

**COMPROBACIÓN Y AJUSTE DE MODELO SOBRE
EFECTO DE LA VARIACIÓN DE TEMPERATURA EN UNA
MICROBALANZA DE CRISTAL DE CUARZO (QCM)**

Modalidad: Exploratorio

TOMÁS GONZÁLEZ CANO

**Trabajo de grado para optar al título de
Ingeniero Biomédico**

Yeison Javier Montagut Ferizzola PhD



**UNIVERSIDAD EIA
INGENIERÍA BIOMÉDICA
ENVIGADO
2020**

AGRADECIMIENTOS

Agradezco al Director del trabajo, Yeison Montagut, por proponer este proyecto tan influyente en un biosensor con tanto potencial.

Agradezco a Andrés Briceño por ser mi compañero durante la toma de datos y mi mano derecha durante el desarrollo de la tesis.

Agradezco al Ingeniero Tomás Echavarría por ayudarnos a configurar el sistema y ponerlo a punto para obtener los datos necesarios.

Agradezco a mi familia que siempre apoyó mi mentalidad curiosa e inquieta por aprender que me ha ayudado a crecer como gran investigador e ingeniero.

Agradezco a mi mejor amiga, Carolina Cardona, por haberme apoyado durante toda la carrera y estar ahí en las buenas y las malas.

Muchas gracias a todos.

CONTENIDO

	pág.
INTRODUCCIÓN	13
1. PRELIMINARES	15
1.1 Planteamiento del problema	15
1.2 Objetivos del proyecto.....	16
1.2.1 Objetivo General.....	16
1.2.2 Objetivos Específicos	16
1.3 Marco de referencia.....	17
1.3.1 Marco teórico	17
2. METODOLOGÍA	22
3. PRESENTACIÓN Y DISCUSIÓN DE RESULTADOS	25
3.1 Entendimiento del Modelo	25
3.2 Traducción y Comprobación del modelo.....	30
3.2.1 Obtención de la MSRF en presencia de un fluido	30
3.2.2 Modelo de inductancia por cambios en la temperatura	30
3.2.3 Modelo de frecuencia, magnitud y fase con respecto al cambio en la temperatura .	32
3.2.4 Modelo con variación de temperatura de una QCM con carga en presencia de un fluido	34
3.3 Obtención y almacenamiento de datos del modelo.....	40
3.4 Creación de ambiente y adquisición de datos del sensor.....	42
3.4.1 Sistema de caracterización y adquisición de datos	42
3.4.2 Ambiente del sensor	44
3.4.3 Aplicación de control	46

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

3.5	Proceso de adquisición y Almacenamiento de datos	47
3.5.1	Puesta a punto del sistema de adquisición	47
3.5.2	Adquisición de los datos	49
3.5.3	Almacenamiento de datos y generación de gráficas	49
3.6	observación de diferencias	50
3.7	Localización de ajuste.....	52
3.8	modificación y comprobación del modelo	52
4.	CONCLUSIONES Y CONSIDERACIONES FINALES.....	54
	REFERENCIAS	55

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

LISTA DE TABLAS

	pág.
Tabla 1: Propiedades del cuarzo en corte AT.....	25
Tabla 2: Variación de la frecuencia con respecto a una carga en función del tiempo.....	36
Tabla 3: Medidas de temperatura vs fase a) Experimental b) Modelado	51
Tabla 4: Barrido de razón de cambio de frecuencia por cada 0.1°C en pasos de 1Hz y cambio en fase por cada 0.1°C resultante.....	52
Tabla 5: Barrido de cambios en frecuencia por cada 0.1°C y cambios en la fase resultantes.	53

LISTA DE FIGURAS

	pág.
Figura 1: Ilustración de cambio de frecuencia debido al cambio de masa en un inmunosensor basado en QCM.....	18
Figura 2: Modelo BVD.....	20
Figura 3: Modelo BVD extendido.....	20
Figura 4: Interfaz Gráfica (GUI) de modelo de una QCM variando su frecuencia en 10 y 100 Hz, hallando valores de L_t	31
Figura 5: Comportamiento de L_t , fase y magnitud con respecto al cambio de f_{sl} : a) Listas de datos numéricos. b) Gráficas de cada uno vs el cambio en frecuencia.....	34
Figura 6: Comportamiento de frecuencia con respecto al tiempo de una QCM cuando se inyecta una carga.....	35
Figura 7: Variación de la fase en función del tiempo en presencia de una carga.....	37
Figura 8: Fases de QCM en presencia de una carga con distintas variaciones de temperatura.....	38
Figura 9: Interfaz de usuario para visualización de posibles fases debido a la variación de temperatura.....	39
Figura 10: Interfaz de usuario con datos de diferencias de fases.....	41
Figura 11: Sistema de caracterización. A la izquierda se generador de señales, arriba el medidor de fases y a la derecha el conversor analógico-digital con Raspberry Pi 2.....	42
Figura 12: Generador de señales.....	43
Figura 13: Circuito de medición de Fases y Magnitudes.....	43
Figura 14: Circuito de adquisición y transmisión de mediciones.....	44
Figura 15: Módulo termoelectrico con sistema de caracterización.....	45
Figura 16: Módulo termoelectrico cerrado con sistema de caracterización dentro.....	45
Figura 17: Aplicación para control de sistema de perfilación y medición.....	46
Figura 18: Mediciones de fase con uso de referencia a 27°C y 10MHz en oscilador.....	48

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Figura 19: Mediciones de fase calibradas en 27°C.....	48
Figura 20: Gráfica de fases y temperaturas vs tiempo en HTML	50

LISTA DE ANEXOS

	pág.
Anexo 1: Algoritmo de Python3 con experimentos y modelación de variaciones de frecuencias	57

RESUMEN

La microbalanza de cristal de cuarzo (QCM por sus siglas en inglés) es un sensor muy útil a la hora de medir pequeños cambios de masa debido a que los puede reconocer a un nivel micro y nanométricos, siendo así un sensor muy útil para distintas aplicaciones industriales, biológicas e ingenieriles a la hora de medir la masa de una sustancia específica en valores que una balanza convencional nunca podrá medir.

Muchos avances se han logrado con esta herramienta para obtener distintas mediciones de masa en las distintas aplicaciones explicadas anteriormente, sin embargo estas mediciones pueden variar mucho con distintas condiciones ambientales haciendo que las mediciones que arroje el sensor sean siempre distintas evitando una medida estándar de masa.

Para resolver este problema se creó un sistema que rodea el sensor estabilizando las condiciones ambientales que afectan al sensor, sin embargo, esto aumenta el tamaño de este sensor, quitando su simplicidad y portabilidad a la hora de usarlo; por esta razón se necesita de un circuito con el cual el sensor pueda autocompensar el efecto de la temperatura sobre sus medidas, por consiguiente se realizó un modelo por el cual pudiera obtenerse el efecto de los cambios de temperatura sobre las medidas del sensor con base en ciertas suposiciones teóricas.

En este trabajo se realizará todo el proceso para comprobar si el modelo supuesto anteriormente puede tomarse en un ambiente el cual no tiene condiciones ideales de temperatura; de no ser válido el modelo, se realizarán los ajustes necesarios para que tal modelo pueda ser tomado para la identificación de variaciones en un ambiente real y no ideal.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Luego de este proceso se obtuvieron resultados reales, se compararon contra los modelados llevando a modificar el modelo ya planteado para un mayor ajuste al comportamiento experimental.

Al final se obtuvo una proporción de cambio en la frecuencia con respecto a variaciones en la temperatura de 0.1°C con la cual se puede predecir de una manera aproximada en cuanto afectará al cristal el cambio en la temperatura.

Palabras clave: Microbalanza, Frecuencia, Fase, Temperatura, Ruido

ABSTRACT

The quartz crystal microbalance (QCM) is a very useful sensor when measuring small changes in mass because it can recognize them at a micro and nanometric level, thus being a very useful sensor for different industrial, biological and engineering applications when measuring the mass of a specific substance in values that a conventional balance will never be able to measure.

Many advances have been achieved with this tool to obtain different mass measurements in the different applications explained above, however these measurements can vary a lot with different environmental conditions making the measurements thrown by the sensor always different avoiding a standard measurement of mass.

To solve this problem, a system was created that surrounds the sensor stabilizing the environmental conditions that affect the sensor, however, this increases the size of this sensor, removing its simplicity and portability when using it; for this reason, a circuit is needed with which the sensor can self-compensate the effect of temperature on its measurements, therefore a model was made by which the effect of temperature changes on the sensor measurements could be obtained based on certain theoretical assumptions.

In this work, the whole process will be carried out to check if the model previously assumed can be taken in an environment which does not have ideal temperature conditions; if the model is not valid, the necessary adjustments will be made so that such model can be taken to identify variations in a real environment and not in an ideal one.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

After this process real results were obtained and compared against the modeling leading to modify the model already raised for a greater adjustment to the experimental behavior.

In the end, a proportion of change in frequency with respect to variations in temperature of 0.1°C was obtained, with which it is possible to predict in an approximate way how much the change in temperature will affect the glass.

Keywords: microbalance, frequency, phase, temperature, noise

INTRODUCCIÓN

El diagnóstico de enfermedades es una herramienta fundamental en la profesión de un funcionario de la salud, ya que, al lograr un buen diagnóstico, esta persona hará uso de sus conocimientos para el tratamiento del paciente; haciendo del diagnóstico preciso una base fuerte para un buen servicio en el área de la salud. De esta importancia nace una tarea fundamental en la profesión del ingeniero biomédico la cual es brindarle al personal de la salud herramientas con las cuales se pueda obtener un diagnóstico con mayor velocidad y precisión para que el tratamiento al paciente sea efectivo.

La microbalanza de cristal de cuarzo (QCM por sus siglas en inglés) ha sido un método últimamente usado para detectar la presencia de entes microscópicos, ya que, según Montagut (2011) este método presenta una resolución comparable con otros métodos para la caracterización de fluidos y la detección de analitos, haciendo de este una gran herramienta para la detección de enfermedades infecciosas en una sustancia biológica como la sangre, logrando un diagnóstico veloz y preciso de las enfermedades antes de que estas se desarrollen.

Debido a la gran resolución de este instrumento, el mismo es afectado por condiciones ambientales como presión, temperatura, viscosidad del aire, etc. Pero cuando se realizan mediciones con una pequeña capa de fluido, puede suponerse que la temperatura es la única variable influyente. (Rahtu & Ritala, 2002)

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Por consiguiente para compensar el efecto de la temperatura se propuso un modelo matemático con el cual se puede medir, en valores electrónicos, como afecta la temperatura a los resultados que este sensor debe otorgar.

En este trabajo se modelará matemáticamente los resultados que brindan distintas temperaturas y luego se obtendrán resultados experimentales en un ambiente controlado, para luego ser comparados ambos resultados para evaluar la certeza del modelo.

1. PRELIMINARES

1.1 PLANTEAMIENTO DEL PROBLEMA

Los inmunosensores son una herramienta que puede ser útil a la hora del diagnóstico de enfermedades infecciosas (neumonía, dengue, tuberculosis, etc.), ya que al haber presencia de anticuerpos relacionados a la enfermedad correspondiente en el sensor, este permitirá evaluar si hay antígenos de alguna de estas enfermedades en la sangre u otro fluido corporal mediante la unión antígeno-anticuerpo y el cambio de masa en el sensor (Pohanka, 2018).

Los inmunosensores basados en QCM (Quartz cristal microbalance) son simples, de bajo costo, necesitan menos tiempo de trabajo, entregan medidas en tiempo real, son seguros y están comercialmente disponibles (Pohanka, 2018; Vashist & Vashist, 2011), siendo una herramienta confiable para conseguir los datos necesarios de un buen diagnóstico. Las QCM funcionan de tal manera que al tener anticuerpos inmovilizados sobre su superficie, sensorán los pequeños cambios de masa que existen durante la reacción inmunológica (Montagut et al., 2011).

Una de las principales razones por la cual se usan las QCM es su gran sensibilidad y precisión a la hora de medir muestras biológicas (Rahtu & Ritala, 2002); sin embargo, son muy variables con respecto a cambios temperatura, lo que lleva a que los resultados sean precisos, pero no exactos, lo cual causa variabilidad en los diagnósticos debido a las variaciones en el clima (humedad, presión, temperatura, etc.) que pueden presentarse en lugares donde se presenten tanto estaciones (Norte o sur del planeta) como en regiones en

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

las cuales se cambie fácilmente de temperatura y otras condiciones ambientales (Distintas regiones de Colombia) mediante los distintos eventos naturales como lluvias, tormentas, entre otros (IDEAM, 2018).

Muchos modelos matemáticos se han propuesto para predecir el comportamiento de este sensor con respecto al cambio de temperatura (Cao-Paz, Rodríguez-Pardo, & Fariña, 2014), con el objetivo de desarrollar un sistema que pueda atenuar estos efectos ambientales sobre el sensor (Ortiz, 2016); el inconveniente con estos modelos es que fueron previamente realizados con bases teóricas e ideales o variaciones de temperatura computarizadas siendo estándares de variación de temperatura ideales pero no reales, por lo tanto, se necesita comprobar y ajustar experimentalmente el efecto de la variación de temperatura en una microbalanza de cristal de cuarzo para que el modelo funcione tanto en simulaciones como en la realidad.

1.2 OBJETIVOS DEL PROYECTO

1.2.1 Objetivo General

Evaluar el modelo de comportamiento de una QCM con respecto a la temperatura de manera experimental para la predicción de resultados aplicables en un entorno real.

1.2.2 Objetivos Específicos

- Establecer las características de los modelos existentes para la obtención de funciones matemáticas y bases aplicables.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

- Evaluar los resultados del modelo con respecto a los cambios experimentales debido a las variaciones de temperatura para la comprobación de su exactitud y precisión.
- Ajustar las características del modelo con base en los resultados reales obtenidos para lograr un acercamiento a la variación real.

1.3 MARCO DE REFERENCIA

1.3.1 Marco teórico

- Biosensor

Los biosensores “son dispositivos analíticos conformados por un elemento biológico de reconocimiento asociado a un mecanismo de detección e interpretación de la señal obtenida de la interacción entre el analito y el dispositivo analítico” (Jiménez & León, 2009), es decir, un biosensor es un dispositivo que contiene una parte biológica la cual mediante, un transductor envía señales medibles, comúnmente eléctricas.

Se compone de 2 elementos:

- Elemento biológico: parte que se compone de algún elemento de la naturaleza que reacciona con el analito a medir (Mandal, 2019).
- Transductor: elemento que traduce los cambios en el sensor biológico como algo medible (Mandal, 2019).

Existen 3 tipos de medición a través de biosensores:

- Sensor de afinidad: el analito se une a un elemento biológico (Mandal, 2019).
- Sensor metabólico: el analito y el elemento biológico se unen y generan un cambio químico por acción del metabolismo en el que contribuyen ambas partes. (Mandal, 2019).
- Sensor catalítico: el elemento biológico convierte el analito en un sustrato auxiliar pero no lo cambia químicamente, es decir, el analito causa reacciones en el elemento biológico sin que este cambie sus características originales (Mandal, 2019).

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

- Inmunosensor piezoeléctrico

Los inmunosensores piezoeléctricos son biosensores en los cuales se fijan anticuerpos (elemento biológico) sobre un cristal de cuarzo (transductor); en esta aplicación, el cristal de cuarzo actúa como micro balanza (QCM) cambiando su frecuencia de resonancia dependiendo de los cambios de masa que sucedan en la reacción inmunológica con los anticuerpos sobre el cristal (Ferizzola et al., 2011).

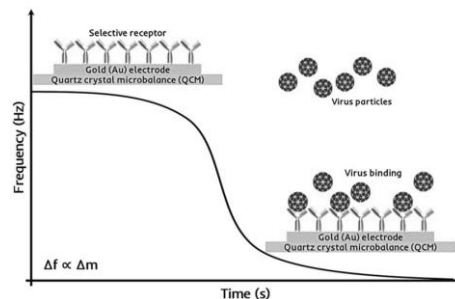


Figura 1: Ilustración de cambio de frecuencia debido al cambio de masa en un inmunosensor basado en QCM

- Efecto de la temperatura sobre cristal

Un inmunosensor piezoeléctrico puede sufrir de muchos ruidos que afectan la medición, entre estos se encuentra las variaciones en la temperatura también conocida como el ruido de Johnson (Raddani, 2015).

El ruido de Johnson se define como la agitación en las cargas (en este caso electrones) la cual puede causar cambios en la transmisión del voltaje y puede ocasionar ruidos en la frecuencia que se espera teóricamente generando imprecisiones en los resultados (Johnson, 1928).

Otras formas por las cuales se puede expresar el ruido las cuales pueden ser por contacto con los líquidos en la superficie del sensor que cambian su propiedades con la temperatura, los cambios por deformación en distintas partes del sensor por el cambio en la temperatura y por último el cambio en las propiedades intrínsecas del sensor ocasionadas por cambios en la temperatura (Edvardsson, 2018).

- Modelo de inmunosensor piezoeléctrico

Para el inmunosensor que se usará, es necesario establecer un modelo eléctrico que representará su comportamiento desde sus condiciones ideales hasta condiciones reales ambientales (temperatura, humedad, propiedades del elemento a medir, etc.) para predecir su reacción frente a este entorno cambiante.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Para representar el sensor se necesita de un modelado basado en la respuesta de la impedancia de este sensor y el modelado que mejor se ajustó al comportamiento del sensor según Montagut (2011) es un derivado del modelo electromecánico continuo donde la impedancia se representa como:

$$Z = \frac{1}{j\omega C_0} \left[1 - \frac{K_q^2}{\alpha_q} \frac{2 \tan\left(\frac{\alpha_q}{2}\right) - j\zeta}{1 - j\zeta \cot(\alpha_q)} \right] \quad (1.1)$$

Luego, simplificando, puede representarse esta ecuación como un circuito que contiene una capacitancia estática (C_0) en paralelo con una impedancia dinámica (Z_m) asociada con la vibración electro-mecánica del sensor (Montagut, 2011), se expresa como:

$$Z = \frac{1}{j\omega C_0} // Z_m = \frac{Z_m}{1 + j\omega C_0 Z_m} \quad (1.2)$$

Luego, se puede obtener que $Z_m = Z_m^q + Z_m^L$ donde la Z_m^q es la impedancia dinámica del cristal sin carga y la Z_m^L es la contribución de la carga en la impedancia dinámica (Montagut, 2011).

