



# Práctica 3

## Exportación de datos en BSM y filtrado de señal

Selene Gonzales, Fernando Ortiz

Carrera: Ingeniería Mecánica y Automotriz

Semestre 8

Profesor: Salvador Martínez-Cruz

San Juan del Río, México

Jueves, 24 de octubre de 2024

**Abstract—**En esta práctica, se explorarán las técnicas de adquisición y procesamiento de datos de las señales de un automóvil eléctrico de la Facultad de Ingeniería Campus San Juan del Río de la Universidad Autónoma de Querétaro. Las señales que se recopilaron son de voltaje y corriente, después se aplicó una técnica de filtrado pasa bajas para eliminar el ruido y mejorar la calidad de estas señales.

### I. INTRODUCCIÓN

La recopilación y análisis de datos provenientes de los sistemas de un vehículo eléctrico es importante para comprender su funcionamiento, analizar sus sistemas y poder mejorar la eficiencia de este. Para realizar este análisis es importante comprender la información que el automóvil manda, recopilando sus señales y filtrándolas para obtener información limpia y lista para procesarla, ya que este proceso de filtrado nos ayuda a resaltar las tendencias y patrones que puedan ser utilizados para la toma de decisiones y la optimización del rendimiento del vehículo.

### II. MARCO TEÓRICO

#### A. Protocolo BMS

El protocolo BMS (Battery Management System) es un conjunto de reglas y regulaciones que rigen el funcionamiento de un sistema de gestión de baterías. Un BMS es un dispositivo electrónico que se encarga de monitorear y controlar el estado de carga de una batería, especialmente en aplicaciones de vehículos eléctricos y sistemas de almacenamiento de energía renovable.

El protocolo BMS se encarga de:

1. Monitorear el voltaje y la corriente de la batería.
2. Controlar la carga y descarga de la batería.
3. Proteger la batería contra sobrecargas y descargas profundas.

4. Optimizar el rendimiento de la batería.
5. Proporcionar información sobre el estado de la batería.

El protocolo BMS se utiliza en diferentes tipos de baterías, como las de iones de litio, plomo-ácido y níquel-metal hidruro. Es importante destacar que cada tipo de batería tiene sus propias características y requerimientos específicos que deben ser considerados en el diseño del protocolo BMS.

En resumen, el protocolo BMS es esencial para garantizar el funcionamiento seguro y eficiente de las baterías en diferentes aplicaciones.

#### B. Función *readtable* Matlab

La función *readtable* de Matlab es una función que se utiliza para leer datos de un archivo y convertirlos en una tabla. Esta función es muy útil para manejar grandes cantidades de datos y realizar operaciones complejas. Aquí te presento una fundamentación teórica de la función *readtable* de Matlab.

La función *readtable* tiene varios parámetros que se pueden personalizar según sea necesario:

- *filename*: nombre del archivo que se va a leer
- *delimiter*: carácter que se utiliza para separar las columnas de los datos
- *headerlines*: número de líneas que se van a saltar al principio del archivo
- *format*: formato de los datos que se van a leer

La función *readtable* devuelve una tabla que contiene los datos leídos del archivo. Esta tabla se puede utilizar para realizar operaciones complejas y análisis de datos.

Algunos de los beneficios de utilizar la función *readtable* de Matlab son:

- Facilita la lectura de archivos grandes y complejos
- Permite realizar operaciones complejas y análisis de datos de manera eficiente
- Se puede personalizar según sea necesario

### C. Filtro pasa bajas

Un filtro de paso bajo es un tipo de dispositivo utilizado en sistemas de procesamiento de señales cuya función principal es permitir el paso de las frecuencias bajas sin alterarlas, mientras que atenúa o rechaza de manera significativa las frecuencias más altas. Un filtro de paso bajo es un tipo de dispositivo utilizado en sistemas de procesamiento de señales cuya función principal es permitir el paso de las frecuencias bajas sin alterarlas, mientras que atenúa o rechaza de manera significativa las frecuencias más altas. El comportamiento de un filtro de paso bajo se caracteriza por su "frecuencia de corte", que define el punto a partir del cual las frecuencias comienzan a ser atenuadas. A medida que la frecuencia aumenta por encima de este umbral, la señal se reduce gradualmente hasta que es prácticamente eliminada [1]. Este filtro reduce el ruido de alta frecuencia por lo que suaviza la señal como se muestra en la Figura 1.

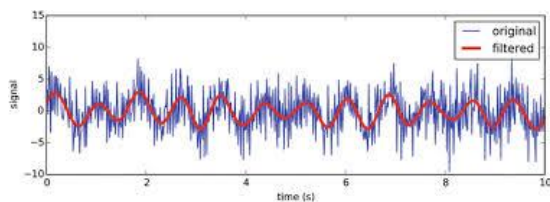


Figura 1. Comparación señal filtrada y señal original

Un filtro digital es solo una fórmula para transformar una señal digital en otra. Puede existir como una ecuación en papel, como un pequeño bucle en una subrutina de computadora.

