

UNIVERSIDAD DE LOS ANDES

PROYECTO DE GRADO

**Puesta en marcha y calibración de un
calorímetro
2277 de ThermoMetric**

Autor:

Juan BARBOSA

Director:

Edgar Francisco VARGAS,

Dr.Sc.

Proyecto de Grado para optar por el título de Químico

Termodinámica de Soluciones

Departamento de Química

2 de diciembre de 2018

Índice general

1. Introducción	1
1. Objetivos	3
1.1. Objetivo general	3
1.2. Objetivos específicos	3
2. Justificación del proyecto	4
3. Metodología	4
2. Armado del equipo	9
3. Control de la temperatura	11
1. Sistema de monitoreo	13
4. Calibración Eléctrica	19
1. Estática	20
2. Dinámica	23
5. Calibración Química	25
1. Preparación de las soluciones de HCl	25
1.1. Solución de HCl 0.25 mM	26
1.2. Solución de KHCO ₃ 0.17 mM	27
2. Sistema de inyección	27
2.1. Control por software	29
2.2. Calibración de la jeringa	30
3. Realización del experimento	31
4. Resultados	32
6. Conclusiones	33

Referencias	35
A. Imágenes de soporte	37
B. Códigos	41
1. Firmware microcontrolador	41
2. Determinación de intervalos estacionarios en la calibración del sistema de monitoreo de temperatura	43
3. Software de Temperatura	44
3.1. Comunicaciones	44
3.2. Interfaz gráfica	48
4. Disco 3D para sensor térmico en el calorímetro	55
5. Determinación de los valores de una calibración estática	56

Índice de figuras

1.1. Componentes exteriores del equipo y partes de los sistemas del calorímetro [12].	5
3.1. Resistencias de década que controlan la temperatura.	11
3.2. Circuito del sistema de monitoreo de temperatura.	13
3.3. Voltajes registrados en la rampa de calentamiento de calibración de los sensores.	15
3.4. Resultados de la calibración del sistema de monitoreo de la temperatura.	16
3.5. Accesorios para el sistema de monitoreo de temperatura.	16
3.6. Efecto de la temperatura del baño externo sobre la temperatura del baño interno.	17
3.7. Efecto de la temperatura del baño externo sobre la temperatura del baño interno.	18
4.1. Interior del cilindro de medición, modificado de [12].	19
4.2. Control del cero y la ganancia del calorímetro	20
4.3. Ejemplos de calibraciones estáticas.	22
4.4. Ejemplo de una calibración dinámica.	23
4.5. Calibraciones dinámica y estática que muestran el correcto funcionamiento del calorímetro.	24
5.1. Determinación de la concentración de una solución de HCl usando valores de densidad reportados en la literatura [21].	26
5.2. Sistema de inyección construido como alternativa al uso de las canulas. .	28
5.3. Configuración de la jeringa en Digitam.	29
5.4. Panel del uso manual del controlador de la jeringa.	30
5.5. Panel de configuración de la jeringa en modo automático.	30
5.6. Curva de calibración de la jeringa usada.	32

A.1. Control térmico del calorímetro. Modificado de [12].	38
A.2. Equivalencia de resistencia y temperatura en una calibración de 1994. . .	39

Índice de tablas

3.1. Valores de las resistencias de década, para una temperatura de 25 °C, según calibración original del calorímetro.	12
3.2. Registro de temperaturas del baño interno en el tiempo para la configuración recomendada por la calibración en la Figura A.2.	12
3.3. Valores de las resistencias de década, para una temperatura de 25 °C, obtenida en octubre.	18
5.1. Densidades y masas medidas para preparar las soluciones con concentraciones 0,25 mM y 0,17 mM para el HCl y KHCO ₃ correspondiente-mente.	28
5.2. Masa dispensada por la jeringa usando diferentes longitudes y un $V_d = 100,000 \mu\text{L}$	31

Introducción

La termodinámica es el estudio de las transformaciones de energía, en la etapa más temprana de esta rama de las ciencias, se pensaba que el calor era una clase de fluido cuya cantidad neta en el universo permanecía siempre constante. El aumento de temperatura de un objeto era explicado a partir de la migración de calor de un objeto a otro [1, 2]. Usando esta teoría del calor como fluido, el ingeniero Sadi Carnot dio origen a la termodinámica con el análisis del problema, sobre cómo generar el mejor y más eficiente motor de vapor en 1824. Posteriormente, Julius von Mayer en 1842 descubrió una equivalencia entre el calor y el trabajo mecánico, trabajo también publicado por James Joule el siguiente año [2]. A raíz de esto hoy se considera el calor como una forma de transferencia de energía [2].

Los resultados de la termodinámica se encuentran implícitos en una serie de declaraciones que reciben el nombre de leyes de la termodinámica, las cuales en conjunto, nos permiten conocer la dirección natural en la que tienen lugar los cambios químicos y físicos en la materia [3]. Históricamente la termodinámica fue desarrollada antes que se tuviera un entendimiento sobre la estructura interna de la materia, además, sus leyes tampoco siguen un orden cronológico [1]. La primera ley surge del trabajo realizado por Mayer en 1842, una de las leyes más importantes de la ciencia en general: la conservación de la energía [1, 2]. La segunda se atribuye a Carnot y Clausius en 1824, y habla sobre la reversibilidad o direccionalidad de los procesos [1].

A pesar que el estudio de las transformaciones de energía parece un tema distante de la química, la termodinámica, estudiada a través de la fisicoquímica, ha demostrado ser de vital importancia tanto en la química como la biología [3]. No sólo permite entender la producción o consumo de energía en las reacciones químicas, además constituye una herramienta fundamental para responder preguntas que se encuentran en el corazón de la bioquímica, por ejemplo sobre cómo fluye la energía en una célula, y qué tan grande puede ser la agrupación de moléculas que forman estructuras complejas como

las células [3]. En otros casos, por ejemplo, existen propiedades de fácil determinación para un analista, como el color, masa y densidad. Sin embargo, hay propiedades que dependen de los enlaces, estructura molecular y naturaleza del material, entre estas se encuentran propiedades termodinámicas de interés químico: capacidad calorífica, entalpía, entropía, etc [4].

Dentro de la fisicoquímica, el estudio y medición de la transferencia de energía en forma de calor se denomina calorimetría, y constituye una de las áreas más viejas de esta rama de la química [5]. Se podría considerar que la historia de esta comienza en junio de 1783, con la presentación de *Memoria del calor* (Mémoire de la Chaleur) por Lavoisier y Laplace a la Academia Francesa [5]. La mayoría de los procesos físicos y químicos están acompañados de absorción o liberación de energía. Lo anterior hace de la calorimetría una técnica con un amplio rango de aplicaciones [6]. Entre estas aplicaciones se tienen titulaciones, flujos, reacciones y estudio de procesos de sorción [4].

El instrumento para realizar este tipo de mediciones se denomina calorímetro, y existen de diversos tipos. Pueden ser clasificados por el tipo de condiciones que imponen al sistema, por ejemplo, volumen constante, temperatura constante, calor constante, etc. Estos calorímetros reciben el nombre de: bombas calorimétricas, isotérmicos, y adiabáticos, correspondientemente [4, 6]. También pueden ser clasificados por el principio de funcionamiento, si compensan los flujos de energía o los acumulan [4]. Finalmente, dependiendo del rango de las potencias medidas, se tienen dos términos comúnmente usados: *microcalorimetría* para el caso de experimentos realizados en el rango de los microvatios [6, 7], mientras que para escalas de nanovatios son usados *nanocalorímetros* [7].

En particular el trabajo a realizar busca armar, poner en marcha y calibrar un microcalorímetro 2277 Thermal Activity Monitor. Esto implica un entendimiento del funcionamiento del calorímetro a nivel interno, desde las partes mecánicas, la etapa electrónica y los principios fisicoquímicos detrás del mismo. En este proyecto, por su naturaleza, se hace imperativo que una vez alcanzado el nivel de operación del calorímetro, se realice una calibración para constatar que el funcionamiento es correcto y concuerda con los resultados obtenidos por la literatura así como con un microcalorímetro moderno disponible en el grupo de investigación *Termodinámica de Soluciones*.

1. Objetivos

1.1. Objetivo general

Poner en funcionamiento el calorímetro 2277 Thermal Activity Monitor con el que se cuenta, en funcionamiento y adicionalmente calibrar el equipo para su uso en las investigaciones activas del grupo **Termodinámica de Soluciones**.

1.2. Objetivos específicos

- Ensamblar el equipo 2277 Thermal Activity Monitor.
- Realizar el cableado y conexiones electrónicas pertinentes al mismo.
- Calibración eléctrica, determinación de las señales de entrada y salida, flujo de las bombas hidráulicas y temperatura del baño.
- Calibración química, determinación de la entalpía molar, energía libre de Gibbs, entropía, y constante de equilibrio, del acomplejamiento del catión bario con éter 18-corona-6.

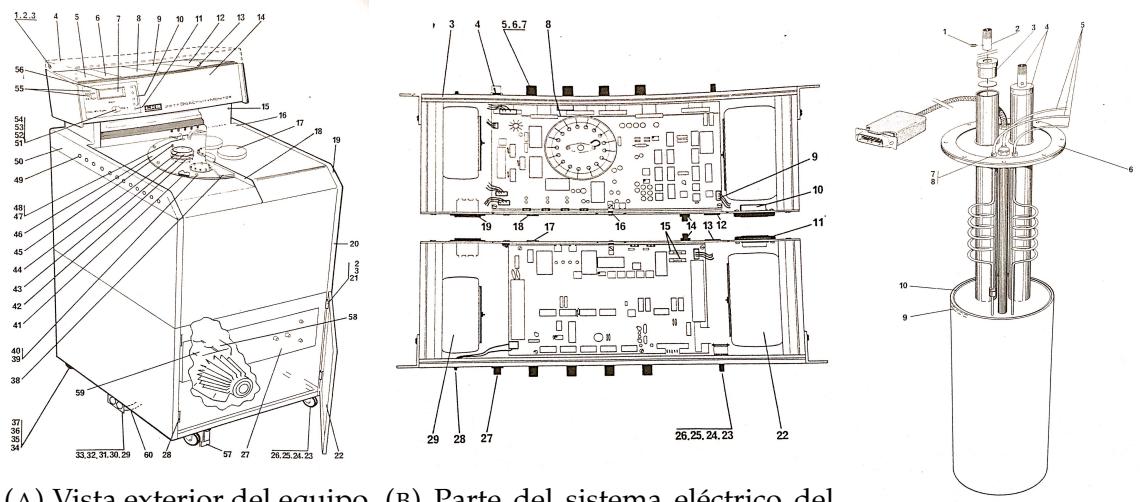
2. Justificación del proyecto

En general la calorimetría permite una gran variedad de análisis, muchos de ellos cuentan con aplicaciones industriales, comerciales, biológicas y químicas, permitiendo el entendimiento de las interacciones moleculares en soluciones [8]. Una ventaja de la calorimetría es que no es específica, ni invasiva además de no depender de las propiedades electroquímicas y ópticas de un sistema dado, siendo esto de vital importancia para las investigaciones de procesos biológicos, el estudio del crecimiento bacteriano y para la detección de compuestos biológicos [9]. Por otro lado la calorimetría también permite el estudio de la termodinámica en sistemas de absorción, bien sea por la universalidad de la absorción física en las superficies como catalizadores, o en procesos industriales como la separación de mezclas de gases [10]. Finalmente la calorimetría es el método clásico para la determinación de propiedades termodinámicas en las muestras, entre estas se encuentran la capacidad calorífica, la entalpía, entropía y energía libre de Gibbs, las cuales constituyen el punto de partida de gran cantidad de estudios teóricos, desarrollos y producción industrial de un compuesto químico [4, 11].

El calorímetro 2277 Thermal Activity Monitor con el que cuenta el grupo de investigación se encuentra desarmado, a la espera de su ensamblaje y puesta en funcionamiento. Este instrumento permite monitorear una gran variedad de reacciones químicas y bioquímicas, lo anterior debido a su capacidad de cuantificar procesos exotérmicos y endotérmicos. Estas reacciones pueden ser estudiadas en el rango de 5 - 80 °C [12]. Este rango de temperaturas se debe al uso de un baño termostatado de 25 litros. Cuatro balones de medición independientes se encuentran sumergidos en este baño, permitiendo medidas con desviaciones de temperatura inferiores a $\pm 2 \times 10^{-4}$ °C, alcanzando de esta manera medidas en el rango de microvarios [12]. Es por esta razón que poner en marcha y calibrar el equipo con el que cuenta el grupo de investigación resulta una contribución importante para el mismo, así como para el Departamento de Química en general.

3. Metodología

Para el ensamblaje del calorímetro se cuenta con el catálogo de partes, y el manual de instrucciones. En el primero se detallan y enumeran las partes del equipo, en la Figura 1.1 se muestran ejemplos de los distintos sistemas con los que el calorímetro cuenta [12].



(A) Vista exterior del equipo con sus componentes. (B) Parte del sistema eléctrico del calorímetro. (C) Celda de medición.

FIGURA 1.1: Componentes exteriores del equipo y partes de los sistemas del calorímetro [12].

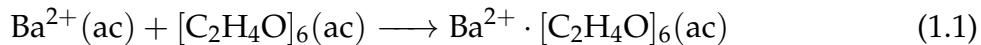
Usando esta información se llevará a cabo el ensamblaje del calorímetro así como las conexiones eléctricas. A nivel electrónico, la liberación o absorción de energía térmica se mide usando una celda Peltier, las cuales tienen una respuesta proporcional en voltaje, de esta forma se obtiene una señal eléctrica a partir de pequeñas variaciones de temperatura en la celda de reacción. La señal eléctrica es procesada por el sistema digital del instrumento [12]. De esta forma y siguiendo las instrucciones es posible realizar una calibración eléctrica del sistema.