Se puede decir que la impedancia dinámica de un sensor sin carga Z_m^q para una frecuencia cercana a la frecuencia de resonancia mecánica a un circuito en serie de elemento concentrados (LEM, Lumped Element Model) (Montagut, 2011), expresándose de la siguiente manera:

$$Z_m^q = \frac{1}{j\omega C_0} \left[\frac{\alpha_q / K_q^2}{2 \tan\left(\frac{\alpha_q}{2}\right)} - 1 \right] = R_m^q + j\omega L_m^q + 1/j\omega C_m^q \quad (1.3)$$

En la última parte de la expresión anterior se describe la impedancia de un circuito equivalente de Butterworth-Van Dyke (BVD) de un resonador sin carga (Montagut, 2011) como se ve en la siguiente figura:

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

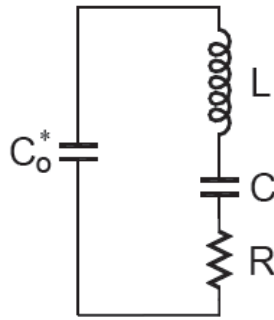


Figura 2: Modelo BVD

En el circuito $C_o^* = C_o + C_p$, donde C_p es la capacitancia externa al cristal de cuarzo, debido al encapsulado, los cables, etc.

Puede demostrarse también que Z_m^L puede aproximarse a un circuito serie compuesto de un elemento resistivo R_m^L , otro de carácter inductivo L_m^L y un último elemento capacitivo C_m^L (Montagut, 2011) generando un modelo equivalente al de la siguiente figura:

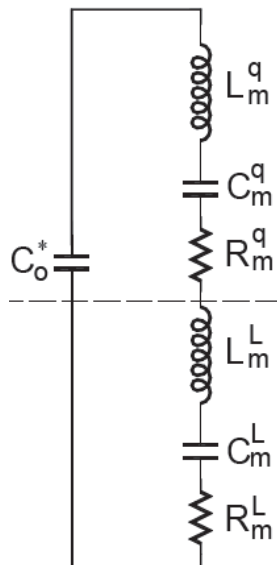


Figura 3: Modelo BVD extendido

Seguido de esto, se formuló una ecuación a partir de este modelo que representará la frecuencia de resonancia de la serie dinámica (MSRF) ya que esta es la que cambiará con respecto a los cambios de masa y esta expresión quedó de la forma:

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

$$f_s = \frac{1}{2\pi\sqrt{(L_m^q + L_m^L) \cdot C_m^q}} \quad (1.4)$$

Debido a que la frecuencia anterior se realizó en un ambiente ideal, no se tomó en cuenta que el dispositivo se medirá en un medio meramente líquido, por esta razón se midió el cambio en la frecuencia que se generaba por la presencia del fluido y se expresó de la siguiente manera:

$$\Delta f_s = -\frac{L_m^L}{2L_m^q} f_s \quad (1.5)$$

Se encontró que los cambios en la MSRF solo eran debidos a cambios en la L_m^L diciendo por lo tanto que las propiedades físicas del fluido se verían influenciadas en este parámetro.

Por consiguiente los cambios que se evidencien en el cristal debido a la temperatura deben ser tomados como un tipo de inductancia siendo esta nuestra variable principal durante el proyecto.

2. METODOLOGÍA

Esta investigación será realizada en tres etapas las cuales, basadas en los objetivos específicos, lograrán la modelación, experimentación y comprobación del modelo propuesto.

1. Establecer las características de los modelos existentes para la obtención de funciones matemáticas y bases aplicables.

- a. **Entendimiento del modelo:** A partir de modificaciones a los modelos ya existentes se obtendrán funciones matemáticas para la modelación numérica.
- b. **Traducción del modelo:** usando el lenguaje Python se planteará un algoritmos que genere los resultados matemáticos del modelo propuesto.
- c. **Comprobación del programa:** se compararán los resultados del modelo programado en Python con respecto al modelo anterior para verificar que haya sido bien programado e implementado.
- d. **Obtención de datos:** se introducirán distintos valores de temperatura en el modelo para la adquisición de las distintas frecuencias generadas por este.
- e. **Almacenamiento de datos del modelo:** se almacenarán las distintas frecuencias obtenidas en el modelo con sus respectivas temperaturas para su posterior análisis.

2. Evaluar los resultados del modelo con respecto a los cambios experimentales debido a las variaciones de temperatura para la comprobación de su exactitud y precisión.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

- a. **Creación de ambiente para el sensor:** se utilizará un sistema de creación de ambientes térmicos, existente en Universidad EIA, con el cual se puede crear varios ambientes a distintas temperaturas generando las condiciones para analizar en el sensor y generar datos comparables con el modelo simulado.
- b. **Adquisición de datos del sensor:** se utilizará un medidor de fase y magnitud, construido en la Universidad EIA, al cual se conectará el sensor y medirá estas propiedades del sensor en tiempo real. Con esto se espera analizar la variación con respecto a los distintos perfiles de temperatura.
- c. **Almacenamiento de los datos obtenidos:** se almacenarán los distintos valores de las variables con sus respectivas temperaturas para su posterior análisis.
- d. **Generación de gráficos:** utilizando los datos obtenidos se generarán las distintas gráficas de ambas listas de los resultados a evaluar.
- e. **Observación de diferencias:** se analizarán las diferencias existentes entre ambas gráficas y sus respectivas temperaturas mediante la observación de los puntos notorios y la localización de estos.
- f. **Análisis de diferencias:** a partir de las diferencias encontradas en el paso anterior se analizarán cual(es) es(son) la(s) causa(s) de estas diferencias.

3. Ajustar las características del modelo con base en los resultados reales obtenidos para lograr un acercamiento a la variación real.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

- a. **Localización de ajuste:** a partir de las diferencias encontradas en el paso anterior se observarán dónde se encuentran las causas de estas diferencias en el modelo.
- b. **Modificación de modelo:** después de localizar las causas se modificará el modelo para guiarlo al resultado más semejante a la variación real.
- c. **Comprobación del modelo:** se realizará todo el proceso con el nuevo modelo ajustado, las veces que sean necesarias para verificar que de correctamente.

3. PRESENTACIÓN Y DISCUSIÓN DE RESULTADOS

3.1 ENTENDIMIENTO DEL MODELO

En un primer lugar, para entender el modelo que se plantea, se adquirió bibliográficamente los valores de las características básicas el sensor a continuación:

Parámetros del cuarzo	Valor	Descripción
ϵ_{22}	$3,982 \times 10^{-11} \text{A}^2 \cdot \text{s}^4 \cdot \text{Kg}^{-1} \cdot \text{m}^{-3}$	Permitividad
η_q	$9,27 \times 10^{-3} \text{ Pa} \cdot \text{s}$	Viscosidad efectiva
C_{66}	$2,947 \times 10^{10} \text{ N} \cdot \text{m}^{-2}$	Módulo de cizalla efectivo.
e_{26}	$9,657 \times 10^{-2} \text{ A} \cdot \text{s} \cdot \text{m}^{-2}$	Constante de esfuerzo piezoeléctrico
ρ_q	$2651 \text{ Kg} \cdot \text{m}^{-3}$	Densidad
A_s	$2,92 \times 10^{-5} \text{ m}^2$	Área efectiva de la superficie del electrodo
h_q	$166,18 \times 10^{-6} \text{ m}$	Espesor

Tabla 1: Propiedades del cuarzo en corte AT

Luego, según la bibliografía, la función de impedancia del cristal con carga en contacto con un fluido newtoniano y sin el efecto de la temperatura sería de la siguiente manera (Montagut, 2011):

$$Z_m = R_m + jX_m = R_q + R_L + j \left(\omega(L_q + L_c + L_L) - \frac{1}{\omega C_q} \right) \quad (3.1)$$

Donde:

R_q : Resistencia dinámica sin carga.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

R_L : Resistencia dinámica del agua.

L_q : Inductancia dinámica sin carga.

L_c : Inductancia dinámica del recubrimiento.

L_L : Inductancia dinámica del agua.

C_q : Capacidad dinámica.

Por consiguiente, teniendo los componentes característicos del modelo de impedancia del cristal, se investigó en la bibliografía las distintas fórmulas para calcularlos y los valores de las variables necesarias en estas fórmulas. Para la obtención de las variables del modelo, se utilizaron las siguientes formulas propuestas por Montagut (2011):

$$\delta_L = \sqrt{\frac{2\eta_L}{\omega\rho_L}} \quad (3.2)$$

$$\omega = \frac{\pi v_q}{h_q} \quad (3.3)$$

$$v_q = \sqrt{\frac{\bar{c}_{66}}{\rho_q}} \quad (3.4)$$

$$h_q = \frac{v_q}{2f_s} \quad (3.5)$$

$$L_q = \frac{\rho_q h_q^3}{8A_s e_{26}^2} \quad (3.6)$$

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

$$K_0^2 = \frac{e_{26}^2}{\varepsilon_{22} \bar{c}_{66}} \quad (3.7)$$

$$Z_{cq} = \sqrt{\rho_q \bar{c}_{66}} = \rho_q \sqrt{\frac{\bar{c}_{66}}{\rho_q}} = \rho_q v_q \quad (3.8)$$

$$\alpha = h_q \omega \sqrt{\frac{\rho_q}{\bar{c}_{66}}} \quad (3.9)$$

$$K_t = \frac{\alpha}{4K_0^2 \omega C_0 Z_{cq}} = \frac{h_q^2}{4e_{26}^2 A_s} \quad (3.10)$$

Donde:

h_q : Grosor del cristal.

A_s : Área del electrodo.

η_q : Viscosidad efectiva del cuarzo.

v_q : Velocidad de propagación de la onda.

ρ_q : Densidad del cuarzo.

e_{26} : Constante piezoeléctrica.

ε_{22} : Permitividad del cuarzo.

ρ_L : Densidad del líquido en contacto con el cristal.

η_L : Viscosidad del líquido en contacto con el cristal.

δ_L : Profundidad de penetración de la onda acústica en el líquido.

f_s : Frecuencia de resonancia.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

ω : Frecuencia angular.

α : Fase de la onda acústica a través del cuarzo.

Z_{cq} : Impedancia característica del cuarzo.

C_0 : Capacidad estática del cristal.

K_0^2 : Factor de acoplo electromecánico.

K_t : Constante electroacústica.

Tomando en cuenta ciertas características del dispositivo, el recubrimiento es el mismo fluido y que el fluido supuesto para el modelo es agua (Montagut, 2011) se definen algunas variables:

$$f_s = 10 \text{ MHz}$$

$$\rho_L = \rho_c = 1000 \frac{\text{kg}}{\text{m}^3}$$

$$\eta_L = 1 \times 10^{-3} \text{ N.s/m}^2$$

$$h_c = 1 \times 10^{-7} \text{ m}$$

Donde:

ρ_c : Densidad del recubrimiento.

h_c : Grosor del recubrimiento.

Luego a partir de la obtención de estas variables, se prosiguió con el cálculo de los componentes faltantes del modelo con las siguientes formulas propuestas por Montagut (2011):

$$m_c = \rho_c h_c \tag{3.11}$$

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

$$m_L = \frac{\rho_L \delta_L}{2} \quad (3.12)$$

$$L_c = K_t m_c \quad (3.13)$$

$$R_L = \omega K_t m_L \quad (3.14)$$

$$L_L = K_t m_L \quad (3.15)$$

Donde:

m_c : Densidad superficial de masa del recubrimiento.

m_L : Densidad superficial de masa equivalente del líquido en contacto con el cristal, asociada al movimiento oscilatorio de la superficie del sensor.

Posterior a esto se procedió a definir la ecuación para hallar la frecuencia de resonancia en presencia de un fluido (f_{sL}) con las variables encontradas, para esto se procedió con el mismo principio de la ecuación 1.4 propuesto por Montagut (2011), estableciendo la siguiente expresión:

$$f_{sL} = \frac{1}{2\pi\sqrt{(L_q + L_L + L_c) * C_q}} \quad (3.16)$$

A partir de esta función se planteó como sería el modelo de variación de temperatura que, según la teoría expuesta anteriormente, expresaría la impedancia de la siguiente manera:

$$Z_m = R_m + jX_m = R_q + R_L + j\left(\omega(L_q + L_c + L_L + L_T) - \frac{1}{\omega C_q}\right) \quad (3.17)$$

Siendo la L_T la inductancia dinámica generada por los cambios en la temperatura.

3.2 TRADUCCIÓN Y COMPROBACIÓN DEL MODELO

3.2.1 Obtención de la MSRF en presencia de un fluido

Basado en las fórmulas y funciones anteriores, se procedió a comprobar el valor de f_{sL} que podría generar el modelo, por lo tanto, mediante un algoritmo realizado con lenguaje Python, se tradujeron todas las constantes y las funciones a calcular para llegar a este valor. A partir de este algoritmo se obtuvo lo siguiente:

$$f_{sL} = 9995721.5 \text{ Hz}$$

Este valor fue el mismo encontrado por Montagut (2011) comprobando así que los valores utilizados para las fórmulas anteriores a esta son correctos. Este valor es la MSRF afectada por la presencia de un recubrimiento de un fluido haciendo que sea distinta a la normal.

3.2.2 Modelo de inductancia por cambios en la temperatura

Basados en el resultado anterior, se varió la f_{sL} para encontrar valores de L_T ; esta primera prueba se realizó basándose en que la temperatura variará principalmente la frecuencia de resonancia, por lo tanto si variamos esta desde un principio, podremos evidenciar los valores posibles de L_T para variaciones tanto positivas como negativas.

Siguiendo esto se varió la frecuencia de resonancia en ± 10 y ± 100 Hz en pasos de 1 y 10 Hz respectivamente; luego tomando en cuenta la metodología de modelo puramente resistivo propuesto por Montagut (2011) se realizaron las debidas operaciones a la ecuación 3.17 para

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

la obtención de una ecuación similar a la ecuación 3.16. A partir de esta última ecuación se despejó L_T para obtenerla en función de la f_{SL} , quedando de la siguiente manera:

$$L_T = \frac{1}{(2\pi f_{SL})^2 \cdot C_q} - (L_q + L_L + L_C) \quad (3.18)$$

Con esta ecuación se utilizaron las distintas variaciones de la f_{SL} para encontrar los valores de L_T ; este primer paso, mostrado en la Figura 4, se hace con el objetivo de adquirir algunos valores de L_T y comprobar la existencia de estos valores dependiendo de la variación.

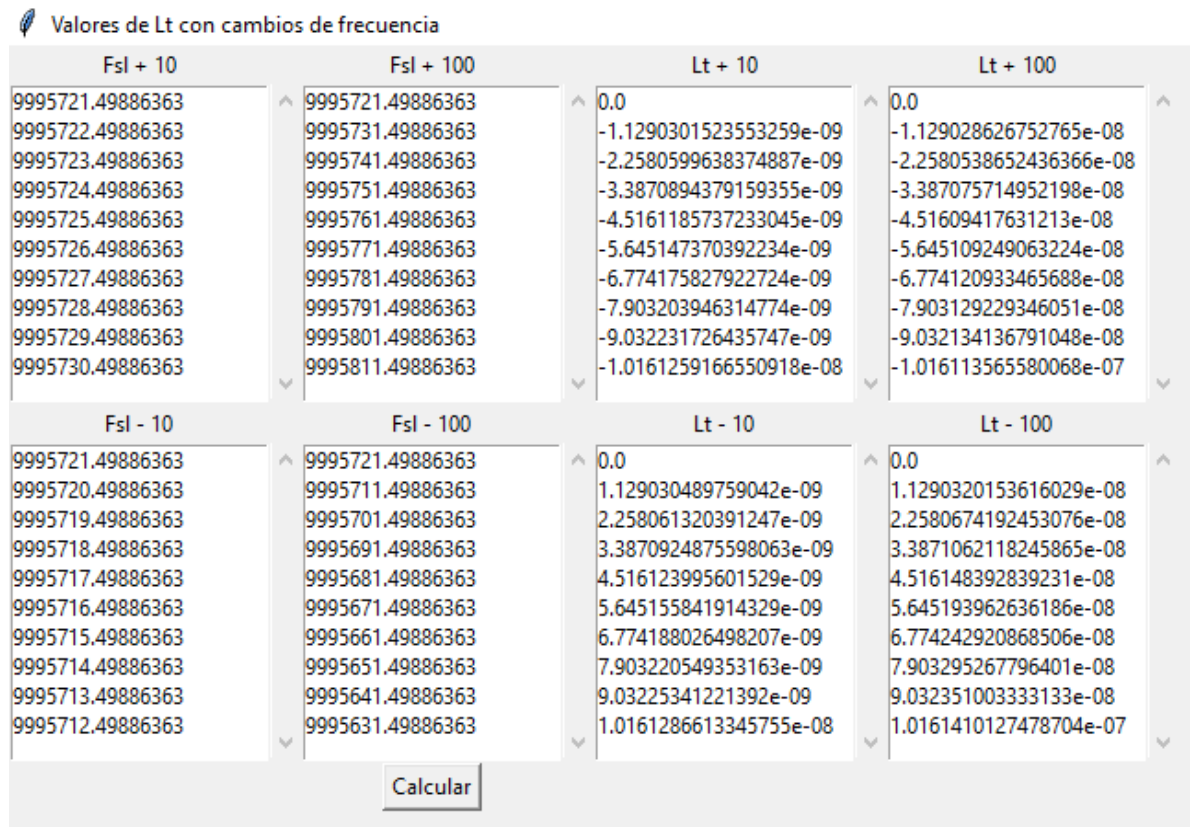


Figura 4: Interfaz Gráfica (GUI) de modelo de una QCM variando su frecuencia en 10 y 100 Hz, hallando valores de L_t

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Se puede observar en los resultados de la Figura 4 que al aplicar la f_{SL} sin cambios, el valor de L_T es nulo, como se tiene propuesto en el modelo; por otra parte, se puede ver que el valor de L_T solo es positivo y real cuando los cambios en la f_{SL} son negativos.

