

Tecnológico de Costa Rica
Sede de Alajuela

Escuela de Ingeniería Electrónica

Programa de Licenciatura en Ingeniería Electrónica



Sistema de caracterización del SoH de celdas de Li-ion, basado en el método de conteo de Coulomb con Arduino

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

María Charel Vega Morales
Carné: 2016072185

Profesor Asesor:
Ing. Sergio Morales Hernández

Alajuela, II Semestre de 2023

[Sistema de caracterización del SoH de celdas de Li-ion, basado en el método de conteo de Coulomb con Arduino](#) © 2023 by María Charel Vega Morales is licensed under [CC BY-NC-ND](#)

[4.0](#)

Declaratoria de Autenticidad

Declaro que el presente documento de proyecto de graduación ha sido realizado enteramente por mi persona, utilizando y aplicando literatura referente al tema e introduciendo conocimientos y resultados experimentales propios.

En los casos donde utilizo material bibliográfico, he señalado e identificado los autores respectivos mediante citas bibliográficas, detalladas al final del documento.

Asumo completa responsabilidad sobre el trabajo realizado y por el correspondiente contenido.

Alajuela, 22 de noviembre de 2023

María Charel Vega Morales

Céd: 1-1732-0814

Tecnológico de Costa Rica
Sede de Alajuela
Escuela de Ingeniería Electrónica
Proyecto de Graduación
Acta de aprobación

Defensa de Proyecto de Graduación
Requisito para optar por el título de Ingeniera en Electrónica
Grado académico de Licenciatura

El Tribunal Evaluador aprueba la defensa del proyecto de graduación denominado *Sistema de caracterización del SoH de celdas de Li-ion, basado en el método de conteo de Coulomb con Arduino*, realizado por la estudiante María Charel Vega Morales, haciendo constar que cumple con las normas establecidas por la escuela de Ingeniería Electrónica del Tecnológico de Costa Rica.

Miembros del Tribunal Evaluador

PhD. Roberto Pereira
Profesor Lector

MSc. Kervin Sánchez Herrera
Profesor Lector

Ing. Sergio Morales Hernández
Profesor Asesor

Alajuela, 22 de noviembre de 2023

Tecnológico de Costa Rica
Sede de Alajuela
Escuela de Ingeniería Electrónica
Proyecto de graduación
Tribunal Evaluador
Acta de evaluación

Defensa de Proyecto de Graduación
Requisito para optar por el título de Ingeniera en Electrónica
Grado académico de Licenciatura

Estudiante: **María Charel Vega Morales**

Carné: 2016072185

Nombre del proyecto: *Sistema de caracterización del SoH de celdas de Li-ion, basado en el método de conteo de Coulomb con Arduino.*

Los miembros de este tribunal hacen constar que este proyecto de graduación ha sido aprobado y cumple con las normas establecidas por la escuela de Ingeniería Electrónica del Tecnológico de Costa Rica y es merecedor de la siguiente calificación:

Nota de proyecto de graduación: _____

Miembros del Tribunal Evaluador

PhD. Roberto Pereira
Profesor Lector

MSc. Kervin Sánchez Herrera
Profesor Lector

Ing. Sergio Morales Hernández
Profesor Asesor

Alajuela, 22 de noviembre de 2023

A mi amada madre y profesora de vida,

Sonia

Agradecimientos

Este logro en mi carrera profesional se debe en gran medida al apoyo y amor inagotable de mi mamá y mi hermana, Verónica, quienes siempre han estado a mi lado y me han brindado la motivación necesaria para seguir adelante. Además, quiero expresar mi profundo e infinito agradecimiento al profesor Sergio Morales, por apoyarme y asignarme un proyecto final de graduación. También a la profesora Aura Ledezma de la escuela de Química, por haberme dado la oportunidad de dar tutorías cuando necesitaba financiar semestre a semestre mi carrera; asimismo, a los profesores Daniel Kohkemper y Javier Pérez por la oportunidad de ser tutora en la escuela de Ingeniería Electrónica.

María Charel Vega Morales

Alajuela, 22 de noviembre de 2023

Resumen

En este documento se presenta la propuesta de proyecto de graduación para optar por el grado de Licenciatura en Ingeniería en Electrónica en el Tecnológico de Costa Rica, la cual es el diseño de un sistema de caracterización del estado de salud (SoH) de celdas de Li-ion basado en el método de conteo de Coulomb; además, utiliza una plataforma para adquirir y procesar datos relevantes que permitan hacer el análisis de la capacidad y vida útil de las celdas Li-ion.

Las celdas de Li-ion han emergido como una solución importante para el almacenamiento de energía en distintas aplicaciones, por lo tanto, se debe desarrollar un sistema de caracterización del SoH confiable que permita evaluar la degradación de las celdas y tomar decisiones informadas en términos de su mantenimiento y reemplazo. El sistema basado en el método de conteo de Coulomb brinda una alternativa eficiente y de bajo presupuesto para la monitorización del SoH.

Palabras clave: Caracterización, Carga, Celda, Coulomb, Descarga, Estado de salud, Li-ion.

Abstract

This document presents the graduation project proposal, to opt for the Licentiate's degree in Electronic Engineering at the Tecnológico de Costa Rica. The topic to be explored in this proposal involves the design of a State of Health (SoH) characterization system for Li-ion cells using the Coulomb counting method. Additionally, it uses a platform to acquire and process relevant data, to enable an accurate analysis of the capacity and lifespan estimation in the Li-ion cells.

The Coulomb counting method-based approach offers an efficient and cost-effective alternative for monitoring the SoH. Li-ion cells have emerged as an important solution for energy storage in different applications, therefore, it's crucial to develop a reliable SoH characterization system that allow the evaluation of cell degradation and facilitates informed decisions regarding maintenance and replacement.

Keywords: Cell, Charge, Coulomb, Discharge, Li-ion, State of Health.

Índice

1. Introducción	1
1.1 Entorno del proyecto	1
1.2 Planteamiento del problema	3
1.3 Objetivos	3
1.3.1 Objetivo General	3
1.3.2 Objetivos específicos	3
1.4 Estructura del documento	4
2. Marco teórico	5
2.1 Celdas de iones de litio	5
2.1.1 Proceso de carga de una batería Li-ion	5
2.1.2 Proceso de descarga de una batería Li-ion	6
2.1.3 Impedancia interna	6
2.1.4 Capacidad	7
2.1.5 Energía contenida	8
2.1.6 Ciclo de vida (vida útil)	8
2.2 Estado de carga (SoC)	9
2.3 Estado de salud (SoH)	9
2.4 Método de conteo de Coulomb	10
2.5 Arduino	11
2.5.1 UNO	13
2.5.2 IDE	16
3. Diseño y verificación del sistema	18
3.1 Configuración	18
3.2 Hardware del sistema	19
3.1.1 Diagrama del circuito	19
3.1.2 Diseño físico del circuito	21
3.2 Código fuente	23
3.2.1 Diagrama de flujo	24
3.2.2 Pseudocódigo	26
3.2.3 Entorno de código	26
3.3 Procesamiento de datos	29
3.3.1 Configuración de Excel	29
4. Resultados y análisis	33
4.1 Variables de respuesta	33
4.2 Factores y niveles	33
4.2.1 Temperatura	34
4.2.2 Capacidad	34
4.2.3 Tiempo	34
4.2.4 Resumen de los factores	34
4.3 Repeticiones del experimento	35

4.4 Valores obtenidos del Arduino.....	35
4.5 Cálculo de energía, carga y tiempo de descarga.....	39
4.6 Cálculo de capacidad, SoC, SoH y resistencia interna	42
5. Conclusiones y recomendaciones.....	46
5.1 Conclusiones	46
5.2 Recomendaciones	47
Referencias	48

Capítulo 1

Introducción

1.1 Entorno del proyecto

El LIVE [1] es el laboratorio de investigación de vehículos eléctricos de la escuela de Ingeniería en Electrónica del Tecnológico de Costa Rica (TEC), en Cartago centro, fundado en el mes de enero del año 2018 por el profesor Sergio Morales como una iniciativa de hacer realidad la movilidad eléctrica y sostenible mediante apoyo en los vehículos eléctricos en Costa Rica. Este laboratorio es parte de las nuevas instalaciones de la escuela de Ingeniería en Electrónica puestas a funcionar en ese mismo año en la sede central del TEC, con el objetivo de otorgar a los docentes y estudiantado un espacio para generar conocimiento útil, de forma colaborativa y accesible al público [1]. Aquí se trabajan diferentes áreas en donde se contempla la parte ambiental y tecnológica del diseño de productos para los vehículos eléctricos, que va de la mano con utilización de celdas de ión-litio (Li-ion) y su correcta deposición [2].



Figura 1: Logo del LIVE. [2]

Según un informe presentado en la revista de IEEE Power & Energy [3], se prevé que en los próximos 50 años suceda un aumento significativo en la demanda de vehículos automotores, pasando a ser el volumen de 3 veces la cantidad de hoy en día. Sin embargo, si se satisface esta demanda con vehículos de combustión interna (ICE), se generará un volumen inmenso de contaminantes cada año. Dada esta situación, se reconoce la importancia de investigar mejoras en la autonomía de los vehículos eléctricos (EV) para ir desplazando poco a poco los ICE. Este es un claro interés de parte de LIVE.

Un punto en la mejora de la autonomía de los EV, es a través de sistemas de gestión de baterías (BMS). Un BMS, a grandes rasgos, es un sistema que controla directamente y realiza la supervisión del estado de salud (SoH) de las celdas y sus arreglos. Algunas de las principales funciones del BMS consisten en el balance de carga y en la estimación del estado de carga [4].

El desbalance de carga se refiere a una situación en la que dos o más celdas dentro de un conjunto se cargan de manera desigual, a pesar de estar conectadas en serie y paralelo, lo que implicaría corrientes iguales de carga y tensiones iguales en los bornes [5]. Cuando ocurre este desbalance, algunas de ellas pueden experimentar sobrecargas o descargas excesivas, lo que resulta en un daño directo en el SoH y, por ende, una reducción en su vida útil [5].

El conocimiento preciso del SoH de las celdas de litio es esencial para garantizar un rendimiento óptimo y seguro. Un diagnóstico del estado de degradación permite una planificación efectiva del mantenimiento y la toma de decisiones informadas sobre el reemplazo de baterías. Además, un sistema de caracterización confiable del SoH puede contribuir a mejorar la sostenibilidad y la eficiencia energética, dándole otros posibles usos a las celdas después de ser consideradas dañadas, así se podrá reducir la generación de residuos electrónicos [1].

A pesar de la importancia de evaluar el SoH de las celdas de Li-ion, existen desafíos significativos en el desarrollo de sistemas de diagnóstico precisos y no invasivos. Diferentes metodologías han sido propuestas y estudiadas en la literatura [6], cada una con sus ventajas y limitaciones. En este contexto, el método de conteo de Coulomb demuestra ser una técnica prometedora para la caracterización del SoH de las celdas de Li-ion debido a su capacidad para evaluar el envejecimiento de manera continua [6].

1.2 Planteamiento del problema

Los vehículos eléctricos están adquiriendo cada vez más posición en las calles; ahora basta con salir al centro de una ciudad para encontrarse con un diseño de vehículo eléctrico circulando por las calles. Sin embargo, muchas personas piensan que tener un vehículo de este tipo es prácticamente imposible por sus costos y, un problema mucho mayor, que “en unos años” la batería se daña y ya prácticamente se convierte en un desecho contaminante del ambiente.

No solo esto, que en un futuro si queremos priorizar la salud del planeta y la nuestra, será necesario hacer uso de un vehículo eléctrico. La manera en cómo se cargan, descargan, capacidad máxima y vida útil de las baterías de Li-ion será vital para poder optimizar el uso de estos.

Es entonces necesario diseñar un método o sistema que caracterice, en una celda de Li-ion sin importar si ya se ha usado o está nueva, su estado de salud (SoH) mediante experimentos de bajo costo que sean no invasivos; es así que, para cuando las baterías de vehículos eléctricos (EV) presenten problemas de capacidad, se cuente ya con un sistema escalable que permita saber cómo está su SoH realmente y así se puedan distinguir las celdas dañadas de las que aún tienen vida útil, previo a desecharlas completamente.

1.3 Objetivos

1.3.1 Objetivo General

Desarrollar un sistema de caracterización confiable y de bajo costo, basado en el método de conteo de Coulomb y la recolección de datos necesarios con Arduino para la estimación real del SoH de celdas de Li-ion.

1.3.2 Objetivos específicos

1. Diseñar una plataforma experimental, de hardware y software, para la obtención de las variables necesarias durante el proceso de descarga de la celda de Li-ion.
2. Monitorear el comportamiento del proceso de descarga de la celda para la verificación de la funcionalidad de la plataforma.

3. Aplicar el algoritmo de conteo de Coulomb en los datos adquiridos para la estimación del SoH de la celda.
4. Realizar severas pruebas y repeticiones experimentales para la validación de la estrategia de caracterización mediante análisis estadístico.

1.4 Estructura del documento

A lo largo de este trabajo, se propone el diseño de un sistema de caracterización de SoH de celdas de Li-ion, basado en el método de conteo de Coulomb utilizando Arduino. La distribución de la estructura del documento se presenta a continuación:

- En el capítulo [2](#) se presenta el marco teórico, donde se abarca la descripción y características que poseen las celdas de Li-ion, cómo determinar su capacidad, su vida útil, qué es la resistencia interna de una celda, el método de conteo de Coulomb y conceptos de SoH y SoC.
- En el capítulo [3](#) se presenta el desarrollo del diseño, tanto en hardware como el código fuente requerido. Asimismo, se muestra el método de recolección de datos, su procesamiento y los entornos usados.
- En el capítulo [4](#) se presentan las distintas variables de respuesta de interés, los factores con sus respectivos niveles. Se realizan las repeticiones del sistema de caracterización, los resultados finales de las réplicas y su análisis.
- En el capítulo [5](#) se finaliza con las conclusiones del trabajo y las recomendaciones ideales para la posterior mejora del sistema de caracterización de las celdas.

Capítulo 2

Marco teórico

2.1 Celdas de iones de litio

Las celdas de Li-ion son celdas secundarias (recargables), que proporcionan una corriente continua manteniendo un voltaje cuyo valor depende de la composición química de la celda. Típicamente, los valores de voltaje de las celdas Li-ion cilíndricas (como el modelo 16850) de manera individual pueden oscilar desde 3,2 V como el mínimo, a 4,2 V como el máximo [7].

El voltaje mínimo real de estas celdas es de 2,8 V [1], pero la zona de seguridad de descarga para una celda como estas es de 3,0 V [8]. Dicho valor mínimo es el que se utilizará en este proyecto. Esto es necesario ya que durante el proyecto se tiene que descargar la celda hasta que marque 3,0 V y ahí se desconectará de la resistencia consumidora a la que esté conectada, para no exceder la zona de seguridad de descarga.

2.1.1 Proceso de carga de una batería Li-ion

Se pueden contemplar dos procesos de carga para la celda Li-ion. Una es que va con corriente constante (a la que se referirá como CC) con un voltaje también constante (a referirse como VC); otra es cuando se tiene VC, pero la corriente sí varía, acercándose a un valor de 120mA. Por lo tanto:

1. CC-VC: primero se carga la batería a corriente constante (CC), hasta que alcance 4,2 V por celda; luego se mantiene el voltaje constante (VC) a 4,2 V hasta que la corriente de carga alcance un 10% de su valor inicial. Esta es la condición en la cual la batería está completamente cargada. Esto quiere decir que la corriente también se debe monitorear.
2. VC: en este caso el voltaje se mantiene constante a 4,2 V durante todo el proceso de carga, hasta que la corriente disminuye a 100 mA o 120 mA [9]. Si la carga de la batería termina cuando el voltaje es de 4,2 V por celda, pero sin haber disminuido a estos puntos la

corriente, entonces solo se habrá cargado la batería entre un 40% a 70% de su capacidad total. Por esta razón, la batería se debe continuar cargando hasta que la corriente disminuya a su valor de terminación [9].

