIO15 / I2C.SDA	DIO14 / I2C.SCL	DGND	DGND	DIO13	DGND	DIO12 / ENC.B	DGND	DIO11 / ENC.A	DGND	UART.TX	DGND	UART.RX	DGND	AO1	AO0	

SPI Pin Map pinmap

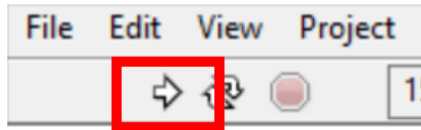
View Connections for: All Pins and Relays

Pin	Site	Instrument	Channel	Multiplexer	Route
CS	0	PXIE-6570_D...	21		
MISO	0	PXIE-6570_D...	25		
MOSI	0	PXIE-6570_D...	23		
SCL	0	PXIE-6570_D...	27		
VDD	<System>	PXIE-6570_D...	4		
VDDIO	<System>	PXIE-6570_D...	5		



2. In LabView 2019, "FPGANoArrayNoStreamPWMdos.vi" has already been compiled and is ready to deploy..

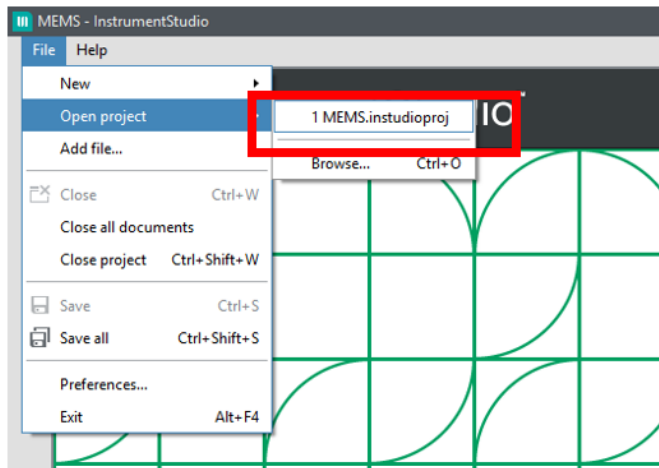
3. Deploy it in the FPGA by clicking the arrow in the top left corner.



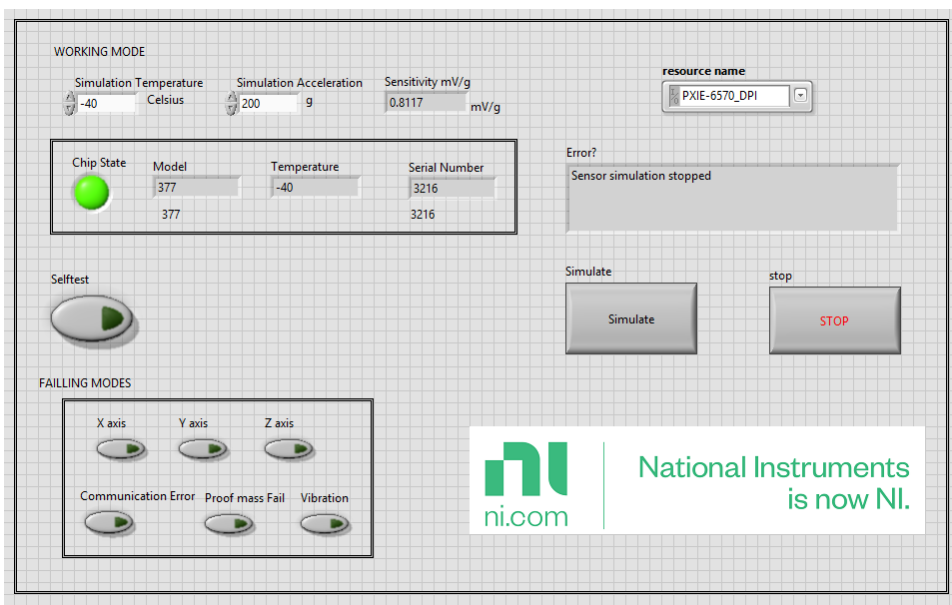
4. Also deploy the code "HostSimulationOutput5" to be run in the real time module of the myRio

For the user interface in PXIe

In the PXIE, open the InstrumentStudio Application and open the file named “MEMS.instudioproj”



1. Activate the “SCOPE.sfp” to view the data
2. Make sure that the probes in channel CH0, CH1 and CH3 are connected to the myRio .
3. In the PXIE, open “MEMSuserInterface2.vi” from MEMS API PROJ. Make sure that the interface is like this one:

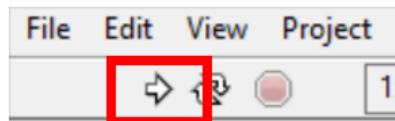


For running the Application

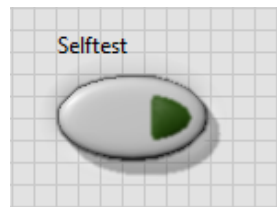
Running in Self-Test Mode

For the self-test option, this condition is done under a 10g acceleration at 25 degrees Celsius.