La calibración química puede realizarse de distintas maneras. En principio cualquier reacción que pueda ser llevada a cabo en un calorímetro, con condiciones controladas y cuyas propiedades termodinámicas sean bien conocidas, puede ser usada como una reacción de calibración [6]. Sin embargo es recomendable que los reactivos involucrados en estas reacciones no requieran de algún tipo de purificación que pueda alterar el análisis [6].

Se propone realizar una calorimetría de titulación como método de calibración química, lo anterior es importante porque una calibración eléctrica puede dar lugar a distribuciones de temperatura distintas en la celda de reacción, o generar patrones de flujo de calor distintos a una reacción química, esto puede generar errores sistemáticos en las medidas realizadas [6]. La titulación es una técnica importante en la medida que permite la determinación simultánea de la entalpía molar y la constante de equilibrio de la reacción, por lo cual también se obtienen el cambio en la energía estándar de Gibbs y el

cambio en la entropía estándar [6].

Para esta calibración es necesario contar con soluciones acuosas de cloruro de bario (BaCl_2) y éter 18-corona-6 ($[\text{C}_2\text{H}_4\text{O}]_6$) [6, 13, 14]. La reacción de acomplejamiento es 1:1, de la siguiente forma:



Dichas soluciones deben encontrarse en el rango de concentraciones de 1 mM a 10 mM para el éter y 10 mM hasta 100 mM para el cloruro de bario. Dentro de este rango, existe evidencia experimental que muestra que no hay una variación significativa de las cantidades termodinámicas medidas [6, 14]. Considerando el volumen de las celdas con las que cuenta el equipo, se propone usar 1.5 mL de la solución de éter. Las soluciones deben ser desgasificadas y termostatadas a 25 °C, con el objetivo de limitar el error experimental introducido por diferencias de temperatura causadas por burbujas [13-15].

En el calorímetro el baño debe encontrarse a 25 °C y deben usarse dos de las cuatro celdas disponibles. La primera se considera como referencia y debe contener al disolvente únicamente, en este caso 1.5 mL de agua [12]. En la segunda se adiciona el mismo volumen de la solución de éter, una vez la línea base del calorímetro se encuentra estable, se adicionan una a una 30 alícuotas de la solución de cloruro de bario, con tiempos entre inyección mayores a $\tau = 6$ minutos, tanto a la celda de referencia como a la de reacción [14, 15]. Cada inyección modifica la temperatura de la celda respecto a la referencia, razón por la cual las termopilas Peltier, deberán compensarla, la energía requerida por las celdas se cuantifica en función del tiempo, esto da lugar a una gráfica de potencia en función del tiempo [15]. Para cada inyección, es posible calcular el cambio en la energía integrando la potencia usada por el sistema. La gráfica de los cambios energéticos en función de la fracción molar ($[\text{Ba}^{2+}]/[\text{éter}]$) permite obtener las propiedades termodinámicas [14, 15]. La diferencia entre el valor mínimo de la isotérmica y la línea base de esta corresponde con la entalpía molar estándar ΔH_m . La derivada de la isotérmica en el punto equimolar, da lugar a la constante de afinidad K_a . Finalmente con esta es posible determinar los cambios en la energía libre de Gibbs y la entropía.

$$\Delta G_m = -RT \ln K_a = \Delta H_m - T\Delta S_m \quad (1.2)$$

Para determinar la capacidad calorífica es necesario realizar el mismo experimento a distintas temperaturas, dado que:

$$C_{p,m} = \left(\frac{\partial H_m}{\partial T} \right)_p \quad (1.3)$$

En el caso de esta titulación los valores reportados por IUPAC corresponden con [6]:

- $\Delta H_m = -(31,42 \pm 0,20) \text{ kJ mol}^{-1}$
- $K_a = (5,90 \pm 0,20) \times 10^3 \text{ mM}$
- $C_{p,m} = 126 \text{ J K}^{-1} \text{ mol}^{-1}$

Armado del equipo

Control de la temperatura

El sistema cuenta con dos calentadores de precisión al interior del baño termostata-
do de 25 litros. Estos calentadores junto con el baño externo determinan la temperatura
de trabajo del equipo. El baño externo se usa con el objetivo de mantener siempre ac-
tivos los calentadores internos en algún nivel de potencia intermedio, pues el baño
externo, al tener una temperatura inferior agregar una carga permanente a los calen-
tadores internos [12]. Para monitorear la temperatura del baño el calorímetro cuenta
con dos termistores, el primero para trabajar a temperaturas inferiores a 50 °C y el se-
gundo para temperaturas superiores a este valor. La señal generada por uno de estos
termistores se compara con un valor de resistencia definido por el usuario. La resis-
tencia y por ende la temperatura del equipo se fijan usando cuatro resistencias de década,
esto es que cada resistencia es una potencia de 10 menor que la anterior, las cuales se
encuentran en la parte inferior del calorímetro, en la Figura 3.1 corresponden A, B, C,
y D, de esta manera se pueden realizar experimentos a temperaturas en el rango de 5
a 80 °C. El diagrama del sistema de control térmico del calorímetro se muestra en el
Apéndice A como Figura A.1.

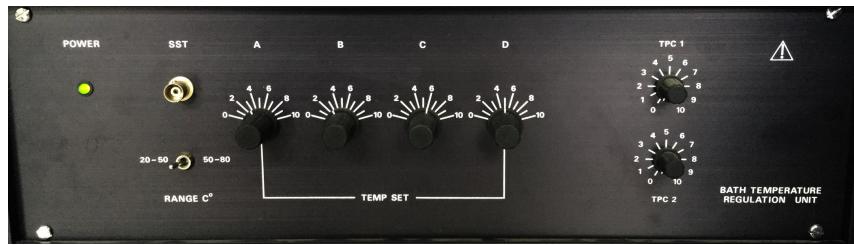


FIGURA 3.1: Resistencias de década que controlan la temperatura.

Para determinar la equivalencia de resistencia y temperatura el calorímetro cuenta
con una tabla que relaciona los valores de resistencia y temperatura de baño externo
que dan lugar a una temperatura constante en el baño interno. Dicha tabla se muestra
en el ?? como Figura A.2, sin embargo, el uso de esa tabla está limitado a temperaturas

ambiente superiores a 20 °C por lo cual, en el caso particular de Bogotá rara vez se cumplen. Lo anterior es relevante dado que la resistencia eléctrica depende de la temperatura, en el caso de materiales metálicos, la resistencia aumenta con la temperatura, y para semiconductores se tiene el efecto contrario [16].

Una vez realizadas las conexiones eléctricas del calorímetro para su funcionamiento a 110 VAC, y antes de realizar una calibración química fue necesario estabilizar el equipo a una temperatura de 25 °C, puesto que el objetivo de la calibración química es contrastar los datos obtenidos con el calorímetro con los reportados en la literatura, los cuales por convención se reportan a estas condiciones. Para alcanzar esta temperatura se probó con las condiciones reportadas en la [Figura A.2](#), en donde los valores de cada resistencia de década se muestra a continuación.

TABLA 3.1: Valores de las resistencias de década, para una temperatura de 25 °C, según calibración original del calorímetro.

Baño interno (°C)	A ($10^4 \Omega$)	B ($10^3 \Omega$)	C ($10^2 \Omega$)	D ($10^1 \Omega$)	Baño externo (°C)
25,0	3	2	4	9	22,0

La temperatura del equipo fue monitoreada por cuatro días, con al menos un registro diario, los resultados se muestran en la [Tabla 3.2](#). En ella se puede observar que cuando inicialmente se creía que la temperatura del baño estaba cerca de estabilizarse a 25 °C, en realidad estaba en aumento de la temperatura, y cuando se creía que finalmente el calorímetro se había estabilizado en 26,86 °C el sistema se encontraba oscilando como lo muestra la última temperatura registrada.

TABLA 3.2: Registro de temperaturas del baño interno en el tiempo para la configuración recomendada por la calibración en la [Figura A.2](#).

Fecha (DD/MM HH:MM)	Temperatura (°C)
07/09 11:13	25,14
07/09 11:50	25,19
10/09 09:43	26,64
11/09 09:22	26,86
11/09 10:00	26,86
12/09 08:30	26,86
12/09 10:30	26,86
12/09 15:48	26,64

Sin embargo, esta no fue la única configuración probada, pues también se trató de estabilizar el equipo usando: 2977 (A,B,C,D), que dio lugar a las siguientes temperaturas de baño interno 25,44 °C, y 28,38 °C luego de 5 días. De manera similar se probaron

3000, 3021, 3040, 3100, 3121, 3140, 3160 ninguna de las cuales presentó una temperatura estable. Debido a la dificultad de estabilizar el equipo a una temperatura determinada, se decidió construir un sistema de monitoreo automatizado para facilitar esta tarea, de esta manera se podría saber el histórico de una configuración, si el sistema está en calentamiento, estable, u oscilando.

1. Sistema de monitoreo

De todas las acciones que se hicieron a lo largo del proyecto, la estabilización de la temperatura es la que llevó más tiempo. Esto se debió a que existen dos variables que determinan una temperatura en el calorímetro, por un lado la configuración de las resistencias de década, y el baño externo, además se debe tener en cuenta el tiempo de estabilización del calorímetro, el cual puede tomar cerca de 6 horas en completarse. Para monitorear el estado de la temperatura del baño interno se construyó un circuito con tres sensores de temperatura, los sensores LM39 fueron elegidos debido a que presentan una respuesta lineal en voltaje, y tienen un rango de trabajo de 2 °C a 120 °C [17]. Cada uno de estos sensores fue conectado a tres cables como se muestra en la Figura 3.2b, y posteriormente fue cubierto con silicona y un cable termoencogible para aislar el sensor del agua.

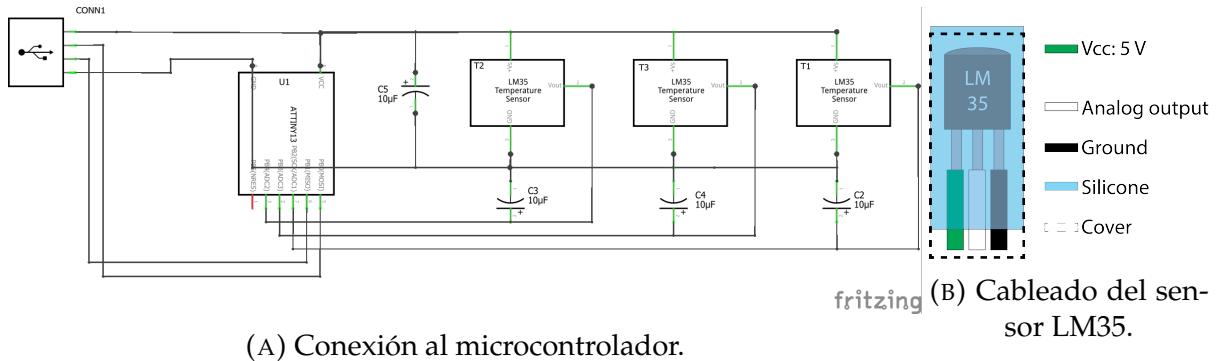


FIGURA 3.2: Circuito del sistema de monitoreo de temperatura.

Los sensores, uno para el baño interno, baño externo y temperatura ambiente, a su vez fueron conectados a un microcontrolador ATtiny 13 [18] (Figura 3.2a), el cual hace las veces de central de procesamiento que unifica las señales de los tres sensores y las envía a un computador usando el protocolo de comunicación UART (*Universal Asynchronous Receiver-Transmitter* por sus siglas en inglés). Para convertir las señales de los sensores a un valor digital se usa el ADC (*Analog to Digital Converter* por sus

siglas en inglés) con el que cuenta el microcontrolador. El ADC compara el voltaje de la señal de entrada con un valor de referencia de voltaje y determina que fracción de voltaje corresponde la señal de entrada, este valor relativo (k) se encuentra entre 0 y 2^n (n es el número de bits del ADC), donde 0 implica un valor igual al de la tierra del circuito, y 2^n tiene lugar cuando la señal de entrada tiene el mismo valor que el voltaje de referencia, los valores intermedios son lineales con los límites, de tal forma que es posible reconstruir el valor de la señal en voltios mediante la siguiente ecuación:

$$V = \left(\frac{k}{2^n} \right) V_{\text{ref}} \quad (3.1)$$

En el caso del ATtiny 13, se cuenta con 10 bits en el ADC con cuatro canales, tres de los cuales son usados para los sensores. Como voltaje de referencia se usa un regulador interno del microcontrolador cuyo valor es 1,1 V [18]. Un voltaje de salida del sensor de 1,1 V en el sensor es cercano a 110 °C y 0,0 V cercano a 0 °C pues el fabricante asegura que por cada grado Celsius el voltaje de salida aumenta en 10 mV. El número de bits del ADC permite una resolución cercana a los 0,1 °C, puesto que se tiene que 110 °C pueden ser divididos en $2^{10} = 1024$ niveles distintos. Sin embargo, para lograr una resolución de 0,01 °C se usó una técnica conocida como sobremuestreo [19]. Esta técnica tiene sus orígenes en el teorema de Nyquist, el cual establece que para reconstruir una señal es necesario muestrear una señal con el doble de la frecuencia de la frecuencia máxima en la señal [20]. En el caso de la temperatura de los baños de agua y ambiente, se espera que la frecuencia máxima sea relativamente baja, del orden del tiempo de respuesta del sensor el cual es del orden de segundos. En particular se tiene que para w bits adicionales en el ADC es necesario muestrear 4^w veces adicionales la señal. Dado que se tiene un rango cercano a los 100 grados y se quiere una resolución de 0,01 °C es necesario diferenciar 10.000 niveles distintos de voltaje, lo cual implica al menos 14 bits ($2^{14} = 16.384$), sin embargo, se eligen muestrear 4096 veces con frecuencia de 150 kHz, lo cual equivale a 16 bits en el ADC.