2.1.2 Proceso de descarga de una batería Li-ion

La corriente suministrada por la batería se rige por la conocida ley de Ohm [7]:

$$I = \frac{V}{R} \quad (1)$$

Donde I es la corriente expresada en Amperios, V es la tensión de la celda o batería en Voltios, y R es la resistencia consumidora (o R de carga) en Ohmios.

La tasa de descarga de la batería se indica a menudo como “C” (viene siendo por C de capacidad) y, esta es la corriente constante que la batería puede entregar durante una hora, expresada en Amperios-hora. Por ejemplo, una batería de 12 V con una capacidad de 20 Ah puede suministrar una corriente constante de 20 A durante 1 hora, o una I de 1 A durante 20 horas. Por lo general, la tasa de carga es menor que la tasa de descarga [7].

Este proceso entonces está condicionado a la corriente que se suministre. Por ello es que los fabricantes [10] especifican capacidad nominal, capacidad típica y mínima capacidad en las características básicas de las baterías de Li-ion, donde contemplan la corriente de descarga según fracción de la capacidad nominal. La celda mostrará una curva de descarga con una pendiente lo más aproximada a 0 si se mantiene una corriente constante suministrada.

2.1.3 Impedancia interna

Los compuestos químicos dentro de la batería y los conductores utilizados como ánodo y cátodo presentan una pequeña resistencia interna (R_{int}) que impide que la corriente fluya libremente dentro de la batería. Esta con normalidad se representa teóricamente como una resistencia real y se coloca entre los terminales de la batería [7].

Cuando la R_{int} no es despreciable, la ley de Ohm aplicada a la celda es:

$$R_{int}(t_n) = \frac{V_{n-1} - V_n}{I_m} ; \text{ con } t_n > 0, n > 0 \quad (2)$$

Respecto a la ecuación 2, n es el número de medición y t_n es el tiempo al momento en el que fue tomado n ; I_m es la corriente promedio suministrada por la celda a la resistencia consumidora durante su proceso de descarga cada 10 mediciones; V_n representa el voltaje medido de la celda en la medición n . La I_m dependerá de i , que es un valor entero mayor que cero.

$$I_m = \frac{\sum_{i=1}^{10} I_i}{10} \quad (3)$$

Como ejemplo se podría considerar que, para una celda de 2 V con una resistencia interna de 100 m Ω conectada a una consumidora de 2 Ω , la tensión operativa en los terminales de la batería es de 1,9 V y la corriente suministrada a la consumidora es de 0,95A [7].

La caída de tensión debido a la resistencia interna de la batería no es constante, sino que depende de la intensidad de la corriente proporcionada por la batería a la consumidora. Para otros tipos de celdas, la resistencia interna se sitúa alrededor de unos pocos miliohmios, por lo que los efectos pueden ser despreciados [7].

En el caso de las celdas 18650 los fabricantes [10] aclaran que la R_{int} debe estar entre un valor menor o igual a 70m Ω , de esta manera es que se considera como “nueva”. Cuando la R_{int} da en centenas de miliohmios, se asume que la celda alcanzó aproximadamente la mitad de su vida útil [1].

El problema de estimar la degradación de la celda utilizando la R_{int} es la prueba que se ejecuta, pues es una comparación con una resistencia de potencia, y, ya que esta última presenta el inconveniente de percibir mucha corriente y una variación de la resistividad (a causa del aumento de la temperatura), presenta resultados no certeros.

2.1.4 Capacidad

Como se mencionó antes, la capacidad de la celda o batería se indica en amperios-hora o miliamperios-hora y representa la corriente que la celda (o batería) puede suministrar en una hora antes de descargarse, o más precisamente, la cantidad de carga contenida en la celda.

La cantidad de carga transferida con intensidad I se mide en Coulomb y está dada por [7]:

$$Q = I_{prom} \cdot t \quad (4)$$

Donde “Q” es la cantidad de carga transferida en Coulomb, “y “t” es el tiempo transcurrido para la transferencia [7]. I_{prom} es la corriente promedio calculada de cada corriente medida hasta el final de la descarga.

La cantidad de carga en una celda completamente “llena” es la capacidad en Coulomb, que puede obtenerse multiplicando “C” (lo que se vio antes como tasa de descarga) expresada en amperios-hora por 3600 (el número de segundos en una hora), independientemente de la tensión nominal de la celda. Por ejemplo, una celda con una capacidad nominal de 20Ah contiene una carga de $20 \times 3600 = 72000$ Coulombs.

2.1.5 Energía contenida

La energía contenida depende de la tensión y se expresa en vatios-hora (o en julios). Para obtener la cantidad de energía contenida, es suficiente multiplicar la capacidad de la batería por la tensión. Por ejemplo, una celda de 12 V con una capacidad de 20 Ah contiene una energía igual a $20 \times 12 = 240$ Wh [7].

Cuando se debe elegir una batería, el requisito principal a considerar es la energía que puede almacenar; son igualmente importantes la tensión y la corriente que puede suministrar a un consumidor o impedancia. En la selección del tipo de batería que se desea utilizar, es necesario comparar los precios en función de la energía almacenada por las celdas en lugar de la cantidad de carga, ya que esto último resulta engañoso [7].

2.1.6 Ciclo de vida (vida útil)

Es el número de veces que puede ser cargada y descargada antes de agotarse. Está fuertemente influenciado por el tipo de composición química de las celdas y las condiciones en las que se utiliza la batería [7].

Para caracterizar la vida útil de una celda de Li-ion, lo que se hace es una descarga desde el valor máximo hasta el valor mínimo de voltaje y luego se realiza el proceso de carga. Hacer esto repetidamente genera los valores necesarios para aplicar el método de conteo de Coulomb [1].

2.2 Estado de carga (SoC)

El estado de carga de una batería (SoC) es la relación porcentual entre la carga existente en la batería en un momento dado y su carga nominal. No se puede medir de forma directa el estado de carga de la batería, por este motivo se emplean métodos de estimación a partir de mediciones de otras variables, como la tensión en bornes y la corriente de la batería [11].

El método más adecuado para estimar el SoC consiste en medir la corriente e integrarla en el tiempo, para encontrar el total de la carga suministrada o utilizada durante un evento (carga o descarga). Sin embargo, este método presenta dos desventajas:

- En primer lugar, dado que no existen mediciones perfectas, las tolerancias de los instrumentos de medición provocan que la estimación del SoC presente errores en mediciones con tiempos prolongados.
- En segundo lugar, la integración en el tiempo de la corriente no cuenta con un punto de referencia inicial, razón por la cual no se puede identificar el SoC, en un momento dado, si no se cuenta con la información del estado inicial de carga [11].

A pesar de su desventaja, se emplea para realizar comparaciones entre otras técnicas de estimación del SoC, ya que las pruebas se pre-acondicionan para garantizar un SOC del 0 % o del 100 % al inicio de cada medición [11].

2.3 Estado de salud (SoH)

Refleja la capacidad actual de almacenamiento de energía de la batería en comparación con una nueva. Una estimación precisa del SoH en las baterías puede prolongar su tiempo de funcionamiento. Sin embargo, el SoH de una celda usada está altamente relacionado con sus condiciones operativas anteriores y es difícil de evaluar y predecir [12]. Es por ello por lo que obtener el SoH depende casi completamente del SoC.

Se han propuesto distintos métodos para estimar el estado de salud (SoH) en la literatura, sin embargo, para este proyecto el método de estimación propuesto se basa en el método de conteo de Coulomb (MCC). Este proporciona una forma precisa y fácil de usar, de la estimación del estado de la batería, sin generar una gran base de datos del tipo específico de batería probada [13]. El SoH

y el SoC de las baterías pueden actualizarse controlando y monitoreando sus comportamientos operativos sin interrumpir la operación del sistema.

Debido al proceso limitado de fabricación, la capacidad real completa de una batería nueva puede diferir de la capacidad nominal de la batería declarada por el fabricante. Por esta razón, la capacidad máxima original, $C_{m\acute{a}x}$, de una batería nueva se obtiene promediando las capacidades completas de varios ciclos de vida útil [12]. En este caso, al contar con celdas usadas se toma como $C_{m\acute{a}x}$ la capacidad nominal dada por el fabricante. Cuando una batería comienza a envejecer conduce a un deterioro del rendimiento, lo que resulta en la disminución de la capacidad máxima de carga/descarga, C_{actual} . El grado de degradación de la capacidad de la batería puede ser indicado por el SoH, que se define como el porcentaje de la capacidad al envejecer en relación con la capacidad de carga máxima original [12].

$$SoH = \frac{C_{actual}}{C_{m\acute{a}x}} \cdot 100\% \quad (5)$$

2.4 Método de conteo de Coulomb

Las mediciones hechas por el método de conteo de Coulomb (MCC) permiten obtener un error de medición de menos del 1%, dejando claro una indicación muy precisa de la energía que queda en la batería. Es independiente de las fluctuaciones de energía de la batería (que provocan caídas de voltaje de la batería), y la precisión permanece constante independientemente si la celda está usada o no [14].

Hacer la monitorización del SoH es fundamental para optimizar el rendimiento y seguridad de celdas Li-ion. Esto se agiliza usando el MCC, pero, como se explicó antes, el SoH depende casi completamente del SoC. Es así que, aplicando el MCC, se calcula el SoC con la siguiente ecuación [15]:

$$SoC(t_1) = SoC(t_0) - \frac{1}{C_{actual}} \int_{t_0}^{t_1} i_{med}(t) \partial t \quad (6)$$

Se despeja el C_{actual} de la ecuación 6 para obtener lo siguiente:

$$C_{actual} = \frac{\int_{t_0}^{t_1} i_{med}(t) \partial t}{SoC(t_0) - SoC(t_1)} \quad (7)$$

Donde i_{med} es la corriente medida (positiva durante la descarga por convención) y $SoC(t_0)$ es el valor inicial del SoC. Básicamente, la idea del algoritmo de conteo de Coulomb es hacer una sumatoria de la corriente medida en descarga, tomando en cuenta el SoC (inicial) desde que la celda estaba nueva hasta mucho tiempo después (el SoC final) y utilizarlos para estimar la capacidad experimental, C_{actual} .

2.5 Arduino

Arduino es una plataforma electrónica de código abierto basada en hardware y software fáciles de usar. Las placas Arduino pueden leer entradas (luz en un sensor, un dedo en un botón o un mensaje de Twitter) y convertirlas en una salida: activar un motor, encender un LED o publicar algo en línea. Se puede configurar enviando un conjunto de instrucciones al microcontrolador de la placa. Para ello se utiliza el lenguaje de programación Arduino (basado en Wiring), y el Software Arduino (IDE), basado en Processing. [16]

A lo largo de los años, Arduino ha estado en miles de proyectos, desde objetos cotidianos hasta complejos instrumentos científicos. Una comunidad mundial de creadores (estudiantes, aficionados, artistas, programadores y profesionales) se ha reunido en torno a esta plataforma de código abierto; sus contribuciones han sumado una increíble cantidad de conocimiento accesible que puede ser de gran ayuda tanto para principiantes como para expertos. [16]

Arduino nació en el Ivrea Interaction Design Institute como una herramienta sencilla para la creación rápida de prototipos en electrónica y programación. Tan pronto como se amplió su mercado, Arduino comenzó a cambiar para adaptarse a nuevas necesidades y desafíos, diferenciando su oferta desde simples placas de 8 bits hasta productos para aplicaciones de IoT, dispositivos portátiles, impresión 3D y entornos integrados. [16]

El software Arduino es sencillo para principiantes, pero lo suficientemente flexible para usuarios avanzados. Se ejecuta en Mac, Windows y Linux. Arduino es una herramienta clave para

aprender cosas nuevas. Cualquiera puede empezar a experimentar simplemente siguiendo las instrucciones paso a paso de un kit o compartiendo ideas en línea con otros miembros de la comunidad Arduino. [16]

Hay muchos otros microcontroladores y plataformas de microcontroladores disponibles para computación física. Parallax Basic Stamp, BX-24 de Netmedia, Phidgets, Handyboard del MIT y muchos otros ofrecen funciones similares. Todas estas herramientas toman los complicados detalles de la programación de microcontroladores y los envuelven en un paquete fácil de usar. Arduino también simplifica el proceso de trabajo con microcontroladores, pero ofrece ciertas ventajas ante otros similares [16]:

- **Económico:** La plataforma es relativamente económica en comparación con otras plataformas de microcontroladores. La versión menos costosa del módulo Arduino se puede ensamblar a mano, e incluso los módulos Arduino preensamblados cuestan menos de 50 dólares.
- **Multiplataforma:** El software Arduino (IDE) se ejecuta en los sistemas operativos Windows, Macintosh y Linux. La mayoría de los sistemas de microcontroladores están limitados a Windows.
- **Entorno de programación simple y claro:** El software Arduino (IDE) es fácil de usar para principiantes, pero lo suficientemente flexible como para que los usuarios avanzados también lo aprovechen. Para los profesores, está convenientemente basado en el entorno de programación Processing, por lo que los estudiantes que aprenden a programar en ese entorno estarán familiarizados con el funcionamiento del IDE de Arduino.
- **Software de código abierto y extensible:** Se publica como herramientas de código abierto, disponibles para que programadores experimentados puedan ampliarlo. El lenguaje se puede ampliar a través de bibliotecas C++, y las personas que quieran comprender los detalles técnicos pueden dar el salto de Arduino al lenguaje de programación AVR C en el que se basa.
- **Hardware extensible y abierto:** los planos de las placas Arduino se publican bajo una licencia Creative Commons, por lo que los diseñadores de circuitos experimentados pueden crear su propia versión del módulo, ampliándolo y mejorándolo. Incluso los usuarios

relativamente inexpertos pueden crear la versión de placa del módulo para comprender cómo funciona y ahorrar dinero.