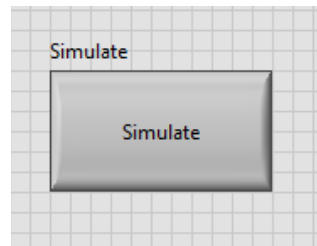
1. Click the run button



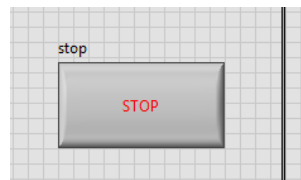
2. Activate the Self-test button, it is not necessary to specify any temperature or acceleration



3. Click the simulate button to burst the parameters specified.



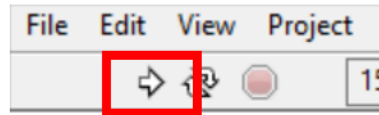
4. In the InstrumentStudio Application there should be a response in the scope according with the self-Test mode
5. Click the stop button and unselect all the previous parameters



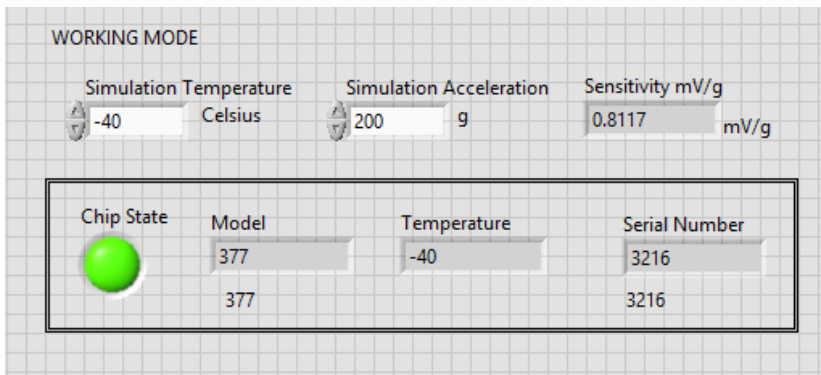
Running in Working Mode

This mode simulates the accelerometer without any error in its outputs.

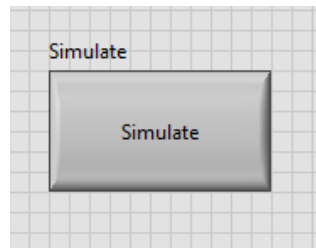
1. Run the code by clicking the arrow in the top left corner.



2. Specify the temperature and acceleration parameters you would like to simulate, use integer values only. The acceleration range is ± 200 , the valid temperature range is -40 to $+85$.

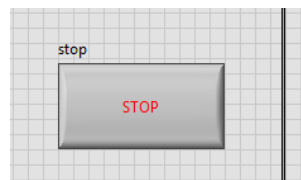


3. Click the simulate button to burst all the parameters specified.



4. In the InstrumentStudio Application there should be a response in the scope according with the acceleration and temperature selected

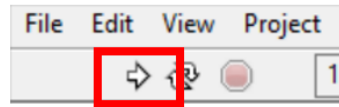
5. Click the stop button and unselect all the previous parameters



Running in Failing Mode

This mode applies a failing parameter in the accelerometer outputs

1. Click the run button



2. Specify the temperature and acceleration parameters you would like to simulate, use integer values only. The acceleration range is ± 200 , the valid temperature range is -40 to $+85$

WORKING MODE

Simulation Temperature: Celsius Simulation Acceleration: g Sensitivity mV/g: mV/g

Chip State	Model	Temperature	Serial Number
	377	-40	3216
	377		3216

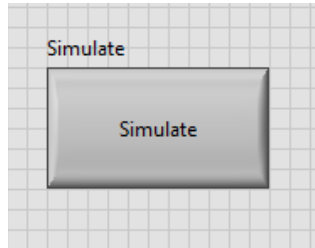
3. Choose the axis that will fail during the simulation, it can fail in each, some, or all outputs. Also select the Failing mode to simulate. Only choose one of the types at a time.

FAILING MODES

X axis 	Y axis 	Z axis
Communication Error 	Proof mass Fail 	Vibration

Communication Error, Proof mass Fail and Vibration

4. Click the simulate button to burst all the parameters specified.



5. In the InstrumentStudio Application there should be a response in the scope according with the acceleration and temperature selected

6. Click the stop button and unselect all the previous parameters



TestStand DUT Sequence

Run any of the modes previously explained, but don't stop the simulation.