## III. DESARROLLO

### A. Adquisición de datos

Se realiza el encendido del E-FACI, una vez que este inicia, se realiza la conexión a la computadora para observar y adquirir los datos en el programa Orion BMS2, Figura 2.



a) Encendido de E-FACI, b) Conexión USB a computadora

Figura 2. a) Encendido de E-FACI, b) Conexión USB a computadora

Una vez que el programa reconoce la conexión, se selecciona la entrada de USB y como rango de transferencia de datos, escogemos 250 kBit/sec. Figura 3.

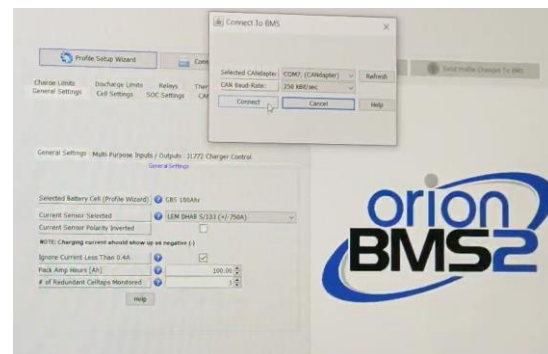


Figura 3. Selección de paquete de transferencia de datos

Cuando se logra que la conexión se exitosa, el software pregunta si se quiere importar los datos ya existentes en el BMS, esto porque se va a realizar una nueva exploración de las señales. Figura 4.

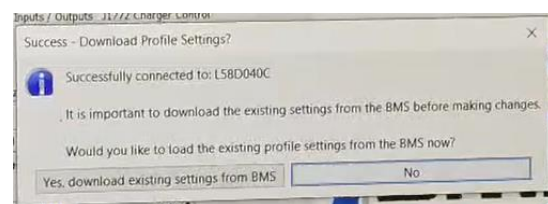


Figura 4. Conexión exitosa del BMS a Orion BMS2

En la pestaña *3rd Party Data*, que es donde encontramos parámetros enviados por dispositivos externos, que en este caso es el controlador del automóvil eléctrico. En esta sección los parámetros van cambiando en tiempo real. Figura 5.



Entrando a la pestaña *Diagnostic Trouble Codes*, encontramos los códigos de fallas que encontró el sistema. Figura 6.



Cambiamos a la pestaña *Live Text Data*, en esta sección del software, nuevamente se observan la variación en tiempo real de los parámetros, sin embargo, aquí se observan los parámetros que envía de manera directa el BMS. Figura 7.



una ventana en la que se pueden observar las variaciones de manera grafica. Figura 8.



En esta ventana encontramos el botón *Select Parameters to Graph*, que nos abrirá otra ventana que muestra todos los parámetros que podemos seleccionar para analizar. Figura 9



Quando se tienen los parámetros que se desean monitorear, se seleccionan aplicar cambios y se va al botón *Start Logging*, este comenzará a guardar los valores de nuestros parámetros con intervalos de tiempo cada 10 ms, para lograr las variaciones, se realiza la aceleración y desaceleración del vehículo, esto se realiza por 1 min aproximadamente, por lo que tendremos más o menos 600 muestras en cada señal. Figura 10



Pasado el minuto se selecciona *Stop Logging* y nos pide donde guardarlo, se selecciona la carpeta donde se va a almacenar el documento con la información del monitoreo y se guarda en formato *.csv*.

#### B. Filtrado de señal

Para el filtrado de la señal obtenida anteriormente se realizó un código en el software Matlab el cual lee los valores del archivo *.exe* y los grafica sin aplicar el filtro, después se aplica el filtro pasabajas y se vuelven a graficar los datos ya filtrados sobre la misma figura para observar la deferencia entre estos. La función *lowpass* requiere los siguientes argumentos para su funcionamiento.

*Filtrado* = *lowpass(signal, Fc, Fs)*, donde  
**signal**: La señal original a la que deseas aplicar el filtro.

**Fc**: La frecuencia de corte del filtro. Frecuencias por debajo de esta pasarán sin atenuamiento, mientras que las superiores serán filtradas.

**Fs**: La frecuencia de muestreo de la señal. Es decir, cuántas muestras de la señal tienes por segundo.

#### IV. RESULTADOS

Se obtuvieron los resultados del BMS en el siguiente archivo *.csv*

	A	B
1	Pack Amperage (Current)	Summed Pack Voltage
2	0.5	105.37
3	0.5	105.38
4	0.5	105.37
5	0.5	105.38
6	0.5	105.38
7	0.5	105.38
8	0.5	105.38
9	0.5	105.38
10	0.5	105.37
11	0.5	105.38
12	0.5	105.36
13	0.8	105.35
14	1.1	105.29
15	1.6	105.24
16	1.5	105.24
17	1.8	105.23
18	1.7	105.21
19	2	105.18
20	3.4	105.01
21	3.9	104.9
22	3.1	104.99
23	2.6	105.1
24	3.6	105.02
25	6.7	104.59
26	6.1	104.64
27	7.2	104.54

Figura 11. Datos de señales

Con el código en Matlab se obtuvieron las siguientes gráficas mostradas figuras a continuación.