2.5.1 UNO

A lo largo de los años, Arduino ha lanzado más de 100 productos de hardware: placas, escudos, soportes, kits y otros accesorios. Es por eso que también se clasifica en “familias”. En la familia “Clásica” se encuentran boards como Arduino UNO y otros como Leonardo y Micro. [17]

Arduino UNO es una placa microcontroladora basada en ATmega328P. Dispone de 14 pines de entrada/salida digitales (de los cuales 6 se pueden utilizar como salidas PWM), 6 entradas analógicas, un resonador cerámico de 16 MHz (CSTCE16M0V53-R0), una conexión USB, un jack de alimentación, un cabezal ICSP y un botón de reinicio. [17]

“Uno” significa uno en italiano y fue elegido para marcar el lanzamiento del software Arduino (IDE) 1.0. La placa UNO y la versión 1.0 del software Arduino (IDE) fueron las versiones de referencia de Arduino, ahora evolucionadas a versiones más recientes. UNO es la primera de una serie de placas USB Arduino y el modelo de referencia para la plataforma Arduino. [17]

La placa se puede alimentar mediante la conexión USB o con una fuente de alimentación externa. La fuente de alimentación se selecciona automáticamente. La alimentación no USB puede provenir de un adaptador de CA a CC o de una batería. El adaptador se puede conectar conectando un enchufe positivo central de 2,1mm al conector de alimentación de la placa. Los cables de una batería se pueden insertar en los cabezales de GND y Vin del conector POWER. [17]

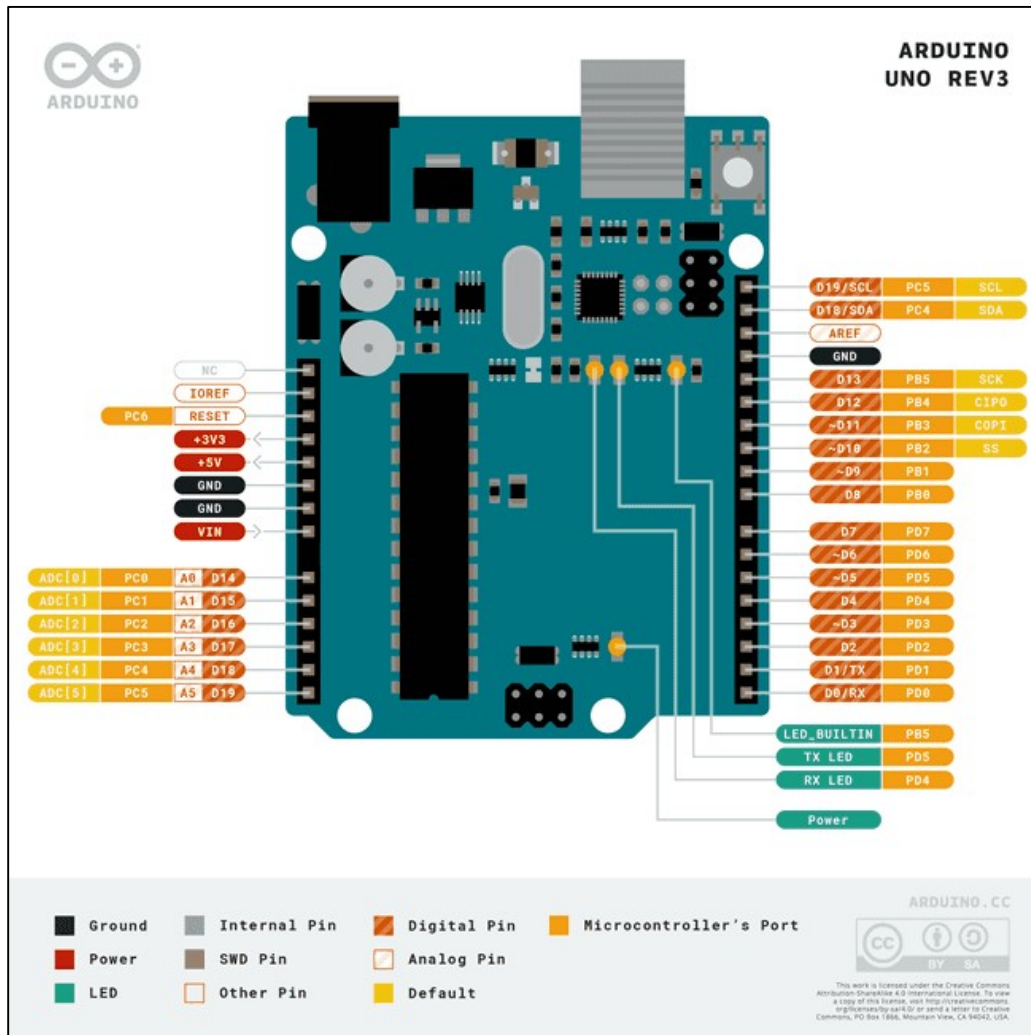


Figura 2. Diagrama de pines del Arduino Uno (Rev3). [17]

Puede funcionar con una alimentación externa de 6 a 20V. Sin embargo, si se suministra menos de 7 V, el pin de 5 V puede suministrar menos de cinco voltios y la placa puede volverse inestable. Si usa más de 12 V, el regulador de voltaje puede sobrecalentarse y dañar la placa. El rango recomendado es de 7 a 12 voltios. Los pines de alimentación son los siguientes [17]:

- Vin: El voltaje de entrada cuando se utiliza una fuente de alimentación externa (a diferencia de los 5V de la conexión USB u otra fuente de alimentación regulada). Puede suministrar voltaje a través de este pin o, si suministra voltaje a través del conector de alimentación, acceder a él a través de este pin.
- 5V: Este pin genera 5V controlados desde el regulador. La placa puede recibir alimentación desde el conector de alimentación de CC (7 - 12 V), el conector USB (5 V) o el pin VIN

de la placa (7-12 V). El suministro de voltaje a través de los pines de 5 V o 3,3 V omite el regulador y puede dañar la placa.

- 3.3V: Un suministro de 3,3V generado por el regulador.
- Tierra: Pasadores de tierra.
- IOREF. Proporciona la referencia de voltaje con el que opera el microcontrolador. Puede leer el voltaje del pin IOREF y seleccionar la fuente de alimentación adecuada o permitir que los traductores de voltaje en las salidas funcionen con 5 V o 3,3 V.

Cada uno de los 14 pines digitales se puede utilizar como entrada o salida, utilizando las funciones `pinMode()`, `digitalWrite()` y `digitalRead()` en el IDE. Funcionan a 5V. Cada pin puede proporcionar o recibir 20 mA y tiene una resistencia pull-up interna (desconectada de forma predeterminada) de 20-50 kΩ. Un máximo de 40 mA no se debe exceder en ningún pin de E/S para evitar daños permanentes al microcontrolador. [17] Además, algunos pines tienen funciones especializadas [17]:

- Serie: 0 (RX) y 1 (TX). Se utiliza para recibir (RX) y transmitir (TX) datos serie TTL.
- Interrupciones externas: 2 y 3. Se pueden configurar para activar una interrupción en un valor low, un flanco ascendente o descendente, o un cambio de valor.
- PWM: 3, 5, 6, 9, 10 y 11. Proporciona salida PWM de 8 bits con la función `analogWrite()`.
- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK), admiten la comunicación SPI.
- LED: 13. Hay un LED incorporado controlado por el pin digital 13. Cuando el pin está HIGH, el LED está encendido, cuando el pin está LOW, está apagado.
- TWI: pin A4 o SDA y pin A5 o SCL, admite comunicación TWI.

El UNO tiene 6 entradas analógicas, etiquetadas de A0 a A5, cada una de las cuales proporciona 10 bits de resolución (es decir, 1024 valores diferentes). Por defecto miden desde tierra hasta 5 voltios, aunque es posible cambiar el extremo superior de su rango usando el pin AREF y la función `analogReference()` [17]. Existen un par de pines más en el tablero [17]:

- AREF. Tensión de referencia para las entradas analógicas. Usado con `analogReference()`.
- Reset: Resetea la información del microcontrolador.

Tabla 1. Especificaciones técnicas del Arduino Uno (Rev3). [17]

Microcontrolador	ATmega328P
Voltaje nominal	5V
Voltaje recomendado	7-12V
Voltaje (límite)	6-20V
Digital I/O Pins	14 (6 proveen PWM output)
PWM Digital I/O Pins	6
Analog I/Pins	6
Corriente CD por I/O Pin	20 mA
Corriente CD en 3.3V Pin	50 mA
Memoria Flash	32 KB
SRAM & EEPROM	2 KB & 1 KB
Frecuencia reloj	16 MHz
LED BUILTIN	13
Largo por ancho	68.6 mm x 53.4 mm
Peso	25 g

2.5.2 IDE

El Entorno de Desarrollo Integrado Arduino - o Software Arduino (IDE) - contiene un editor de texto para escribir código, un área de mensajes, una consola de texto, una barra de herramientas con botones para funciones comunes y una serie de menús. Se conecta al hardware Arduino para cargar programas y comunicarse con ellos. [18]

Los programas escritos con el software Arduino (IDE) se denominan bocetos/sketches. Estos bocetos se escriben en el editor de texto y se guardan con la extensión de archivo .ino. El editor tiene funciones para cortar/pegar y buscar/reemplazar texto. El área de mensajes brinda comentarios al guardar y exportar y también muestra errores. [18]

La consola muestra el texto generado por el software Arduino (IDE), incluidos mensajes de error completos y otra información. La esquina inferior derecha de la ventana muestra la placa configurada y el puerto serie. Los botones de la barra de herramientas permiten verificar y cargar programas, crear, abrir y guardar bocetos y abrir el monitor serie. [18]



Figura 3. Botones de la barra de herramientas en IDE.

De izquierda a derecha de la Figura 3, los botones son los de “Verify”, “Upload”, “New file”, “Open”, “Save file” y “Serial monitor”. Verify revisa el código en búsqueda de errores mientras lo compila, Upload compila el código y luego lo sube a la board de Arduino que esté conectada, New file crea un nuevo sketch, Save file guarda el sketch y el Serial Monitor es el que despliega los valores al compilar el código (viene siendo como el entorno del IDE). [18]

Para desplegar en el Serial Monitor se escoge la velocidad en baudios en el menú desplegable que coincida con la velocidad pasada a Serial.begin del sketch. Se debe tomar en cuenta que con cualquier sistema de computadora (Windows, Mac o Linux) la placa se reiniciará (volverá a ejecutar el sketch) cuando se active el monitor serie. [18]

Capítulo 3

Diseño y verificación del sistema

En este capítulo se describen especificaciones técnicas del hardware seleccionado, incluyendo la lista de componentes del circuito necesario para el sistema de adquisición de datos; conexión del circuito con el Arduino, la vinculación entre el código fuente desarrollado en IDE con el Arduino UNO. Para el monitoreo del comportamiento del proceso de descarga, se hace uso de Microsoft Excel, con la extensión DataStreamer [19], que permite conectar el Arduino UNO al Excel, para obtener los datos que mide con ayuda del código.

3.1 Configuración

El proyecto se centra en el diseño de un sistema de caracterización del SoH para celdas de Li-ion, entonces hay que definir el punto de partida y conocer el estado de las celdas seleccionadas (las celdas 18650 en este caso) para que sean representativas y estén en condiciones adecuadas. Los criterios para seleccionar estas celdas serán los siguientes:

- Capacidad nominal: Este criterio es para garantizar una comparación significativa. Esta se toma de los datos del fabricante, dato verificable en la hoja de datos de la celda disponible en el repositorio [20], siendo de 2600mAh.
- Carga final: Este criterio es para asegurar condiciones uniformes de finalización y para bienestar de la celda. La carga final se estima en 3,0 V.
- Estado físico de salud: Este criterio se usa para determinar las celdas dañadas físicamente previo al experimento (por ejemplo, si vienen abiertas). Se escogen 2 celdas (Con nombres asignados de A1A y A1B) de un banco de celdas desechado.
- Carga inicial: Este criterio es para asegurar condiciones uniformes de inicio. La carga inicial se estima en 4,2 V, según hoja de datos.

3.2 Hardware del sistema

Para construir el hardware del sistema fue necesario contar con una base de batería en la que cupiera la celda 18650, cables o jumpers pequeños, una base donde soldar el circuito de manera que no se esté moviendo, un interruptor, un regulador de voltaje, un disipador de calor, conectores, 2 resistencias, 2 relés, 1 sensor de corriente y materiales de soldadura.

Es necesario contar un Arduino UNO para hacer la plataforma experimental que tomará los datos de interés de las celdas. La celda a utilizar será una LIR18650 de 2600 mAh y 4,2 V, tales como la de la Figura 4. Su hoja de datos está disponible en el repositorio [20].



Figura 4. Celda 18650 usada.

3.1.1 Diagrama del circuito

Para entender la funcionalidad del circuito y lo que se quiso conseguir se realizó un diagrama del mismo previo a su construcción, de manera que se pueda entender las posiciones correspondientes de cada componente usado:

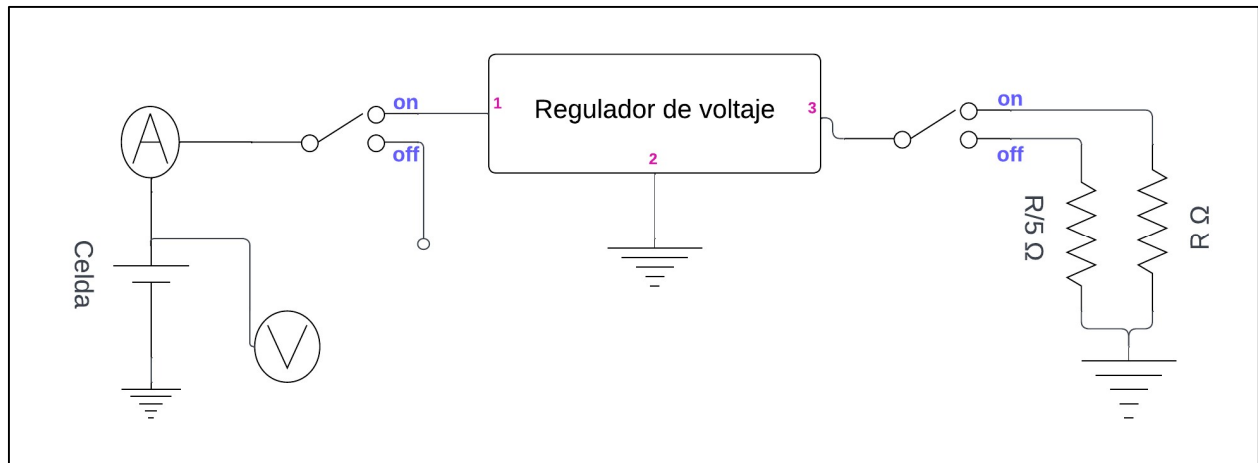


Figura 5. Diagrama del circuito diseñado. Elaboración propia [21].

A la celda se le coloca en paralelo un voltímetro y en serie un amperímetro. De ahí es que se tiene un switch que la permite conectarse al regulador de voltaje o bien, que la desconecte para proteger su SoC. El regulador de voltaje se encarga de proveer el voltaje necesario para mantener una corriente constante de descarga; siendo que un primer pin es la entrada, el segundo pin sería el pin necesario de ajuste, pero en este caso se conecta a tierra y, el tercer pin, sería la salida. El interruptor después del regulador permite “escoger” entre 2 resistencias de manera que se pueda descargar la celda con valores diferentes para distintas pruebas (esto se cumpliría con un segundo relé).

La resistencia R necesita ser una resistencia de potencia; idealmente, para este circuito su valor de impedancia no debe exceder 5Ω , ya que se quiere moldear un proceso de caracterización detallado y que permita ver caídas de voltaje lentas durante el proceso de la descarga. La segunda resistencia, de valor de 1Ω , es la que se usa puesto que logra un proceso más estable y minucioso a la hora de la recopilación de datos necesarios.

Entonces, de una manera generalizada, la resistencia consumidora del circuito debe ser de un valor óhmico pequeño. Si se quiere apreciar diferencias como una descarga más rápida, es recomendable usar una resistencia 5 veces mayor, tal y como en la Figura 5. El proceso de descarga será más lento con la $R/5$ que con la R .

3.1.2 Diseño físico del circuito

Lo que es representado como un amperímetro y el voltímetro en la Figura 5 son los cables de salida del circuito que se conectan al sensor de corriente y a uno de los relés en el hardware físico, que a su vez se convierten en entradas a los pines del Arduino. El primer interruptor es moldeado con un switch real para mantener aislada la celda del resto del circuito para prevenir un desgaste de esta por falta de control, así no es necesario retirar la celda de la base para apagar el circuito.