In the PXIE TestStand 2021 run the "MEMS-TestSequence" sequence. It will look like this:

MEMS - TestSequence.seq		
Steps: MainSequence		
STEP	DESCRIPTION	SETTINGS
- Setup (1)		
L Initialize Scope	Action, InitializeScope.vi	Result Recording: Disabled
<End Group>		
- Main (22)		
L ChipState	Numeric Limit Test, x == 1, DataCaptureTest.vi	Post Action
L SerialNumber	Numeric Limit Test, x == 3216, DataCaptureTest.vi	
L Model	Numeric Limit Test, x == 377, DataCaptureTest.vi	
L VoltageLevel X	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Post Action, Additional Results
L VoltageLevel Y	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Additional Results
L VoltageLevel Z	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Additional Results
* f	Locals ErrorX==True	
L VoltageLevel X Communication	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Post Action, Additional Results
L VoltageLevel X ProofMass	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results
L VoltageLevel X Vibration	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results
End		
* f	Locals ErrorY==True	
L VoltageLevel Y Communication	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Post Action, Additional Results
L VoltageLevel Y ProofMass	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results
L VoltageLevel Y Vibration	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results
End		
* f	Locals ErrorZ==True	
L VoltageLevel Z Communication	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Post Action, Additional Results
L VoltageLevel Z ProofMass	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results
L VoltageLevel Z Vibration	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results
End		
Additional Results	Records 6 Additional Result(s)	
<End Group>		
- Cleanup (1)		
L Shut Down Scope	Action, CloseScope.vi	
<End Group>		

After running the sequence a report will be generated and the steps status will be shown next to each step.

Single Pass - MEMS - TestSequence.s... MEMS - TestSequence.seq			
Steps			
STEP	DESCRIPTION	SETTINGS	STATUS
- Setup (1)			
L Initialize Scope	Action, InitializeScope.vi	Result Recording: Disabled	Done
<End Group>			
- Main (22)			
L ChipState	(1), Numeric Limit Test, x == 1, DataCaptureTest.vi	Post Action	Passed
L SerialNumber	(3216), Numeric Limit Test, x == 3216, DataCaptureTest.vi		Passed
L Model	(377), Numeric Limit Test, x == 377, DataCaptureTest.vi		Passed
L VoltageLevel X	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Post Action, Additional Results	Failed
L VoltageLevel Y	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Additional Results	Passed
L VoltageLevel Z	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Additional Results	Passed
* f	Locals ErrorX==True		Done
L VoltageLevel X Communication	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Post Action, Additional Results	Passed
L VoltageLevel X ProofMass	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results	Failed
L VoltageLevel X Vibration	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results	Skipped
End			Done
* f	Locals ErrorY==True		Done
L VoltageLevel Y Communication	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Post Action, Additional Results	
L VoltageLevel Y ProofMass	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results	
L VoltageLevel Y Vibration	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results	
End			Done
* f	Locals ErrorZ==True		Done
L VoltageLevel Z Communication	Pass/Fail Test, VoltageLevelTest.vi	Loop, Post Expression, Post Action, Additional Results	
L VoltageLevel Z ProofMass	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results	
L VoltageLevel Z Vibration	Pass/Fail Test, VoltageLevelTest.vi	Loop, Precondition, Post Expression, Post Action, Additional Results	
End			Done
Additional Results	Records 6 Additional Result(s)		Done
<End Group>			
- Cleanup (1)			
L Shut Down Scope	Action, CloseScope.vi		Done
<End Group>			

For each axis the result data will show if there occurred an error and which type it was.

Step	Status	Measurement	Units	Limits			
				Nominal Value	Low Limit	High Limit	Comparison Type
End(If)	Done						
Additional Results	Done						
TestResults/Data							
Error X:	True						
Tipo Error X:	Proof Mass						
Error Y:	False						
Tipo Error Y:	NO Error						
Error Z:	True						
Tipo Error Z:	Proof Mass						
Shut Down Scope	Done						

After the test sequence is done click the stop button in the API running on labView and unselect all the previous parameters



A.12 Manual de usuario para el sistema caracteriza- ción

Manual de operación

Sistema de caracterización de un acelerómetro MEMS

-Carla Corina Gómez Carrasquilla

Configuraciones Iniciales	3
Consideraciones de Seguridad	3
Sistema aceleración	3
Eje de medición	4
Proceso de toma de datos	4
Inicio de sistema adquisición de datos	4
Inicio del sistema de aceleración	5
Extracción de los datos del sistema	6

Configuraciones Iniciales

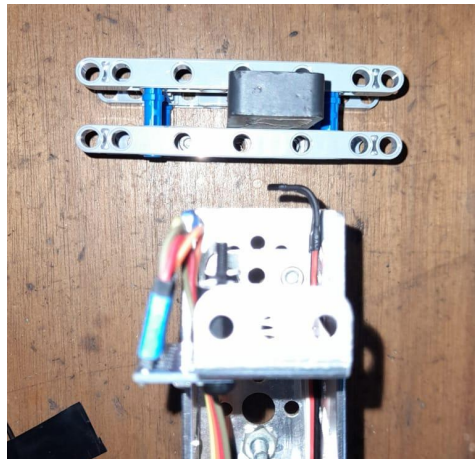
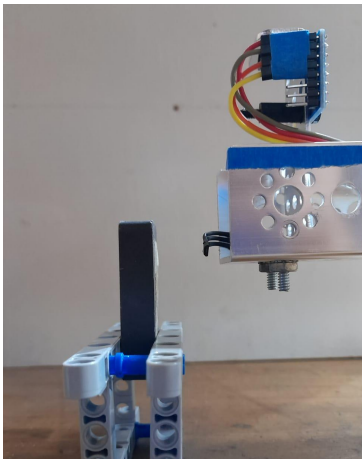
Consideraciones de Seguridad

Dado que el sistema gira a más de 250 RPM, por razones de seguridad no se debe utilizar ropa holgada o de mangas anchas, ya que estas pueden ser capturadas por el sistema giratorio.