Los sensores fueron calibrados usando el baño externo, y el termómetro de mercurio previamente usado en la verificación de la lectura de temperatura del sensor interno del calorímetro. Para esto se realizó una rampa de calentamiento discreta de 10 °C hasta 50 °C con pasos de 5 °C y tiempos de estabilización de 20 minutos. Se escribió un algoritmo que calcula el valor promedio y desviación estándar de una lectura de voltaje para los intervalos estacionarios de la rampa. Este algoritmo realiza lo siguiente:

1. Para cada instante de tiempo considera el promedio de voltaje de los tres sensores.

2. Saca el valor absoluto de la derivada de los datos en el tiempo.
3. Aplica un filtro de medianas con kernel 507 sobre los datos anteriores (líneas continuas rojas en la [Figura 3.3](#)).
4. Los puntos con valores menores a 0,05 se les asigna el valor 0, para los otros valores se asigna 0,5 (lineas segmentadas en la [Figura 3.3](#)).
5. Se considera que los puntos donde el valor anterior corresponde a 0 son estables, sobre estos intervalos calcula el promedio y desviación estándar de cada sensor.

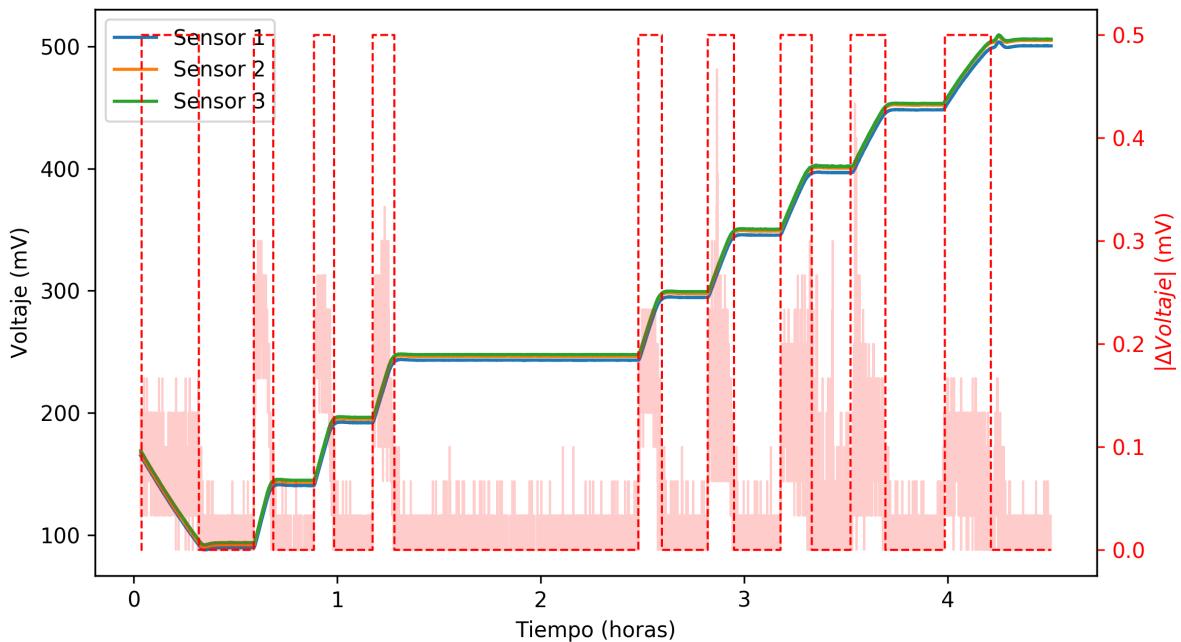
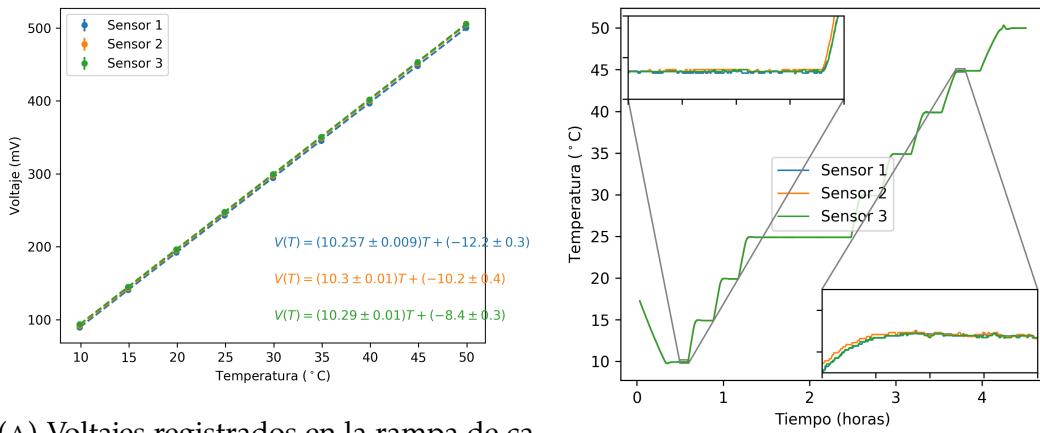


FIGURA 3.3: Voltajes registrados en la rampa de calentamiento de calibración de los sensores.

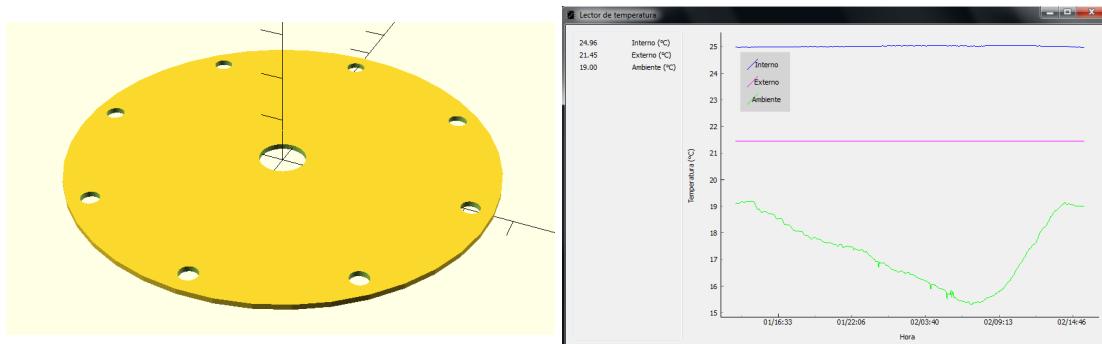
Los valores de voltaje de los intervalos estacionarios de la rampa fueron relacionados con la temperatura del termómetro de mercurio en la [Figura 3.4](#), en donde se observó una tendencia lineal, cuya pendiente es cercana a 10 mV, lo cual concuerda con lo reportado por el fabricante [17]. A partir de las relaciones obtenidas con la regresión lineal, se reconstruyó la [Figura 3.3](#) en función de la temperatura.



(A) Voltajes registrados en la rampa de calentamiento de calibración de los sensores. (B) Voltajes registrados en la rampa de calentamiento de calibración de los sensores.

FIGURA 3.4: Resultados de la calibración del sistema de monitoreo de la temperatura.

Una vez calibrados los sensores, se escribió un programa en Python para facilitar la visualización de los datos en tiempo real, cuyo código se encuentra en la [Sección 3](#). Además, para introducir el sensor usado en el baño interno del calorímetro sin que este estuviera expuesto directamente a la temperatura ambiente, se imprimió una tapa en 3D con un filamento de PLA (ácido poliláctico) para usar uno de los espacios reservados para un cilindro de medición del calorímetro.

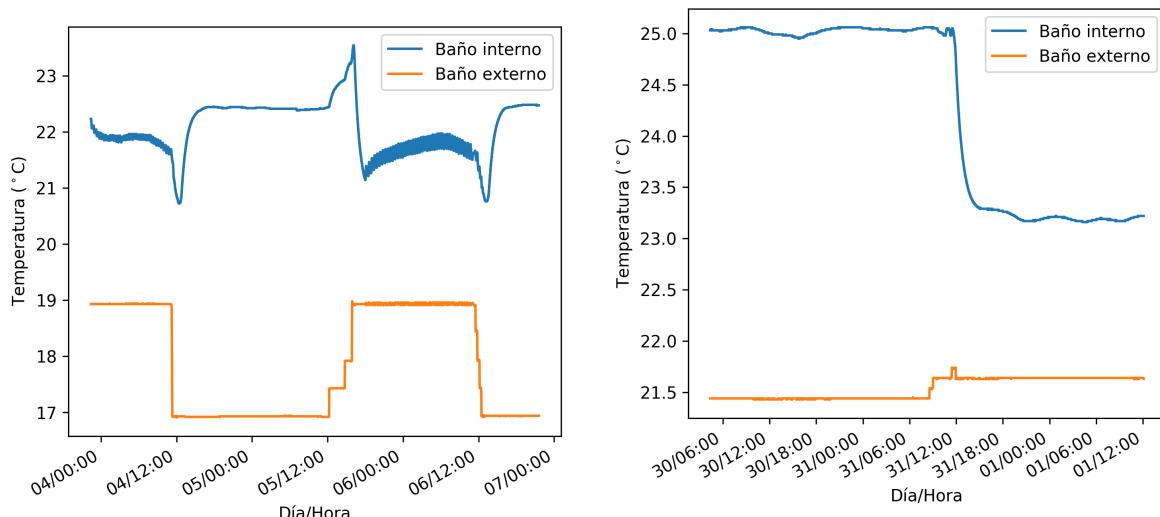


(A) Disco impreso para facilitar el acceso del sensor al baño térmico del calorímetro. (B) Interfaz gráfica para el monitoreo de la temperatura en tiempo real.

FIGURA 3.5: Accesorios para el sistema de monitoreo de temperatura.

Con el monitoreo de temperatura establecido fue posible determinar el efecto del baño térmico externo sobre la temperatura del calorímetro. Por ejemplo, para el caso de las oscilaciones que se muestran en la [Tabla 3.2](#), se encuentran relacionadas con la

temperatura del baño externo, para el caso de temperaturas muy bajas del baño externo se observa que la temperatura del baño interno se encuentra estable, sin embargo los calefactores se encuentran trabajando fuera del rango completamente prendidos, y a temperaturas del baño externo intermedias, la temperatura del baño interno oscila. Esto se puede entender si se considera que los sistemas térmicos presentan inercia, por lo cual mientras el sensor de temperatura registra una temperatura menor a la deseada, aplicará sobre los calentadores potencia, una vez se sobre pase este valor los calentadores se apagarán pero el agua de los alrededores se continuará calentando por algún tiempo, posteriormente el sistema detectará que la temperatura ha disminuido prendiendo nuevamente los calentadores haciendo que la temperatura en general oscile como se muestra en la [Figura 3.6a](#).



(A) Oscilaciones debidas a una temperatura del baño externo intermedia.
 (B) Caída de la temperatura del baño interno debido al aumento de 0,20 °C en el baño externo.

FIGURA 3.6: Efecto de la temperatura del baño externo sobre la temperatura del baño interno.

En la [Figura 3.6b](#) se observa cómo la temperatura del calorímetro puede variar bruscamente cuando los calentadores trabajan fuera de rango. En el caso de esta figura, la temperatura del calorímetro se encontraba estable, el calentador fino siempre en el rango de trabajo, mientras que el precalentador ocasionalmente salía del rango de trabajo, sin embargo, al aumentar la temperatura del baño 0,20 °C el equipo disminuyó su temperatura y no volvió aumentar.

En general la [Figura 3.6](#) muestra la dificultad de estabilizar el equipo a una temperatura determinada, puesto que se debe encontrar la pareja correcta de valores en las resistencias de décadas, junto con la temperatura del baño externo, siendo los dos particularmente sensibles al valor del otro. Pues como fue mencionado anteriormente, aunque en la [Figura 3.6b](#) el sistema parezca estable, los calentadores internos están funcionando fuera del rango.

Finalmente, se logró determinar la configuración en donde los calentadores se encuentran en el rango de trabajo y el equipo es estable a $25,02 \pm 0,03$ °C, esta configuración se muestra en la [Tabla 3.3](#). Además, el sensor de temperatura ambiente, probó ser útil, pues la pequeña variación en la temperatura interna del calorímetro se encuentra correlacionada con la temperatura ambiente, pues en la [Figura 3.7](#) se observa que los máximos de temperatura interna ocurren en los mínimos de temperatura ambiente.

TABLA 3.3: Valores de las resistencias de década, para una temperatura de 25 °C, obtenida en octubre.