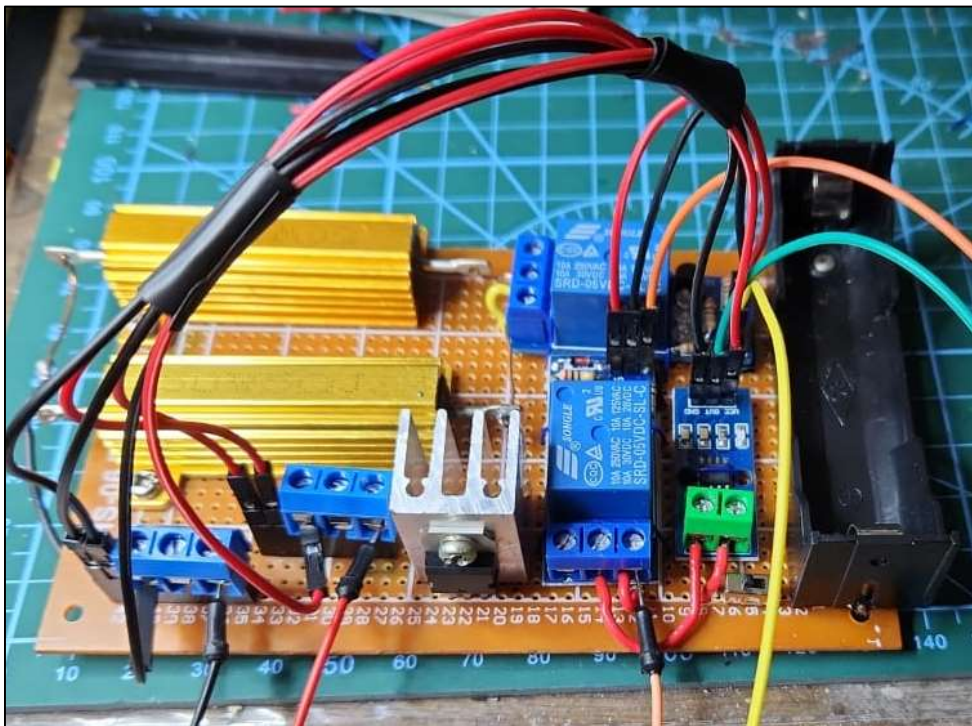


Figura 6. Prototipo del hardware del circuito apagado.

A partir de la Figura 5, se construye el hardware físico, tal y como se muestra en la Figura 6. La base de la celda en su lado positivo se conecta al interruptor pequeño; soldado por debajo, se realizan el resto de las conexiones hacia los relés, el sensor de corriente y las resistencias. Se produce una tierra común entre el lado negativo de la celda y las resistencias, que también se unen a la tierra del Arduino con ayuda de los jumpers.

El hardware considera 2 relés porque uno conecta a las resistencias (R y $R/5$) y el otro es el que permite capturar las medidas del voltímetro. En las pruebas ejecutadas, el circuito va

conectado a la resistencia consumidora de 1Ω . Al regulador de voltaje se le añadió además un disipador de calor para proteger todo el sistema.

Los cables rojos representan la señal de alimentación (en la mayoría de los componentes como Vcc) mientras que los cables negros representan la tierra común de todo el circuito. El cable naranja de la parte inferior de la imagen es el que conduce al pin A0 del Arduino (donde se mide la corriente) y el cable amarillo es el que conduce al pin A1 del Arduino (donde se mide el voltaje).

Para el regulador de voltaje se usó el LM317 [20], este por ser de uso muy común, que facilita conseguirlo y a un costo bajo, además de tener la ventaja de limitar la corriente. En la Figura 7, el primer pin de izquierda a derecha es el pin de ajuste, el segundo el de la salida y el tercero el de la entrada. Su hoja de datos se encuentra en el repositorio.

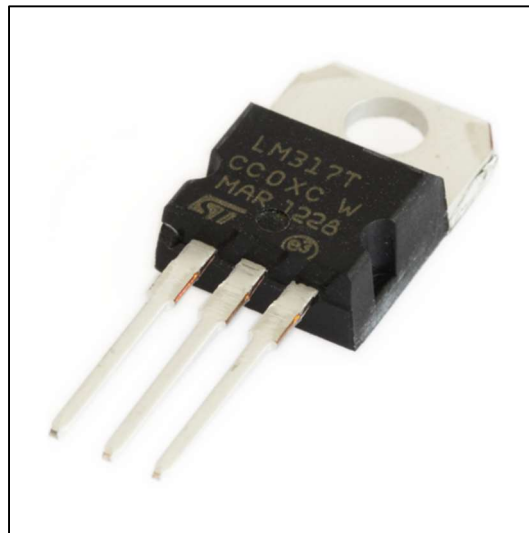


Figura 7. Regulador de voltaje LM317. [21]

El sensor de corriente está dado por el ACS712, que es un módulo para Arduino para medir tanto corriente continua como alterna. Es útil también porque es apto para uso en PCB. El mismo se puede apreciar en la Figura 8. Este sensor también es de bajo costo y su resolución (de 0.185) es bastante pequeña, permitiendo así obtener medidas con un error de cuantización mínimo. Su hoja de datos se encuentra en el repositorio. [20]

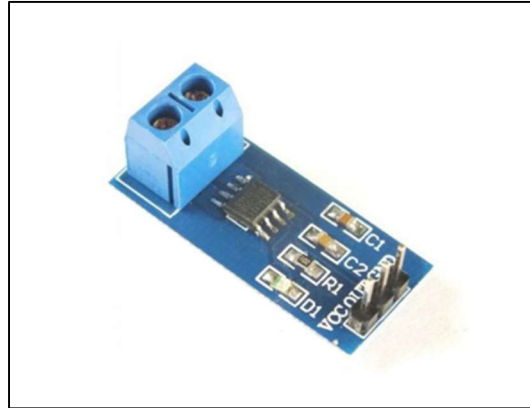


Figura 8. Sensor de corriente. Obtenido de [21].

Para los relés se usan ambos SRD-05VDC-SL-C [20], ya que su uso es común que da lugar a bajo costo, es seguro por su construcción sellada y como es de tamaño pequeño permite la adaptación futura a una PCB. Además, su diseño permite la facilidad de interconexión con el Arduino UNO. Su resolución es de 0.0049 al funcionar como ADC en la toma de datos. Se puede apreciar en la Figura 9.

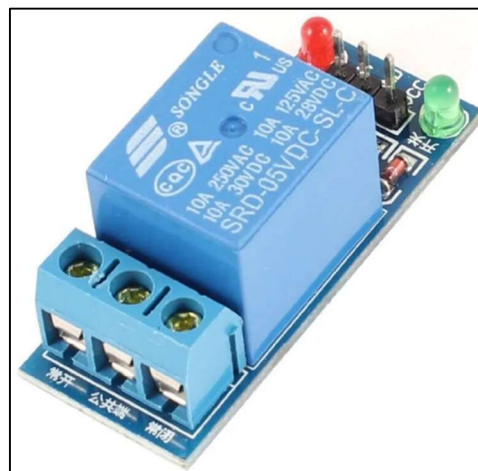


Figura 9. Relay Songle. Obtenido de [21].

3.2 Código fuente

Como el circuito elaborado del punto anterior va conectado a un Arduino UNO que se encargó de la recopilación de datos, se requiere usar el software de Arduino (IDE). Para obtener

esto, fue necesario la descarga de la aplicación de Arduino IDE para Windows 7 o más, 64bits, en su versión 1.8.19 [22].

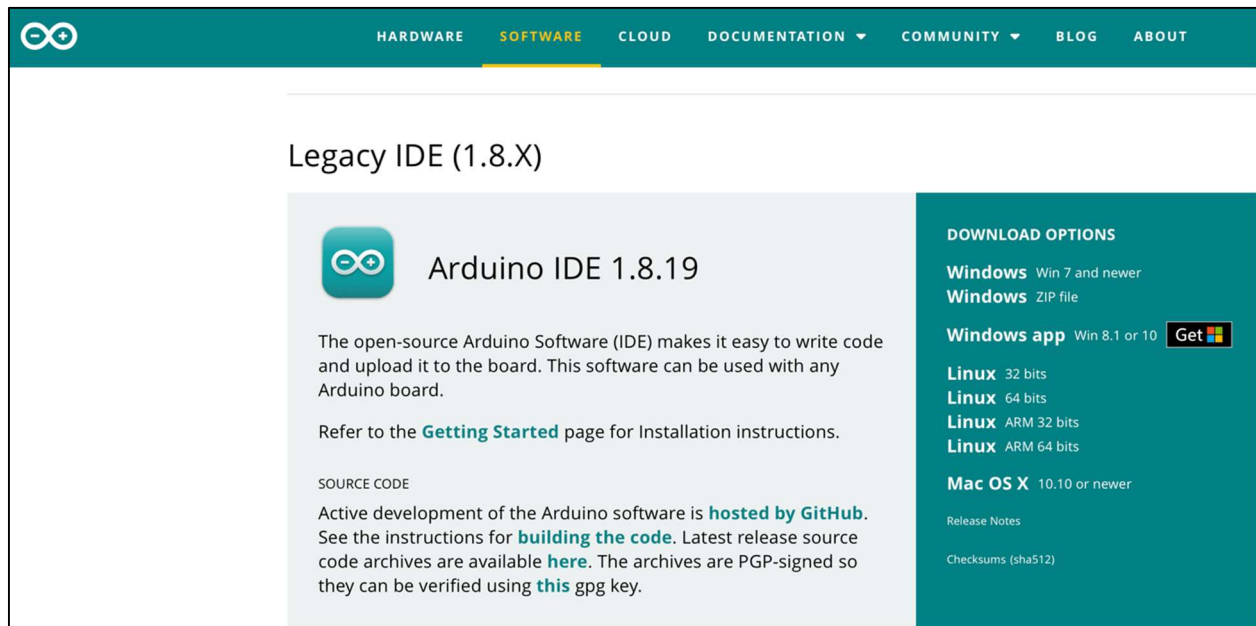


Figura 10. Página de Arduino, sección de Software. [22]

Se descargó en su modelo de Legacy, ya que la versión 2.2.1 que salió en agosto 2023 presentaba al momento de ejecución del sistema muchos bugs.

3.2.1 Diagrama de flujo

Para la creación del código se elaboró primeramente el diagrama de flujo de la Figura 11. Los bloques naranjas representan eventos sin acción posterior/anterior; los bloques grises representan el proceso/flujo del código; los bloques amarillos representan subprocessos dentro del código (las funciones); las líneas azules representan una repetición del proceso con una condición distinta a un sí/no; finalmente, las líneas verdes representan el flujo de las decisiones cuando la respuesta es un sí y las líneas rojas representan cuando la respuesta es un no.

En esencia, se declaran las variables necesarias para el código, luego se inicializan algunas de esas variables como la resolución a usar en cada medición. En código de Arduino siempre se cuenta con la función de `setup()` que es donde se inicializan los pines y se asignan, por lo que en este caso se configuran los pines a usar que son los pines A1 (voltaje) y A0 (corriente), además de inicializar el estado del relay en el pin 3.

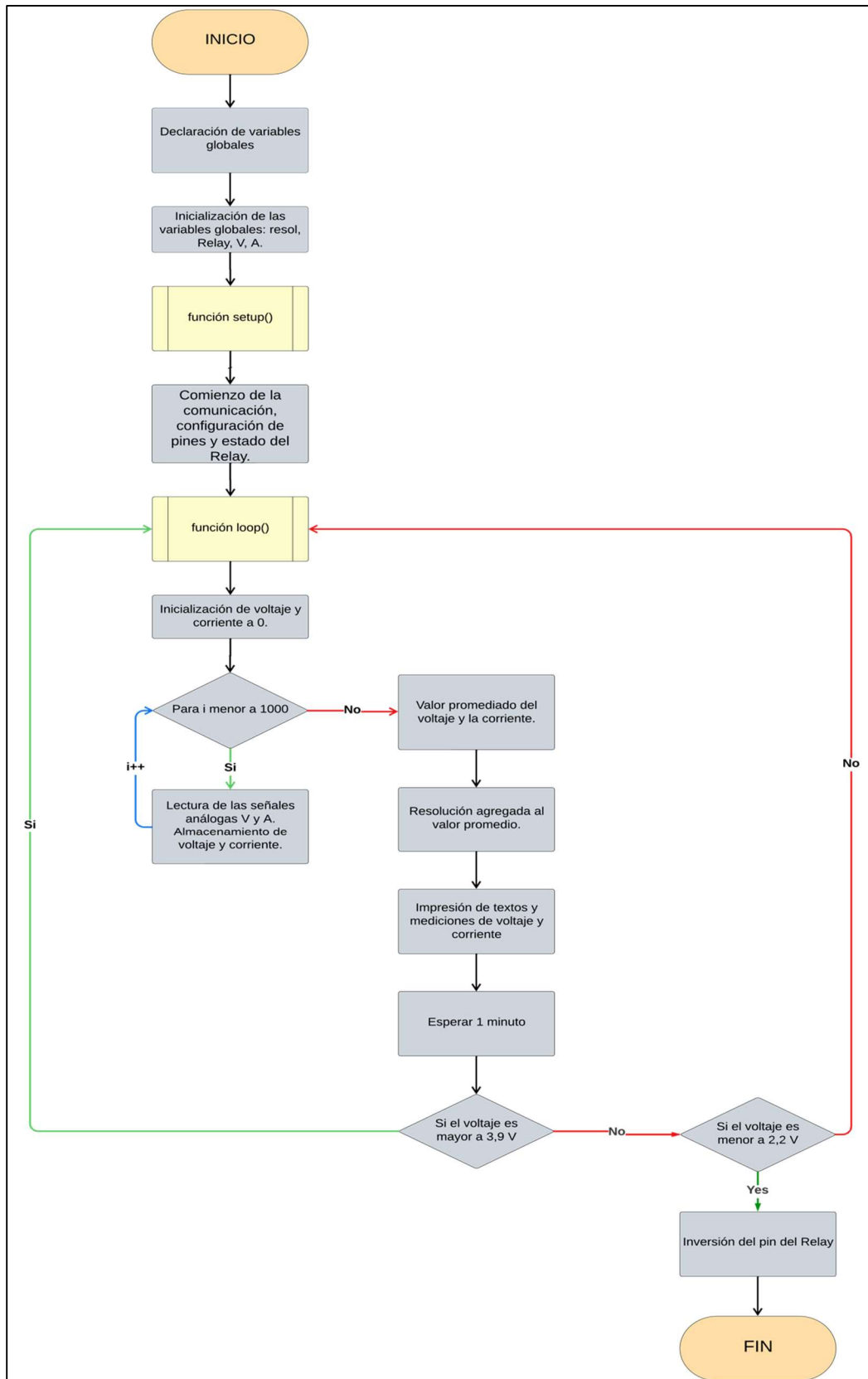


Figura 11. Diagrama de flujo del código. [23]

Asimismo, en el Arduino IDE se cuenta siempre con la función `loop()`, que es lo que el Arduino va a correr continuamente si no es detenido por el usuario. Dentro de la función `loop()`, se inicializa el voltaje y la corriente, de manera que se pueda leer las señales análogas de voltaje y corriente, cuando el contador sea menor a 1000, y se puedan almacenar para usar ese valor final y promediarlo a lo largo de todas las muestras tomadas. Se hace la conversión entre ADC y luego se agrega la resolución a ese valor promedio para sacar un valor real tanto para corriente como para el voltaje. Seguidamente, se imprimen esos valores promediados acompañados de la leyenda de “voltaje” y “corriente” para que el usuario pueda distinguir cada medición.

El código deberá esperar 1 minuto antes de hacer la nueva toma de muestras para permitir una diferencia visible en los resultados obtenidos, pues el proceso de descarga en la zona de seguridad para la celda consume varias horas. Luego de esperar el minuto, el código verifica si la celda está dentro de un margen de SoC cercano al valor nominal y, si es así, repite el ciclo; de lo contrario, el código revisa si es que la celda aún está dentro de un rango seguro de operación; si se obtiene un valor de voltaje menor al de la zona segura lo que hace es invertir el estado del relay para “apagar” el flujo del circuito físico, de lo contrario guía al código a la función `loop()` otra vez.

3.2.2 Pseudocódigo

Previo a la redacción del código fuente, se estableció un pseudocódigo generado a partir del diagrama de flujo. El pseudocódigo se puede encontrar en la Figura 12, el mismo fue redactado en una aplicación en línea. El contenido del código fuente generado después de diseñar el pseudocódigo y se encuentra en el repositorio de GitHub [20].