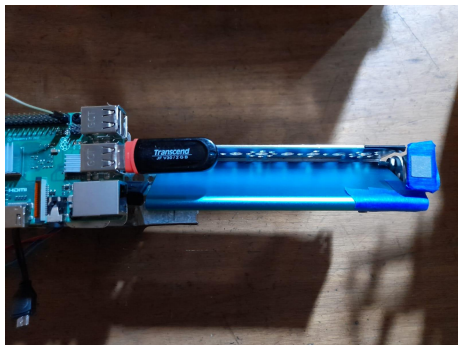
1. Utilizar anteojos de seguridad y no utilizar ningún tipo de adorno colgante (aretes largos, collares, pulseras) a la hora de trabajar con el dispositivo.
2. No acercarse al sistema mientras esté en funcionamiento
3. Mantener las tapas de la cámara de aislamiento cerradas mientras el sistema esté conectado.
4. Únicamente utilizar el botón para encender y apagar el motor. Si el botón no está apagado no conectar o desconectar la Raspberry Pi ni extraer la llave malla.

Sistema aceleración

1. Verificar que el eje del motor esté bien acoplado mediante el prisionero en el acople.
2. Girar el brazo y asegurarse que el sensor de efecto Hall pase por el campo magnético (Sur) del imán en la mesa.



3. Al colocar la batería verificar que esté sujeta por sus extremos y esté dentro se la viga en U.

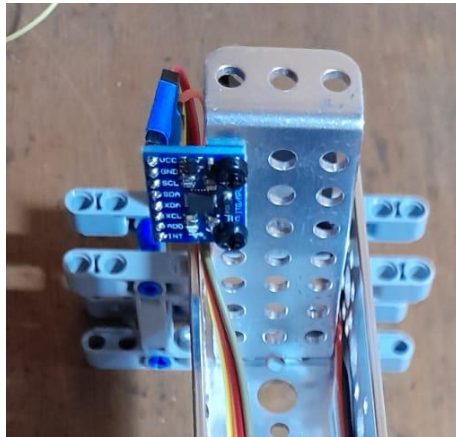
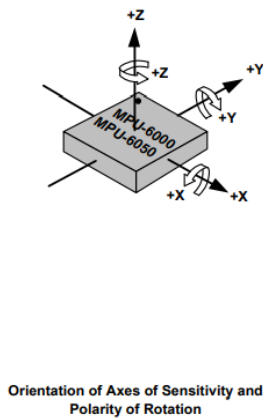


Eje de medición

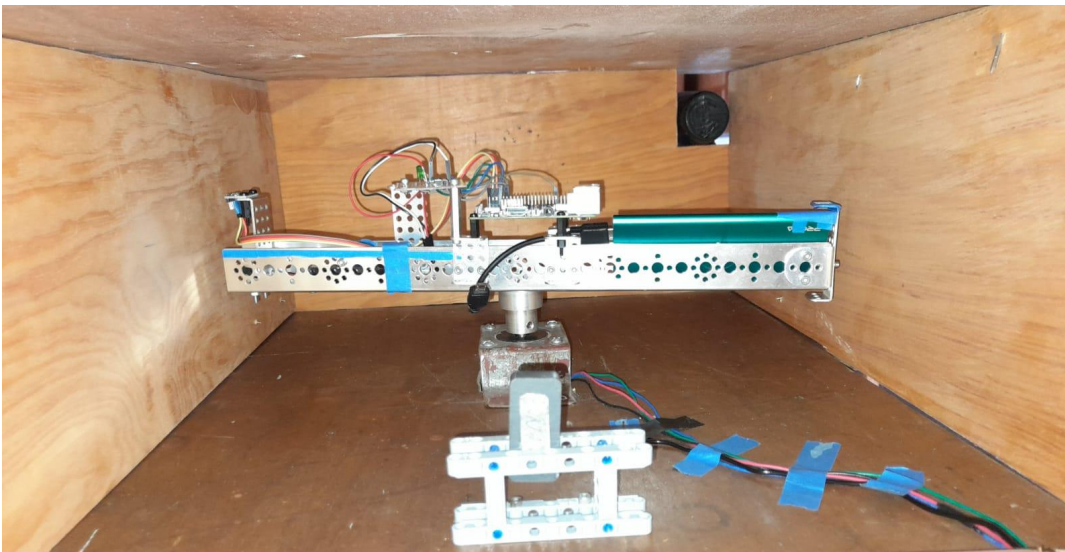
Según sea el eje que se quiera caracterizar se debe direccionar el mismo hacia afuera del sistema de caracterización, con el fin de obtener la medición de la aceleración centrífuga en el eje a caracterizar.