3.2.3 Modelo de frecuencia, magnitud y fase con respecto al cambio en la temperatura

A partir de los resultados anteriores y tomando la temperatura de referencia de 25°C, se hizo el supuesto de que la f_{SL} variaba 5 Hz por cada 0.1°C; por consiguiente se hizo un barrido de frecuencias desde los 25°C hasta los 50°C en pasos de 0.1°C, es decir, desde f_{SL} hasta $f_{SL} - 1250$ Hz en pasos de 5 Hz, obteniendo la magnitud y la fase de cada una de estas variaciones con las siguientes formulas fasoriales:

$$\varphi_Z = \tan^{-1} \left(\frac{\omega(L_q + L_c + L_L + L_T) - \frac{1}{\omega C_q}}{R_q + R_L} \right) \quad (3.19)$$

$$|Z| = \sqrt{(R_q + R_L)^2 + \left(\omega(L_q + L_c + L_L + L_T) - \frac{1}{\omega C_q} \right)^2} \quad (3.20)$$

Donde:

φ_Z : Fase de la impedancia dinámica.

$|Z|$: Magnitud de la impedancia dinámica.

ω : Frecuencia angular = $2 * \pi * f_{SL}$.

Esta simulación se hizo con el objetivo de observar los cambios en las variables de Frecuencia, magnitud y fase; por cambio en la temperatura ambiente del sensor; por otro

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

lado, se tomó que la frecuencia solo podía variar negativamente, debido a que si variaba positivamente, no podría analizarse de manera real los valores dados en L_T .

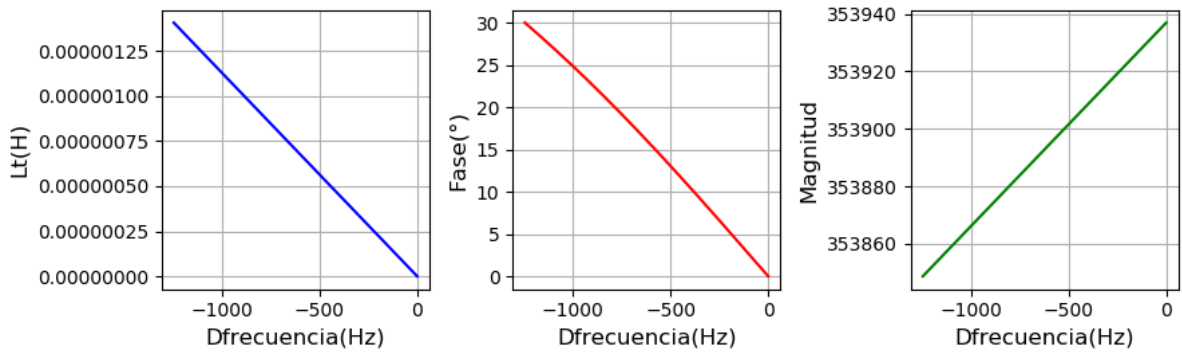
Siguiente a esto, se creó una interfaz gráfica para la visualización de los datos de manera sencilla y se graficó cada uno de ellos con respecto al cambio de frecuencia como se ve en la Figura 5a y 5b.

Fo -	Lt -	Q(Z)
9995721.49886363	0.0	5.279092265192746e-07
9995716.49886363	5.645155841914329e-09	0.002323310937739781
9995711.49886363	1.1290320153616029e-08	0.004646072381174978
9995706.49886363	1.6935492937707186e-08	0.006968787178348907
9995701.49886363	2.2580674192453076e-08	0.009291430268810184
9995696.49886363	2.8225863919588423e-08	0.011613976597893275
9995691.49886363	3.3871062118245865e-08	0.013936401116814375
9995686.49886363	3.951626878755804e-08	0.016258678784291738
9995681.49886363	4.516148392839231e-08	0.01858078456892719
9995676.49886363	5.08067075424834e-08	0.020902693451585748
9995671.49886363	5.645193962636186e-08	0.023224380422818148
DFo	Z	
0.0	353936.9764789098	
-5.0	353936.62178334035	
-10.0	353936.2670872387	
-15.0	353935.9123906048	
-20.0	353935.5576934386	
-25.0	353935.2029957401	
-30.0	353934.8482975093	
-35.0	353934.4935987463	
-40.0	353934.13889945095	
-45.0	353933.7841996233	
-50.0	353933.42949976347	

Calcular

(a)

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.



(b)

Figura 5: Comportamiento de L_T , fase y magnitud con respecto al cambio de f_{sL} : a) Listas de datos numéricos. b) Gráficas de cada uno vs el cambio en frecuencia

Se puede observar que el valor de L_T solo existe cuando se da una variación en la f_{sL} ; por otro lado también se observa como se ven afectadas la fase y la magnitud al momento en el que se varía la f_{sL} de manera positiva y negativa respectivamente. En todas las gráficas se puede observar también la linealidad que presentan las variaciones sugiriendo un gran potencial de predicción en el modelo.

3.2.4 Modelo con variación de temperatura de una QCM con carga en presencia de un fluido

En el capítulo anterior se determinó un modelo con el cual se encuentran los distintos valores que podría tomar la inductancia L_T ; en el presente capítulo se realizará un análisis al comportamiento de la QCM con una carga de ejemplo en presencia de un fluido viendo como este es afectado por los distintos valores que toma la L_T , esto con el objetivo de verificar la funcionalidad del modelo con un ejemplo real.

En la Figura 6, se observa un ejemplo de variaciones en frecuencia de la QCM con respecto a un intervalo de tiempo, en el cual, la QCM recibe la carga y logra estabilizarse. Con este

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

ejemplo se probó el modelo para observar si el comportamiento del cristal cambiaba con cada valor calculado de L_T .

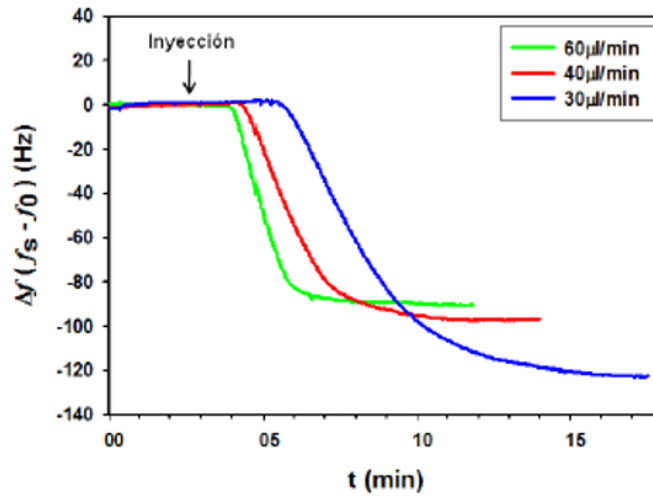


Figura 6: Comportamiento de frecuencia con respecto al tiempo de una QCM cuando se inyecta una carga.

Para la toma de valores de frecuencia se tomó la curva azul de la figura 6 y se tomaron valores aproximados de frecuencia durante cada minuto tomando un total de 17 valores aproximados desde el minuto 0 como se ve en la Tabla 2.

Tiempo (t) en minutos	Variación en frecuencia (Δf) en Hz
0	0
1	0
2	0
3	0
4	0
5	0
6	-9
7	-37
8	-61
9	-82

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

10	-99.5
11	-109
12	-113.5
13	-117
14	-119.5
15	-121
16	-121

Tabla 2: Variación de la frecuencia con respecto a una carga en función del tiempo.

A partir de estas variaciones de frecuencia se utilizó el modelo de impedancia de QCM en presencia de un fluido con carga sin cambios por la temperatura descrito en la ecuación 3.21.

$$Z_m = R_m + jX_m = R_q + R_L + j \left(\omega(L_q + L_c + L_L + L_m) - \frac{1}{\omega C_q} \right) \quad (3.21)$$

Donde:

L_m : Inductancia dinámica de la carga.

Con base en este modelo se aplicó el método de modelo puramente resistivo obteniendo la ecuación 3.22 con la cual se podrá obtener la inductancia generada por la carga en función del tiempo.

$$L_m = \frac{1}{(2\pi f_{SL})^2 \cdot C_q} - (L_q + L_L + L_c) \quad (3.22)$$

A partir de estos valores de L_m obtenidos, se calculó el comportamiento de la fase con respecto al tiempo como se puede observar en la Figura 7.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

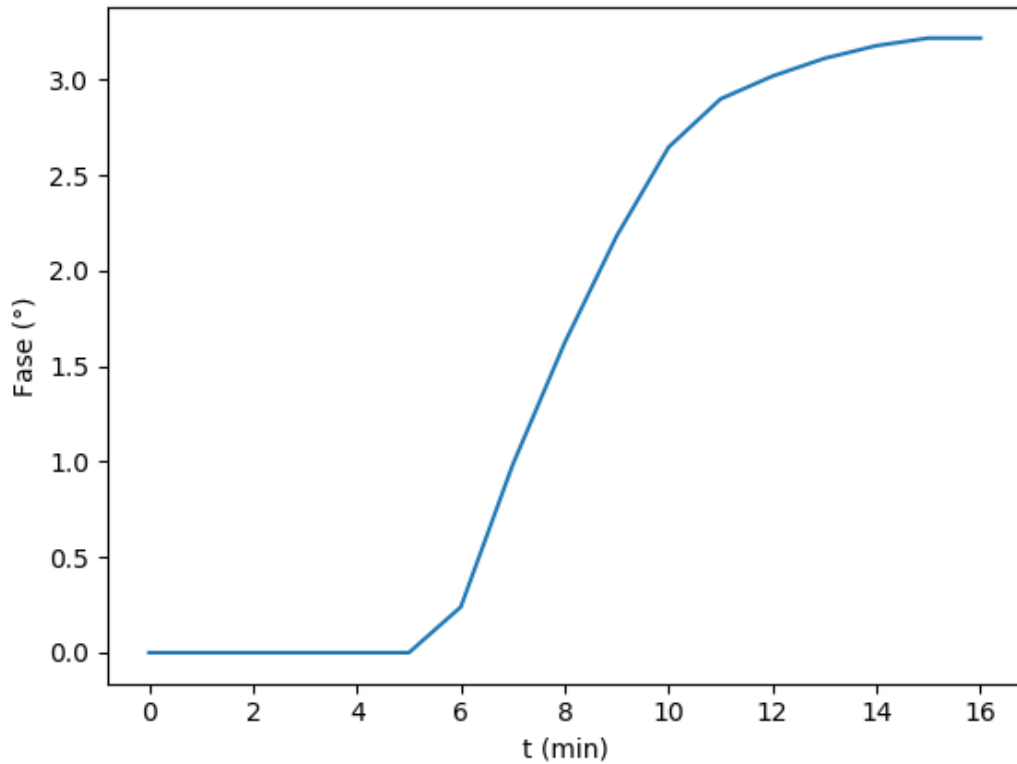


Figura 7: Variación de la fase en función del tiempo en presencia de una carga

Después de este procedimiento se obtuvieron valores de L_T a partir de la variación de la frecuencia entre ± 10 Hz en pasos de 1 Hz aplicando la ecuación 3.18, todo esto con el objetivo de modelar el sensor en presencia de una carga siendo afectado por la variación de temperatura como se describe en la ecuación 3.23.

$$Z_m = R_m + jX_m = R_q + R_L + j \left(\omega(L_q + L_c + L_L + L_m + L_T) - \frac{1}{\omega C_q} \right) \quad (3.23)$$

Luego con cada valor obtenido de L_m y los valores obtenidos de L_T se calculó la influencia de la temperatura en el comportamiento de la fase en función tiempo. Para esto se mantuvo

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

constante el comportamiento de la L_m y se calculó con cada valor de la L_T las posibles fases resultantes durante ese intervalo de tiempo. En la Figura 8 se puede evidenciar en Negro la fase sin variación causada por la L_T y alrededor de esta sus posibilidades causadas por variaciones en la temperatura.

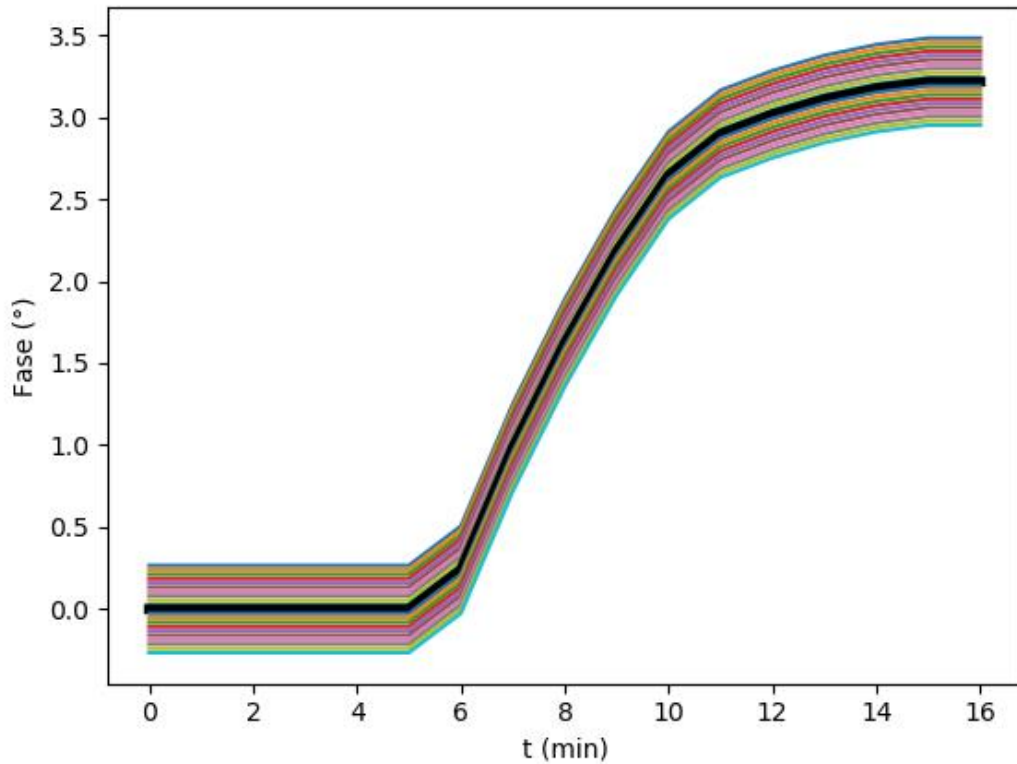


Figura 8: Fases de QCM en presencia de una carga con distintas variaciones de temperatura

Como se observa en la Figura 8 el cambio en la fase es meramente vertical indicando la predictibilidad de los posibles resultados ocasionados por variaciones en la temperatura.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Para ver los valores precisos de cada variación se realizó una interfaz, como se ve en la Figura 9, con la cual se podían ver los datos para tener una forma más detallada de compararlos posteriormente.