Baño interno (°C)	A (10^4 Ω)	B (10^3 Ω)	C (10^2 Ω)	D (10^1 Ω)	Baño externo (°C)
$25,02 \pm 0,03$	3	3	6	5	21,5

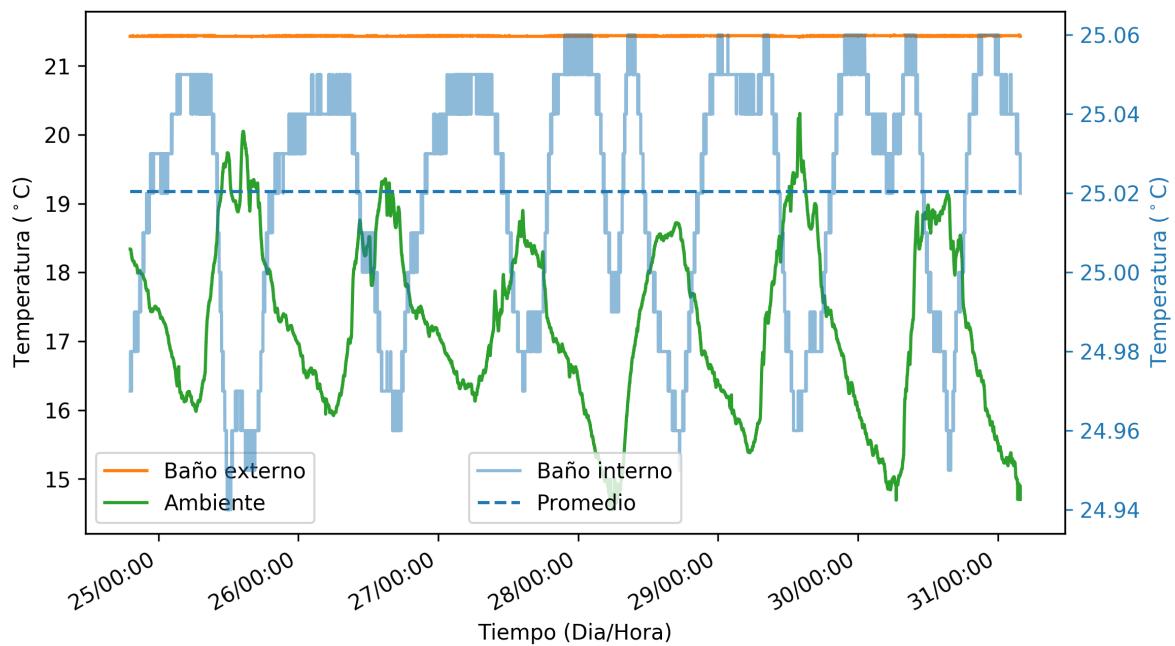


FIGURA 3.7: Efecto de la temperatura del baño externo sobre la temperatura del baño interno.

Calibración Eléctrica

Con el objetivo de asegurar que la información registrada por el calorímetro corresponde con un valor específico de potencia, es necesario realizar una calibración eléctrica, la cual es específica para cada canal de medición. Para ello, el equipo cuenta con una resistencia de precisión que envuelve el contenedor de la celda y permite simular, lo mejor posible, la energía liberada en forma de calor cuando una reacción química tiene lugar en la celda, la ubicación de esta resistencia se muestra en la [Figura 4.1 \[12\]](#).

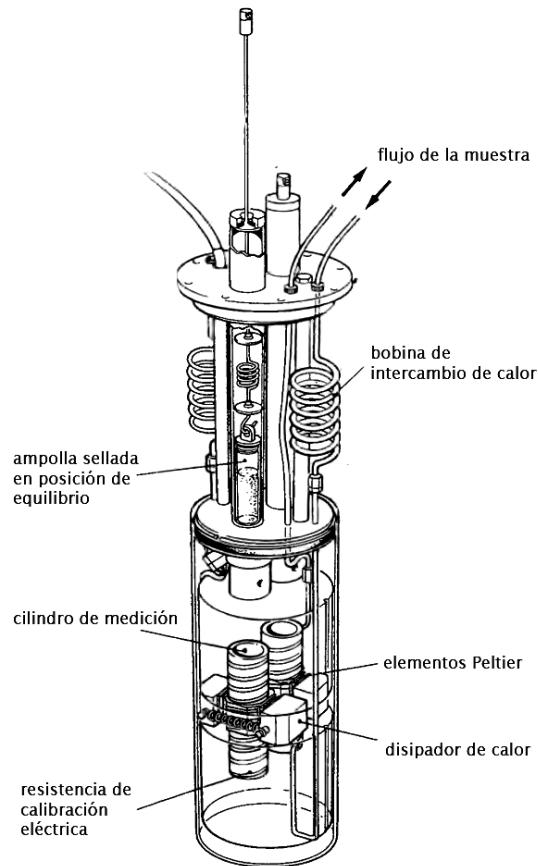


FIGURA 4.1: Interior del cilindro de medición, modificado de [12].

Para calibrar el sistema se realizan lecturas consecutivas de la línea base, esto es la potencia que registra el calorímetro en ausencia de perturbación, y a partir de esto se ajusta el cero del canal, pues se busca que la línea base sea lo más cercana a $0 \mu\text{W}$. Posteriormente, se aplica una potencia conocida sobre la resistencia y hacen lecturas de la potencia registrada por el calorímetro, a partir de esto se ajusta la ganancia, hasta que la potencia aplicada y la registrada tengan el valor más cercano posible. La potencia conocida debe concordar con el valor de amplificación del selector RANGE en la [Figura 4.2](#).



FIGURA 4.2: Control del cero y la ganancia del calorímetro

Existen dos tipos de calibraciones, en la primera se modifican los valores de cero y ganancia en el equipo directamente, mientras que en la segunda los valores corresponden a ajustes en el software. Estos métodos de calibración reciben el nombre de estática y dinámica correspondientemente, y se debe asegurar que al momento de realizar una calibración dinámica, una calibración estática haya sido realizada previamente, pues los valores registrados por software son digitales (discretos), entonces la resolución de estos será determinada por la ganancia del circuito analógico, lo cual sólo es posible de controlar en el calorímetro directamente.

1. Estática

En la calibración estática, se debe manipular el equipo directamente. En primer lugar como fue mencionado anteriormente se ajusta el cero del canal en ausencia de perturbaciones. Para esto se manipula el potenciómetro de diez vueltas del lado izquierdo el cual está marcado como ZERO en la [Figura 4.2](#), girando en sentido horario el valor

de potencia registrado será cada vez menor. Una vez el cero ha sido ajustado, se aplica sobre la resistencia una potencia conocida, por un tiempo determinado por el usuario, y se espera que la señal registrada por el calorímetro en su estado estacionario coincida con el valor aplicado de potencia. Para esto se debe esperar cerca de 20 minutos para que la resistencia alcance el estado estacionario y posteriormente se manipula el potenciómetro derecho (FINE en la [Figura 4.2](#)), donde giros en el sentido horario aumentan el valor de potencia registrado por el equipo, de esta forma se configura la ganancia del equipo.

Para activar la resistencia es posible hacerlo manualmente a través de los botones blancos que se observan en la [Figura 4.2](#). En primer lugar se selecciona el canal sobre el cual se quiere realizar la calibración, para esto se oprime el botón correspondiente a cada canal (canal 1, botón izquierdo) del panel SELECT, esto ocasionará que la luz asociada a ese canal comience a oscilar. Posteriormente, si la luz continúa en este estado se debe volver a presionar el botón de SELECT asociado a este canal. El lado de la calibración A ó B se selecciona oprimiendo el botón CAL A/B donde la luz prendida indica A, y la potencia oprimiendo los botones de RANGE cuyo producto da el valor de potencia deseado en μW . Finalmente, para activar la calibración se oprimen de manera simultánea el botón de SELECT del canal y el botón CAL A/B. Una vez se haya calibrado el estado estacionario, la resistencia se apaga oprimiendo el botón del canal en SELECT y posteriormente, al mismo tiempo este botón y CAL A/B.

Otra manera de hacerlo es a través del software Digitam. En donde se realiza un método experimental que contenga la calibración eléctrica, este método ya se encuentra creado bajo el nombre staticCalibration.tam, el cual:

- Mide por 30 minutos la línea base.
- Aplica 300 μW sobre el lado B del canal 1 (referencia) por 35 minutos.
- Espera por 45 minutos a que la potencia se estabilice en la línea base.

Resultados de calibraciones estáticas se muestran en la [Figura 4.3](#). En particular, para el caso de la [Figura 4.3a](#) se tiene que el cero está bien ajustado, sin embargo el valor de la amplitud no corresponde, pues el sensor se satura a un valor de 213,9 μW , sin una transición suave al estado estacionario. El efecto contrario se observa en la [Figura 4.3b](#), pues la potencia aplicada corresponde con la leída, sin embargo, la línea base se encuentra a -80,1 μW .

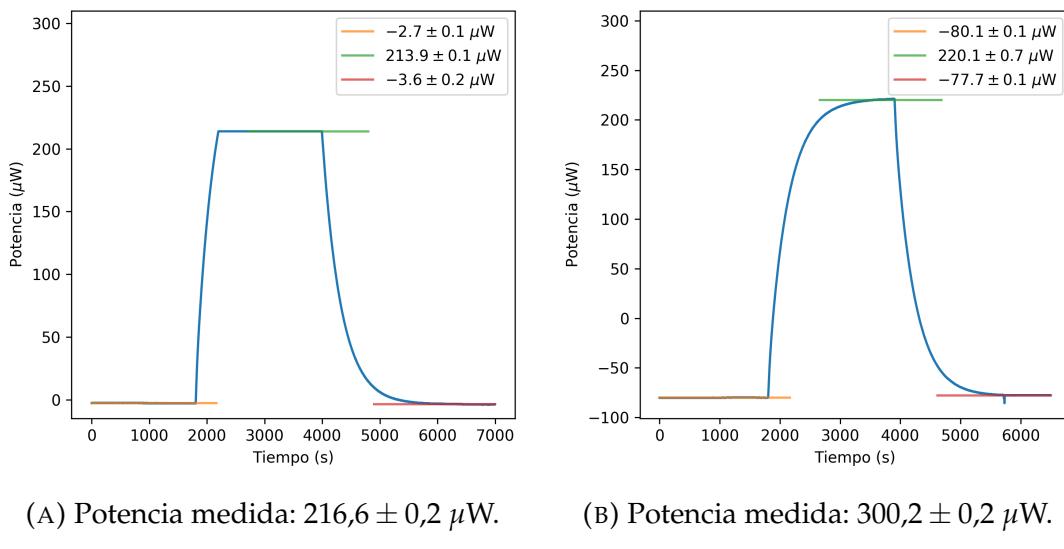


FIGURA 4.3: Ejemplos de calibraciones estáticas.

Para determinar las desviaciones a la línea base así como el valor del estado estacionario se implementó un algoritmo que se muestra en la [Sección 5](#) y que realiza lo siguiente:

1. Toma el valor absoluto de la derivada de la potencia.
2. Realiza un filtro de medianas sobre los resultados anteriores, con un kernel de 101 puntos.
3. A los puntos con valores mayores o iguales a 0,5 se les asigna un valor de 1, y los restantes 0.
4. Se toma la derivada de los valores anteriores, de tal forma que aumentos corresponden con 1, y caídas con -1. Para el caso del estado estacionario y la última línea base, se toman el último 30 % de los datos, en donde ya se encuentran estables estos estados. Las fronteras de estos valores determinan las fronteras de las líneas base y el estado estacionario.
5. El promedio y desviación estándar se toma sobre cada uno de estos intervalos.

2. Dinámica

En la calibración dinámica el sistema registra la línea base por 1 minuto, posteriormente aplica 40 % de la potencia seleccionada sobre la resistencia, dando lugar a una pendiente constante por tres minutos, posteriormente, incremente nuevamente la potencia, mientras registra las pendientes observadas. Finalmente, la potencia se lleva hasta el 95 % y se obtiene un corto estado estacionario, de esta forma el software ajusta los valores de ganancia y cero del canal [12].

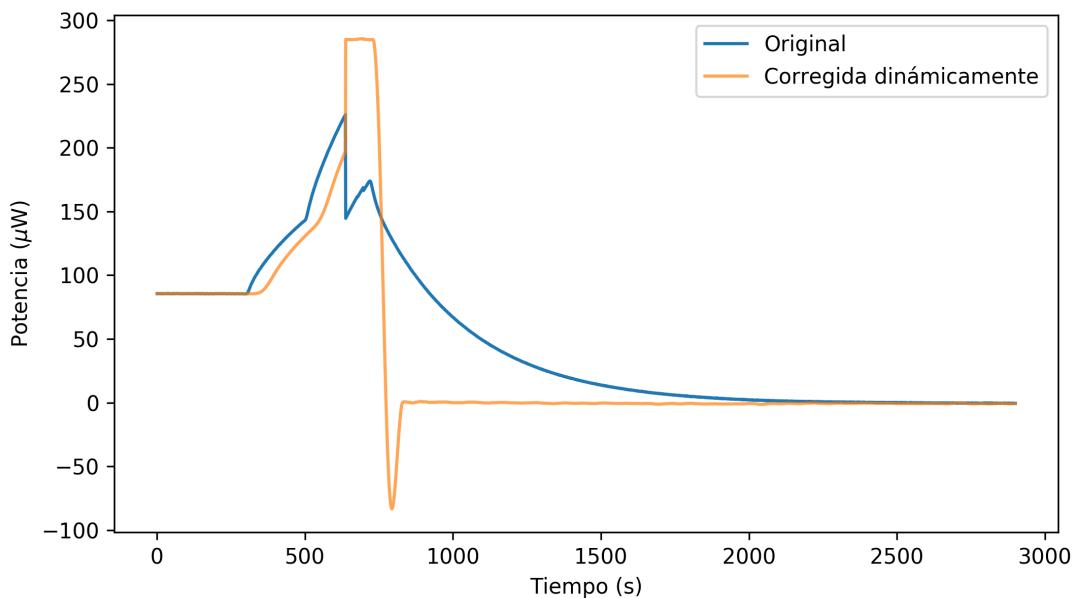


FIGURA 4.4: Ejemplo de una calibración dinámica.