Por ejemplo:

Para caracterizar el eje X del sensor, color el sensor con el Eje X en dirección hacia afuera del sistema de rotación, con esto se obtiene datos desde 0 a 16g en este eje.



Con esto queda el sistema listo para tomar datos.

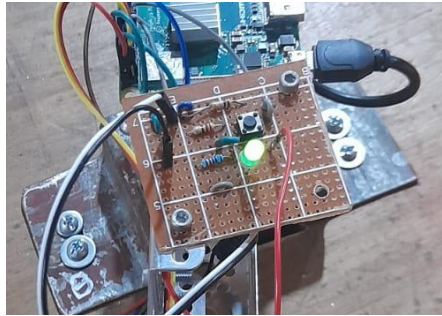


Proceso de toma de datos

Inicio de sistema adquisición de datos

1. Asegurarse que no haya ningún cable suelto en la Raspberry Pi
2. Conectar el cable de alimentación de la Raspberry Pi

3. Verificar que se enciendan las luces de Power en la Raspberry Pi
4. Una vez se encienda el LED verde del circuito se ha iniciado el código para almacenar datos del sensor.



Inicio del sistema de aceleración

1. Inmediatamente se haya encendido el LED verde del circuito se debe cerrar la cámara de aislamiento térmico.



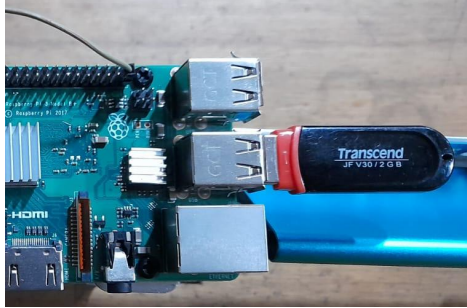
2. Activar manualmente la fuente que alimenta el sistema de control del motor fuera de la cámara térmica..
3. Esto activa el motor a pasos, el sistema gira a favor de las manecillas del reloj.

Atención: El motor empieza a girar inmediatamente, por lo que no se debe cerrar la tapa de la cámara térmica antes y no colocar ningún elemento dentro de esta.

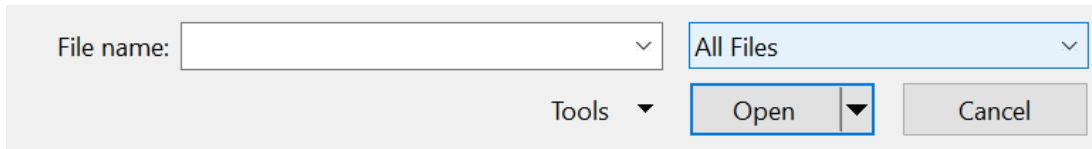
4. El sistema empezará a subir su velocidad hasta llegar a 270 RPM y luego disminuirá su velocidad hasta detenerse.
5. Una vez el sistema se haya detenido, apagar la manualmente la fuente que alimenta el sistema de control del motor.
6. Abrir la tapa de cámara térmica
7. Presionar el botón en el circuito para guardar los datos obtenidos en la llave maya y apagar automáticamente la Raspberry Pi.
8. Desconectar el cable de alimentación de la Raspberry Pi.

Extracción de los datos del sistema

1. Extraer la llave maya de la Raspberry Pi.



2. Conectar la llave maya en un computador personal.
3. Abrir Excel
4. File>>Open>>Browse
5. Ir a la llave maya y habilitar todos los archivos



6. Seleccionar el archivo con la fecha y hora a la que se inició la toma de datos
7. Seleccionar ok/si
8. Next
9. Comma
10. Next
11. Finish
12. Con esto se pueden visualizar los datos en columnas.
13. Guardarlo en formato Excel Workbook para poder editar los valores y graficarlos.

Con esto se ha logrado extraer los datos del sensor luego de ser acelerado en el sistema de caracterización.

Anexos

A.13 Hoja de Datos myRio

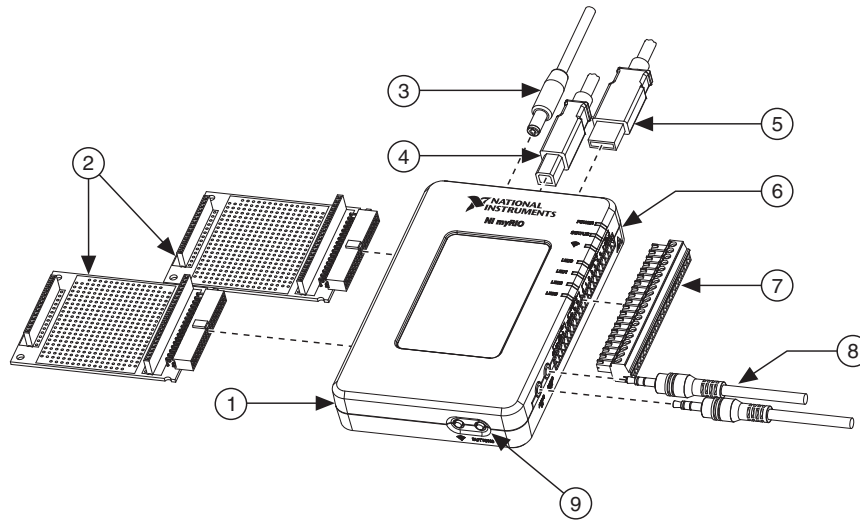
Seguidamente se adjunta una muestra de la hoja de datos del myRio, el documento completo se encuentra en [\[12\]](#)

USER GUIDE AND SPECIFICATIONS

NI myRIO-1900

The National Instruments myRIO-1900 is a portable reconfigurable I/O (RIO) device that students can use to design control, robotics, and mechatronics systems. This document contains pinouts, connectivity information, dimensions, mounting instructions, and specifications for the NI myRIO-1900.