Fase fs-10	Fase fs-9	Fase fs-8	Fase fs-7	Fase fs-6	Fase fs
0.26617009241396017	0.2395533746364237	0.2129365614823969	0.18631966437410347	0.15970269486508024	
0.26617009241396017	0.2395533746364237	0.2129365614823969	0.18631966437410347	0.15970269486508024	
0.26617009241396017	0.2395533746364237	0.2129365614823969	0.18631966437410347	0.15970269486508024	
0.26617009241396017	0.2395533746364237	0.2129365614823969	0.18631966437410347	0.15970269486508024	
0.26617009241396017	0.2395533746364237	0.2129365614823969	0.18631966437410347	0.15970269486508024	
0.50571364476634	0.47909837445645653	0.4524829053525244	0.42586724893892736	0.399251416742315	
1.2508136799891991	1.224208858648286	1.1976035177099926	1.170997667983155	1.144391321209618	
1.8891487872125685	1.8625600717557766	1.8359705617677746	1.809380266666367	1.7827892038713802	
2.447316615437057	2.420747386729696	2.3941771242672703	2.3676058393527297	2.3410335434225256	
2.81140866710277	2.80550706408500	2.800000140200105	2.822457700602077	2.805004677562067	
Fase fs-5	Fase fs-4	Fase fs-3	Fase fs-2	Fase fs-1	Fase fs
0.13308566439980493	0.10646858446661187	0.0798514665539632	0.05323432221596822	0.026617162832008587	0.0
0.13308566439980493	0.10646858446661187	0.0798514665539632	0.05323432221596822	0.026617162832008587	0.0
0.13308566439980493	0.10646858446661187	0.0798514665539632	0.05323432221596822	0.026617162832008587	0.0
0.13308566439980493	0.10646858446661187	0.0798514665539632	0.05323432221596822	0.026617162832008587	0.0
0.13308566439980493	0.10646858446661187	0.0798514665539632	0.05323432221596822	0.026617162832008587	0.0
0.37263542023191315	0.34601927084402484	0.3194029800885606	0.2927865595175789	0.2661700205086831	0.2395533746364237
1.1177844887630846	1.0911771821277167	1.064569412767062	1.0379611922113925	1.0113525318174104	0.9847434431613783
1.7561973787828051	1.7296048048244224	1.7030114934219471	1.6764174560684904	1.6498227040843954	1.6232272490101587
2.3144602478066565	2.2878859638813167	2.261310703025251	2.2347344766851687	2.2081572961357407	2.181579172872263
2.770250653071067	2.757305647060665	2.736300623040304	2.706091735576153	2.671130110266440	2.646563171270016
Fase fs+1	Fase fs+2	Fase fs+3	Fase fs+4	Fase fs+5	
-0.02661715487877855	-0.05323429022827251	-0.07985139466912351	-0.10646845664725715	-0.1330854646742576	
-0.02661715487877855	-0.05323429022827251	-0.07985139466912351	-0.10646845664725715	-0.1330854646742576	
-0.02661715487877855	-0.05323429022827251	-0.07985139466912351	-0.10646845664725715	-0.1330854646742576	
-0.02661715487877855	-0.05323429022827251	-0.07985139466912351	-0.10646845664725715	-0.1330854646742576	
-0.02661715487877855	-0.05323429022827251	-0.07985139466912351	-0.10646845664725715	-0.1330854646742576	
0.21293663334453408	0.18631980818624042	0.1597029105620414	0.13308595196002027	0.10646894393396628	
0.9581339376022413	0.9315240267402968	0.9049137219366136	0.8783030346624966	0.8516919764557738	
1.5966311021916126	1.570034275129199	1.543436779215955	1.5168386258903106	1.490239826592395	
2.1550001181961083	2.1284201435637744	2.10183926032504	2.075257479875665	2.0486748136137205	
2.520001001202165	2.501401060708276	2.4800076006701777	2.45011630164300	2.42177700100713	
Fase fs+6	Fase fs+7	Fase fs+8	Fase fs+9	Fase fs+10	
-0.15970240728369814	-0.18631927294377332	-0.21293605023212359	-0.23955272759552254	-0.26616929354655067	
-0.15970240728369814	-0.18631927294377332	-0.21293605023212359	-0.23955272759552254	-0.26616929354655067	
-0.15970240728369814	-0.18631927294377332	-0.21293605023212359	-0.23955272759552254	-0.26616929354655067	
-0.15970240728369814	-0.18631927294377332	-0.21293605023212359	-0.23955272759552254	-0.26616929354655067	
-0.15970240728369814	-0.18631927294377332	-0.21293605023212359	-0.23955272759552254	-0.26616929354655067	
0.07985189795039938	0.053234825475937636	0.02661773799911718	6.470740615811579e-07	-0.026616435876193736	
0.825080558767857	0.7984687930510871	0.7718566908023959	0.7452442634977414	0.7186315226139229	
1.463640392807677	1.4370403359359263	1.4104396674440278	1.3838383987786125	1.3572365413878635	
2.0220912729832086	1.9955068693431015	1.968921614120063	1.9423355187211322	1.9157485945555166	

Figura 9: Interfaz de usuario para visualización de posibles fases debido a la variación de temperatura

Aquí se ha comprobado el funcionamiento del modelo para calcular las posibles consecuencias de la variación de la temperatura sobre las mediciones que realiza la QCM en

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

este ejemplo predefinido; posteriormente se ajustará de tal manera que la QCM y el modelo esté en condiciones similares para así compararlos y según estos resultados ajustar el modelo.

3.3 OBTENCIÓN Y ALMACENAMIENTO DE DATOS DEL MODELO

En las mediciones experimentales que se van a realizar posteriormente, no se obtendrán valores absolutos de fase sino diferencias de fase entre el cristal sin carga y el cristal con carga durante el intervalo definido en la Tabla 2.

Para obtener las diferencias de fase ($\Delta\varphi$ (φ carga – φ sin carga)), se compararon la del modelo descrito en la ecuación 3.17 y el modelo de la ecuación 3.23 durante el intervalo de tiempo definido; en cada comparación ambos modelos tienen una misma L_T . Este proceso se realiza para evidenciar solo el cambio causado por la carga en el cristal y obtener diferencias de fase teóricas para luego ser comparadas en el ámbito experimental.

Para visualizar los datos de cada comparación realizada con su respectiva L_T , se realizó una interfaz de usuario similar a la ilustrada en la Figura 9 en donde se enseñarían los distintos cambios de la fase (en grados) para cada una de las respectivas L_T .

Como se puede evidenciar en la Figura 10, los primeros minutos son nulos debido a que todavía no se evidencia un cambio en la fase debido a la carga, sin embargo después de esos primeros minutos se comienza a evidenciar una diferencia.

Estos datos fueron almacenados para ser comparados posteriormente con los datos experimentales. En caso tal de que las condiciones iniciales experimentales sean distintas a las del modelo, se realizarán los cambios oportunos.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Estos resultados al ser obtenidos por el modelo propuesto serán usados para verificar la fiabilidad de este al momento simular el posible efecto de la variación de temperatura en una QCM.

Fase fs-10	Fase fs-9	Fase fs-8	Fase fs-7	Fase fs-6						
0.0	0.0	0.0	0.0	0.0						
0.0	0.0	0.0	0.0	0.0						
0.0	0.0	0.0	0.0	0.0						
0.0	0.0	0.0	0.0	0.0						
0.0	0.0	0.0	0.0	0.0						
0.23954355235237978	0.23954499982003283	0.2395463438701275	0.2395475845648239	0.23954872187915124						
0.984643587575239	0.9846554842284049	0.9846669562275957	0.9846780036090517	0.9846886263445377						
1.6229786947986082	1.6230066971193529	1.6230340002853778	1.6230606042922635	1.6230865090063						
2.181146523023097	2.1811940120932722	2.181240562784873	2.1812861749786263	2.181330848557445						
2.646020774204312	2.646067218621655	2.646113753204324	2.646160232030942	2.6462067159509154						
Fase fs-5	Fase fs-4	Fase fs-3	Fase fs-2	Fase fs-1	Fase fs					
0.0	0.0	0.0	0.0	0.0	0.0					
0.0	0.0	0.0	0.0	0.0	0.0					
0.0	0.0	0.0	0.0	0.0	0.0					
0.0	0.0	0.0	0.0	0.0	0.0					
0.0	0.0	0.0	0.0	0.0	0.0					
0.23954975583210822	0.23955068637741297	0.23955151353459742	0.2395522373016107	0.2395528576766745	0.239553746364237					
0.9846988243632796	0.9847085976611049	0.9847179462130987	0.9847268699954242	0.9847353689854018	0.9847434431613783					
1.6231117143830003	1.6231362203578106	1.623160026867984	1.6231831338525222	1.6232055412523867	1.6232272490101587					
2.1813745834068516	2.1814173794147047	2.1814592364712877	2.1815001544692003	2.181540133303732	2.181579172872263					
2.646254101807302	2.6463006732074405	2.6463471462075788	2.6463936176181170	2.6464400890282552	2.6464865700383935					
Fase fs+1	Fase fs+2	Fase fs+3	Fase fs+4	Fase fs+5						
0.0	0.0	0.0	0.0	0.0						
0.0	0.0	0.0	0.0	0.0						
0.0	0.0	0.0	0.0	0.0						
0.0	0.0	0.0	0.0	0.0						
0.0	0.0	0.0	0.0	0.0						
0.23955378822331264	0.23955409841451292	0.23955430523116494	0.23955440860727742	0.23955440860822388						
0.9847510924810199	0.9847583169685693	0.9847651166057371	0.9847714913097537	0.9847774411300314						
1.6232482570703912	1.6232685653574714	1.6232881738850784	1.6233070825375677	1.6233252912666527						
2.181617273074887	2.1816544337920467	2.181690654994164	2.1817259365229225	2.181760278287978						
2.646502323014095	2.646548795206743	2.6465952672004074	2.6466417392039430	2.6466882105085004						
Fase fs+6	Fase fs+7	Fase fs+8	Fase fs+9	Fase fs+10						
0.0	0.0	0.0	0.0	0.0						
0.0	0.0	0.0	0.0	0.0						
0.0	0.0	0.0	0.0	0.0						
0.0	0.0	0.0	0.0	0.0						
0.0	0.0	0.0	0.0	0.0						
0.23955430523409751	0.23955409841971095	0.23955378823124077	0.23955337466958412	0.23955285767035694						
0.9847829660515551	0.9847880659948605	0.9847927410345195	0.9847969910932639	0.9848008161604735						
1.6233428000913752	1.6233596088796995	1.6233757176761514	1.623391126374135	1.6234058349344143						
2.1817936802669067	2.1818261422868748	2.1818576643521865	2.181888246316655	2.1819178881020673						

Figura 10: Interfaz de usuario con datos de diferencias de fases.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

3.4 CREACIÓN DE AMBIENTE Y ADQUISICIÓN DE DATOS DEL SENSOR

3.4.1 Sistema de caracterización y adquisición de datos

El sistema de caracterización se compone de tres partes como se ve en la Figura 11: un generador de señales senoidal y cosenoidal, un circuito de medición de fases y un conversor analógico digital conectado a una Raspberry Pi 2 por la cual se transmitirán los datos medidos al computador.

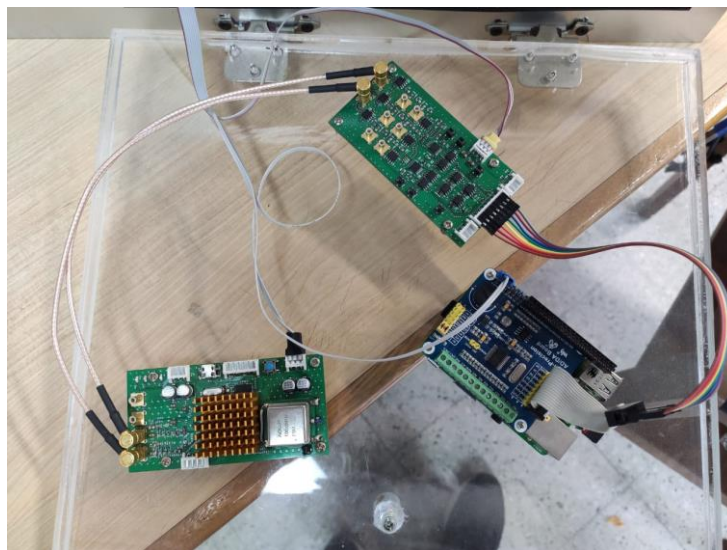


Figura 11: Sistema de caracterización. A la izquierda se encuentra el generador de señales, arriba el medidor de fases y a la derecha el conversor analógico-digital con Raspberry Pi 2

En una primera parte se encuentra el generador de señales. Este es un circuito integrado ya fabricado anteriormente como se puede ver en la Figura 12, con el cual se puede obtener una señal senoidal y cosenoidal con frecuencia modulable por medio de una USB conectada al computador y en el computador con un programa ya realizado anteriormente.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.



Figura 12: Generador de señales

En una segunda parte las señales obtenidas entran en un circuito de medición de fases y magnitudes como se ve en la Figura 13, este ya había sido construido anteriormente. A este circuito se conectan los cristales y una referencia con la cual se medirán y obtendrán los valores de fase y magnitud diferencial entre ambos cristales para el análisis del modelo.

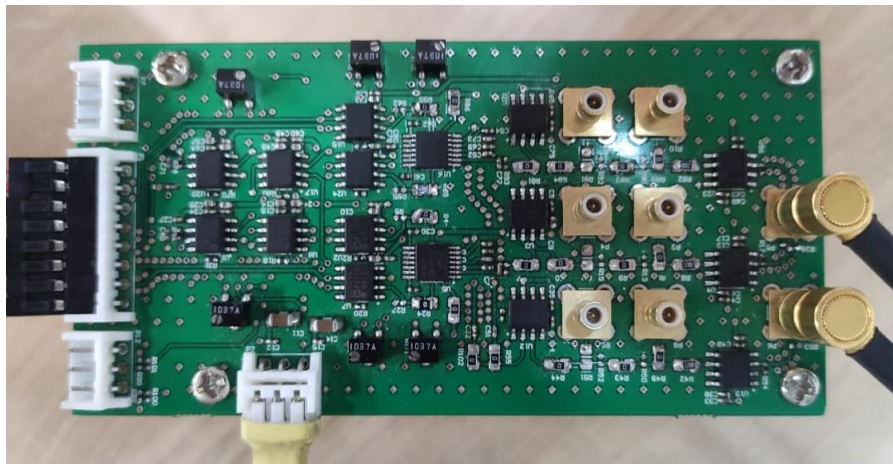


Figura 13: Circuito de medición de Fases y Magnitudes

En la tercera y última parte se encuentra un conversor analógico-digital el cual tomará las señales que se generan eléctricamente por la medición de la fase y la magnitud y las pasará a

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

una Raspberry Pi la cual enviará los datos vía Wifi al computador y al programa que usamos para la medición para así obtener los datos y las gráficas. El dispositivo completo se puede observar en la Figura 14 donde la placa de arriba es el conversor analógico-digital y la placa de abajo es la tarjeta Raspberry Pi.

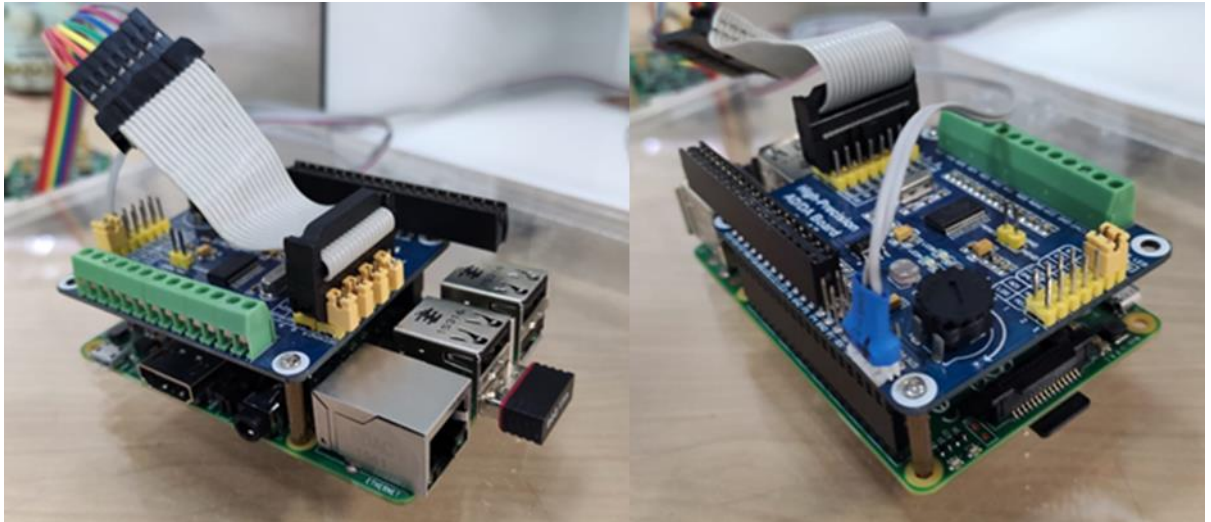


Figura 14: Circuito de adquisición y transmisión de mediciones

3.4.2 Ambiente del sensor

Para la generación de los perfiles de temperatura se utilizó un modulador de termoelectrico que ya estaba construido como se puede observar en la Figura 15. En el compartimiento irá todo el sistema de caracterización y un par de baterías que harán de fuentes eléctricas para alimentar los circuitos de este sistema.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

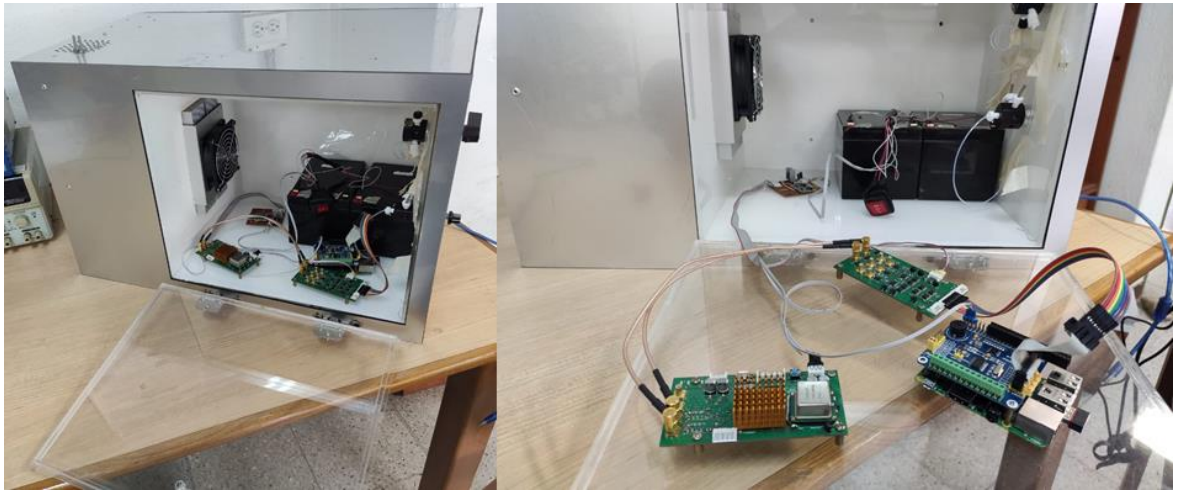


Figura 15: Módulo termoelectrico con sistema de caracterización

Todos los componentes del sistema están dentro del compartimiento como se presenta en la Figura 16 debido a que, se necesita completo aislamiento de la temperatura externa para evitar inestabilidad en el perfil de temperatura.