3.2.3 Entorno de código

Una vez sabiendo lo que se necesita conseguir con ayuda del diagrama de flujo y la construcción del pseudocódigo se procede a generar el código fuente en Arduino IDE. En la Figura 14 se aprecia el código sin comentar en el sketch “*codigoTFGsincomentar.ino*”. Para visualizar el código comentado, el mismo se encuentra en el repositorio. [20]

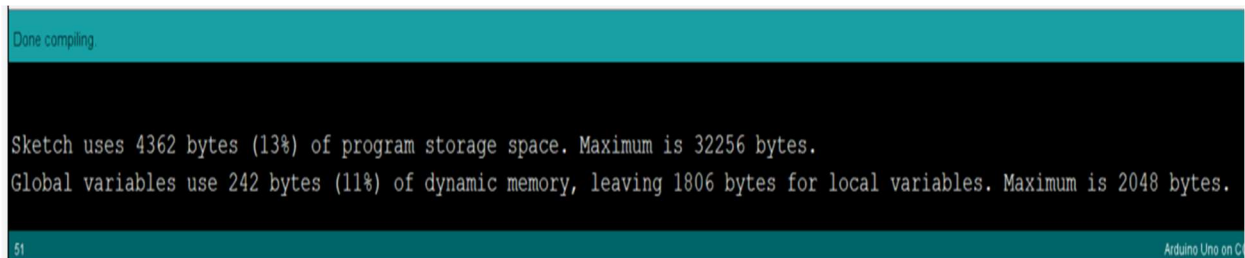
```

1  Algoritmo CodigoTFG //Realizado por Maria Charel Vega 2023
2  // Declarar variables globales
3  Leer lecturaV, volt, voltaje, lecturaA, amp, corriente, i
4  resol ← 0.0049
5  Relay ← 3
6  V ← A1
7  A ← A0
8  Funcion void ← setup (vacio)
9      definir velocidad de comunicacion
10     modalidad del pin A1
11     modalidad del pin A0
12     estado del Relay
13 Fin Funcion
14 Funcion void ← loop (vacio)
15     voltaje ← 0
16     corriente ← 0
17     Para i←0 Hasta 1000 Con Paso i++ Hacer
18         Leer analogo lecturaV. analogo lecturaA
19         Asignar lectura a voltaie
20         Asignar lectura a corriente
21     Fin Para
22     Promediar voltaie
23     Promediar corriente.
24     Imprimir datos necesarios y medidas tomadas
25     Retraso de un minuto
26     Si voltaie mavor a 3.9 Entonces
27         mantener estado del Relay.
28     Fin Si
29     Si voltaie menor 2.2 Entonces
30         invertir estado del Relay.
31     Fin Si
32 Fin Funcion
33 FinAlgoritmo

```

Figura 12. Pseudocódigo del código fuente. Elaborado en [24].

Cuando el código se carga o compila, si no presenta errores de escritura, la consola muestra la salida que se puede ver en la Figura 13.



```

Done compiling.

Sketch uses 4362 bytes (13%) of program storage space. Maximum is 32256 bytes.
Global variables use 242 bytes (11%) of dynamic memory, leaving 1806 bytes for local variables. Maximum is 2048 bytes.

51 Arduino Uno on C

```

Figura 13. Salida de la consola en IDE al compilar.

```
codigoTFGsincomentar.ino | Arduino 1.8.19
File Edit Sketch Tools Help
✓ ↻ 📄 ⬆️ ⬇️
codigoTFGsincomentar.ino
float lecturaV;
float volt;
float voltaje;
float lecturaA;
float amp;
float corriente;
float resol = 0.0049;
int Relay = 3;
int V = A1;
int A = A0;
int i;

void setup() {
  Serial.begin(9600);
  pinMode(V, INPUT);
  pinMode(A, INPUT);
  pinMode(Relay, OUTPUT);
  digitalWrite(Relay, LOW);
}

void loop() {
  voltaje = 0;
  corriente = 0;
  for (i = 0; i < 1000; i++){
    lecturaV = analogRead(V);
    lecturaA = analogRead(A);
    voltaje = lecturaV + voltaje;
    corriente = lecturaA + corriente;
  }
  voltaje = voltaje / 1000.000;
  volt = voltaje * resol;
  corriente = corriente / 1000.000;
  amp = corriente * resol;
  corriente = (2.50 - amp) / 0.185;
  Serial.print("Voltaje: ");
  Serial.print(",");
  Serial.print(volt, 4);
  Serial.print(",");
  Serial.print("Corriente: ");
  Serial.print(",");
  Serial.println(corriente, 4);
  delay(60000);
  if (voltaje > 3.9000)
  {
    digitalWrite(Relay, LOW);
  }
  if (voltaje < 2.2000)
  {
    digitalWrite(Relay, HIGH);
  }
}
```

Figura 14. Código fuente escrito en IDE.

3.3 Procesamiento de datos

Una vez configurado el circuito tanto en hardware como en software, se procede a ejecutar el código y empezar con el procesamiento de los datos.

3.3.1 Configuración de Excel

Si bien el Arduino posee una terminal para observar las muestras de cada una de las pruebas, es necesario pasar estas muestras y sus resultados a una herramienta que permita realizarle un análisis de datos. Para esto, se escoge la herramienta Excel. Es necesario contar con la licencia de Excel, que por consiguiente se utiliza la cuenta institucional del TEC.

Con el Excel abierto, hay que seguir la guía [25] para conseguir la extensión de Data Streamer que se impregna a Excel. Esta herramienta es la que permite extraer datos medidos con el Arduino UNO a tiempo real y a su vez transformarlos a celdas de Excel. Una vez que la extensión ya está instalada, se procede a abrir un libro de Excel y realizar la conexión correspondiente con el dispositivo USB.

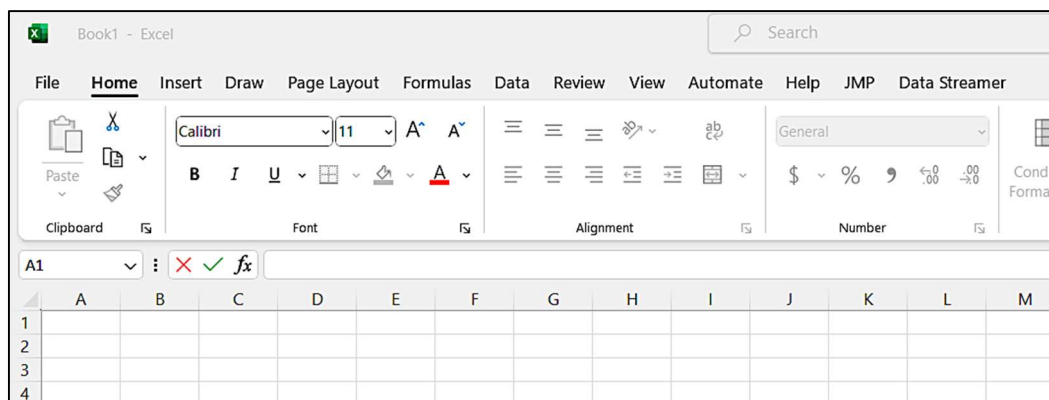


Figura 15. Vista al abrir un nuevo libro de Excel con la extensión DataStreamr ya instalada.

Se debe dar click en el apartado de Data Streamer visible sobre la barra de herramientas del libro de Excel. Una vez allí, en la barra de tareas, será necesario dar click en “Conectar un dispositivo/ Connect a device” para poder seleccionar el dispositivo que se encuentra conectado al puerto USB con el que se puede usar Data Streamer. Aclarar que, la extensión no está disponible en Microsoft Office Excel de MacOS.

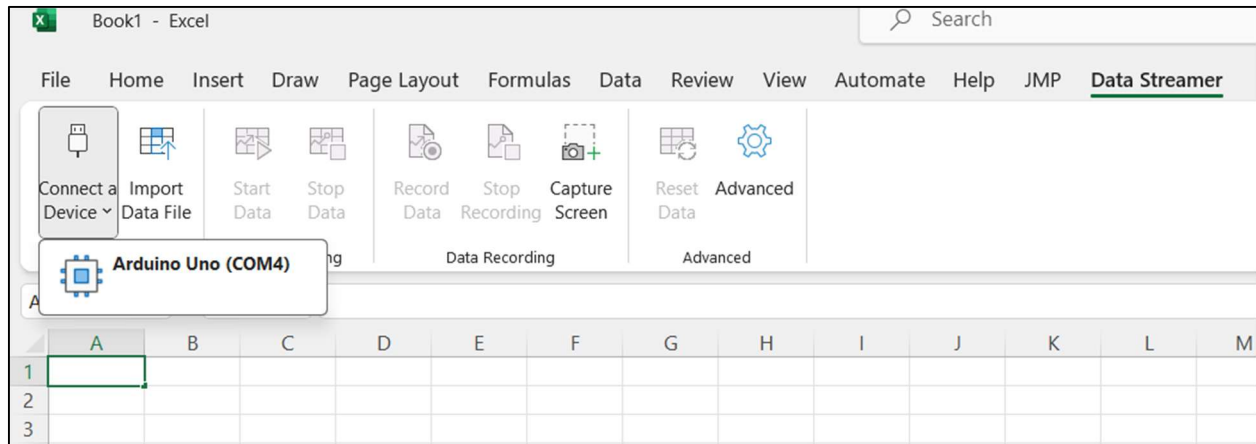


Figura 16. Vista del libro de Excel al configurar el dispositivo.

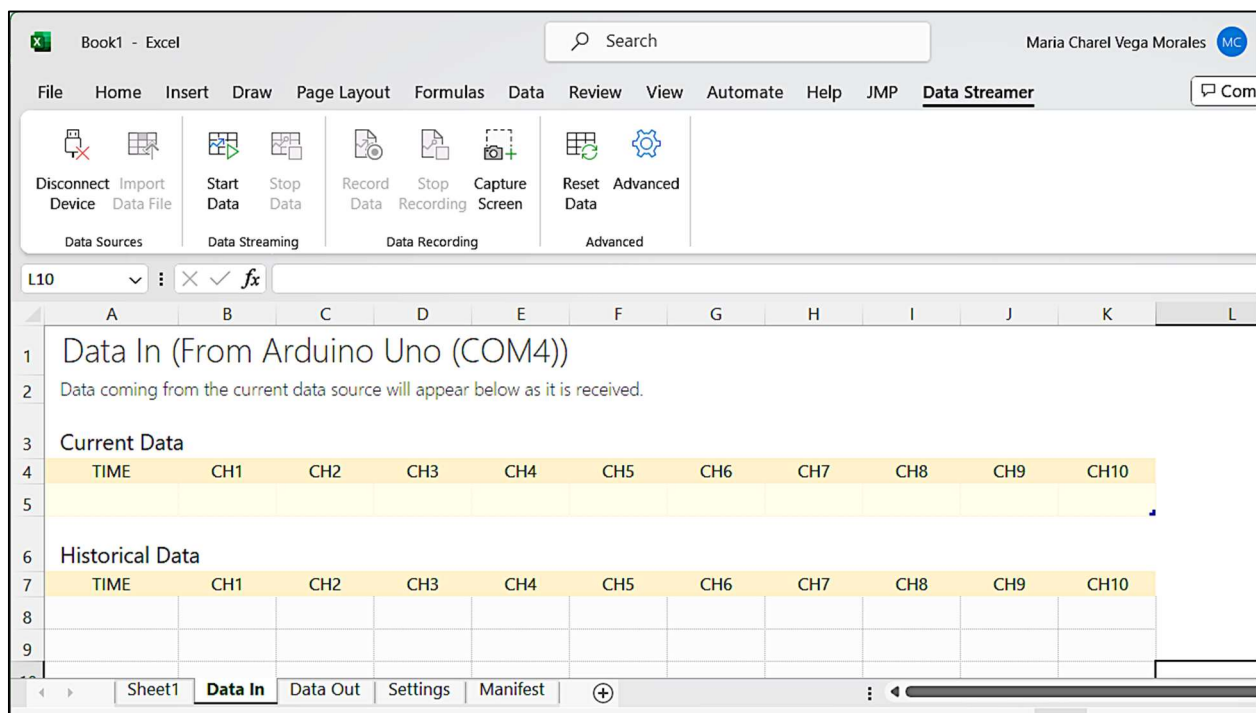


Figura 17. Vista del libro de Excel al enlazar exitosamente el dispositivo.

De manera predefinida por la extensión, la cantidad de canales que se despliegan son 10 y, la cantidad de filas son 15. En las columnas A-K, fila 5 despliega los datos de la última muestra, así como la hora en que los capturó. En las columnas A-K fila 8 y sucesivas, se muestran los datos tomados previamente. El orden de si es el más reciente primero o el más viejo primero se configura en la hoja del libro que se llama “Settings”.

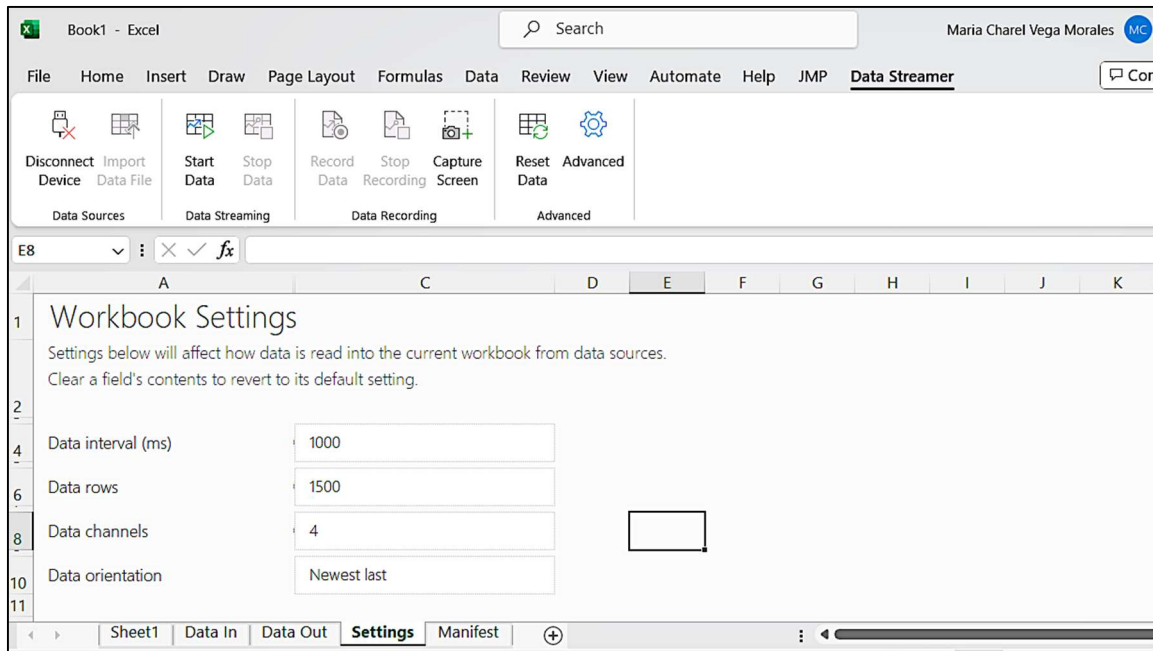


Figura 18. Configuración correcta de la hoja Settings.