Figure 1. NI myRIO-1900



- | | |
|--|---|
| 1 NI myRIO-1900 | 6 LEDs |
| 2 myRIO Expansion Port (MXP) Breakouts (One Included in Kit) | 7 Mini System Port (MSP) Screw-Terminal Connector |
| 3 Power Input Cable | 8 Audio In/Out Cables (One Included in Kit) |
| 4 USB Device Cable | 9 Button0 |
| 5 USB Host Cable (Not Included in Kit) | |

Safety Information



Caution Do not operate the hardware in a manner not specified in this document and in the user documentation. Misuse of the hardware can result in a hazard. You can compromise the safety protection if the hardware is damaged in any way. If the hardware is damaged, return it to National Instruments for repair.

Clean the hardware with a soft, nonmetallic brush. Make sure that the hardware is completely dry and free from contaminants before returning it to service.

Electromagnetic Compatibility Guidelines

This product was tested and complies with the regulatory requirements and limits for electromagnetic compatibility (EMC) stated in the product specifications. These requirements and limits provide reasonable protection against harmful interference when the product is operated in the intended operational electromagnetic environment.

This product is intended for use in commercial locations. There is no guarantee that harmful interference will not occur in a particular installation or when the product is connected to a test object. To minimize interference with radio and television reception and prevent unacceptable performance degradation, install and use this product in strict accordance with the instructions in the product documentation.

Furthermore, any modifications to the product not expressly approved by National Instruments could void your authority to operate it under your local regulatory rules.



Caution This product was tested for EMC compliance using myRIO application software. The maximum length for USB cables is 2.0 m (6.6 ft), and the maximum length for signal wires is 30.0 cm (11.8 in.).



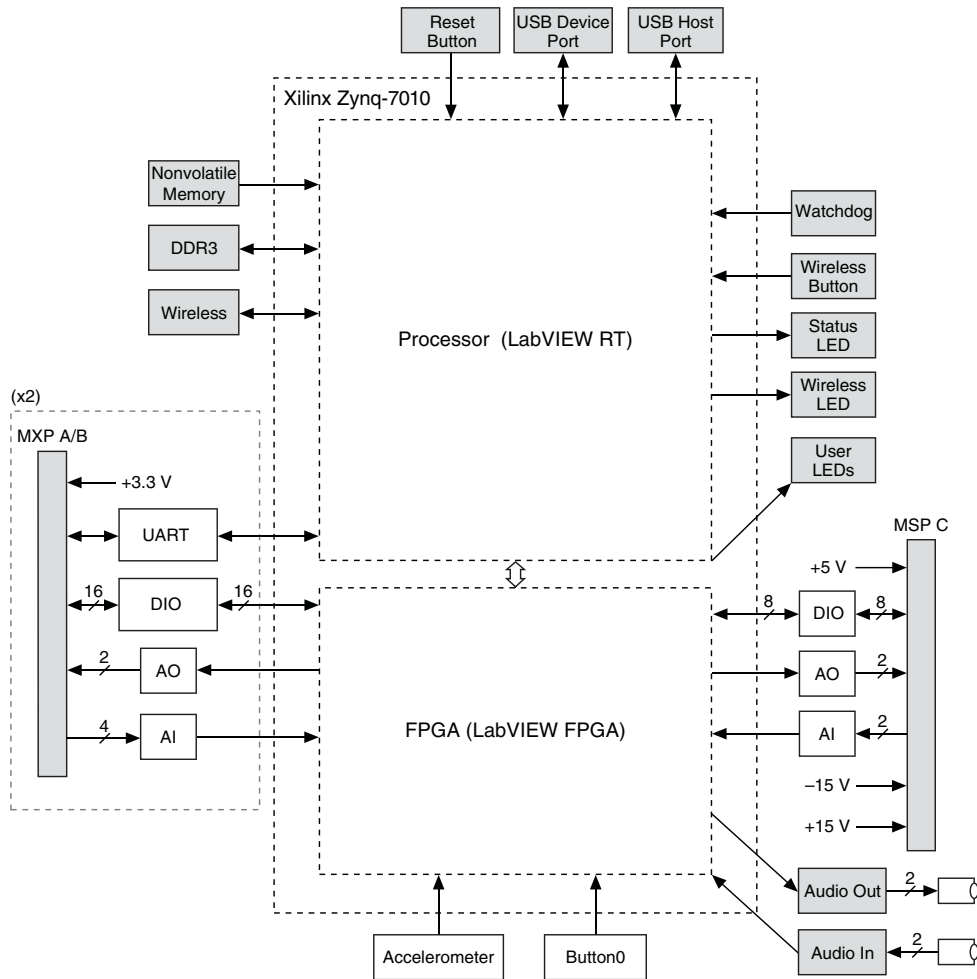
Caution The mounting keyholes on the back of the NI myRIO-1900 are sensitive to electrostatic discharge (ESD). When handling the device, be careful not to touch inside the keyholes.