Figura 16: Módulo termoelectrico cerrado con sistema de caracterización dentro

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

3.4.3 Aplicación de control

Todo el sistema mostrado anteriormente se controla mediante una aplicación, la cual fue diseñada anteriormente, la cual permite observar la variación en tiempo real de la fase (en voltios), la magnitud (en voltios) y la temperatura (en grados centígrados) con respecto al tiempo. En la Figura 17 se puede observar la aplicación donde a la izquierda se observa el segmento donde se realiza la configuración inicial y la derecha el segmento donde se ven las gráficas del sistema en tiempo real.

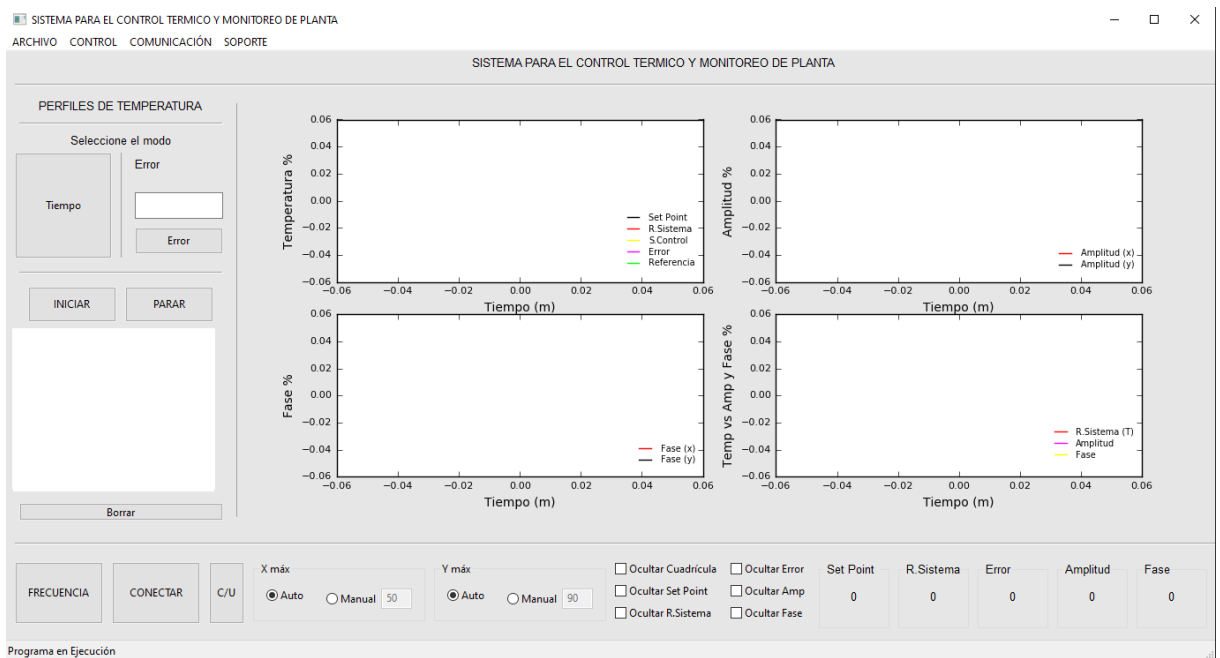


Figura 17: Aplicación para control de sistema de perfilación y medición

En el segmento de la izquierda se puede observar arriba las opciones para perfilación donde la opción “Tiempo” controla una temperatura durante un tiempo determinado con error indeterminado y la opción “Error” controla el sistema hasta generar una temperatura definida dentro del margen de error impuesto en un tiempo indeterminado; por otro lado en el medio

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

se encuentra los botones para iniciar y detener la perfilación de temperatura, y un recuadro donde se muestran los errores durante la configuración inicial al principio y luego se mostrarán los valores numéricos de las variables. Por último en la parte inferior se encuentran tres botones: “Frecuencia” para configurar la frecuencia del oscilador, “Conectar” para comunicar el computador con el modulador de temperatura y por último “C/U” el cual abre otra ventana donde se muestra gráficas de cada una de las variables en tiempo real.

En el segmento derecho superior se encuentran las gráficas de la temperatura, la magnitud, la fase y un último recuadro que muestra todas juntas al mismo tiempo; en la parte inferior se puede ver distintas selecciones para decidir qué ocultar en las gráficas y manipulación de ejes, por otro lado, se observan los valores de temperatura a la que se quiere llegar, temperatura actual, error, magnitud y fase en tiempo real.

3.5 PROCESO DE ADQUISICIÓN Y ALMACENAMIENTO DE DATOS

3.5.1 Puesta a punto del sistema de adquisición

Para la adquisición de la señal, el sistema utiliza dos señales diferenciales que compara un cristal 1 con la referencia y lo envía como fase X, y un cristal 2 con la misma referencia que envía como fase Y. La referencia usada se compone de una resistencia y un capacitor en paralelo.

Se obtuvieron referencias las cuales son un circuito en paralelo entre un resistencia 220Ω y un capacitor de 10 pF que cumplen la función de simular un cristal sumergido en agua con la gran diferencia de que estos no varían significativamente con la temperatura.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

Siendo así, la calibración del sistema se realizó conectando estas referencias a las tres entradas y ajustando las mediciones del sistema a aproximadamente cero, ya que, como se muestra en la Figura 18, cuando se empezaron las mediciones, el sistema se encontraba descalibrado ocasionando que las medidas tomadas tuvieran un offset sobre el valor requerido. La calibración se realizó manipulando los potenciómetros de calibración que tiene integrado el sistema de caracterización.

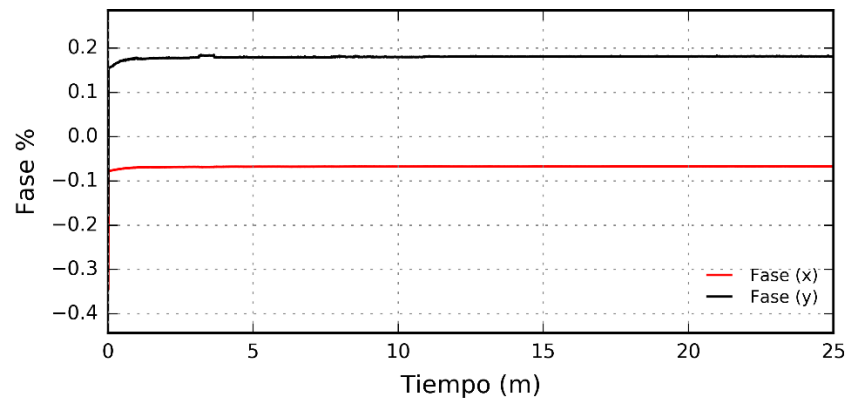


Figura 18: Mediciones de fase con uso de referencia a 27°C y 10MHz en oscilador

Luego de la calibración apropiada, las fases quedaron en cero como se puede evidenciar en la Figura 19.

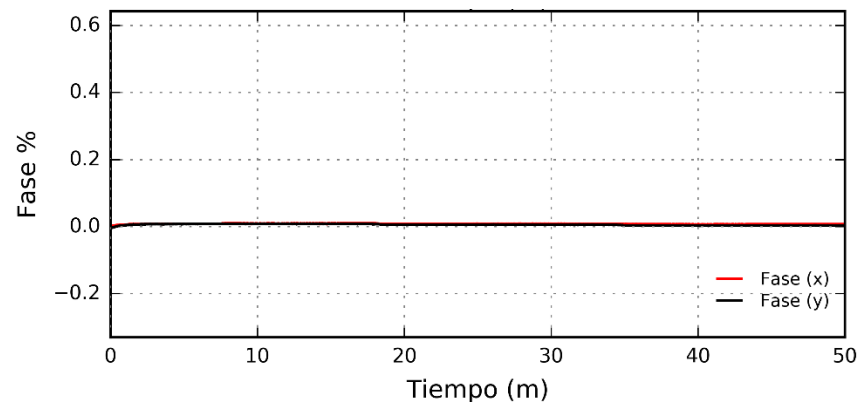


Figura 19: Mediciones de fase calibradas en 27°C

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

A partir de esta calibración se conectaron los cristales de 10MHz en sus celdas, sumergidos en agua Milli-Q, esto ocasiona una variación en la fase haciendo que se desvíe del cero, por lo tanto, se varió la frecuencia del oscilador hasta encontrar la frecuencia en la cual la fase de ambos cristales estuviera más cercana a cero; en este caso la frecuencia encontrada donde la fase fue más cercana a cero fue 9.889766 MHz. Esta frecuencia hallada será la que se usará en el oscilador ya que sería aproximadamente la frecuencia de resonancia de los cristales sumergidos en agua.

3.5.2 Adquisición de los datos

Para el proceso de adquisición primero se debía estabilizar el sistema, por lo tanto, antes de iniciar, se programa un perfil de 27°C durante un tiempo de 1000 segundos permitiendo al sistema estabilizarse aproximadamente en esta temperatura.

Luego de esto, se empezaron a tomar perfiles de temperatura desde 27°C hasta 34°C en pasos de un 1°C en grupos de 3 perfiles, debido a que si se toman más de 3 el programa no guarda los datos. Cada perfil se programó con una duración de 1000 segundos para permitir al sistema la estabilización de esta temperatura y la toma de medidas correcta en cada perfil programado.

3.5.3 Almacenamiento de datos y generación de gráficas

Al final de cada proceso de tres perfiles, el programa genera una gráfica centrada en HTML como se puede ver en la Figura 20 donde se muestran las temperaturas (línea roja) y las

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

mediciones de fase (líneas azul y amarilla) vs el tiempo, por otro lado, el programa también genera un CSV con los valores específicos de cada variable durante el tiempo.

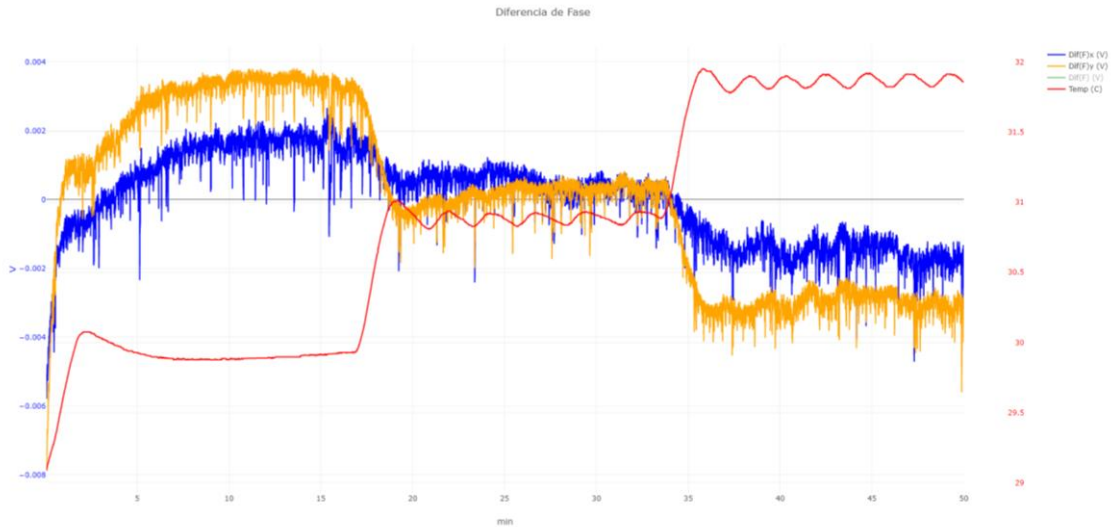


Figura 20: Gráfica de fases y temperaturas vs tiempo en HTML

3.6 OBSERVACIÓN DE DIFERENCIAS

Para la comparación de diferencias entre el modelo y el experimento se eligió primero un rango trabajo, entre todos los perfiles de temperatura, para realizar el análisis. A partir de lo observado en las gráficas se tomaron los perfiles de 28 a 29°C.

Los datos de tomados van desde los 28°C hasta los 29.1°C; estos datos se ordenaron de la menor a la mayor temperatura, luego de esto a razón de que el modelo está basado en cambios de 0.1°C se procesaron los datos de la siguiente manera:

1. Se toman las fases que tengan temperaturas entre $X^{\circ}\text{C}$ hasta $X + 0.09^{\circ}\text{C}$

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

2. Se realiza el promedio de estos datos y se le asignan a la temperatura X°C

Esta aproximación se realizó debido a que el sistema tomaba variaciones de temperatura menores a 0.1°C generando complicaciones a la hora de comparar con el modelo, por lo tanto, usando este método se obtiene un dato más estable para generar una comparación con el modelo propuesto.

A partir de este proceso se obtuvieron los datos en la Tabla 3(a) y se comparó con los datos del modelo obtenidos los cuales se pueden observar en la tabla 3(b); estos datos se refieren al cambio de fase por cada variación en la temperatura de 0.1°C.

Temperatura (°C)	Δ Fase (°)
28-28.1	0.075699717
28.1-28.2	0.023343289
28.2-28.3	0.018504373
28.3-28.4	-0.00689415
28.4-28.5	0.001328276
28.5-28.6	0.009796154
28.6-28.7	0.026977602
28.7-28.8	0.004758922
28.8-28.9	0.026768017
28.9-29	0.057555682
29-29.1	0.012453484

a)

Temperatura (°C)	Δ Fase (°)
28-28.1	0.132
28.1-28.2	0.132
28.2-28.3	0.132
28.3-28.4	0.132
28.4-28.5	0.132
28.5-28.6	0.132
28.6-28.7	0.132
28.7-28.8	0.132
28.8-28.9	0.132
28.9-29	0.132
29-29.1	0.132

b)

Tabla 3: Medidas de temperatura vs fase a) Experimental b) Modelado

A partir de los resultados obtenidos se puede evidenciar que la gran diferencia entre lo modelado y el experimento es que el cambio de fase por cada 0.1°C es menor en el caso real, ya que, en el experimento, el cambio en promedio fue de 0.0227 grados por cada 0.1°C mientras que en el modelo 0.132 grados por cada 0.1°C.

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

3.7 LOCALIZACIÓN DE AJUSTE

A partir de los resultados mencionados anteriormente se puede evidenciar que el cambio en la fase por cada variación de 0.1°C es mucho más pequeño que en el modelo supuesto, por lo tanto, el ajuste del modelo se realizó en la proporción de cambio de 5Hz por cada 0.1°C para acercarnos más a relación experimental que se obtuvo.

3.8 MODIFICACIÓN Y COMPROBACIÓN DEL MODELO

Se realizó un barrido de razones de cambio de 1-4Hz por cada 0.1°C y se obtuvieron los resultados que se pueden observar en la Tabla 4; se puede observar la menor razón de cambio en la frecuencia hay una menor razón de cambio promedio en la fase.

Cambio en frecuencia	Cambio en fase promedio (grados)
4	0.106
3	0.0796
2	0.0531
1	0.0266

Tabla 4: Barrido de razón de cambio de frecuencia por cada 0.1°C en pasos de 1Hz y cambio en fase por cada 0.1°C resultante

Por los resultados obtenidos se determinó que la razón de cambio que se necesita se encuentra entre 0 y 1Hz por lo tanto se realizó un barrido de 0.5-0.9Hz obteniendo los resultados de la Tabla 5; mostrando que la razón de cambio se encuentra entre 0.9 y 0.8 Hz por cada variación en la temperatura de 0.1°C .

Cambio en frecuencia	Cambio en fase promedio (grados)
0.9	0.0239
0.8	0.0213
0.7	0.0186
0.6	0.0159
0.5	0.0133

Tabla 5: Barrido de cambios en frecuencia por cada 0.1°C y cambios en la fase resultantes

Siguiendo este método se logró llegar a la razón de cambio de 0.854Hz por cada variación en temperatura de 0.1°C, que este genera una razón de cambio en la fase de 0.0227 grados por cada variación en la temperatura de 0.1°C.

Se comprobó el modelo calculando en otro rango de trabajo el promedio de cambio de fase en la variación de 0.1°C de temperatura. El rango escogido fue de 30.8 a 31.9°C y se halló que el cambio promedio de fase en una variación en la temperatura de 0.1°C fue 0.0207 grados presentado un porcentaje de error del 8.8% con respecto al medido originalmente siendo muy cercano al valor del rango de trabajo analizado anteriormente.

Para finalizar el modelo de variación de temperatura del cristal se debe analizar con una razón de cambio de la frecuencia de 0.854Hz por cada variación en 0.1°C en la temperatura.

4. CONCLUSIONES Y CONSIDERACIONES FINALES

Con este modelo modificado y comprobado experimentalmente se podrá hacer de una manera más exacta un sistema de compensación de temperatura debido a que se tiene un valor aproximado a la razón de cambio de la frecuencia con respecto a la variación de la temperatura el cual fue obtenido en condiciones reales y no ideales.

Durante la medición, el sistema de caracterización debe estar aislado de cualquier ruido ambiental (vibración, movimientos), debido a que la medición puede ser muy sensible a estos ruidos generando datos inestables o incorrectos.

El sistema fue modificado para no presentar complicaciones con una red pública de internet mediante la generación de una red cerrada entre el sistema y el computador asegurando la estabilidad de la transmisión de los datos.

Debe actualizarse el algoritmo del sistema de perfilación de temperaturas y caracterización ya que, este se encuentra en una versión antigua del lenguaje de programación Python y esto generará complicaciones para futuras mediciones con este sistema.