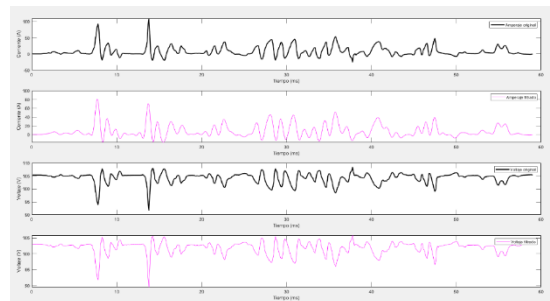


Figura 12. Comparativa datos obtenidos y datos filtrados

Para observar de una mejor manera se realizó la comparativa de los valores sobre el valor filtrado y se obtuvieron las siguientes gráficas.

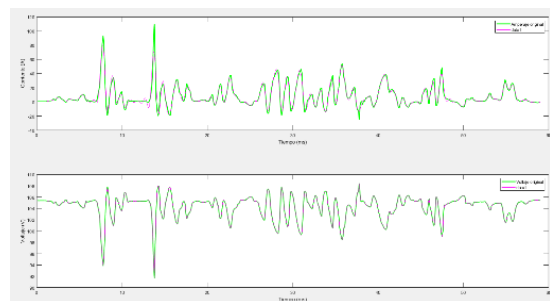


Figura 13. Comparativa valores

En la figura 13 se alcanza a notar que los valores obtenidos de corriente y de voltaje son los inversos del otro.

Tomando la parte de la señal original de corriente y la filtrada, alcanzamos a notar como se hace una disminución del pico de amperaje. Figura 14.

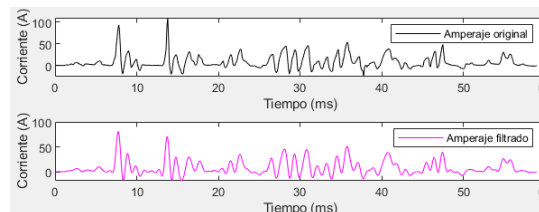


Figura 14. Comparación de señal original y filtrada de corriente

Esta disminución de amplitud es debido a la eliminación de ruido de la señal.



## V. CONCLUSIONES

**Fernando:** Se realizó con éxito la adquisición de datos mediante el software BMS, con estos datos realizamos un filtro pasa bajas para eliminar el ruido de alta frecuencia que puede provenir de la adquisición de datos, además de realizar la comparativa de estos datos una vez que se realizó su procesamiento.

**Selene:** El objetivo de esta práctica es el aplicar los conocimientos adquiridos en la adquisición de datos de y el procesamiento de estos, se logró gracias al uso de adquisición de datos en BMS y las señales de los parámetros que se midieron se filtraron para que pudiéramos observar un su comportamiento de una manera más limpia, además de realizar una comparación con las originales.

## VI. REFERENCIAS

1. *The simplest Lowpass Filter / Introduction to digital Filters.* (n.d.).  
[https://www.dsprelated.com/freebooks/filters/Simplest\\_Lowpass\\_Filter\\_1.html](https://www.dsprelated.com/freebooks/filters/Simplest_Lowpass_Filter_1.html)
2.  
[https://www.uco.es/grupos/giie/cirweb/teoria/tema\\_12/tema\\_12\\_02.pdf](https://www.uco.es/grupos/giie/cirweb/teoria/tema_12/tema_12_02.pdf)

## VII. ANEXOS

## A. Código Matlab

```
clc;
clear all;
close all;
% frecuencia de muestreo, muestras que
% toma por segundo ( en este caso cada
% 10)
% periodo, tiempo que tarda en tomar
% una muestra
t=0.1:0.1:59;
BMS_data=readtable('corriente_voltaje.csv','VariableNamingRule','preserve');
% Nombre del archivo, Para que lea los
% encabezados pero no los guarde como
% datos, para que los mantenga y no
% los elimine
Amp = table2array(BMS_data(:,1));
```

```
Volt = table2array(BMS_data(:,2));
Amp_f = lowpass(Amp,50*pi/180,10);
Volt_f = lowpass(Volt,59*pi/180,10);
% Señal ,frecuencia de corte,
% frecuencia de muestreo
subplot(4,1,1)
plot(t,Amp)
legend('Amperaje original')
xlabel("Tiempo (ms)")
ylabel("Corriente (A)")
subplot(4,1,2)
plot(t,Amp_f)
legend('Amperaje filtrado')
xlabel("Tiempo (ms)")
ylabel("Corriente (A)")
subplot(4,1,3)
plot(t,Volt)
legend('Voltaje original')
xlabel("Tiempo (ms)")
ylabel("Voltaje (V)")
subplot(4,1,4)
plot(t,Volt_f)
legend('Voltaje filtrado')
xlabel("Tiempo (ms)")
ylabel("Voltaje (V)")
```