Hardware Overview

The NI myRIO-1900 provides analog input (AI), analog output (AO), digital input and output (DIO), audio, and power output in a compact embedded device. The NI myRIO-1900 connects to a host computer over USB and wireless 802.11b,g,n.

The following figure shows the arrangement and functions of NI myRIO-1900 components.

Figure 2. NI myRIO-1900 Hardware Block Diagram



A.14 Hoja de Datos MPU-6050

Seguidamente se adjunta una muestra de la hoja de datos del MPU-6050, el documento completo se encuentra en [\[22\]](#)

	MPU-6000/MPU-6050 Product Specification	Document Number: PS-MPU-6000A-00 Revision: 3.3 Release Date: 5/16/2012
---	--	--

3 Product Overview

3.1 MPU-60X0 Overview

MotionInterface™ is becoming a “must-have” function being adopted by smartphone and tablet manufacturers due to the enormous value it adds to the end user experience. In smartphones, it finds use in applications such as gesture commands for applications and phone control, enhanced gaming, augmented reality, panoramic photo capture and viewing, and pedestrian and vehicle navigation. With its ability to precisely and accurately track user motions, MotionTracking technology can convert handsets and tablets into powerful 3D intelligent devices that can be used in applications ranging from health and fitness monitoring to location-based services. Key requirements for MotionInterface enabled devices are small package size, low power consumption, high accuracy and repeatability, high shock tolerance, and application specific performance programmability – all at a low consumer price point.

The MPU-60X0 is the world's first integrated 6-axis MotionTracking device that combines a 3-axis gyroscope, 3-axis accelerometer, and a Digital Motion Processor™ (DMP) all in a small 4x4x0.9mm package. With its dedicated I²C sensor bus, it directly accepts inputs from an external 3-axis compass to provide a complete 9-axis MotionFusion™ output. The MPU-60X0 MotionTracking device, with its 6-axis integration, on-board MotionFusion™, and run-time calibration firmware, enables manufacturers to eliminate the costly and complex selection, qualification, and system level integration of discrete devices, guaranteeing optimal motion performance for consumers. The MPU-60X0 is also designed to interface with multiple non-inertial digital sensors, such as pressure sensors, on its auxiliary I²C port. The MPU-60X0 is footprint compatible with the MPU-30X0 family.

The MPU-60X0 features three 16-bit analog-to-digital converters (ADCs) for digitizing the gyroscope outputs and three 16-bit ADCs for digitizing the accelerometer outputs. For precision tracking of both fast and slow motions, the parts feature a user-programmable gyroscope full-scale range of ± 250 , ± 500 , ± 1000 , and $\pm 2000^\circ/\text{sec}$ (dps) and a user-programmable accelerometer full-scale range of $\pm 2g$, $\pm 4g$, $\pm 8g$, and $\pm 16g$.

An on-chip 1024 Byte FIFO buffer helps lower system power consumption by allowing the system processor to read the sensor data in bursts and then enter a low-power mode as the MPU collects more data. With all the necessary on-chip processing and sensor components required to support many motion-based use cases, the MPU-60X0 uniquely enables low-power MotionInterface applications in portable applications with reduced processing requirements for the system processor. By providing an integrated MotionFusion output, the DMP in the MPU-60X0 offloads the intensive MotionProcessing computation requirements from the system processor, minimizing the need for frequent polling of the motion sensor output.

Communication with all registers of the device is performed using either I²C at 400kHz or SPI at 1MHz (MPU-6000 only). For applications requiring faster communications, the sensor and interrupt registers may be read using SPI at 20MHz (MPU-6000 only). Additional features include an embedded temperature sensor and an on-chip oscillator with $\pm 1\%$ variation over the operating temperature range.

By leveraging its patented and volume-proven Nasiri-Fabrication platform, which integrates MEMS wafers with companion CMOS electronics through wafer-level bonding, InvenSense has driven the MPU-60X0 package size down to a revolutionary footprint of 4x4x0.9mm (QFN), while providing the highest performance, lowest noise, and the lowest cost semiconductor packaging required for handheld consumer electronic devices. The part features a robust 10,000g shock tolerance, and has programmable low-pass filters for the gyroscopes, accelerometers, and the on-chip temperature sensor.

For power supply flexibility, the MPU-60X0 operates from VDD power supply voltage range of 2.375V-3.46V. Additionally, the MPU-6050 provides a VLOGIC reference pin (in addition to its analog supply pin: VDD), which sets the logic levels of its I²C interface. The VLOGIC voltage may be 1.8V $\pm 5\%$ or VDD.