En el caso del intervalo de muestras, que se encuentra en milisegundos, se configura en la columna C fila 4 a 1000 ms, de esta manera se hará cada 1 segundo. La cantidad de filas de muestras se configura a 1500 tal que se pueda contar con un espacio amplio de recolecta y no se vayan a quedar muestras por fuera. Este valor de filas designado no es una cantidad fija de filas de muestras, puesto que el rango de descarga de las celdas dura entre 10 y 14 horas y esto se traduce a $10h \times 60min/muestras = 600$ hasta $14h \times 60min/muestras = 840$. Los canales del Data Streamer se configuran a 4, ya que el código despliega cuatro valores (la palabra voltaje, la palabra corriente, el valor de corriente medido y el valor de voltaje medido).

La configuración de cuáles datos se muestra en la parte superior (si los más viejos o los más nuevos) no resulta consecuente en este muestreo, por lo que ambas configuraciones son aceptables. Sin embargo, la configuración de mostrar los más viejos en la parte superior permite mostrar el primer valor medido del proceso de descarga, de manera que se puede ubicar fácilmente si el proceso de descarga de la celda comienza en un valor correcto o no.

Una vez que se cuenta con los aspectos anteriores configurados correctamente se procede a dar click en “Start data/Empezar transmisión” que se encuentra en la barra de herramientas en el mismo apartado de la extensión, como en la Figura 19.

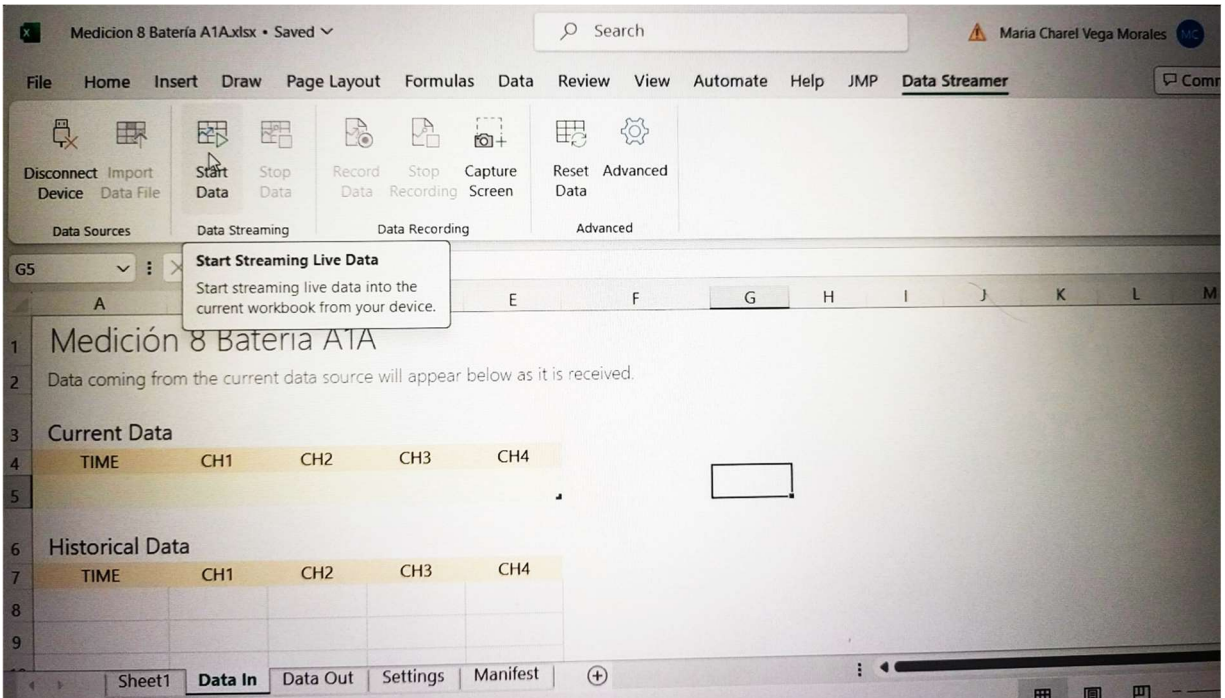


Figura 19. Vista del Start Data en el apartado de la extensión dentro del libro de Excel.

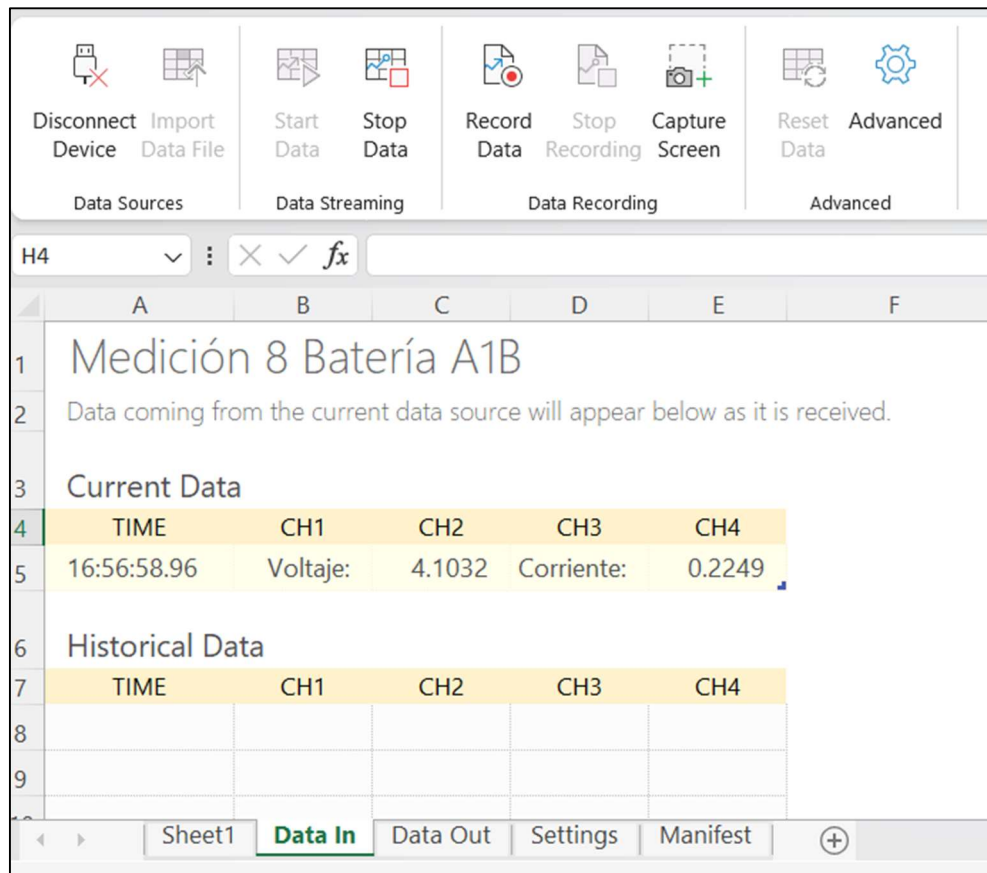


Figura 20. Vista del Data Streamer en funcionamiento.

Capítulo 4

Resultados y análisis

En este capítulo se recopilan los datos tomados por el Arduino UNO y los resultados de la aplicación del algoritmo de conteo de Coulomb, explicando su funcionamiento y los cálculos involucrados en la estimación del SoH de las celdas. También se realizan gráficas para analizar los datos adquiridos.

Se obtendrá la tasa de porcentaje en el cálculo de SoH a partir de las pruebas realizadas mediante análisis estadístico; además, se obtendrá un valor promedio de R_{int} de todo el proceso en cada medición para validar la estrategia de caracterización y relacionarla con el SoH.

4.1 Variables de respuesta

De acuerdo con los aspectos descritos de cada capítulo previo y a un arduo análisis objetivo en función de las limitaciones de tiempo y presupuesto, apegado a la viabilidad, se definieron las siguientes variables de la selección:

1. Corriente de descarga
2. Tiempo de descarga.
3. Voltaje de la celda.
4. Pérdida de capacidad.
5. Curva de voltaje/tiempo.

4.2 Factores y niveles

En diseño de experimentos, se conoce como factor a cada componente del sistema que pueda influenciar de cualquier manera las variables de respuesta. Se muestra a continuación el detalle de los factores y cada uno de sus niveles elegidos para el proyecto.

4.2.1 Temperatura

La idea de variar la temperatura a la cual se encuentran las celdas es debido a que este factor determina el proceso de degradación. Se deseó establecer los rangos de configuración que pueden ser implementados en el sistema mediante la variación de temperatura. Para este factor, se definió 1 nivel:

·ambiente: $25^{\circ}\text{C} \pm 5^{\circ}\text{C}$

4.2.2 Capacidad

La idea de calcular la tasa de descarga a la cual se encuentran las celdas es debido a que este factor determina el proceso de degradación [26]. Se deseó establecer los rangos de configuración que pueden ser determinados en el sistema mediante la estimación de tasa de descarga correcta para las pruebas. Para este factor, se definió 3 niveles:

·baja: menos de 2Ah

·media: entre 2Ah y 2,5Ah

·alta: más de 2,5Ah

4.2.3 Tiempo

La necesidad de determinar el tiempo al cual se descargan las celdas a corriente constante es debido a que este factor afecta directamente la capacidad. Se deseó establecer los rangos de configuración que pueden ser determinados en el sistema para garantizar su rendimiento óptimo y evitar degradaciones innecesarias. Para este factor, se definió 3 niveles de impacto:

·mínimo: 12h

·notable: entre 12,1 y 13,99h

·máximo: 14 h

4.2.4 Resumen de los factores

La Tabla 2 muestra el resumen de los factores y sus niveles del sistema.

Tabla 2. Factores y niveles.

Temperatura	Capacidad	Tiempo
Ambiente	Baja	Mínimo
	Media	Notable
	Alta	Máximo

4.3 Repeticiones del experimento

En cuanto a las combinaciones posibles, según los datos de la Tabla 2, se tuvo el siguiente cálculo:

$$1 \times 3 \times 3 = 9 \text{ combinaciones}$$

Dado que, para experimentos multifactoriales se requieren al menos 2 repeticiones de cada combinación, se propuso una cantidad de 2 repeticiones, siendo suficientes según [27]; con lo que se tendrán un total de:

$$9 \times 2 = 18 \text{ réplicas}$$

4.4 Valores obtenidos del Arduino

Para esta sección, se utilizarán 2 celdas usadas 18650, con los nombres designados de A1A y A1B. Teniendo en cuenta que son 2 celdas, se realizarán mediciones que permitan recoger 9 réplicas bajo los factores y niveles esperados citados también en la sección 4.3. Esto dará un total de 18 réplicas. Las réplicas se ejecutaron con la resistencia consumidora de 1Ω de la Figura 6.

En el proceso de descarga de cada medición, se tomaron en cuenta 11 cálculos, de manera que se pudiera describir el sistema de caracterización de cada celda. Es requisito que la corriente se mantenga constante durante el proceso, sin embargo, como el sistema presenta componentes en el circuito que tienen resolución, se procedió a promediar la misma en cada una de las mediciones.

Por último, a lo largo de la recopilación de datos, se tomaron en cuenta condiciones iniciales tales como: voltaje inicial de cada celda en 4,2V; la capacidad en 2600mAh; voltaje final en 3,0V y R_{int} menor o igual a $70m\Omega$, [10].

1. Gráficas de la celda A1A

Para las 9 réplicas de esta celda, se prosiguió a graficar la curva de voltaje contra tiempo, de manera que se puedan visualizar mejor los datos recopilados del proceso de descarga, ya que al ser un sistema tan minucioso resultó en muchas muestras. Estas gráficas se encuentran en el repositorio. [20]

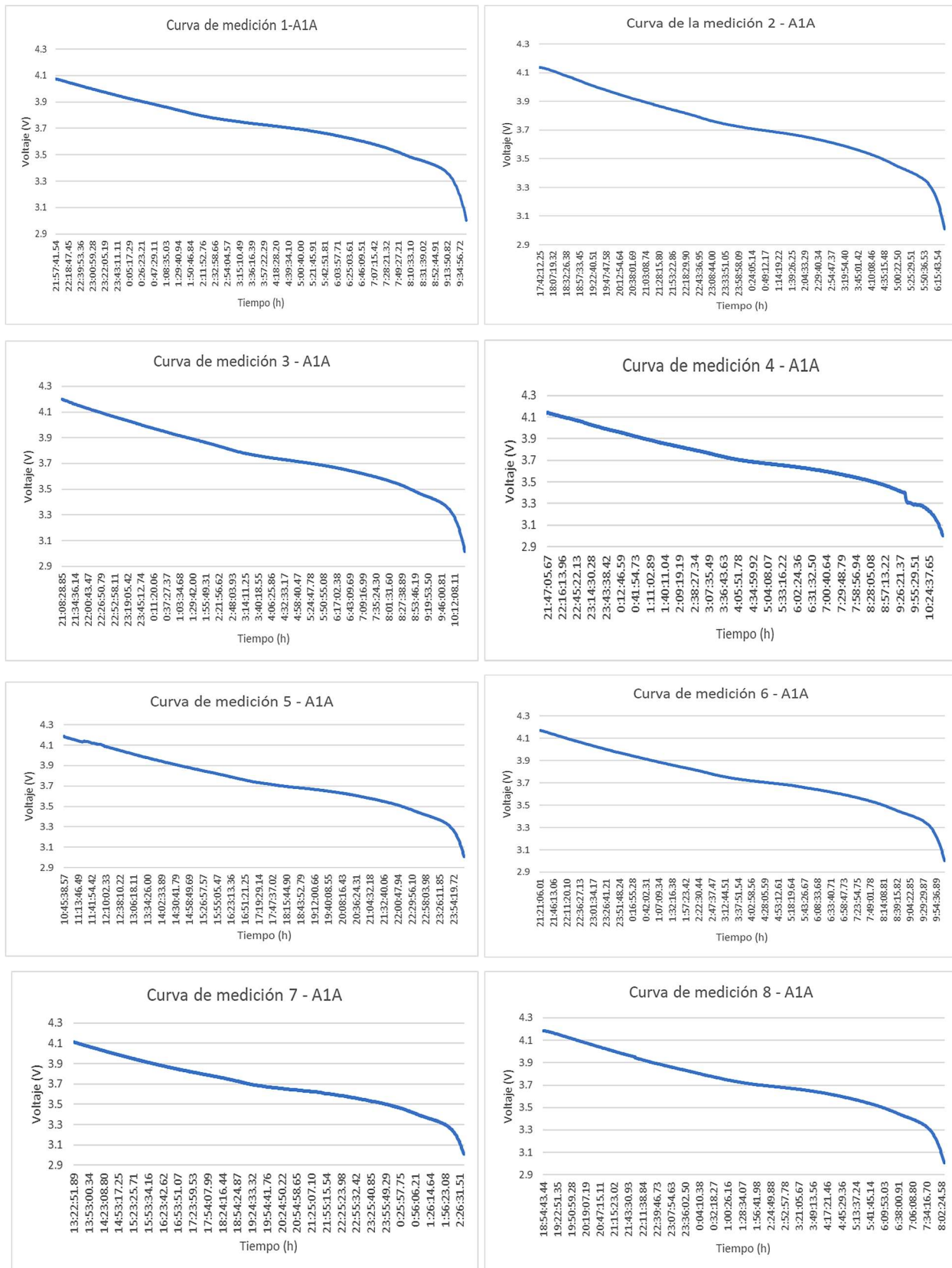


Figura 21. Curvas de voltaje contra tiempo de la celda A1A.

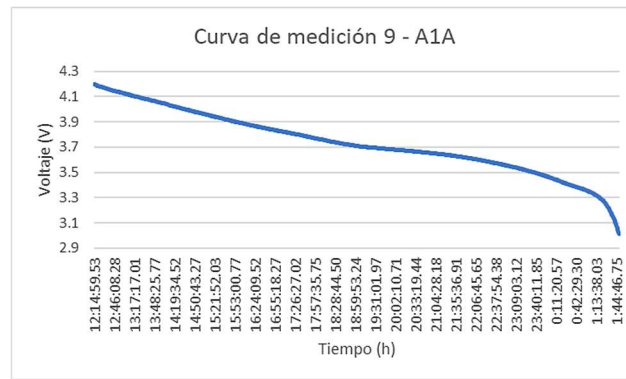


Figura 22. Curvas de voltaje contra tiempo de la celda A1A (continuación).