En la Figura 4.4 se muestra un ejemplo de una calibración dinámica. En ella, la línea base se encuentra en $85,5 \mu\text{W}$, luego de la calibración la línea base se estabiliza en $-0,4 \mu\text{W}$. La calibración dinámica permite caracterizar dos constantes que describen la inercia que presenta el calorímetro. Esto es que existe un tiempo de respuesta del equipo ante una perturbación. Luego de la calibración dinámica es posible tener una señal corregida dinámicamente, en donde se puede observar el estado estacionario al 95 % de la potencia aplicada, y los cambios de pendiente que fueron antes mencionados.

Finalmente, la calibración dinámica asegura la reproducibilidad de un experimento, como se muestra en la Figura 4.5, en donde luego de haber realizado una calibración estática, se hace una dinámica y posteriormente para confirmar sus resultados se ejecuta una nueva calibración estática, en donde no se modifican ninguno de los parámetros en

el calorímetro. En ella se observa como el inicio de las curvas de la calibración dinámica no coinciden, sin embargo, la parte final de estas se solapa, haciendo que las medidas sean reproducibles.

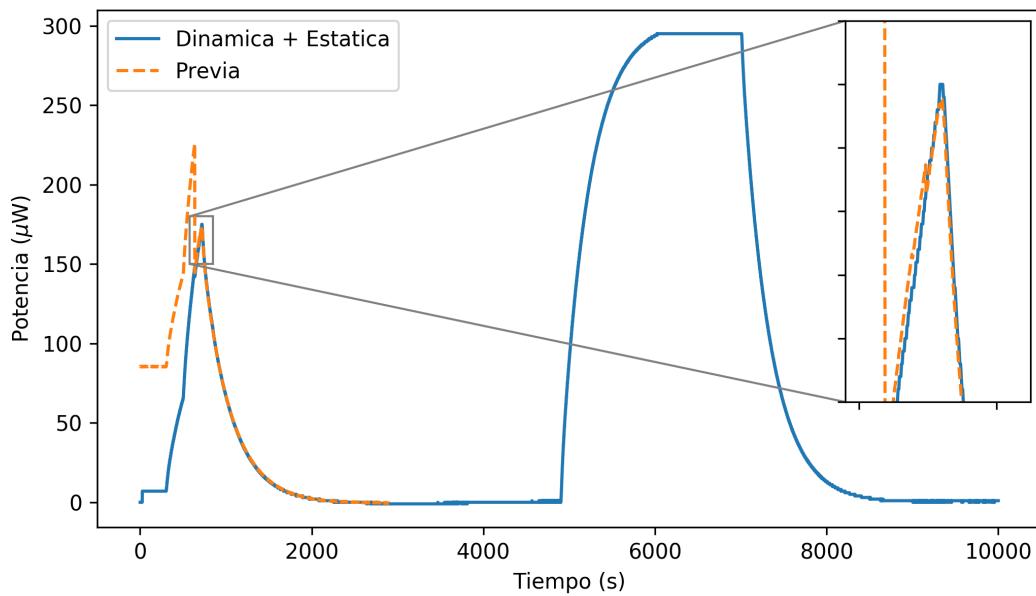


FIGURA 4.5: Calibraciones dinámica y estática que muestran el correcto funcionamiento del calorímetro.

Calibración Química

La calibración química consiste en contrastar las propiedades termodinámicas de un sistema químico, obtenidas usando el calorímetro con aquellas reportadas en la literatura. Para esto se estudia la reacción del ácido clorhídrico con bicarbonato de potasio. Esta reacción presenta varias ventajas: por un lado hace uso de reactivos de fácil acceso, es una reacción con esqueometría 1:1, ha sido estudiada previamente, y además, es usada en calibraciones de equipos calorimétricos., como el calorímetro de titulación NanoITC de *TA instruments*.

1. Preparación de las soluciones de HCl

Para la preparación de las soluciones, se hace uso de forma sistemática de una balanza Ohaus Analytical Plus (AP250D) y un densímetro Anton Paar DSA5000M. La balanza presenta incertidumbres de 1×10^{-5} y 1×10^{-4} g dependiendo del rango usado, para el primer caso la masa medida debe ser inferior a 80 g y en el segundo no puede superar los 250 g, siendo esta la capacidad máxima de la balanza. En el caso del densímetro, son necesarios volúmenes cercanos a 2 mL de una muestra líquida, con esto el equipo determina la densidad de la sustancia con incertidumbres de 1×10^{-6} g/cm⁻³ para una temperatura determinada por el usuario. Además, en el proceso de preparación se hace necesaria la toma de dos alícuotas de 30 μ L y 170 μ L, para el HCl y KHCO₃ correspondientemente, por lo cual se usa una micropipeta pipet4u Performance con rango de 20 a 200 μ L.

1.1. Solución de HCl 0.25 mM

Para determinar la concentración de una solución de 1,0 mL de HCl concentrado en 25,0 mL de agua tipo 1, se midió la densidad de la solución, posteriormente usando como referencia los datos reportados en la literatura fue realizada una regresión lineal que permitió relacionar la concentración con la densidad de la solución ρ [21]. De esta manera se estableció el valor de la concentración en: $3,02 \pm 0,05\%$ (fracción de masa $[w_t]$), donde la incertidumbre se obtiene de la pendiente (m) e intercepto (b) de la regresión lineal que se muestra en la [Figura 5.1](#).

$$\delta[w_t] = \sqrt{\left(\frac{\rho - b}{m^2} \delta m\right)^2 + \left(\frac{\delta b}{m}\right)^2 + \left(\frac{\delta \rho}{m}\right)^2} \quad (5.1)$$

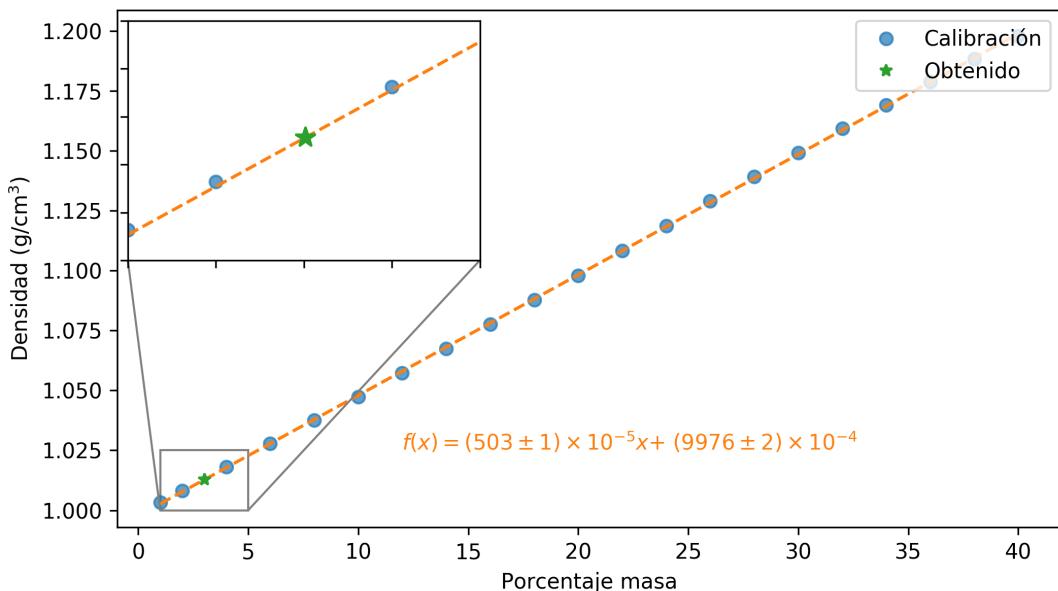


FIGURA 5.1: Determinación de la concentración de una solución de HCl usando valores de densidad reportados en la literatura [21].

Para obtener la concentración $0,84 \pm 0,04\text{ M}$, se usa la siguiente ecuación, la cual relaciona la concentración en fracción de masa con la molaridad $[M]$:

$$[M] = 10 \frac{[w_t]\rho}{m_m} \quad m_m \text{ la masa molecular del HCl} \quad (5.2)$$

Se tiene entonces que la incertidumbre en la concentración estará dada por:

$$\delta[M] = [M] \sqrt{\delta[w_t]^2 + \delta\rho^2} \quad (5.3)$$

Una alícuota de 0,0297 g de esta solución fue disuelta en 99,5553 g de agua, dando lugar a una solución 0,2469 mM. Donde la concentración final se calcula a partir de las densidades de las soluciones inicial (ρ_s) y final (ρ_f), las masas de agua ($m_{\text{H}_2\text{O}}$) y de solución inicial (m_s) usando la siguiente ecuación:

$$[M]_f = [M]_i \frac{V_s}{V_f} = [M]_i \frac{m_s / \rho_s}{(m_s + m_{\text{H}_2\text{O}}) / \rho_f} = [M]_i \left(\frac{m_s}{m_s + m_{\text{H}_2\text{O}}} \right) \left(\frac{\rho_s}{\rho_f} \right) \quad (5.4)$$

Las densidades de las soluciones inicial y final se muestran en la [Tabla 5.1](#).

1.2. Solución de KHCO_3 0,17 mM

En un balón aforado de 10 mL fueron adicionados 0,10143 g de KHCO_3 , junto con 9,93551 g de H_2O . La densidad fue medida a 25 °C y su valor fue 1,003662 g/cm³. La concentración de esta solución se calculó usando la siguiente ecuación:

$$[M] = \frac{n}{V} = \frac{m\rho}{m_m(m + m_{\text{H}_2\text{O}})} \quad (5.5)$$

Obteniendo un valor de $0,10131 \pm 0,00001$ M, donde la incertidumbre se calcula usando:

$$\delta[M] = \sqrt{\frac{m^2}{m_m^2(m + m_{\text{H}_2\text{O}})^2} \delta\rho^2 + \frac{\rho^2 m_{\text{H}_2\text{O}}^2}{m_m^2(m + m_{\text{H}_2\text{O}})^4} \delta m^2 + \frac{\rho^2 m^2}{m_m^2(m + m_{\text{H}_2\text{O}})^4} \delta m_{\text{H}_2\text{O}}^2} \quad (5.6)$$

Posteriormente se tomó una alícuota de 0,1709 g, la cual fue diluida en 99,5657 g de agua. Usando la [Ecuación 5.4](#), se obtiene una concentración de 0,17265 mM. El resumen de las cantidades usadas para la dilución de las soluciones de ácido y bicarbonato, así como las densidades obtenidas a 20 °C se muestran en la [Tabla 5.1](#).

2. Sistema de inyección

El sistema de inyección consiste en un motor de pasos acoplado a un tornillo de precisión, el cual controla el desplazamiento del émbolo de la jeringa de inyección. El

TABLA 5.1: Densidades y masas medidas para preparar las soluciones con concentraciones 0,25 mM y 0,17 mM para el HCl y KHCO₃ correspondientemente.

	[M] _i (M)	m _s (g)	m _{H2O} (g)	ρ_s (g/cm ³)	ρ_f (g/cm ³)	[M] _f (mM)
HCl	0,84 ± 0,04	0,0297	99,5553	1,012832	0,998205	0,25
KHCO ₃	0,10131 ± 0,00001	0,1709	99,5657	1,003662	0,998215	0,17265

fluido saliente de la jeringa se desvía usando una manguera de cromatografía líquida de acero inoxidable, el cual se conecta en el otro extremo a un canal que lleva el fluido hasta la celda de medición.

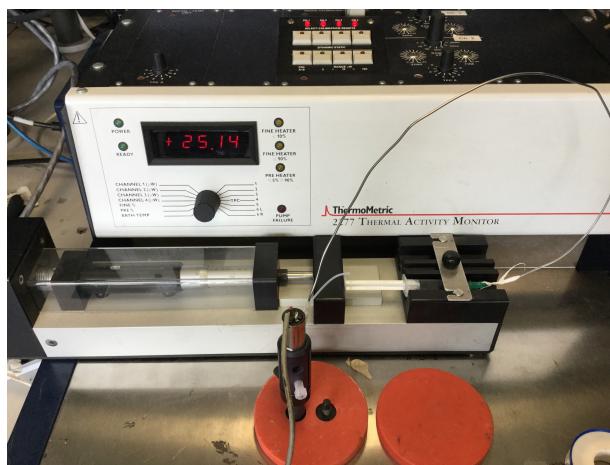


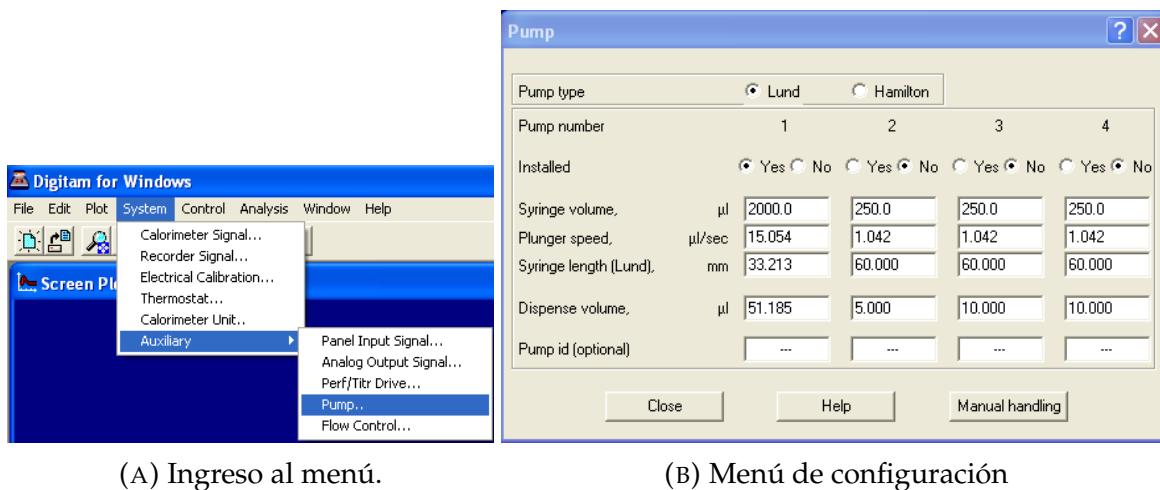
FIGURA 5.2: Sistema de inyección construido como alternativa al uso de las canulas.