En futuros trabajos se podrá analizar de manera más detallada el comportamiento de este modelo

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

REFERENCIAS

- Cao-Paz, A., Rodríguez-Pardo, L., & Fariña, J. (2014). Temperature compensation of QCM sensors in liquid media. *Sensors and Actuators B: Chemical*, *193*, 78–81. <https://doi.org/10.1016/j.snb.2013.11.044>
- Afzal, A., Mujahid, A., Schirhagl, R., Bajwa, S., Latif, U., & Feroz, S. (2017). Gravimetric Viral Diagnostics: QCM Based Biosensors for Early Detection of Viruses. *Chemosensors*, *5*(1), 7. doi:10.3390/chemosensors5010007
- Edvardsson, M. (2018). Why temperature stability is critical in QCM-measurements.
- Ferizzola, Y. J. M., Narbon, J. V. G., Jiménez, Y. J., Iborra, C. M., Baidés, A. M., Villa, R. A., & Vives, A. A. (2011). Oscillator for biosensors based on quartz crystal microbalance (QCM) | Oscilador para biosensores basado en microbalanza de cristal de cuarzo (QCM). *Revista Facultad de Ingeniería*, (61), 114–122. Recuperado de <http://aprendeenlinea.udea.edu.co/revistas/index.php/ingenieria/article/view/13543>
- IDEAM. (2018). *Variabilidad Climática y Cambio Climático en Colombia*. Recuperado de <http://documentacion.ideam.gov.co/openbiblio/bvirtual/023778/variabilidad.pdf>
- Jimenez, C., & León, D. E. (2009). Biosensores: aplicaciones y perspectivas en el control y calidad de procesos y productos alimenticios. *Vitae*, *16*(1), 144–154. Recuperado de http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0121-40042009000100017
- Johnson, J. B. (1928). Thermal Agitation of Electricity in Conductors. *Physical Review*, *32*(1), 97–109. <https://doi.org/10.1103/PhysRev.32.97>
- Mandal, A. (2019). Principios del biosensor. Recuperado de [https://www.news-medical.net/health/Biosensor-Principles-\(Spanish\).aspx](https://www.news-medical.net/health/Biosensor-Principles-(Spanish).aspx)
- Montagut, Y. J. (2011). *Sistema Oscilador Mejorado para Aplicaciones de Microbalanza (QCM) en Medios Líquidos y Propuesta de un Nuevo Método de Caracterización para Biosensores Piezoeléctricos*. UNIVERSIDAD POLITÉCNICA DE VALENCIA.
- Ortiz, D. (2016). *SISTEMA PARA LA COMPENSACIÓN DE TEMPERATURA EN BIOSENSORES PIEZOELÉCTRICOS (ESTUDIO DE SIMULACIÓN)* (p. 106). p. 106. Envigado.
- Pohanka, M. (2018). Overview of Piezoelectric Biosensors, Immunosensors and DNA Sensors and Their Applications. *Materials*, *11*(3), 448. <https://doi.org/10.3390/ma11030448>

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

- Raddani, P. (2015). *Quartz Crystal Microbalance: Temperature Effects Analysis* (POLITECNICO DI MILANO). Recuperado de https://www.politesi.polimi.it/bitstream/10589/117500/3/Tesi_PaoloRaddani_823087.pdf
- Rahtu, A., & Ritala, M. (2002). Compensation of temperature effects in quartz crystal microbalance measurements. *Applied Physics Letters*, 80(3), 521–523. <https://doi.org/10.1063/1.1433904>
- Vashist, S. K., & Vashist, P. (2011). Recent Advances in Quartz Crystal Microbalance-Based Sensors. *Journal of Sensors*, 2011, 1–13. <https://doi.org/10.1155/2011/571405>

ANEXO 1: ALGORITMO DE PYTHON3 CON EXPERIMENTOS Y MODELACIÓN DE VARIACIONES DE FRECUENCIAS

```
# -*- coding: utf-8 -*-  
"""
```

```
Created on Mon Oct 28 18:15:08 2019
```

```
@author: Tomas  
"""
```

```
import matplotlib.pyplot as plt  
import numpy as np  
import tkinter as tk
```

```
# Toma de variables iniciales a usar en el modelo
```

```
fs = 10e6 #Hz #Frecuencia de resonancia  
fsl = 9995721.5  
C66 = 2.947e10 #Modulo de cizalla efectivo  
e26 = 9.657e-2 #Constante de esfuerzo piezoeléctrico  
pq = 2651 #Kgm^-3 #Densidad del cristal  
As = 2.92e-5 #Área del electrodo  
E22 = 3.982e-11 #Permitividad  
nq = 9.27e-3 #Viscosidad efectiva del cuarzo  
nl = 1e-3 #Viscosidad del líquido  
pL = 1000 #Densidad líquido  
pc = 1000 #Densidad del recubrimiento  
hc = 1e-7 #Espesor base del recubrimiento  
Rq = 9.63 #Resistencia del cristal
```

```
vq = (C66/pq)**0.5 #Velocidad de propagación de la onda  
hq = vq/(2*fs) #Espesor del cristal  
Kt = (hq**2)/(4*(e26**2)*As) #Constante electroacústica  
w = (vq*np.pi)/hq #Frecuencia angular  
a = hq*w*(pq/C66)**0.5 #Fase de onda acústica a través del cuarzo  
dl = ((2*nl)/(w*pL))**0.5 #Profundidad de penetración de la onda acústica en el líquido  
Lq = (pq*(hq**3))/(8*(e26**2)*As) #Inductancia del cristal  
Ko2 = (e26**2)/(E22*C66) #Factor de acoplo electromecánico  
ml = (pL*dl)/2 #Densidad superficial de masa equivalente del líquido en contacto con el cristal  
LI = ml*Kt #Inductancia del líquido  
Co = E22*As/hq #Capacidad estática del cristal  
Cq = (8*Ko2*Co)/(np.pi**2) #Capacidad dinámica del cristal  
mc = pc*hc #Densidad superficial de masa del recubrimiento  
Lc = Kt*mc #Inductancia del recubrimiento
```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```
RI = w*Kt*ml #Resistencia del líquido
wl = 9995721.5*2*np.pi
```

```
"""
```

```
Experimento 1: Suponiendo que la variación de la temperatura cambia
el grosor del recubrimiento.
```

```
"""
```

```
"Llenado de listas"
```

```
# Deltas hc
```

```
Dhcpos = np.arange(0,5e-8,1e-10)
```

```
Dhcneg = np.arange(0,-5e-8,-1e-10)
```

```
#Hc sumados y restados
```

```
Hcpos = []
```

```
Hcneg = []
```

```
for j in range (0,500):
```

```
    Hcpos.append(hc + Dhcpos[j])
```

```
    Hcneg.append(hc + Dhcneg[j])
```

```
# mc resultantes en Kg/m^2
```

```
mcpos = []
```

```
mcneg = []
```

```
for j in range (0,500):
```

```
    mcpos.append(pc*(Hcpos[j]))
```

```
    mcneg.append(pc*(Hcneg[j]))
```

```
#Lc resultantes en F
```

```
Lcpos = []
```

```
Lcneg = []
```

```
for j in range (0,500):
```

```
    Lcpos.append(Kt*mcpos[j])
```

```
    Lcneg.append(Kt*mcneg[j])
```

```
#Frecuencias resultantes en Hz
```

```
Fspos = []
```

```
Fsneg = []
```

```
for j in range (0,500):
```

```
    Fspos.append(1/(2*np.pi*((Lcpos[j]+Lq+Ll)*Cq)**(1/2)))
```

```
    Fsneg.append(1/(2*np.pi*((Lcneg[j]+Lq+Ll)*Cq)**(1/2)))
```

```
#Delta con fs en MHz
```

```
Dfspos = []
```

```
Dfsneg = []
```

```
for j in range (0,500):
```

```
    Dfspos.append((Fspos[j])-9995721.5)
```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```

Dfsneg.append((Fsneg[j])-9995721.5)

#Lt a partir de frecuencias
Ltps = []
Ltng = []
for j in range (0,500):
    Ltps.append((1/(((Fspos[j])**2)*Cq*(4*(np.pi**2))))-(Lq+Ll+Lc))
    Ltng.append((1/(((Fsneg[j])**2)*Cq*(4*(np.pi**2))))-(Lq+Ll+Lc))

Zfpos = []
Zfneg = []
for j in range (0,500):
    Zfpos.append(np.arctan(((wl*(Lcpos[j]+Lq+Ll))-(1/(wl*Cq)))/(Rq+Rl)))
    Zfneg.append(np.arctan(((wl*(Lcneg[j]+Lq+Ll))-(1/(wl*Cq)))/(Rq+Rl)))

#plt.plot(Zfpos,Dfspos)
#plt.plot(Zfneg,Dfsneg)

"Interfaz de usuario"

win = tk.Tk()
win.title("Variación de frecuencia debido al grosor")

#Listas
scrollbar3 = tk.Scrollbar(win)
scrollbar3.grid(column=3,row=1,sticky = tk.N,rowspan=50,ipady=60)
Dhcp = tk.Listbox(win, yscrollcommand = scrollbar3.set)
Dhcp.grid(column=2,row=1,ipady=5,ipadx=10)

scrollbar4 = tk.Scrollbar(win)
scrollbar4.grid(column=3,row=3,sticky = tk.N,rowspan=50,ipady=60)
Dhcn = tk.Listbox(win, yscrollcommand = scrollbar4.set)
Dhcn.grid(column=2,row=3,ipady=5,ipadx=10)

scrollbar5 = tk.Scrollbar(win)
scrollbar5.grid(column=5,row=1,sticky = tk.N,rowspan=50,ipady=60)
Dmcp = tk.Listbox(win, yscrollcommand = scrollbar5.set)
Dmcp.grid(column=4,row=1,ipady=5,ipadx=10)

scrollbar6 = tk.Scrollbar(win)
scrollbar6.grid(column=5,row=3,sticky = tk.N,rowspan=50,ipady=60)
Dmcn = tk.Listbox(win, yscrollcommand = scrollbar6.set)
Dmcn.grid(column=4,row=3,ipady=5,ipadx=10)

scrollbar7 = tk.Scrollbar(win)
scrollbar7.grid(column=7,row=1,sticky = tk.N,rowspan=50,ipady=60)
Dlcp = tk.Listbox(win, yscrollcommand = scrollbar7.set)

```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```

Dlcp.grid(column=6,row=1,ipady=5,ipadx=10)

scrollbar8 = tk.Scrollbar(win)
scrollbar8.grid(column=7,row=3,sticky = tk.N,rowspan=50,ipady=60)
Dlcn = tk.Listbox(win, yscrollcommand = scrollbar8.set)
Dlcn.grid(column=6,row=3,ipady=5,ipadx=10)

scrollbar9 = tk.Scrollbar(win)
scrollbar9.grid(column=9,row=1,sticky = tk.N,rowspan=50,ipady=60)
Fslp = tk.Listbox(win, yscrollcommand = scrollbar9.set)
Fslp.grid(column=8,row=1,ipady=5,ipadx=10)

scrollbar10 = tk.Scrollbar(win)
scrollbar10.grid(column=9,row=3,sticky = tk.N,rowspan=50,ipady=60)
Fsln = tk.Listbox(win, yscrollcommand = scrollbar10.set)
Fsln.grid(column=8,row=3,ipady=5,ipadx=10)

scrollbar11 = tk.Scrollbar(win)
scrollbar11.grid(column=11,row=1,sticky = tk.N,rowspan=50,ipady=60)
Dfslp = tk.Listbox(win, yscrollcommand = scrollbar11.set)
Dfslp.grid(column=10,row=1,ipady=5,ipadx=10)

scrollbar12 = tk.Scrollbar(win)
scrollbar12.grid(column=11,row=3,sticky = tk.N,rowspan=50,ipady=60)
Dfsln = tk.Listbox(win, yscrollcommand = scrollbar12.set)
Dfsln.grid(column=10,row=3,ipady=5,ipadx=10)

scrollbar13 = tk.Scrollbar(win)
scrollbar13.grid(column=13,row=1,sticky =tk.N,rowspan=50,ipady=60)
Ltp = tk.Listbox(win, yscrollcommand = scrollbar13.set)
Ltp.grid(column=12,row=1,ipady=5,ipadx=10)

scrollbar14 = tk.Scrollbar(win)
scrollbar14.grid(column=13,row=3,sticky = tk.N,rowspan=50,ipady=60)
Ltn = tk.Listbox(win, yscrollcommand = scrollbar14.set)
Ltn.grid(column=12,row=3,ipady=5,ipadx=10)

#titulos
tit4 =tk.Label(win, text="Dhc+")
tit4.grid(column=2,row=0)
tit5 =tk.Label(win, text="Dhc-")
tit5.grid(column=2,row=2)
tit6 =tk.Label(win, text="Dmc+")
tit6.grid(column=4,row=0)
tit7 =tk.Label(win, text="Dmc-")
tit7.grid(column=4,row=2)

```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```

tit8 =tk.Label(win, text="DLc+")
tit8.grid(column=6,row=0)
tit9 =tk.Label(win, text="DLc-")
tit9.grid(column=6,row=2)
tit10 =tk.Label(win, text="Fsl+")
tit10.grid(column=8,row=0)
tit11 =tk.Label(win, text="Fsl-")
tit11.grid(column=8,row=2)
tit12 =tk.Label(win, text="DFsl+")
tit12.grid(column=10,row=0)
tit13 =tk.Label(win, text="DFsl-")
tit13.grid(column=10,row=2)
tit14 =tk.Label(win, text="Lt+")
tit14.grid(column=12,row=0)
tit15 =tk.Label(win, text="Lt-")
tit15.grid(column=12,row=2)

```

Subprogramas

```

def clicked():
    for ind in range(0,500):
        Dhcp.insert(tk.END,str(Dhcpos[ind]))
        Dhcn.insert(tk.END,str(Dhcneg[ind]))
        Dmcp.insert(tk.END,str(mcp[0:ind]))
        Dmcp.insert(tk.END,str(mcpneg[ind]))
        Dlcp.insert(tk.END,str(Lcpos[ind]))
        Dlcn.insert(tk.END,str(Lcneg[ind]))
        Fslp.insert(tk.END,str(Fspos[ind]))
        Fsln.insert(tk.END,str(Fsneg[ind]))
        Dfslp.insert(tk.END,str(Dfspos[ind]))
        Dfsln.insert(tk.END,str(Dfsneg[ind]))
        Ltp.insert(tk.END,str(Ltps[ind]))
        Ltn.insert(tk.END,str(Ltng[ind]))
        scrollbar3.config(command = Dhcp.yview)
        scrollbar4.config(command = Dhcn.yview)
        scrollbar5.config(command = Dmcp.yview)
        scrollbar6.config(command = Dmcp.yview)
        scrollbar7.config(command = Dlcp.yview)
        scrollbar8.config(command = Dlcn.yview)
        scrollbar9.config(command = Fslp.yview)
        scrollbar10.config(command = Fsln.yview)
        scrollbar11.config(command = Dfslp.yview)
        scrollbar12.config(command = Dfsln.yview)
        scrollbar13.config(command = Ltp.yview)
        scrollbar14.config(command = Ltn.yview)
    """ f = open('Lt.txt','w')
    for j in range (0,500):
        f.write(str(Ltpos[j])+'\n')

```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```

f.close()"""

# Botones
calcular = tk.Button(win,text="Calcular", command=clicked)
calcular.grid(column=2,row=4)

#win.mainloop()

"Fin de experimento 1"

"""
Experimento 2: Calcular valores de Lt variando la frecuencia de resonancia en
distintos rangos y distintos pasos
"""

"Llenado de listas"

fsl = Fspos[0]

#Variaciones de frecuencia
Fp10 = np.arange(fsl,fsl+11,1)
Fn10 = np.arange(fsl,fsl-11,-1)
Fp100 = np.arange(fsl,fsl+110,10)
Fn100 = np.arange(fsl,fsl-110,-10)

#Lt's dependiendo de la var en frecuencia
Ltfp10 = []
Ltfn10 = []
Ltfp100 = []
Ltfn100 = []
for j in range(0,11):
    Ltfp10.append((1/(((Fp10[j])**2)*Cq*(4*(np.pi**2))))-(Lq+Ll+Lc))
    Ltfn10.append((1/(((Fn10[j])**2)*Cq*(4*(np.pi**2))))-(Lq+Ll+Lc))
    Ltfp100.append((1/(((Fp100[j])**2)*Cq*(4*(np.pi**2))))-(Lq+Ll+Lc))
    Ltfn100.append((1/(((Fn100[j])**2)*Cq*(4*(np.pi**2))))-(Lq+Ll+Lc))

"Interfaz de usuario"

win1 = tk.Tk()
win1.title("Valores de Lt con cambios de frecuencia")

scrollbar20 = tk.Scrollbar(win1)
scrollbar20.grid(column=1,row=1,sticky = tk.N,rowspan=50,ipady=60)
Fpos10 = tk.Listbox(win1, yscrollcommand = scrollbar20.set)
Fpos10.grid(column=0,row=1,ipady=5,ipadx=9)

scrollbar21 = tk.Scrollbar(win1)

```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```
scrollbar21.grid(column=1,row=3,sticky = tk.N,rowspan=50,ipady=60)
Fneg10 = tk.Listbox(win1, yscrollcommand = scrollbar21.set)
Fneg10.grid(column=0,row=3,ipady=5,ipadx=9)
```

```
scrollbar22 = tk.Scrollbar(win1)
scrollbar22.grid(column=3,row=1,sticky = tk.N,rowspan=50,ipady=60)
Fpos100 = tk.Listbox(win1, yscrollcommand = scrollbar22.set)
Fpos100.grid(column=2,row=1,ipady=5,ipadx=9)
```

```
scrollbar23 = tk.Scrollbar(win1)
scrollbar23.grid(column=3,row=3,sticky = tk.N,rowspan=50,ipady=60)
Fneg100 = tk.Listbox(win1, yscrollcommand = scrollbar23.set)
Fneg100.grid(column=2,row=3,ipady=5,ipadx=9)
```

```
scrollbar24 = tk.Scrollbar(win1)
scrollbar24.grid(column=5,row=1,sticky = tk.N,rowspan=50,ipady=60)
Ltpos10 = tk.Listbox(win1, yscrollcommand = scrollbar24.set)
Ltpos10.grid(column=4,row=1,ipady=5,ipadx=9)
```

```
scrollbar25 = tk.Scrollbar(win1)
scrollbar25.grid(column=5,row=3,sticky = tk.N,rowspan=50,ipady=60)
Ltneg10 = tk.Listbox(win1, yscrollcommand = scrollbar25.set)
Ltneg10.grid(column=4,row=3,ipady=5,ipadx=9)
```

```
scrollbar26 = tk.Scrollbar(win1)
scrollbar26.grid(column=7,row=1,sticky = tk.N,rowspan=50,ipady=60)
Ltpos100 = tk.Listbox(win1, yscrollcommand = scrollbar26.set)
Ltpos100.grid(column=6,row=1,ipady=5,ipadx=9)
```

```
scrollbar27 = tk.Scrollbar(win1)
scrollbar27.grid(column=7,row=3,sticky = tk.N,rowspan=50,ipady=60)
Ltneg100 = tk.Listbox(win1, yscrollcommand = scrollbar27.set)
Ltneg100.grid(column=6,row=3,ipady=5,ipadx=9)
```