Comparando las 9 curvas de medición, se ve que, a pesar de que no todas las réplicas tienen el mismo tiempo de descarga, su curva de voltaje contra el tiempo presenta el mismo patrón del proceso. Esto quiere decir que la celda A1A siguió un proceso de descarga correcto sin importar si su estado de carga inicial no fue el máximo en todas las mediciones y por tanto que el sistema funciona. Por ejemplo, según la curva de medición 1, su tiempo de descarga fue aproximadamente 12 horas, donde se ve en la primera muestra que su voltaje estaba muy por debajo del nominal, pero en el proceso alcanza a las 7 horas un voltaje de 3,7V, tal y como en el resto de curvas de descarga. Esta comparación es más visible al realizarlo con las curvas 3, 4 y 6, que tienen horas de inicio similares.

Según la forma de las curvas de mediciones de la celda A1A, se puede verificar que la misma es una celda usada, puesto que el patrón de su curva presenta una decadencia más pronunciada que una celda nueva. Es decir, esta celda comparada con una celda nueva no mantiene su voltaje tan constante. Las celdas nuevas poseen una curva de descarga lineal con una pendiente apenas reconocible (idealmente de 0) hasta llegar a su valor de voltaje de “zona segura”, donde decaen exponencialmente; es por esto que las nuevas pueden proveer un voltaje constante por más tiempo que las celdas usadas. Se ve reflejado la existencia de una pendiente negativa y más grande en el patrón de las curvas medidas.

2. Gráficas de la celda A1B

Para las 9 réplicas de esta celda, se prosiguió a graficar la curva de voltaje contra tiempo, de manera que se puedan visualizar mejor los datos recopilados del proceso de descarga, ya que al ser un sistema tan minucioso resultó en muchas muestras.

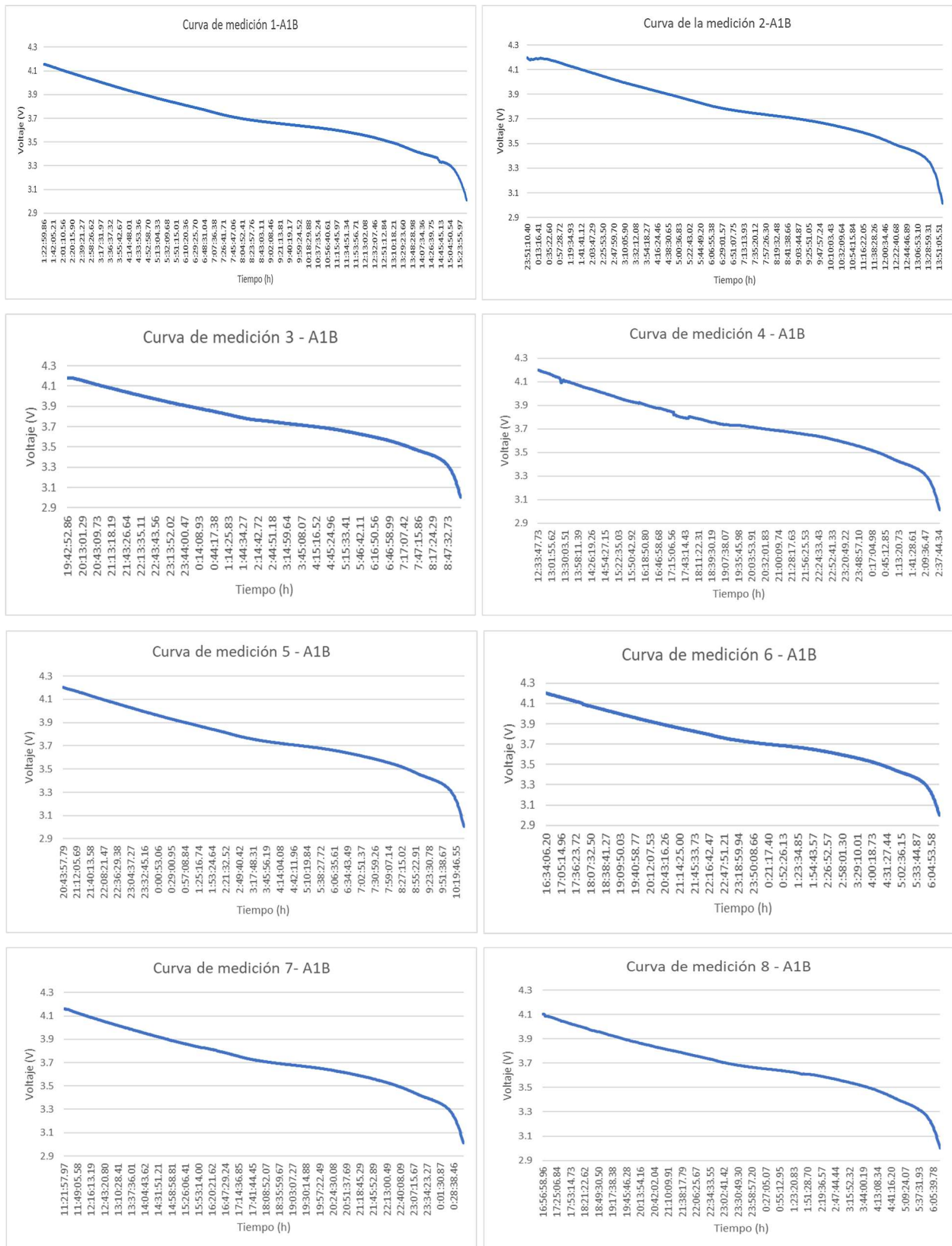


Figura 23. Curvas de voltaje contra tiempo de la celda A1B.

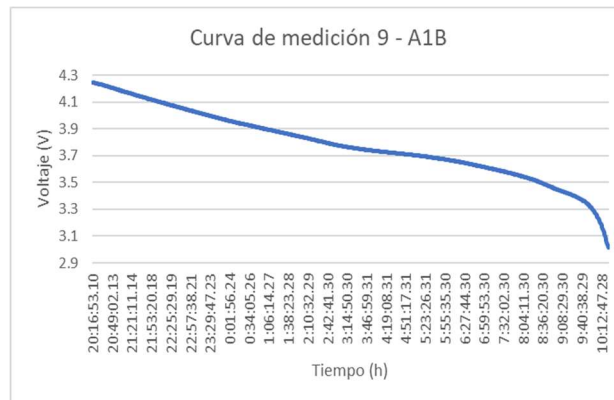


Figura 24. Curvas de voltaje contra tiempo de la celda A1B (continuación).

Comparando las 9 curvas de medición, se ve que, a pesar de que no todas las réplicas tienen el mismo tiempo de descarga, su curva de voltaje contra el tiempo presenta el mismo patrón del proceso. Esto quiere decir que la celda A1B siguió un proceso de descarga correcto, sin importar si su estado de carga inicial no fue el máximo en todas las mediciones y, por tanto, que el sistema funciona. Sin embargo, de la curva de medición 1 se puede verificar que, a pesar de ser la curva que recopiló más muestras de entre las 9 réplicas, es la que decae más rápido; así que no se puede asumir que el hecho que el proceso de descarga tome más tiempo signifique que es porque el voltaje se mantiene más constante; un factor que pudo haber afectado esto es que la corriente de descarga sea un valor menor al esperado.

4.5 Cálculo de energía, carga y tiempo de descarga

A partir de las secciones 2.1.4 y 2.1.5, se procede a realizar el cálculo correspondiente a la energía mínima, energía máxima y carga de la celda, en cada una de las mediciones. Estos cálculos se realizaron en libros de Excel, ya que allí se conservaron los datos transmitidos por el Arduino UNO, como indica el capítulo 3. Los libros de Excel se encuentran disponibles en el repositorio, en un archivo zip. [\[20\]](#)

Lo que es el tiempo de descarga se calcula desde la hora de la primera muestra hasta la hora del último muestreo realizado. Todos estos datos son necesarios para el sistema de caracterización de cada celda. Al final de las 9 réplicas de cada celda se calcula un promedio de

cada uno de los aspectos medidos para estimar un aproximado total que permita establecer una relación entre sí. En la Tabla 3 se contempla un resumen de los datos de las réplicas.

La energía mínima se obtiene del último valor de voltaje medido (en el instante antes de salir de su zona de seguridad) y la energía máxima del primer valor de voltaje medido. El tiempo de descarga se mide desde el momento en que se activa el Data Streamer hasta el momento en el que el voltaje alcanza los 3,0V.

Tabla 3. Resumen de cálculo de energía, carga y tiempo de descarga de las 18 réplicas.

Medición (#)	Muestras (#)	I prom (A)	Energía min (Wh)	Energía máx (Wh)	Carga (C)	Tiempo de descarga (h)
1-A1A	706	0,1522	7,2872	7,2997	6447,618	12,0
2-A1A	765	0,1441	5,5324	7,6008	6614,790	12,5
3-A1A	802	0,1529	6,1654	8,5838	7357,728	13,5
4-A1A	782	0,1683	6,5807	9,0712	7894,521	13,0
5-A1A	807	0,1356	5,4883	7,6455	6567,360	13,5
6-A1A	767	0,1613	6,1952	8,6052	7424,140	12,8
7-A1A	789	0,1900	7,5224	10,2682	8993,862	13,0
8-A1A	791	0,1593	6,3167	8,7844	7560,660	13,0
9-A1A	811	0,1404	5,7180	7,9718	6833,796	13,5
Promedio	780	0,1560	6,3118	8,4256	7299,386	12,98
1-A1B	851	0,1664	7,1150	9,8234	8508,786	14,0
2-A1B	845	0,1800	7,6443	10,6615	9129,060	14,0
3-A1B	808	0,1934	7,8239	10,8835	9377,794	13,5
4-A1B	843	0,1841	7,7909	10,8659	9312,762	14,0
5-A1B	829	0,1860	7,7229	10,7961	9252,072	13,9
6-A1B	824	0,1728	7,1220	9,9654	8541,562	13,9
7-A1B	803	0,1778	7,1632	9,8979	8567,508	13,4
8-A1B	799	0,1787	7,1438	9,7657	8568,042	13,0
9-A1B	844	0,1783	7,5628	10,6479	9030,222	14,0
Promedio	827	0,1797	7,4543	10,3675	8920,868	13,7

Idealmente, si una celda 18650 nueva, con capacidad(C) de 2600mAh, se descarga a una corriente constante de 0,2A (esto representa el 0.1C) significa que la cantidad de muestras ideales son 780, ya que su tiempo de descarga sería 13h. Sabiendo esto, se nota a partir de las réplicas de la celda A1A que el promedio de muestras son 780, lo que alcanza el valor ideal, pero como el

promedio de corriente se encuentra por 0,1560A deja saber que la celda no presenta la misma capacidad que tenía en un principio cuando estaba “nueva”. La celda ahora tarda lo mismo en descargarse dando una corriente menor.

Se distingue de las mediciones de la celda A1A que entre mayor corriente entregue la celda, mayor será su energía mínima y su energía máxima. No es lo mismo para la carga medida en Coulombs, que no se ve estrechamente relacionada con la cantidad de corriente suministrada, tal y como se puede comparar entre la medición 1-A1A y la medición 2-A1A. La cantidad de carga es menor en la 1-A1A por la cantidad de tiempo de descarga que tuvo, no porque la corriente sea mayor que la de 2-A1A.

Es notable al relacionar las mediciones 4-A1A, 7-A1A y 8-A1A que, al ejecutarse su proceso de descarga en el tiempo ideal, resulta en una mayor energía, tanto máxima como mínima.

De entre estos datos, la medición 7-A1A evidencia un mejor comportamiento de la réplica al conseguir 0.1900A, el valor más cercano a la corriente constante de descarga de 0.1C.

A partir de los datos tomados de las 9 réplicas de la celda A1A se resalta que sí afecta en los resultados el hecho que inicien el proceso de descarga a un voltaje diferente del nominal, ya que se consigue menos tiempo de descarga, menos corriente, menos energía y menos Q.

Para lo que serían las mediciones de la celda A1B, se destaca que esta celda contó con el voltaje nominal al inicio de todas sus mediciones, de una manera más aproximada. En las 9 réplicas la corriente nunca bajó de 0,1664A, por lo que se espera una mejor respuesta en cuanto a capacidad total y SoH. De obtener el valor de corriente más cercano a 0,2A, se destaca que para esta celda su energía mínima y máxima en las 9 réplicas no resultan ser tan inestables como sucedió en las 9 réplicas de la celda A1A.

Por la comparación de las celdas A1A y A1B, según la corriente entregada y la energía máxima entregable, se espera un mejor SoH de la celda A1B.

4.6 Cálculo de capacidad, SoC, SoH y resistencia interna

Para el cálculo de estas características, se procede según el capítulo 2. El conteo de muestras y la I_{prom} de cada medición o réplica permite proceder con el cálculo de la capacidad. Los libros de Excel de las réplicas se encuentran disponibles en el repositorio. [\[20\]](#)

Tabla 4. Resumen de cálculo de SoC, resistencia interna y MCC para el SoH, de las 18 réplicas.

Medición (#)	Muestras (#)	Capacidad (Ah)	SoC inicial (%)	SoC exp final (%)	SoH (%)	Rint (mΩ)
1-A1A	706	1,7910	97,0428	71,5095	68,8848	81,4548
2-A1A	765	1,8374	98,4905	71,6881	70,6708	85,5169
3-A1A	802	2,0438	99,9976	71,8238	78,6082	85,0597
4-A1A	782	2,1929	98,4905	71,4500	84,3432	77,3314
5-A1A	807	1,8243	99,7857	71,6310	70,1641	89,9545
6-A1A	767	2,0623	99,3500	71,5262	79,3177	78,1397
7-A1A	789	2,4983	97,8595	71,6905	96,0883	66,0695
8-A1A	791	2,1002	99,5881	71,6119	80,7763	82,9952
9-A1A	811	1,8983	99,9881	71,7190	73,0106	83,0050
Promedio	780	2,0276	98,9548	71,6278	79,1224	80,2128
1-A1B	851	2,3636	98,9571	71,6738	90,9058	66,8401
2-A1B	844	2,5327	99,9476	71,7738	97,4118	66,2640
3-A1B	808	2,6049	99,4762	71,5119	100,000	60,5972
4-A1B	843	2,5869	100,000	71,7071	99,4953	60,6688
5-A1B	829	2,5700	100,000	71,5476	98,8469	61,8929
6-A1B	824	2,3727	100,000	71,4690	91,2560	65,0079
7-A1B	803	2,3799	99,0238	71,6643	91,5332	66,8858
8-A1B	799	2,3800	97,6952	71,4667	91,5389	65,2708
9-A1B	844	2,5084	100,00	71,7857	96,4767	62,9136
Promedio	827	2,4777	99,4555	71,6222	95,2738	64,0379

De los resultados de la capacidad en cada una de las réplicas de ambas celdas se distingue que es necesario que la celda empiece el proceso de descarga con al menos 98% de SoC, ya que tomar mediciones por debajo de este porcentaje dará resultado muestras inestables y no se podrá ejecutar el sistema de caracterización de manera correcta.