Tanto la aguja de la jeringa como la manguera de cromatografía, corresponden con una alternativa para ingresar el fluido hasta la celda que difiere del mecanismo original en los volúmenes usados y el tipo de jeringas. Originalmente, para introducir una sustancia en la celda se hace uso de una jeringa de vidrio Hammilton de 250 μ L la cual cuenta con una canula soldada en la punta de esta. Al momento de realizar el experimento, no se contaba con una jeringa de este tipo con la canula conectada, y a pesar que varios intentos fueron realizados para soldar la punta con la canula de oro y acero inoxidable no fue posible juntar las partes, en parte dado que el diámetro de la canula es considerablemente pequeño, siendo difícil de ver a simple vista. Lo anterior tiene consideraciones especiales, pues los diámetros de la jeringa y la manguera no son compatibles, por lo cual se hace necesario usar cinta de teflón para evitar al máximo fugas en el sistema. Además el volumen interno de la manguera cromatográfica es mucho

mayor al de la canula, por lo cual se debe tener especial cuidado por los volúmenes introducidos, pues no se debe exceder la capacidad de 4 mL de la celda.

2.1. Control por software

Para acceder al control de la jeringa, en el menú superior: System >Auxiliary >Pump. En el momento se cuenta con un único agitador para la celda, por lo cual sólo se encuentra instalado uno de los controladores de jeringa tipo Lund, por esta razón el sistema debe detectar automáticamente únicamente el primer controlador (Installed = yes). La configuración de una jeringa consiste en escribir el volumen de esta (Syringe volume), la velocidad con la que se quiere mover el émbolo (Plunger speed), su longitud (Syringe length) y el volumen por inyección (Dispense volume).



(A) Ingreso al menú.

(B) Menú de configuración

FIGURA 5.3: Configuración de la jeringa en Digitam.

Una vez se encuentra configurada la jeringa, es posible usarla de dos maneras distintas. Por un lado se tiene el modo manual, donde el usuario tiene la posibilidad de avanzar rápidamente (Fast forward), por ejemplo para disminuir al distancia entre el émbolo y el pistón. También es posible aumentar esta distancia (Fast backward), avanzar un milímetro (One millimeter forward) y moverse la distancia requerida para que la jeringa dispense el volumen por inyección (Dispense). El control manual resulta útil antes de iniciar un experimento, pues con frecuencia será necesario ajustar la distancia entre el pistón y el émbolo para que haya contacto. Además en el caso de ser requerida una calibración de la jeringa es posible usar el botón de dispensar para determinar si el volumen dispensado corresponde con el volumen configurado.

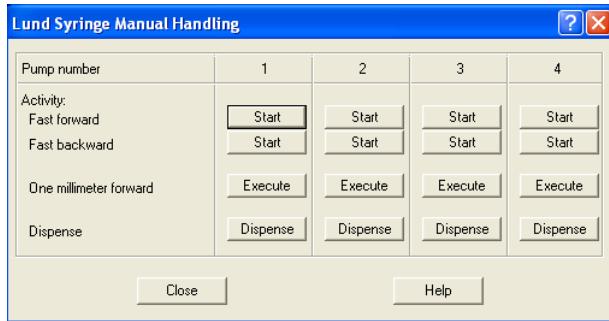


FIGURA 5.4: Panel del uso manual del controlador de la jeringa.

El modo automático se configura al momento de definir el método del experimento. Para esto es necesario expandir la opción de **Auxiliary system** en el panel izquierdo del menú y seleccionar ítem **Pump and flow control**. Se debe tener en cuenta la sección del experimento que se está configurando. En el caso de la [Figura 5.5](#), sólo existe la sección de **Baseline**, en esta sección se busca realizar 20 inyecciones cada una con el volumen de dispensación y velocidad configurados en la [Figura 5.3b](#) y 300 segundos de espera entre cada inyección.

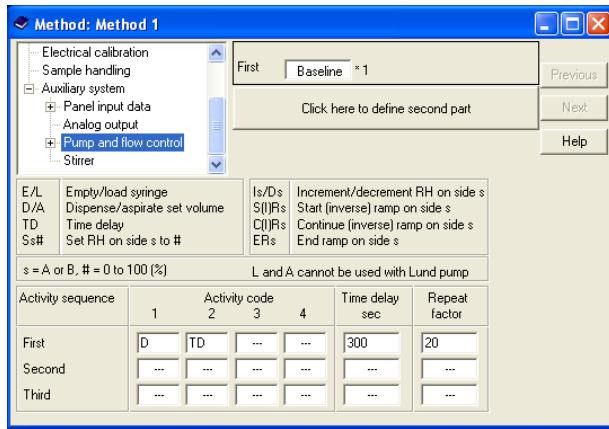


FIGURA 5.5: Panel de configuración de la jeringa en modo automático.

2.2. Calibración de la jeringa

Si bien el volumen de una jeringa es bien conocido, la longitud del émbolo es un parámetro que con frecuencia no se encuentra fácilmente. El controlador de la jeringa depende de este valor para relacionar la longitud que debe expandir el pistón para dispensar el volumen V_d . Por esta razón, para determinar la longitud correcta de la jeringa que debe ser configurada, se usaron distintos valores de esta y se midió la masa

de agua tipo 1 desplazada, para cada valor de longitud se tomaron 3 medidas, los cuales se muestran en la [Tabla 5.2](#).

TABLA 5.2: Masa dispensada por la jeringa usando diferentes longitudes y un $V_d = 100,000 \mu\text{L}$

Longitud (mm)	Masa 1 (g)	Masa 2 (g)	Masa 3 (g)	Promedio (g)	Desviacion (g)
28,000	0,08194	0,08196	0,08262	0,0822	0,0004
29,000	0,08573	0,08673	0,08648	0,0863	0,0005
30,000	0,09003	0,08884	0,08970	0,0895	0,0006
31,000	0,09270	0,09322	0,09263	0,0928	0,0003
32,000	0,09551	0,09536	0,09538	0,0954	0,0001
33,000	0,09895	0,09825	0,09870	0,0986	0,0004
34,000	0,10172	0,10176	0,10137	0,1016	0,0002
35,000	0,10409	0,10318	0,10422	0,1038	0,0006
36,000	0,10679	0,10593	0,10757	0,1068	0,0008
37,000	0,10958	0,11015	0,10993	0,1099	0,0003
38,000	0,11384	0,11327	0,11360	0,1136	0,0003

Con el termómetro de mercurio usado en la calibración térmica de los sensores de temperatura se midió la temperatura del agua usada en la calibración de la jeringa, dando un valor de 17.7 °C. El densímetro usado en la [Sección 1](#) fue configurado para realizar una lectura a esta misma temperatura, con lo cual se obtuvo un valor de densidad de 0.998679 g/cm³. Con esta información es posible construir una curva que permite inferir la configuración a usar de la jeringa, esta curva junto con el valor escogido se muestran en la [Figura 5.6](#). Para un volumen de dispensación de 100 μL, se obtuvo una longitud de 33,213 mm.

3. Realización del experimento

El método experimental está dividido en cuatro partes:

1. **Pause:** En la etapa inicial del experimento se busca medir el estado de la línea base por 120 minutos consecutivos.
2. **Baseline:** Luego de tener datos sobre el estado sin perturbar del sistema, se realiza una calibración dinámica para realizar un ajuste fino de los parámetros de ganancia y nivel del cero de la señal.

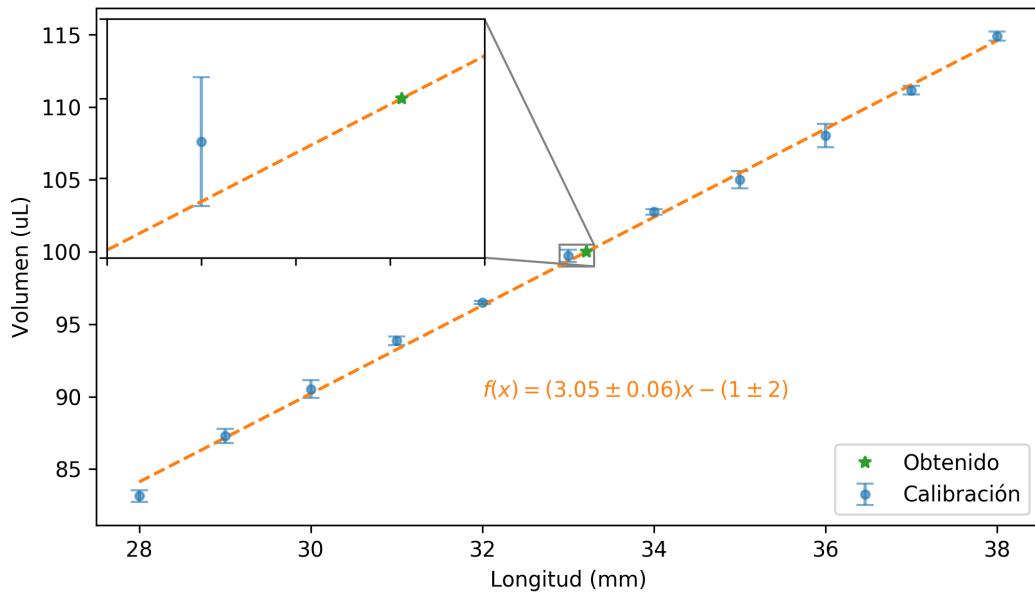


FIGURA 5.6: Curva de calibración de la jeringa usada.

3. **Pause:** En este punto se realiza una calibración estática para confirmar que la calibración dinámica fue correcta. Para esto, se aplican $300 \mu\text{W}$ sobre la celda por 30 minutos, posteriormente se retira la potencia y se esperan 50 minutos para la estabilización de la linea base.
4. **Main:** Una vez estabilizada la linea base, se realizan 30 inyecciones sucesivas con 10 minutos de espera entre cada inyección, la velocidad de inyección es de $50,018 \mu\text{/s}$ y el volumen de inyección es de $51,185 \mu\text{L}$.

En la celda de medición fueron adicionados 1.6029 g de la solución de KHCO_3 0.17265 mM , y la jeringa fue cargada con 2.0 mL de la solución de $\text{HCl} 0.2469 \text{ mM}$, y la temperatura del baño térmico se mantuvo en $25,05 \pm ^\circ\text{C}$ a lo largo del experimento.

4. Resultados

Conclusiones

Referencias

- (1) Feynman, R. P.; Leighton, R. B. y Sands, M., *The Feynman lectures on physics, Vol. I: The new millennium edition: mainly mechanics, radiation, and heat*; Basic books: 2011; vol. 1.
- (2) Fermi, E., *Notes on thermodynamics and statistics*; University of Chicago Press: 1986.
- (3) Atkins, P. y De Paula, J., *Physical chemistry for the life sciences*; Oxford University Press, USA: 2011.
- (4) Gaisford, S.; Kett, V. y Haines, P., *Principles of thermal analysis and calorimetry*; Royal society of chemistry: 2016.
- (5) Zielenkiewicz, W. y Margas, E., *Theory of calorimetry*; Springer Science & Business Media: 2006; vol. 2.
- (6) Wadsö, I. y Goldberg, R. N. *Pure and Applied Chemistry* **2001**, 73, 1625-1639.
- (7) Wadsö, I. y Wadsö, L. *Thermochimica acta* **2003**, 405, 15-20.
- (8) Blandamer, M. J.; Briggs, B.; Cullis, P. M.; Irlam, K. D.; Engberts, J. B. y Kevelam, J. *Journal of the Chemical Society, Faraday Transactions* **1998**, 94, 259-266.
- (9) Winkelmann, M; Hüttl, R y Wolf, G *Thermochimica acta* **2004**, 415, 75-82.
- (10) Morrison, J. *Pure and applied chemistry* **1987**, 59, 7-14.
- (11) Wang, M.-H.; Tan, Z.-C.; Sun, X.-H.; Zhang, H.-T.; Liu, B.-P.; Sun, L.-X. y Zhang, T. *Journal of Chemical and Engineering Data* **2005**, 50, 270-273.
- (12) Suurkuusk, J. *2277 Thermal Activity Monitor*; inf. téc.; Järfälla: Termometric AB.
- (13) Tellinghuisen, J. *The Journal of Physical Chemistry B* **2007**, 111, 11531-11537.
- (14) Mizoue, L. S. y Tellinghuisen, J. *Biophysical chemistry* **2004**, 110, 15-24.
- (15) Duff, M. R. y col. *Journal of visualized experiments: JoVE* **2011**.
- (16) Simon, S. H., *The Oxford solid state basics*; OUP Oxford: 2013.