The MPU-6000 and MPU-6050 are identical, except that the MPU-6050 supports the I²C serial interface only, and has a separate VLOGIC reference pin. The MPU-6000 supports both I²C and SPI interfaces and has a single supply pin, VDD, which is both the device's logic reference supply and the analog supply for the part. The table below outlines these differences:

	MPU-6000/MPU-6050 Product Specification	Document Number: PS-MPU-6000A-00 Revision: 3.3 Release Date: 5/16/2012
---	--	--

Primary Differences between MPU-6000 and MPU-6050

Part / Item	MPU-6000	MPU-6050
VDD	2.375V-3.46V	2.375V-3.46V
VLOGIC	n/a	1.71V to VDD
Serial Interfaces Supported	I ² C, SPI	I ² C
Pin 8	/CS	VLOGIC
Pin 9	AD0/SDO	AD0
Pin 23	SCL/SCLK	SCL
Pin 24	SDA/SDI	SDA

	MPU-6000/MPU-6050 Product Specification	Document Number: PS-MPU-6000A-00 Revision: 3.3 Release Date: 5/16/2012
---	--	--

4 Applications

- *BlurFree™* technology (for Video/Still Image Stabilization)
- *AirSign™* technology (for Security/Authentication)
- *TouchAnywhere™* technology (for "no touch" UI Application Control/Navigation)
- *MotionCommand™* technology (for Gesture Short-cuts)
- Motion-enabled game and application framework
- InstantGesture™ iG™ gesture recognition
- Location based services, points of interest, and dead reckoning
- Handset and portable gaming
- Motion-based game controllers
- 3D remote controls for Internet connected DTVs and set top boxes, 3D mice
- Wearable sensors for health, fitness and sports
- Toys

	MPU-6000/MPU-6050 Product Specification	Document Number: PS-MPU-6000A-00 Revision: 3.3 Release Date: 5/16/2012
---	--	--

5 Features

5.1 Gyroscope Features

The triple-axis MEMS gyroscope in the MPU-60X0 includes a wide range of features:

- Digital-output X-, Y-, and Z-Axis angular rate sensors (gyroscopes) with a user-programmable full-scale range of ± 250 , ± 500 , ± 1000 , and $\pm 2000^\circ/\text{sec}$
- External sync signal connected to the FSYNC pin supports image, video and GPS synchronization
- Integrated 16-bit ADCs enable simultaneous sampling of gyros
- Enhanced bias and sensitivity temperature stability reduces the need for user calibration
- Improved low-frequency noise performance
- Digitally-programmable low-pass filter
- Gyroscope operating current: 3.6mA
- Standby current: 5 μ A
- Factory calibrated sensitivity scale factor
- User self-test

5.2 Accelerometer Features

The triple-axis MEMS accelerometer in MPU-60X0 includes a wide range of features:

- Digital-output triple-axis accelerometer with a programmable full scale range of $\pm 2g$, $\pm 4g$, $\pm 8g$ and $\pm 16g$
- Integrated 16-bit ADCs enable simultaneous sampling of accelerometers while requiring no external multiplexer
- Accelerometer normal operating current: 500 μ A
- Low power accelerometer mode current: 10 μ A at 1.25Hz, 20 μ A at 5Hz, 60 μ A at 20Hz, 110 μ A at 40Hz
- Orientation detection and signaling
- Tap detection
- User-programmable interrupts
- High-G interrupt
- User self-test

5.3 Additional Features

The MPU-60X0 includes the following additional features:

- 9-Axis MotionFusion by the on-chip Digital Motion Processor (DMP)
- Auxiliary master I²C bus for reading data from external sensors (e.g., magnetometer)
- 3.9mA operating current when all 6 motion sensing axes and the DMP are enabled
- VDD supply voltage range of 2.375V-3.46V
- Flexible VLOGIC reference voltage supports multiple I²C interface voltages (MPU-6050 only)
- Smallest and thinnest QFN package for portable devices: 4x4x0.9mm
- Minimal cross-axis sensitivity between the accelerometer and gyroscope axes
- 1024 byte FIFO buffer reduces power consumption by allowing host processor to read the data in bursts and then go into a low-power mode as the MPU collects more data
- Digital-output temperature sensor
- User-programmable digital filters for gyroscope, accelerometer, and temp sensor
- 10,000 g shock tolerant
- 400kHz Fast Mode I²C for communicating with all registers
- 1MHz SPI serial interface for communicating with all registers (MPU-6000 only)
- 20MHz SPI serial interface for reading sensor and interrupt registers (MPU-6000 only)

Índice alfabético

aceleración, 7
acelerómetros, 8
API, 18

caracterización, 29
centrífuga, 46
comercial, 28
comunicación, 59
costo, 91

datos, 49

FPGA, 16

MEMS, 5
metricas, 31
modelo, 12
modos, 61

objetivos, 2

problemas, 85
prototipo, 40

simulación, 52
simulador, 36
SPI, 20

velocidad, 48