Titulos

```
tit1 =tk.Label(win1, text="Fsl + 10")
tit1.grid(column=0,row=0)
tit2 =tk.Label(win1, text="Fsl - 10")
tit2.grid(column=0,row=2)
tit3 =tk.Label(win1, text="Fsl + 100")
tit3.grid(column=2,row=0)
tit4 =tk.Label(win1, text="Fsl - 100")
tit4.grid(column=2,row=2)
tit5 =tk.Label(win1, text="Lt + 10")
tit5.grid(column=4,row=0)
tit6 =tk.Label(win1, text="Lt - 10")
tit6.grid(column=4,row=2)
```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```

tit7 =tk.Label(win1, text="Lt + 100")
tit7.grid(column=6,row=0)
tit8 =tk.Label(win1, text="Lt - 100")
tit8.grid(column=6,row=2)

# Subprogramas
def clicked1():
    for ind in range(0,10):
        Fpos10.insert(tk.END,str(Fp10[ind]))
        Fneg10.insert(tk.END,str(Fn10[ind]))
        Fpos100.insert(tk.END,str(Fp100[ind]))
        Fneg100.insert(tk.END,str(Fn100[ind]))
        Ltpos10.insert(tk.END,str(Ltfn10[ind]))
        Ltneg10.insert(tk.END,str(Ltfn10[ind]))
        Ltpos100.insert(tk.END,str(Ltfn100[ind]))
        Ltneg100.insert(tk.END,str(Ltfn100[ind]))
        scrollbar20.config(command = Fpos10.yview)
        scrollbar21.config(command = Fneg10.yview)
        scrollbar22.config(command = Fpos100.yview)
        scrollbar23.config(command = Fneg100.yview)
        scrollbar24.config(command = Ltpos10.yview)
        scrollbar25.config(command = Ltneg10.yview)
        scrollbar26.config(command = Ltpos100.yview)
        scrollbar27.config(command = Ltneg100.yview)

# Botones
calcular1 = tk.Button(win1,text="Calcular", command=clicked1)
calcular1.grid(column=2,row=4)

#win1.mainloop()

"Fin de Experimento 2"

"""
Experimento 3: Creación y Lectura de archivos de texto para
el futuro análisis de los resultados
"""

"""win2 = tk.Tk()
win2.title("Lectura de archivos de texto")

Ltpdata=[]

a = open('Lt.txt','r')
for linea in a.readlines():
    Ltpdata.append(linea)
a.close()

```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.


```

scrollbar28 = tk.Scrollbar(win2)
scrollbar28.grid(column=1,row=1,sticky = tk.N,rowspan=50,ipady=60)
DatLtpos = tk.Listbox(win2, yscrollcommand = scrollbar28.set)
DatLtpos.grid(column=0,row=1,ipady=5,ipadx=9)

tit9 = tk.Label(win2,text = "Datos Lt +")
tit9.grid(column=0,row=0)

def clicked2():
    for j in range (0,500):
        DatLtpos.insert(tk.END,Ltpdata[j])
        scrollbar28.config(command = DatLtpos.yview)

calcular2 = tk.Button(win2,text="Leer",command=clicked2)
calcular2.grid(column=2,row=1)

#win2.mainloop()"""

"Fin de Experimento 3"

"""
Experimento 4: Calcular valores de Lt variando la fs en un mayor rango
y con un paso más pequeño
"""

fsl = Fspos[0] #Esta es la fs de la QCM en presencia de agua, al usar LI debe usarse esta
frecuencia

"Llenado de listas"

#Variaciones de frecuencia
Fp = np.arange(fsl,fsl+1250,5)
Fn = np.arange(fsl,fsl-1250,-5)

#Lt's dependiendo de la var en frecuencia
Ltfp = []
Ltfn = []
Zf = []
Zm = []
Dfsl =[]

for j in range(0,250):
    Ltfp.append((1/(((Fp[j])**2)*Cq*(4*(np.pi**2))))-(Lq+LI+Lc))
    Ltfn.append((1/(((Fn[j])**2)*Cq*(4*(np.pi**2))))-(Lq+LI+Lc))
    Zm.append((((RI+Rq)**2) + (((w*(LI+Lc+Ltfn[j])) - 1/(w*Cq))**2))**0.5)
    Zf.append((180/np.pi)*np.arctan(((wl*(LI+Lc+Ltfn[j]+Lq)) - 1/(wl*Cq))/(RI+Rq)))

```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```

Dfsl.append(Fn[j]-fsl)

fig = plt.figure(constrained_layout=True)
gs = fig.add_gridspec(1,3)
LT = fig.add_subplot(gs[0,0])
LT.set_ylabel('Lt(H)',fontsize=12)
LT.set_xlabel('Dfrecuencia(Hz)',fontsize=12)
FaseLt = fig.add_subplot(gs[0,1])
FaseLt.set_ylabel('Fase(°)',fontsize=12)
FaseLt.set_xlabel('Dfrecuencia(Hz)',fontsize=12)
MagLt = fig.add_subplot(gs[0,2])
MagLt.set_ylabel('Magnitud',fontsize=12)
MagLt.set_xlabel('Dfrecuencia(Hz)',fontsize=12)
LT.plot(Dfsl,Lfn,color='blue')
FaseLt.plot(Dfsl,Zf,color='red')
MagLt.plot(Dfsl,Zm,color='green')
LT.grid()
FaseLt.grid()
MagLt.grid()
plt.show()

```

"Interfaz de usuario"

```

win3 = tk.Tk()
win3.title("Lt a partir de variacion de frecuencia")

scrollbar29 = tk.Scrollbar(win3)
scrollbar29.grid(column=1,row=1,sticky = tk.N,rowspan=50,ipady=60)
Fneg = tk.Listbox(win3, yscrollcommand = scrollbar29.set)
Fneg.grid(column=0,row=1,ipady=5,ipadx=9)

scrollbar30 = tk.Scrollbar(win3)
scrollbar30.grid(column=1,row=3,sticky = tk.N,rowspan=50,ipady=60)
Dfrec = tk.Listbox(win3, yscrollcommand = scrollbar30.set)
Dfrec.grid(column=0,row=3,ipady=5,ipadx=9)

scrollbar31 = tk.Scrollbar(win3)
scrollbar31.grid(column=3,row=1,sticky = tk.N,rowspan=50,ipady=60)
Ltneg = tk.Listbox(win3, yscrollcommand = scrollbar31.set)
Ltneg.grid(column=2,row=1,ipady=5,ipadx=9)

scrollbar32 = tk.Scrollbar(win3)
scrollbar32.grid(column=3,row=3,sticky = tk.N,rowspan=50,ipady=60)
Zmag = tk.Listbox(win3, yscrollcommand = scrollbar32.set)
Zmag.grid(column=2,row=3,ipady=5,ipadx=9)

```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```

scrollbar33 = tk.Scrollbar(win3)
scrollbar33.grid(column=5,row=1,sticky = tk.N,rowspan=50,ipady=60)
Zfase = tk.Listbox(win3, yscrollcommand = scrollbar33.set)
Zfase.grid(column=4,row=1,ipady=5,ipadx=9)

```

```
# Titulos
```

```

tit1 =tk.Label(win3, text="Fo - ")
tit1.grid(column=0,row=0)
tit2 =tk.Label(win3, text="DFo ")
tit2.grid(column=0,row=2)
tit5 =tk.Label(win3, text="Lt - ")
tit5.grid(column=2,row=0)
tit6 =tk.Label(win3, text="|Z|")
tit6.grid(column=2,row=2)
tit7 =tk.Label(win3, text="Q(Z)")
tit7.grid(column=4,row=0)

```

```
# Subprogramas
```

```

def clicked1():
    for ind in range(0,250):
        Fneg.insert(tk.END,str(Fn[ind]))
        Dfrec.insert(tk.END,str(Dfsl[ind]))
        Ltneg.insert(tk.END,str(Ltn[ind]))
        Zmag.insert(tk.END,str(Zm[ind]))
        Zfase.insert(tk.END,str(Zf[ind]))
    scrollbar29.config(command = Fneg.yview)
    scrollbar30.config(command = Dfrec.yview)
    scrollbar31.config(command = Ltneg.yview)
    scrollbar32.config(command = Zmag.yview)
    scrollbar33.config(command = Zfase.yview)

```

```
# Botones
```

```

calcular1 = tk.Button(win3,text="Calcular", command=clicked1)
calcular1.grid(column=2,row=4)

```

```
#win3.mainloop()
```

```
"""
```

```
Fin de Experimento 4
```

```
Se concluye que la frecuencia solo puede variar negativamente con respecto a la temperatura ya que si varia positivamente la Lt sería negativa
```

```
"""
```

```
'''
```

```
Experimento 5: A partir de una gráfica de comportamiento de la frecuencia con
```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

respecto al tiempo se obtiene el valor de Lm durante ese tiempo, luego con esos valores de Lm graficamos el comportamiento de la fase y luego con cada uno de los valores de Lt obtenidos anteriormente se obtienen los distintos comportamientos de fase dependiendo del valor que tenga la Lt.

```
'''
wl = fsl*2*np.pi
```

```
#Puntos de la gráfica delta de frecuencia vs tiempo tomado cada segundo
Df1 = [0,0,0,0,0,0,-9,-37,-61,-82,-99.5,-109,-113.5,-117,-119.5,-121,-121]
```

```
f1 = []
```

```
for j in range (0,17):
    f1.append(fsl + Df1[j])
```

```
#Obtención de Lm
```

```
Lm = []
```

```
for j in range (0,17):
    Lm.append((1/(((f1[j])**2)*Cq*(4*(np.pi**2))))-(Lq+Ll+Lc))
```

```
#Obtención de Fase
```

```
Q = []
```

```
for j in range(0,17):
    Q.append(np.arctan(((wl*(Ll+Lc+Lm[j])+Lq)) - 1/(wl*Cq))/(Rl+Rq))
```

```
QR = []
```

```
for j in range (0,21):
```

```
    if j<=10:
```

```
        QR.append((180/np.pi)*np.arctan(((wl*(Ll+Lc+Ltfp10[10-j])+Lq)) - 1/(wl*Cq))/(Rl+Rq))
```

```
    else:
```

```
        QR.append((180/np.pi)*np.arctan(((wl*(Ll+Lc+Ltfp10[j-10])+Lq)) - 1/(wl*Cq))/(Rl+Rq))
```

```
#Lista de valores de Lt a usar
```

```
Lt1 = []
```

```
for j in range(0,17):
    Lt1.append(Ltfn[j])
```

```
#Fase con cada valor de Lt
```

```
Q1 = [];Q2 = [];Q3 = [];Q4 = [];Q5 = [];Q6 = [];Q7 = [];Q8 = [];Q9 = [];Q10 = []
```

```
Q11 = []#Fase sin Lt, ideal
```

```
Q12 = [];Q13 = [];Q14 = [];Q15 = [];Q16 = [];Q17 = [];Q18 = [];Q19 = [];Q20 = [];Q21 = []
```

```
for j in range (0,17):
```

```
    Q1.append((180/np.pi)*np.arctan(((wl*(Ll+Lc+Lm[j])+Lq+Ltfp10[10])) - 1/(wl*Cq))/(Rl+Rq))
```

```
    Q2.append((180/np.pi)*np.arctan(((wl*(Ll+Lc+Lm[j])+Lq+Ltfp10[9])) - 1/(wl*Cq))/(Rl+Rq))
```

```
    Q3.append((180/np.pi)*np.arctan(((wl*(Ll+Lc+Lm[j])+Lq+Ltfp10[8])) - 1/(wl*Cq))/(Rl+Rq))
```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```

Q4.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltn10[7]) - 1/(wl*Cq))/(Ri+Rq)))
Q5.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltn10[6]) - 1/(wl*Cq))/(Ri+Rq)))
Q6.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltn10[5]) - 1/(wl*Cq))/(Ri+Rq)))
Q7.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltn10[4]) - 1/(wl*Cq))/(Ri+Rq)))
Q8.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltn10[3]) - 1/(wl*Cq))/(Ri+Rq)))
Q9.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltn10[2]) - 1/(wl*Cq))/(Ri+Rq)))
Q10.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltn10[1]) - 1/(wl*Cq))/(Ri+Rq)))
Q11.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltn10[0]) - 1/(wl*Cq))/(Ri+Rq)))
Q12.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltfp10[1]) - 1/(wl*Cq))/(Ri+Rq)))
Q13.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltfp10[2]) - 1/(wl*Cq))/(Ri+Rq)))
Q14.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltfp10[3]) - 1/(wl*Cq))/(Ri+Rq)))
Q15.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltfp10[4]) - 1/(wl*Cq))/(Ri+Rq)))
Q16.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltfp10[5]) - 1/(wl*Cq))/(Ri+Rq)))
Q17.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltfp10[6]) - 1/(wl*Cq))/(Ri+Rq)))
Q18.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltfp10[7]) - 1/(wl*Cq))/(Ri+Rq)))
Q19.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltfp10[8]) - 1/(wl*Cq))/(Ri+Rq)))
Q20.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltfp10[9]) - 1/(wl*Cq))/(Ri+Rq)))
Q21.append((180/np.pi)*np.arctan(((wl*(Li+Lc+Lm[j])+Lq+Ltfp10[10]) - 1/(wl*Cq))/(Ri+Rq)))

```

```

dQ1 = [];dQ2 = [];dQ3 = [];dQ4 = [];dQ5 = [];dQ6 = [];dQ7 = [];dQ8 = [];dQ9 = [];dQ10 = []
dQ11 = []#Fases sin Lt, ideal
dQ12 = [];dQ13 = [];dQ14 = [];dQ15 = [];dQ16 = [];dQ17 = [];dQ18 = [];dQ19 = [];dQ20 = []
dQ21 = []

```

```

for j in range (0,17):
    dQ1.append(Q1[j]-QR[0])
    dQ2.append(Q2[j]-QR[1])
    dQ3.append(Q3[j]-QR[2])
    dQ4.append(Q4[j]-QR[3])
    dQ5.append(Q5[j]-QR[4])
    dQ6.append(Q6[j]-QR[5])
    dQ7.append(Q7[j]-QR[6])
    dQ8.append(Q8[j]-QR[7])
    dQ9.append(Q9[j]-QR[8])
    dQ10.append(Q10[j]-QR[9])

```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```
dQ11.append(Q11[j]-QR[10])
dQ12.append(Q12[j]-QR[11])
dQ13.append(Q13[j]-QR[12])
dQ14.append(Q14[j]-QR[13])
dQ15.append(Q15[j]-QR[14])
dQ16.append(Q16[j]-QR[15])
dQ17.append(Q17[j]-QR[16])
dQ18.append(Q18[j]-QR[17])
dQ19.append(Q19[j]-QR[18])
dQ20.append(Q20[j]-QR[19])
dQ21.append(Q21[j]-QR[20])
```

#Voltajes que debe obtener el sensor de fase

```
VQ1 = [];VQ2 = [];VQ3 = [];VQ4 = [];VQ5 = [];VQ6 = [];VQ7 = [];VQ8 = [];VQ9 = [];VQ10 = []
VQ11 = []#Voltaje sin Lt
VQ12 = [];VQ13 = [];VQ14 = [];VQ15 = [];VQ16 = [];VQ17 = [];VQ18 = [];VQ19 = [];VQ20 =
[];VQ21 = []
```

for j in range (0,17):

```
VQ1.append(1800-(10*(Q1[j])))
VQ2.append(1800-(10*(Q2[j])))
VQ3.append(1800-(10*(Q3[j])))
VQ4.append(1800-(10*(Q4[j])))
VQ5.append(1800-(10*(Q5[j])))
VQ6.append(1800-(10*(Q6[j])))
VQ7.append(1800-(10*(Q7[j])))
VQ8.append(1800-(10*(Q8[j])))
VQ9.append(1800-(10*(Q9[j])))
VQ10.append(1800-(10*(Q10[j])))
VQ11.append(1800-(10*(dQ11[j])))
VQ12.append(1800-(10*(Q12[j])))
VQ13.append(1800-(10*(Q13[j])))
VQ14.append(1800-(10*(Q14[j])))
VQ15.append(1800-(10*(Q15[j])))
VQ16.append(1800-(10*(Q16[j])))
VQ17.append(1800-(10*(Q17[j])))
VQ18.append(1800-(10*(Q18[j])))
VQ19.append(1800-(10*(Q19[j])))
VQ20.append(1800-(10*(Q20[j])))
VQ21.append(1800-(10*(Q21[j])))
```

#Gráficas de fases