- (17) Instruments, T. *LM35 Precision Centigrade Temperature Sensors*; inf. téc.; 1999.
- (18) AVR, M. *8-bit AVR microcontroller with 1K Bytes In-System Programmable Flash*; inf. téc.; 2010.
- (19) Grewal, H. *Oversampling the ADC12 for higher resolution*; inf. téc.; 2006.
- (20) Alexander, C. K.; Sadiku, M. N. y Sadiku, M., *Fundamentals of electric circuits*; McGraw-Hill New York: 2009; vol. 3.
- (21) Perry, R. H.; Green, D. W.; Maloney, J. O. y col. *Perry's chemical engineers' handbook.*, 200722.

Imágenes de soporte

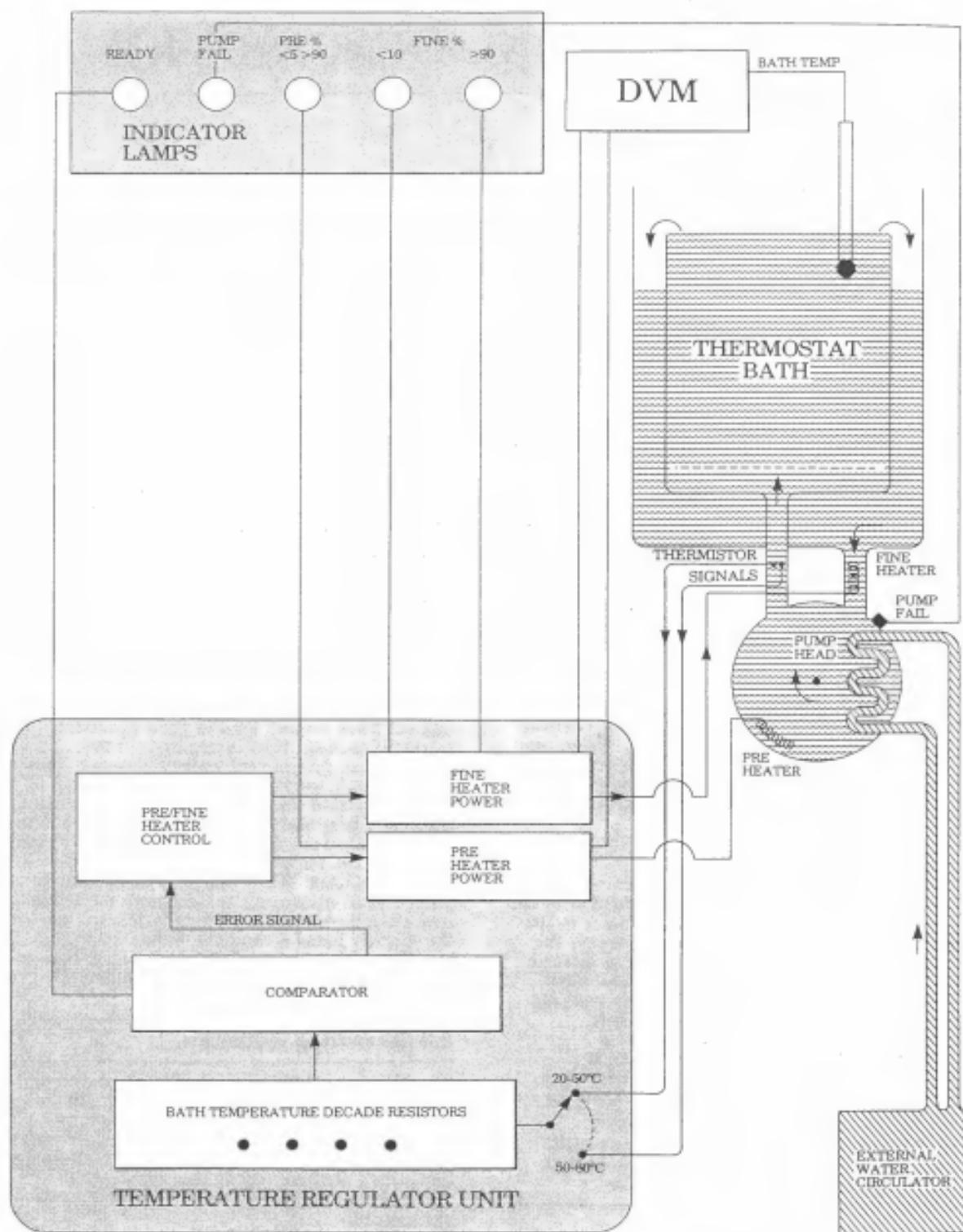


FIGURA A.1: Control térmico del calorímetro. Modificado de [12].

Thermal Activity Monitor

No. 404

Date: 05-17-1994

Sensor #: 411B

Lowtemp

Calibration data

Resistans	Temp	Error	Bath temp	Decade set	Circ temp	Bath temp	Decade set	Circ temp
Ω	°C	°C	°C	Ω/10	°C	°C	Ω/10	°C
40607.50	19.999	0.000	10	6474	6.0	30	2616	27.0
25895.50	30.235	-0.000	11	6171	7.0	31	2507	28.0
17268.70	39.999	0.000	12	5884	8.5	32	2403	29.0
11656.50	50.006	-0.000	13	5612	9.5	33	2304	30.0
			14	5354	10.5	34	2209	31.0
			15	5109	11.5	35	2119	32.0
			16	4877	12.5	36	2033	33.0
			17	4657	13.5	37	1951	34.5
			18	4448	14.5	38	1873	35.5
			19	4249	15.5	39	1798	36.5
A=0.11332690E-01			20	4061	16.5	40	1727	37.5
B= 4957.60			21	3881	17.5	41	1659	38.5
C= -156384.66			22	3711	18.5	42	1594	39.5
T=temp. in Kelvin			23	3549	19.5	43	1531	40.5
			24	3395	21.0	44	1472	41.5
			25	3249	22.0	45	1415	42.5
			26	3109	23.0	46	1361	43.5
			27	2977	24.0	47	1309	44.5
			28	2850	25.0	48	1259	45.5
			29	2730	26.0	49	1211	46.5

Sensor #: 406A

Hightemp

Calibration data

Resistans	Temp	Error	Bath temp	Decade set	Circ temp	Bath temp	Decade set	Circ temp
Ω	°C	°C	°C	Ω/10	°C	°C	Ω/10	°C
61596.80	50.026	0.000	50	6166	47.5	70	2765	68.0
40850.50	60.007	-0.000	51	5912	48.5	71	2663	69.0
27224.10	70.413	0.000	52	5669	49.5	72	2564	70.0
19224.30	79.800	-0.000	53	5438	51.0	73	2469	71.5
			54	5217	52.0	74	2379	72.5
			55	5006	53.0	75	2292	73.5
			56	4805	54.0	76	2209	74.5
			57	4613	55.0	77	2129	75.5
			58	4429	56.0	78	2052	76.5
			59	4254	57.0	79	1979	77.5
A=0.12585980E-01			60	4086	58.0	80	1909	78.5
B= 5542.77			61	3926	59.0	81	1841	79.5
C= -182502.18			62	3773	60.0	82	1776	80.5
T=temp. in Kelvin			63	3627	61.0	83	1714	81.5
			64	3487	62.0	84	1654	82.5
			65	3353	63.0	85	1597	83.5
			66	3225	64.0	86	1541	84.5
			67	3103	65.0	87	1488	85.5
			68	2985	66.0	88	1437	86.5
			69	2873	67.0	89	1388	87.5

FIGURA A.2: Equivalencia de resistencia y temperatura en una calibración de 1994.

Códigos

1. Firmware microcontrolador

```
#include "uart.h"
#include <util/delay.h>
#include <avr/interrupt.h>

#define STOP 0x01
#define START 0x02
#define SET_CHANNEL_0 0x03
#define SET_CHANNEL_1 0x04
#define SET_CHANNEL_2 0x05
#define SET_CHANNEL_3 0x06
#define NAME 0x0f

#define N_MEAN 4096 // 4**6

void setupADC(void);
void setupUART(void);
void setChannel(uint8_t ch);
void sendADC(uint16_t value);

int main(void)
{
    setupADC();
    setupUART();

    uint8_t val;
    uint16_t i;
    uint32_t mean;
```

```

sei();

while(1)
{
    val = uart_getc() // & (0x7f);

    if(val == START)
    {
        mean = 0;
        for(i = 0; i < N_MEAN; i++)
        {
            ADCSRA |= (1 << ADSC); // start conversion
            while(ADCSRA & (1 << ADSC));
            mean += ADC;
        }
        sendADC(mean / 64); // 2**10 * 4**6 / 2**16
    }
    else if(val == NAME)
    {
        uart_puts("Rutherford\n");
    }
    else if((val >= SET_CHANNEL_0) && (val <= SET_CHANNEL_3))
    {
        setChannel(val - SET_CHANNEL_0);
        uart_puts("C\n");
    }
    else
    {
        uart_putc(val);
    }
}
return 0;
}

void sendADC(uint16_t value)
{
    uint8_t high, low;
    high = value >> 8;
    low = value;

    uart_putc(high);
    uart_putc(low);
}

```

```

void setChannel(uint8_t ch)
{
    uint8_t mask = ~((1 << MUX1) | (1 << MUX0));
    ADMUX = (ADMUX & mask) | ch;
}

void setupADC(void)
{
    /*setup ADC*/
    ADMUX |= (1 << REFS0); // internal reference
    //~ ADCSRA |= (1 << ADPS2) | (1 << ADPS1) | (1 << ADPS0); // set
    division factor to 128
    ADCSRA |= (1 << ADPS2) | (1 << ADPS1); // set division factor to 64 //
    default
    //~ ADCSRA |= (1 << ADPS1) | (1 << ADPS0); // set division factor to 8
    ADCSRA |= (1 << ADEN);
}

void setupUART(void)
{
    PORTB &= ~(1 << UART_RX); // set low
    DDRB &= ~(1 << UART_RX); // input

    PORTB |= (1 << UART_TX); // set high
    DDRB |= (1 << UART_TX); // output
}

```

2. Determinación de intervalos estacionarios en la calibración del sistema de monitoreo de temperatura

```

import numpy as np
import matplotlib.pyplot as plt
from scipy.signal import medfilt

def findZeros(ch1, ch2, ch3):
    y = np.zeros((len(ch1), 3))
    y[:, 0] = ch1[:, 1]
    y[:, 1] = ch2[:, 1]
    y[:, 2] = ch3[:, 1]

```

```
y = np.mean(y, axis = 1)
dy = abs(np.diff(y))
abs_dy = np.zeros(len(dy) + 1)
abs_dy[0] = dy[1]
abs_dy[1:] = dy

dy = medfilt(abs_dy, kernel_size = 507)
pos = (dy > 5e-2)
dy[pos] = 0.5
dy[np.logical_not(pos)] = 0
return abs_dy, dy
```

3. Software de Temperatura

3.1. Comunicaciones

```
from time import sleep
from serial import Serial
import serial.tools.list_ports as find_ports

GET_ADC = 0x02
SET_CHANNEL_0 = 0x03
SET_CHANNEL_1 = 0x04
SET_CHANNEL_2 = 0x05
SET_CHANNEL_3 = 0x06
NAME = 0x0f

SLOPES = [10.257, 10.3, 10.29]
INTERCEPTS = [-12.2, -10.2, -8.4]

BAUDRATE = 115200
TIMEOUT = 0.5

SAMPLING_TIME = 2
ADJUST_TIME = 0.4

SERIAL = None

V_REF = 1.1

LAST_CHANNEL = -1
```

```
class NoActiveSerialException(Exception):
    """
        There is no serial port active.
    """

    def __init__(self):
        super(Exception, self).__init__("There_is_no_serial_port_active.")


class InvalidCalibrationPort(Exception):
    """
        Not a calibration port.
    """

    def __init__(self, port):
        super(Exception, self).__init__("{}_is_not_a_calibration_port.".format(port))


class ADCException(Exception):
    """
        Critical error. It was not possible to retrieve a valid ADC value.
    """

    def __init__(self):
        super(Exception, self).__init__("Critical_error._It_was_not_possible_to_retrieve_a_valid_ADC_value.")


def testName(serial):
    serial.write([NAME])
    sleep(0.2)
    ans = serial.readline()
    try:
        ans = ans.decode()
    except:
        return False
    if "Rutherford" in ans:
        return True
    return False


def findDevices():
    ports = list(find_ports.comports())
    devs = []
    for port in ports:
        port = port.device
        try:
            ser = initPort(port)
```

```
        devs.append(port)
        ser.close()
    except Exception as e:
        print(e)
        # pass
    return devs

def initPort(port):
    ser = Serial(port = port, baudrate = BAUDRATE, timeout = TIMEOUT)
    if testName(ser):
        return ser
    else:
        ser.close()
        raise(InvalidCalibrationPort(port))

def setChannel(channel):
    global SERIAL, LAST_CHANNEL
    if (channel <= 3) and (channel >= 0):
        try:
            for i in range(10):
                SERIAL.write([SET_CHANNEL_0 + channel])
                ans = SERIAL.readline()
                try:
                    ans = ans.decode()
                    if "C" in ans:
                        break
                except: pass
                sleep(0.01)
        except AttributeError:
            raise(NoActiveSerialException())
    else:
        raise(Exception("%d is not a valid channel." %channel))

def getADC():
    for i in range(5):
        SERIAL.write([GET_ADC])
        try:
            high = SERIAL.read()[0]
            # if high <= 3:
            low = SERIAL.read()[0]
            value = (high << 8) | low
            return value
        except IndexError:
```

```
        pass
    raise(ADCException())

def getVoltage():
    global V_REF
    v = (V_REF * getADC()) / 0xFFFF
    return v

def getTemperatures():
    global SAMPLING_TIME, ADJUST_TIME
    t = [0]*3
    adjust = SAMPLING_TIME / 3 - ADJUST_TIME
    if adjust < 0: adjust = 0
    for i in range(3):
        setChannel(i + 1)
        val = (1000*getVoltage() - INTERCEPTS[i]) / SLOPES[i]
        t[i] = round(val, 2)
        sleep(adjust)
    return t

def setGlobalSerial(serial):
    global SERIAL
    SERIAL = serial

def isSerialNone():
    global SERIAL
    if SERIAL == None:
        return True
    return False

def close():
    global SERIAL
    try:
        SERIAL.close()
        SERIAL = None
    except: pass

if __name__ == '__main__':
    devs = findDevices()
    print(devs)
    if len(devs) == 1:
        SERIAL = initPort(devs[0])
```

```
while True:  
    try:  
        text = []  
        for i in range(4):  
            setChannel(i)  
            val = getADC()  
            text.append("C%d:\u2022%d" %(i, val))  
        text = "\t".join(text)  
        print(text)  
  
        sleep(5e-2)  
    except KeyboardInterrupt:  
        break  
  
SERIAL.close()
```