La capacidad (C) depende tanto del tiempo de descarga como de la corriente de descarga, por lo tanto, en el caso de la medición 5-A1A a pesar de que empezó con un voltaje casi de 4,2V la corriente promedio de descarga dio alrededor de 0,13A por una reducción de la corriente a las 2 horas de iniciar su proceso de descarga. Revisando las muestras de esta medición, esto pudo deberse a algún problema de comunicación de los dispositivos al momento del proceso de descarga o un movimiento brusco durante el proceso que afectó las medidas de corriente que estaba proporcionando el Arduino, puesto que pasó de estar midiendo en promedio 0,16A a 0,13A.

El promedio de la R_{int} de la celda A1A permite saber que la celda aún tiene vida útil y que puede realizarse más ciclos de carga/descarga, pues su valor no está en valor de centiohmios, sino que se mantiene dentro del rango de los miliohmios, tal y como se discutió en el capítulo 2. De todas las réplicas ejecutadas, tanto para la celda A1A como para la A1B, se puede verificar que a menor R_{int} mayor será el SoH.

La medición 1-A1A representa una medición ejecutada de una manera no correcta, puesto que las condiciones iniciales en esta medición son diferentes a las establecidas y por tanto demuestra un resultado incorrecto y que no se puede utilizar para evaluar el funcionamiento correcto del sistema. Es por esto que en el promedio calculado del SoH y de R_{int} para la celda A1A no se contemplan los valores de dichas respuestas de la medición 1-A1A.

El SoC es el porcentaje de carga (voltaje inicial) respecto al voltaje nominal de la celda (la condición inicial). La capacidad (C) depende tanto del tiempo de descarga como de la corriente de descarga, por lo tanto, en el caso de la medición 5-A1A a pesar de que empezó con un voltaje casi de 4,2V (un SoC de 99%) la corriente promedio de descarga dio alrededor de 0,13A y esto fue lo que bajó su capacidad. Al revisar las muestras de esta medición, se detecta que hay un bajo de corriente a las 2 horas de que empezó su proceso de descarga. Esto puede deberse a algún problema de comunicación de los dispositivos al momento del proceso de descarga o un movimiento brusco durante el proceso que afectó las medidas de corriente que estaba proporcionando el Arduino, puesto que pasó de estar midiendo en promedio 0,16A a 0,13A.

Se debe asociar que la capacidad medida por el sistema de caracterización parte de tanto la cantidad de corriente como la cantidad de voltaje inicial, no por el tiempo que le tome a la celda descargarse. Esto quiere decir que, a pesar de que una cantidad de muestras ideal resulta en 780 debe ir de la mano con haber empezado el proceso de descarga a voltaje nominal y a corriente

constante de 0,2A (el 10% de la corriente de descarga dada por el fabricante), de lo contrario, la cantidad de muestras tomadas no tienen relevancia en la determinación del SoH.

El SoH es correctamente determinado cuando se cuenta con una aproximación y estabilidad cercana a un SoC de 100%, un muestreo de 780 valores y una corriente de 0,2A, siendo que de las 3 variables la que más peso demuestra es la corriente constante. A raíz de esto, se analiza que la determinación del SoH es más exacta en la celda A1B.

El SoC experimental final de la Tabla 4 representa el porcentaje de carga al que cayó la celda en cada réplica, que representaba el límite de la zona segura de descarga según el fabricante, permitiendo analizar que el valor de zona de seguridad de una celda 18650 se estima en un 71,62% de SoC. Si una celda marca un SoC menor a este porcentaje significará que ya no puede almacenar carga como una celda en buenas condiciones y deberá desecharse.

Tabla 5. Error absoluto y relativo para el SoH y R_{int} de las 18 réplicas.

Medición (#)	Error absoluto SoH	Error relativo SoH (%)	Error absoluto R_{int}	Error relativo R_{int} (%)
1-A1A	10,24	12,94	1,24	1,55
2-A1A	8,45	10,68	5,30	6,61
3-A1A	0,51	0,65	4,85	6,04
4-A1A	5,22	6,60	2,88	3,59
5-A1A	8,96	11,32	9,74	12,14
6-A1A	0,20	0,25	2,07	2,58
7-A1A	16,97	21,44	20,51	25,57
8-A1A	1,65	2,09	2,78	3,47
9-A1A	6,11	7,72	2,79	3,48
1-A1B	4,37	4,58	2,80	4,38
2-A1B	2,14	2,24	2,23	3,48
3-A1B	4,73	4,96	3,44	5,37
4-A1B	4,22	4,43	3,37	5,26
5-A1B	3,57	3,75	2,15	3,35
6-A1B	4,02	4,22	0,97	1,51
7-A1B	3,74	3,93	2,85	4,45
8-A1B	3,73	3,92	1,23	1,93
9-A1B	1,20	1,26	1,12	1,76

El cálculo de la medición 7-A1A es la réplica que presenta mejores resultados para esta celda, ya que entrega el valor más bajo de resistencia interna, el SoH más alto, así como la capacidad más grande y la cantidad de muestras más cercana al valor de 780. El cálculo de la

medición 3-A1B es la réplica que presenta mejores resultados para esta celda, ya que entrega el valor más bajo de resistencia interna, el SoH más alto, así como la capacidad más grande y la corriente más cercana a 0,2A.

La lectura 6 de la celda A1A es la que representa mejor el estado de esta, ya que el porcentaje de error relativo del SoH es de 0,25% y el de la R_{int} es de 2,58%. La lectura 7 de la celda A1A es la que menos representa el estado actual de la celda, ya que sus porcentajes de error relativo sobrepasan el 20%. Esto comprueba que para que el sistema de caracterización demuestre resultados confiables, el SoC no debe ser inferior a 98% respecto al valor nominal.

La lectura 9 de la celda A1B es la que mejor representa el estado de esta, ya que el porcentaje de error relativo del SoH es de 1,26% y el de la R_{int} es de 1,76%. Todas las lecturas de esta celda representan valores fiables de resultados, ya que no superan un porcentaje de error mayor al 5%. Sin embargo, la medición 3 se encuentra en el máximo con lo que respecta a la R_{int} .

Los porcentajes de error obtenidos de ambas celdas son confiables y demuestran que el sistema funciona. En el caso de la celda A1A, se define que su condición de estado se encuentra alrededor del 80% respecto a cuando estaba nueva y que la celda A1B, ronda el 95% respecto a cuando estaba nueva, según los datos confiables del sistema.

Capítulo 5

Conclusiones y recomendaciones

5.1 Conclusiones

El prototipo del sistema de caracterización del SoH diseñado consigue medir las diferentes variables necesarias y logra determinar las principales características necesarias para describir el SoH de una celda Li-ión usada.

El diseño realizado responde a la necesidad de diferenciar entre una celda 18650 de Li-ion usada dañada y una en buenas condiciones.

El porcentaje de error para el SoH alcanza valores menores al 5%, esto indica que, el sistema de caracterización es capaz de entregar datos confiables respecto al SoH, por lo que refleja su viable uso en una adaptación para distintos tamaños de celdas y configuraciones de baterías.

El proyecto desarrollado presenta una solución accesible, ideal para alcanzar una población que esté en búsqueda de ayudar al planeta y requiera brindar un segundo uso a celdas de Li-ion usadas, por ejemplo, en un almacenamiento de energía solar proveniente de paneles.

Los componentes y sensores implementados en el sistema de caracterización muestran un funcionamiento adecuado al momento de realizar las muestras de cada réplica, así también como el código fuente; sin embargo, después de ejecutar todo el proceso y análisis se percibe que, a pesar de tener un sistema estable, se debe contar con monitoreo por parte de una persona para detener el muestreo y para realizar los cálculos del sistema.

El uso de Excel en el proyecto de investigación presenta una interfaz adecuada para poder visualizar los datos obtenidos por el Arduino UNO y realizar los cálculos del sistema de caracterización, mostrando las unidades de cada variable (voltaje y corriente) transmitidos por el Arduino UNO, así como también permite la creación de las gráficas que son necesarias para comparar el proceso de descarga de cada réplica y validar los datos registrados.

Se logra aplicar el método de conteo de Coulomb en la estimación del SoH de una celda y se demuestra su efectividad en la misma, a partir de los resultados y cálculos obtenidos de los experimentos.

De las 2 celdas utilizadas por el sistema de caracterización en este proyecto y sus experimentos, la celda A1B es la que está en mejores condiciones, ya que su SoH es el más alto, su R_{int} es la más baja, así como su capacidad se aproxima más a la nominal y la cantidad de energía capaz de suministrar es mayor.

5.2 Recomendaciones

- Una mejora en el diseño del hardware sería incorporar los componentes necesarios para realizar la conexión de la celda a un proceso de carga, de manera que mediante un interruptor se pueda realizar la conexión al circuito de descarga y luego al de carga o viceversa.
- Para un mayor aprovechamiento del sistema de caracterización se podría incluir un sensor de temperatura, de manera que se pueda proteger la celda en caso de sobrecalentamiento, esto más que todo cuando se pretenda extender el sistema a un banco de celdas más grande que incluya configuraciones en serie o paralelo.
- El hardware del sistema de caracterización usado con una celda 18650 se podría optimizar si se diseñara sobre una PCB, para mejorar aspectos de diseño, seguridad y tamaño.
- Un factor importante para disminuir el tiempo de análisis de datos sería diseñar una plantilla en Excel con las fórmulas necesarias ya incorporadas; también, el proceso de descarga podría modificarse a una corriente constante equivalente a $0.5C$, donde C es la capacidad de la celda a utilizar.
- Un mejor control de tiempo de encendido y de apagado se conseguiría con la vinculación de alguna aplicación de teléfono que permita dar seguimiento mediante notificaciones.

Referencias

- [1] S. Morales, Interviewee, *Laboratorio de vehículos eléctricos*. [Entrevista]. 30 Nov 2021.
- [2] LIVE, «Publicaciones oficiales y fotos,» 2021. [En línea]. Available: <https://www.facebook.com/LIVE.ITCR/>. [Último acceso: 30 Nov 2021].
- [3] C. Chan y Y. Wong, «Electric vehicles charge forward,» *IEEE Power and Energy Magazine*, vol. II, nº 6, pp. 24-33, Nov 2014.
- [4] Y. Hu, «Identification and state estimation for linear parameter varying systems with application to battery management system design,» Ohio, 2010.
- [5] M. Charkhgard y M. Farrokhi, «State-of-charge estimation for lithium-ion batteries using neural networks and ekf,» *IEEE Transactions on*, vol. 57, nº 12, pp. 4178-4187, Dec 2017.
- [6] H. Rahimi-Eichi y M. Chow, «Adaptive parameter identification and state-of-charge estimation of lithium-ion batteries,» de *IECON - 38th Annual Conference*, 2012.
- [7] L. Buccolini, «Methods and Tools for Characterization and Development of Lithium Batteries,» <https://hdl.handle.net/11566/266495>, 2019.
- [8] I. Buchmann, «Types of Lithium-ion,» <https://batteryuniversity.com/article/bu-205-types-of-lithium-ion>, 2019.
- [9] G. Castillo, «Diseño y construcción de un ciclador de baterías,» <https://dx.doi.org/10.21754/tecnia.v21i1.1101>, Perú, 2021.
- [10] EEMB, «Lithium-ion Battery DATA SHEET Battery Model : LIR18650 2600mAh,» Nov 2010. [En línea]. Available: <https://www.ineltro.ch/media/downloads/SAItem/45/45958/36e3e7f3-2049-4adb-a2a7-79c654d92915.pdf>.
- [11] C. Terán, «Aplicacion de metodos algebraicos para la estimacion del estado de carga de baterias de litio en aplicaciones de vehiculos electricos,» Bogotá, 2017.
- [12] H. Wu, C. Wang, S. Moo y A. Kawamura, «State of charge and state of health estimating method for lithium-ion batteries,» 10.1109/COMPEL.2016.7556688., Norway, 2016.
- [13] C. S. M. Y. P. C. a. Y. C. H. K. S. Ng, «Enhanced Coulomb Counting Method for Estimating State-of-Charge and State-of-Health of Lithium-Ion Batteries,» *J. Applied Energy*, 2009.
- [14] BSLBATT, «Medición del estado de carga (SoC) de iones de litio,» BSLBATT, s.f. [En línea]. Available: <https://www.lithium-battery-factory.com/es/lithium-battery-state-of-charge/>. [Último acceso: 07 Julio 23].
- [15] E. V. M. C. G. S. a. G. S. G. Nobile, «Estimation of SOH for Battery Packs: A Real-Time Mixed Algorithm based on Coulomb Counting Method and Parameter-Varying Circuit Modeling,» *IEEE*

- 11th International Symposium on Power Electronics for Distributed Generation Systems (PEDG)*, pp. 536-541, 2020.
- [16] Arduino, «¿Qué es Arduino?,» 05 Febrero 2018. [En línea]. Available: <https://www.arduino.cc/en/Guide/Introduction>. [Último acceso: 01 Noviembre 2023].
- [17] Arduino.cc, «Arduino UNO Rev3 with Long Pins,» 31 Octubre 2023. [En línea]. Available: <https://docs.arduino.cc/retired/boards/arduino-uno-rev3-with-long-pins>. [Último acceso: 01 Noviembre 2023].
- [18] Arduino.cc, «Overview of the Arduino IDE 1,» s.f. [En línea]. Available: <https://docs.arduino.cc/software/ide-v1/tutorials/Environment>. [Último acceso: 01 Noviembre 2023].
- [19] Microsoft, «What is Data Streamer?,» [En línea]. Available: <https://support.microsoft.com/en-us/office/what-is-data-streamer-1d52ffce-261c-4d7b-8017-89e8ee2b806f>.
- [20] M. Vega, «TFG Repositorio,» 10 Octubre 2023. [En línea]. Available: <https://github.com/msvm16/TFG.git>.
- [21] LucidChart, «lucid.app,» 01 11 2023. [En línea]. Available: https://lucid.app/lucidchart/178fd1b9-9a7c-4353-b9ac-ffd2a504c6c6/edit?view_items=umLKmJ-KgoFf&invitationId=inv_3e84ff9e-46fc-4de8-b49f-3e84bba23f27.
- [22] Arduino.cc, «Software downloads,» 31 Agosto 2023. [En línea]. Available: <https://www.arduino.cc/en/software>.
- [23] LucidChart, «Diagrama de Flujo,» 01 Noviembre 2023. [En línea]. Available: https://lucid.app/lucidchart/7c977a8d-1fe3-44a8-80d0-f0769faec0e0/edit?viewport_loc=-201%2C-1488%2C4361%2C2394%2C4RWK8WeH_OQo&invitationId=inv_1ee8cd49-9b33-4af0-b025-05867360de8d.
- [24] rollApp, «Pselnt,» 17 Mayo 2023. [En línea]. Available: <https://www.rollapp.com/app/pseint>. [Último acceso: 01 Noviembre 2023].
- [25] Microsoft, «Enable the Data Streamer add-in,» s.f. [En línea]. Available: <https://support.microsoft.com/en-us/office/enable-the-data-streamer-add-in-70052b28-3b00-41e7-8ab6-8a9f142dffeb#:~:text=Go%20to%20File%20%3E%20Options%20%3E%20Add,%2Din%2C%20then%20click%20OK..>
- [26] H. Y. C. Z. L. a. L. Z. X. Hu, «Co-estimation of state of charge and state of health for lithium-ion batteries based on fractional-order calculus,» *IEEE Trans. Veh. Technol*, vol. 67, nº 11, pp. 10319-10329, Nov 2018.
- [27] L. D. e. a. Henry Pedroza, «Sistema de analisis estadisticos con SPSSA.,» 2007.

- [28] lekkernet, «PILA BATERIA UNIROS LIR18650 RECARGABLE 2500mAh 3,7V PROTECTED BLISTER 1 (ecotasa incl),» 2023. [En línea]. Available: <https://lekkernet.es/producto/pila-bateria-uniros-lir18650-recargable-2500mah-37v-protected-blister-1-ecotasa-incl/> .