```
plt.figure()
plt.xlabel('t (min)')
plt.ylabel('Fase (°)')
```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```

plt.plot(Q1);plt.plot(Q2);plt.plot(Q3);plt.plot(Q4);plt.plot(Q5);plt.plot(Q6)
plt.plot(Q7);plt.plot(Q8);plt.plot(Q9);plt.plot(Q10)
plt.plot(Q11,'k',linewidth=4)
plt.plot(Q12);plt.plot(Q13);plt.plot(Q14);plt.plot(Q15);plt.plot(Q16);plt.plot(Q17)
plt.plot(Q18);plt.plot(Q19);plt.plot(Q20);plt.plot(Q21)
plt.figure()
plt.xlabel('t (min)')
plt.ylabel('Fase (°)')
plt.plot(VQ11-(1800-(10*(QR[10]))))

```

```

plt.figure()
plt.xlabel('t (min)')
plt.ylabel('dFase (°)')
plt.plot(dQ1);plt.plot(dQ2);plt.plot(dQ3);plt.plot(dQ4);plt.plot(dQ5);plt.plot(dQ6)
plt.plot(dQ7);plt.plot(dQ8);plt.plot(dQ9);plt.plot(dQ10)
plt.plot(dQ11,'k',linewidth = 3)
plt.plot(dQ12);plt.plot(dQ13);plt.plot(dQ14);plt.plot(dQ15);plt.plot(dQ16)
plt.plot(dQ17);plt.plot(dQ18);plt.plot(dQ19);plt.plot(dQ20);plt.plot(dQ21)

```

#Gráficas de voltaje del sensor

```

plt.figure()
plt.plot(VQ1)
plt.plot(VQ2)
plt.plot(VQ3)
plt.plot(VQ4)
plt.plot(VQ5)
plt.plot(VQ6)
plt.plot(VQ7)
plt.plot(VQ8)
plt.plot(VQ9)
plt.plot(VQ10)
plt.plot(VQ11,'k',linewidth=3)
plt.plot(VQ12)
plt.plot(VQ13)
plt.plot(VQ14)
plt.plot(VQ15)
plt.plot(VQ16)
plt.plot(VQ17)
plt.plot(VQ18)
plt.plot(VQ19)
plt.plot(VQ20)
plt.plot(VQ21)

```

"Interfaz de Usuario"

```

win4 = tk.Tk()
win4.title("Comprobación de Fase")

```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```
# Titulos
tit1 =tk.Label(win4, text="Fase fs-10")
tit1.grid(column=0,row=0)
tit2 =tk.Label(win4, text="Fase fs-9")
tit2.grid(column=2,row=0)
tit3 =tk.Label(win4, text="Fase fs-8")
tit3.grid(column=4,row=0)
tit4 =tk.Label(win4, text="Fase fs-7")
tit4.grid(column=6,row=0)
tit5 =tk.Label(win4, text="Fase fs-6")
tit5.grid(column=8,row=0)
tit6 =tk.Label(win4, text="Fase fs-5")
tit6.grid(column=0,row=2)
tit7 =tk.Label(win4, text="Fase fs-4")
tit7.grid(column=2,row=2)
tit8 =tk.Label(win4, text="Fase fs-3")
tit8.grid(column=4,row=2)
tit9 =tk.Label(win4, text="Fase fs-2")
tit9.grid(column=6,row=2)
tit10 =tk.Label(win4, text="Fase fs-1")
tit10.grid(column=8,row=2)
tit11 =tk.Label(win4, text="Fase fs")
tit11.grid(column=11,row=2)
tit12 =tk.Label(win4, text="Fase fs+1")
tit12.grid(column=0,row=4)
tit13 =tk.Label(win4, text="Fase fs+2")
tit13.grid(column=2,row=4)
tit14 =tk.Label(win4, text="Fase fs+3")
tit14.grid(column=4,row=4)
tit15 =tk.Label(win4, text="Fase fs+4")
tit15.grid(column=6,row=4)
tit16 =tk.Label(win4, text="Fase fs+5")
tit16.grid(column=8,row=4)
tit17 =tk.Label(win4, text="Fase fs+6")
tit17.grid(column=0,row=6)
tit18 =tk.Label(win4, text="Fase fs+7")
tit18.grid(column=2,row=6)
tit19 =tk.Label(win4, text="Fase fs+8")
tit19.grid(column=4,row=6)
tit20 =tk.Label(win4, text="Fase fs+9")
tit20.grid(column=6,row=6)
tit21 =tk.Label(win4, text="Fase fs+10")
tit21.grid(column=8,row=6)
```

```
scrollbar35 = tk.Scrollbar(win4)
```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.


```
scrollbar35.grid(column=1,row=1,sticky = tk.N,rowspan=50,ipady=60)
Fase1 = tk.Listbox(win4, yscrollcommand = scrollbar35.set)
Fase1.grid(column=0,row=1,ipady=5,ipadx=9)
```

```
scrollbar36 = tk.Scrollbar(win4)
scrollbar36.grid(column=3,row=1,sticky = tk.N,rowspan=50,ipady=60)
Fase2 = tk.Listbox(win4, yscrollcommand = scrollbar36.set)
Fase2.grid(column=2,row=1,ipady=5,ipadx=9)
```

```
scrollbar37 = tk.Scrollbar(win4)
scrollbar37.grid(column=5,row=1,sticky = tk.N,rowspan=50,ipady=60)
Fase3 = tk.Listbox(win4, yscrollcommand = scrollbar37.set)
Fase3.grid(column=4,row=1,ipady=5,ipadx=9)
```

```
scrollbar38 = tk.Scrollbar(win4)
scrollbar38.grid(column=7,row=1,sticky = tk.N,rowspan=50,ipady=60)
Fase4 = tk.Listbox(win4, yscrollcommand = scrollbar38.set)
Fase4.grid(column=6,row=1,ipady=5,ipadx=9)
```

```
scrollbar39 = tk.Scrollbar(win4)
scrollbar39.grid(column=9,row=1,sticky = tk.N,rowspan=50,ipady=60)
Fase5 = tk.Listbox(win4, yscrollcommand = scrollbar39.set)
Fase5.grid(column=8,row=1,ipady=5,ipadx=9)
```

```
scrollbar40 = tk.Scrollbar(win4)
scrollbar40.grid(column=1,row=3,sticky = tk.N,rowspan=50,ipady=60)
Fase6 = tk.Listbox(win4, yscrollcommand = scrollbar40.set)
Fase6.grid(column=0,row=3,ipady=5,ipadx=9)
```

```
scrollbar41 = tk.Scrollbar(win4)
scrollbar41.grid(column=3,row=3,sticky = tk.N,rowspan=50,ipady=60)
Fase7 = tk.Listbox(win4, yscrollcommand = scrollbar41.set)
Fase7.grid(column=2,row=3,ipady=5,ipadx=9)
```

```
scrollbar42 = tk.Scrollbar(win4)
scrollbar42.grid(column=5,row=3,sticky = tk.N,rowspan=50,ipady=60)
Fase8 = tk.Listbox(win4, yscrollcommand = scrollbar42.set)
Fase8.grid(column=4,row=3,ipady=5,ipadx=9)
```

```
scrollbar43 = tk.Scrollbar(win4)
scrollbar43.grid(column=7,row=3,sticky = tk.N,rowspan=50,ipady=60)
Fase9 = tk.Listbox(win4, yscrollcommand = scrollbar43.set)
Fase9.grid(column=6,row=3,ipady=5,ipadx=9)
```

```
scrollbar44 = tk.Scrollbar(win4)
scrollbar44.grid(column=9,row=3,sticky = tk.N,rowspan=50,ipady=60)
Fase10 = tk.Listbox(win4, yscrollcommand = scrollbar44.set)
```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```
Fase10.grid(column=8,row=3,ipady=5,ipadx=9)
```

```
scrollbar45 = tk.Scrollbar(win4)
```

```
scrollbar45.grid(column=12,row=3,sticky = tk.N,rowspan=50,ipady=60)
```

```
Fase11 = tk.Listbox(win4, yscrollcommand = scrollbar45.set)
```

```
Fase11.grid(column=11,row=3,ipady=5,ipadx=9)
```

```
scrollbar46 = tk.Scrollbar(win4)
```

```
scrollbar46.grid(column=1,row=5,sticky = tk.N,rowspan=50,ipady=60)
```

```
Fase12 = tk.Listbox(win4, yscrollcommand = scrollbar46.set)
```

```
Fase12.grid(column=0,row=5,ipady=5,ipadx=9)
```

```
scrollbar47 = tk.Scrollbar(win4)
```

```
scrollbar47.grid(column=3,row=5,sticky = tk.N,rowspan=50,ipady=60)
```

```
Fase13 = tk.Listbox(win4, yscrollcommand = scrollbar47.set)
```

```
Fase13.grid(column=2,row=5,ipady=5,ipadx=9)
```

```
scrollbar48 = tk.Scrollbar(win4)
```

```
scrollbar48.grid(column=5,row=5,sticky = tk.N,rowspan=50,ipady=60)
```

```
Fase14 = tk.Listbox(win4, yscrollcommand = scrollbar48.set)
```

```
Fase14.grid(column=4,row=5,ipady=5,ipadx=9)
```

```
scrollbar49 = tk.Scrollbar(win4)
```

```
scrollbar49.grid(column=7,row=5,sticky = tk.N,rowspan=50,ipady=60)
```

```
Fase15 = tk.Listbox(win4, yscrollcommand = scrollbar49.set)
```

```
Fase15.grid(column=6,row=5,ipady=5,ipadx=9)
```

```
scrollbar50 = tk.Scrollbar(win4)
```

```
scrollbar50.grid(column=9,row=5,sticky = tk.N,rowspan=50,ipady=60)
```

```
Fase16 = tk.Listbox(win4, yscrollcommand = scrollbar50.set)
```

```
Fase16.grid(column=8,row=5,ipady=5,ipadx=9)
```

```
scrollbar51 = tk.Scrollbar(win4)
```

```
scrollbar51.grid(column=1,row=7,sticky = tk.N,rowspan=50,ipady=60)
```

```
Fase17 = tk.Listbox(win4, yscrollcommand = scrollbar51.set)
```

```
Fase17.grid(column=0,row=7,ipady=5,ipadx=9)
```

```
scrollbar52 = tk.Scrollbar(win4)
```

```
scrollbar52.grid(column=3,row=7,sticky = tk.N,rowspan=50,ipady=60)
```

```
Fase18 = tk.Listbox(win4, yscrollcommand = scrollbar52.set)
```

```
Fase18.grid(column=2,row=7,ipady=5,ipadx=9)
```

```
scrollbar53 = tk.Scrollbar(win4)
```

```
scrollbar53.grid(column=5,row=7,sticky = tk.N,rowspan=50,ipady=60)
```

```
Fase19 = tk.Listbox(win4, yscrollcommand = scrollbar53.set)
```

```
Fase19.grid(column=4,row=7,ipady=5,ipadx=9)
```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```
scrollbar54 = tk.Scrollbar(win4)
scrollbar54.grid(column=7,row=7,sticky = tk.N,rowspan=50,ipady=60)
Fase20 = tk.Listbox(win4, yscrollcommand = scrollbar54.set)
Fase20.grid(column=6,row=7,ipady=5,ipadx=9)
```

```
scrollbar55 = tk.Scrollbar(win4)
scrollbar55.grid(column=9,row=7,sticky = tk.N,rowspan=50,ipady=60)
Fase21 = tk.Listbox(win4, yscrollcommand = scrollbar55.set)
Fase21.grid(column=8,row=7,ipady=5,ipadx=9)
```

```
def clicked2():
    for ind in range(0,17):
        Fase1.insert(tk.END,str(Q1[ind]))
        Fase2.insert(tk.END,str(Q2[ind]))
        Fase3.insert(tk.END,str(Q3[ind]))
        Fase4.insert(tk.END,str(Q4[ind]))
        Fase5.insert(tk.END,str(Q5[ind]))
        Fase6.insert(tk.END,str(Q6[ind]))
        Fase7.insert(tk.END,str(Q7[ind]))
        Fase8.insert(tk.END,str(Q8[ind]))
        Fase9.insert(tk.END,str(Q9[ind]))
        Fase10.insert(tk.END,str(Q10[ind]))
        Fase11.insert(tk.END,str(Q11[ind]))
        Fase12.insert(tk.END,str(Q12[ind]))
        Fase13.insert(tk.END,str(Q13[ind]))
        Fase14.insert(tk.END,str(Q14[ind]))
        Fase15.insert(tk.END,str(Q15[ind]))
        Fase16.insert(tk.END,str(Q16[ind]))
        Fase17.insert(tk.END,str(Q17[ind]))
        Fase18.insert(tk.END,str(Q18[ind]))
        Fase19.insert(tk.END,str(Q19[ind]))
        Fase20.insert(tk.END,str(Q20[ind]))
        Fase21.insert(tk.END,str(Q21[ind]))
    scrollbar35.config(command = Fase1.yview)
    scrollbar36.config(command = Fase2.yview)
    scrollbar37.config(command = Fase3.yview)
    scrollbar38.config(command = Fase4.yview)
    scrollbar39.config(command = Fase5.yview)
    scrollbar40.config(command = Fase6.yview)
    scrollbar41.config(command = Fase7.yview)
    scrollbar42.config(command = Fase8.yview)
    scrollbar43.config(command = Fase9.yview)
    scrollbar44.config(command = Fase10.yview)
    scrollbar45.config(command = Fase11.yview)
    scrollbar46.config(command = Fase12.yview)
    scrollbar47.config(command = Fase13.yview)
    scrollbar48.config(command = Fase14.yview)
```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```

scrollbar49.config(command = Fase15.yview)
scrollbar50.config(command = Fase16.yview)
scrollbar51.config(command = Fase17.yview)
scrollbar52.config(command = Fase18.yview)
scrollbar53.config(command = Fase19.yview)
scrollbar54.config(command = Fase20.yview)
scrollbar55.config(command = Fase21.yview)

calcular2 = tk.Button(win4,text="Calcular", command=clicked2)
calcular2.grid(column=11,row=5)

"Fin de Experimento 5"
razdecamb = 0.854

fslinit = (fsl-((28-25)/0.1)*razdecamb) #Frecuencia inicial en al que se supone el modelo
donde hay un cambio de 5Hz por cada 0.1 grados

"Llenado de listas"

#Variaciones de frecuencia
Fbarrido = np.arange(fslinit,fslinit-(razdecamb*12),-razdecamb)# se llenará la lista desde la
frecuencia en 28.0 hasta la frecuencia en 29.1

#Lt's dependiendo de la var en frecuencia
Lt = []
Zf = []
Temp = []
Dif = []
dZf =[]

for frec in Fbarrido:
    Lt.append((1/(((frec)**2)*Cq*(4*(np.pi**2))))-(Lq+Ll+Lc))
for j in range(12):
    Zf.append((180/np.pi)*np.arctan(((wl*(Ll+Lc+Lt[j]+Lq)) - 1/(wl*Cq))/(Rl+Rq)))
    Temp.append(28 + (j*0.1))
for j in range(1,12):
    dZf.append(Zf[j]-Zf[j-1])

"Interfaz de usuario"

win5 = tk.Tk()
win5.title("Lt a partir de variacion de frecuencia")

scrollbar56 = tk.Scrollbar(win5)
scrollbar56.grid(column=1,row=1,sticky = tk.N,rowspan=50,ipady=60)
Tempe = tk.Listbox(win5, yscrollcommand = scrollbar56.set)

```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.

```

Tempe.grid(column=0,row=1,ipady=5,ipadx=9)

scrollbar57 = tk.Scrollbar(win5)
scrollbar57.grid(column=1,row=3,sticky = tk.N,rowspan=50,ipady=60)
Ltlista = tk.Listbox(win5, yscrollcommand = scrollbar57.set)
Ltlista.grid(column=0,row=3,ipady=5,ipadx=9)

scrollbar58 = tk.Scrollbar(win5)
scrollbar58.grid(column=3,row=1,sticky = tk.N,rowspan=50,ipady=60)
FaseLista = tk.Listbox(win5, yscrollcommand = scrollbar58.set)
FaseLista.grid(column=2,row=1,ipady=5,ipadx=9)

scrollbar59 = tk.Scrollbar(win5)
scrollbar59.grid(column=3,row=3,sticky = tk.N,rowspan=50,ipady=60)
deltaFase = tk.Listbox(win5, yscrollcommand = scrollbar59.set)
deltaFase.grid(column=2,row=3,ipady=5,ipadx=9)

# Titulos
tit1 =tk.Label(win5, text="Temp")
tit1.grid(column=0,row=0)
tit2 =tk.Label(win5, text="Lt")
tit2.grid(column=0,row=2)
tit3 =tk.Label(win5, text="Fase(Grados)")
tit3.grid(column=2,row=0)
tit4 =tk.Label(win5, text="delta Fase(Grados)")
tit4.grid(column=2,row=2)

# Subprogramas
def clicked1():
    for ind in range(12):
        Tempe.insert(tk.END,str(Temp[ind]))
        Ltlista.insert(tk.END,str(Lt[ind]))
        FaseLista.insert(tk.END,str(Zf[ind]))

    for indx in range(11):
        deltaFase.insert(tk.END,str(dZf[indx]))

    scrollbar56.config(command = Tempe.yview)
    scrollbar57.config(command = Ltlista.yview)
    scrollbar58.config(command = FaseLista.yview)
    scrollbar59.config(command = deltaFase.yview)

# Botones
calcular1 = tk.Button(win5,text="Calcular", command=clicked1)
calcular1.grid(column=2,row=4)

win5.mainloop()

```

La información presentada en este documento es de exclusiva responsabilidad de los autores y no compromete a la EIA.