3.2. Interfaz gráfica

```
import sys  
import time  
  
import __images__  
from com import initPort, findDevices, setChannel, getTemperatures,  
    setGlobalSerial, close, isSerialNone  
  
import pyqtgraph as pg  
try:  
    from PyQt5 import QtCore, QtGui  
    from PyQt5.QtWidgets import QLabel, QWidget, QMainWindow, QBoxLayout,\  
        QGroupBox, QFormLayout, QSystemTrayIcon, QApplication, QMenu, \  
        QStyleFactory, QMessageBox  
except ImportError:  
    from PyQt4 import QtCore, QtGui  
    from PyQt4.QtGui import QLabel, QWidget, QMainWindow, QBoxLayout,\  
        QGroupBox, QFormLayout, QSystemTrayIcon, QApplication, QMenu,\  
        QStyleFactory, QMessageBox  
  
import datetime  
import numpy as np  
  
START_TIME = 0  
KERNEL_FILTER = 3
```

```
MAX HOLDER = 45e3

# https://gist.github.com/iverasp/9349dff42aeffb32e48a0868edfa32d

class TimeAxisItem(pg.AxisItem):
    def __init__(self, *args, **kwargs):
        super().__init__(*args, **kwargs)
        self.setLabel(text='Time', units=None)
        self.enableAutoSIPrefix(False)

    def tickStrings(self, values, scale, spacing):
        return [datetime.datetime.fromtimestamp(value).strftime("%d/%H %M")
for value in values]

class RingBuffer(object):
    def __init__(self, size_max):
        self.max = int(size_max)
        self.data = []

    class __Full:
        """ class that implements a full buffer """
        def append(self, x):
            """ Append an element overwriting the oldest one. """
            self.data[self.cur] = x
            self.cur = (self.cur+1) % self.max

        def get(self):
            """ return list of elements in correct order """
            return self.data[self.cur:]+self.data[:self.cur]

        def append(self, x):
            """append an element at the end of the buffer"""
            self.data.append(x)
            if len(self.data) == self.max:
                self.cur = 0
                self.__class__ = self.__Full # Permanently change self's class
from non-full to full

        def get(self):
            """ Return a list of elements from the oldest to the newest. """
            return self.data

class DataHolder(object):
```

```
def __init__(self):
    self.x = RingBuffer(MAX HOLDER)
    self.y = [RingBuffer(MAX HOLDER), RingBuffer(MAX HOLDER), RingBuffer(MAX HOLDER)]
```

```
def timestamp(self):
    return time.mktime(datetime.datetime.now().timetuple())
```

```
def addValues(self, temperatures):
    now = time.localtime()
    self.x.append(self.timestamp())
    for i in range(3):
        y = self.y[i]
        y.append(temperatures[i])
        if (len(y.get()) > KERNEL_FILTER) and (KERNEL_FILTER > 0):
            temp = sorted(y.get()[-KERNEL_FILTER:])[(KERNEL_FILTER - 1)
// 2]
            y.get()[-(KERNEL_FILTER - 1)] = temp
```

```
    self.save(now)
```

```
def save(self, now):
    with open("TemperatureData.txt", "a") as file:
        now = time.strftime("%A %m-%d %H:%M %S", now)
        data = ["%.2f" % self.getY(i)[-1] for i in range(3)]
        line = [now] + data
        file.write("\t".join(line) + "\r\n")
```

```
def getX(self):
    return self.x.get()
```

```
def getY(self, i):
    return self.y[i].get()
```

```
def clear(self):
    self.x = RingBuffer(MAX HOLDER)
    self.y = [RingBuffer(MAX HOLDER), RingBuffer(MAX HOLDER), RingBuffer(MAX HOLDER)]
```

```
def __len__(self):
    return len(self.x.get())
```

```
class FindDevicesThread(QtCore.QThread):
```

```
def __init__(self):
    super(QtCore.QThread, self).__init__()

def run(self):
    while True:
        if isSerialNone():
            devs = findDevices()
            if len(devs) == 1:
                try:
                    setGlobalSerial(initPort(devs[0]))
                except:
                    setGlobalSerial(None)
            else:
                setGlobalSerial(None)
        else:
            time.sleep(10)

class requestDataThread(QtCore.QThread):
    def __init__(self, holder):
        super(QtCore.QThread, self).__init__()
        self.holder = holder
        self.exception = None

    def run(self):
        global START_TIME
        while True:
            if not isSerialNone():
                try:
                    self.holder.addValues(getTemperatures())
                except Exception as e:
                    self.stop()
                    self.exception = e
            else:
                time.sleep(5)

    def stop(self):
        close()

class MainWindow(QMainWindow):
    SAMPLING_DEFAULT = 1 # seconds
    MINIMUM_PLOT_UPDATE = 2000

    def __init__(self):
```

```
super(QMainWindow, self).__init__()
self.setWindowTitle("Lector_de_temperatura")
widget = QWidget()
self.setCentralWidget(widget)

self.main_layout = QBoxLayout(widget)
self.main_layout.setContentsMargins(11, 11, 11, 11)
self.main_layout.setSpacing(6)

self.settings_frame = QGroupBox()
self.settings_layout = QFormLayout(self.settings_frame)

self.label_0 = QLabel("00.00")
self.label_1 = QLabel("00.00")
self.label_2 = QLabel("00.00")
self.settings_layout.addRow(self.label_0, QLabel("\tInterno_(C)"))
self.settings_layout.addRow(self.label_1, QLabel("\tExterno_(C)"))
self.settings_layout.addRow(self.label_2, QLabel("\tAmbiente_(C)"))

self.main_layout.addWidget(self.settings_frame)
### pyqtgraph
pg.setConfigOptions(leftButtonPan = False, foreground = 'k',
background = None)
# pg.setConfigOptions(, antialias = True)
self.temperature_plot = pg.PlotWidget(
    labels={'left': 'Temperatura_(C)', 'bottom': 'Hora'},
    axisItems={'bottom': TimeAxisItem(orientation='bottom')}
)
self.temperature_plot.addLegend()
self.main_layout.addWidget(self.temperature_plot)

symbol = None #
symbolSize = 3
self.data0_line = self.temperature_plot.plot(pen = "b", symbol =
symbol, symbolPen = "b", symbolBrush="b", symbolSize=symbolSize, name=""
Interno")
self.data1_line = self.temperature_plot.plot(pen = "m", symbol =
symbol, symbolPen = "m", symbolBrush="m", symbolSize=symbolSize, name=""
Externo")
self.data2_line = self.temperature_plot.plot(pen = "g", symbol =
symbol, symbolPen = "g", symbolBrush="g", symbolSize=symbolSize, name=""
Ambiente")
```

```
#### signals
self.update_plots_timer = QtCore.QTimer()
self.update_plots_timer.setInterval(self.MINIMUM_PLOT_UPDATE)
self.update_plots_timer.timeout.connect(self.updatePlots)

self.find_thread = FindDevicesThread()
self.find_thread.start()

self.data = DataHolder()
self.data_thread = RequestDataThread(self.data)
self.data_thread.start()
self.update_plots_timer.start()

def updatePlots(self):
    if self.data_thread.exception != None:
        self.errorWindow(self.data_thread.exception)
    else:
        try:
            x = np.array(self.data.getX())
            for i in range(3):
                label = getattr(self, "label_%d" % i)
                plot = getattr(self, "data%d_line" % i)
                data = np.array(self.data.getY(i))
                plot.setData(x, data)
                label.setText("%.2f" % data[-1])
        except Exception as e:
            print(e)

def deviceConnection(self):
    if self.find_thread.success:
        self.deviceExists()
    else:
        self.noDevice()

def noDevice(self):
    self.data_thread.stop()
    self.update_plots_timer.stop()

def warning(self, text):
    msg = QMessageBox()
    msg.setIcon(QMessageBox.Warning)
    msg.setText('Warning.\n%s' % text)
    msg.setWindowTitle("Warning")
```

```
msg.setStandardButtons(QMessageBox.Ok)
msg.exec_()

def errorWindow(self, exception):
    self.noDevice()
    text = str(exception)
    msg = QMessageBox()
    msg.setIcon(QMessageBox.Warning)
    msg.setText('An_error_occurred.\n%s' % text)
    msg.setWindowTitle("Error")
    msg.setStandardButtons(QMessageBox.Ok)
    msg.exec_()

def closeEvent(self, event):
    event.ignore()
    self.hide()

class SystemTrayIcon(QSystemTrayIcon):
    def __init__(self, icon, parent = None):
        QSystemTrayIcon.__init__(self, icon, parent)
        menu = QMenu(parent)
        openAction = menu.addAction("Open")
        exitAction = menu.addAction("Exit")
        self.setContextMenu(menu)

        openAction.triggered.connect(self.openMain)
        exitAction.triggered.connect(self.exit)
        self.activated.connect(self.systemIcon)

    self.main_window = MainWindow()
    self.openMain()

    def exit(self):
        QtCore.QCoreApplication.exit()

    def systemIcon(self, reason):
        if reason == QSystemTrayIcon.Trigger:
            self.openMain()

    def openMain(self):
        self.main_window.show()

if __name__ == '__main__':
```

```
app = QApplication(sys.argv)
QApplication.setStyle(QStyleFactory.create('Fusion'))
app.setWindowIcon(icon)
app.processEvents()

w = QWidget()
trayIcon = SystemTrayIcon(icon, w)

trayIcon.show()

app.exec_()

close()
```

4. Disco 3D para sensor térmico en el calorímetro

```
thick = 1.2;
d_screw = 4;
r_screw = (86.5 - d_screw) / 2;
$fn = 40;

difference()
{
    cylinder(d = 92, h = thick, center = true);

    translate([0, r_screw, 0])
    cylinder(d = d_screw, h = 2*thick, center = true);

    translate([r_screw, 0, 0])
    cylinder(d = d_screw, h = 2*thick, center = true);

    translate([(r_screw)/sqrt(2), (r_screw)/sqrt(2), 0])
    cylinder(d = d_screw, h = 2*thick, center = true);

    translate([0, -r_screw, 0])
    cylinder(d = d_screw, h = 2*thick, center = true);

    translate([-r_screw, 0, 0])
    cylinder(d = d_screw, h = 2*thick, center = true);

    translate([- (r_screw)/sqrt(2), -(r_screw)/sqrt(2), 0])
    cylinder(d = d_screw, h = 2*thick, center = true);
```

```
translate([- (r_screw) / sqrt(2), (r_screw) / sqrt(2), 0])
cylinder(d = d_screw, h = 2 * thick, center = true);

translate([(r_screw) / sqrt(2), -(r_screw) / sqrt(2), 0])
cylinder(d = d_screw, h = 2 * thick, center = true);

cylinder(d = 10, h = 2 * thick, center = true);
}
```

5. Determinación de los valores de una calibración estática

```
import numpy as np
from scipy.signal import medfilt

def getStaticData(name, from_ = 0, to_ = -1):
    sCal0 = np.genfromtxt(name, skip_header = 2)
    sCal0 = sCal0[from_:to_, :2] - sCal0[0, 0]
    return sCal0.T

def getIntervals(p, threshold = 0.5):
    d = np.diff(p)
    ad = medfilt(abs(d), kernel_size = 101)

    filtered = ad.copy()
    up = ad >= threshold
    low = ad < threshold
    filtered[up] = 1
    filtered[low] = 0

    changes = np.diff(filtered)
    ones = np.where(changes == 1)[0]
    negative = np.where(changes == -1)[0]

    t = 0.7
    baseline = (0, ones[0])
    top = int(t * (ones[1] - negative[0])) + negative[0], ones[1]
    baseline2 = int(t * (len(p) - negative[1])) + negative[1], len(p)

    return baseline, top, baseline2
```

```
def getStatistics(p, intervals):
    data = [p[i[0] : i[1]] for i in intervals]
    return [(np.mean(d), np.std(d)) for d in data]
```