



# **MATRIX** | ELECTRICAL MACHINES

## Electrical Machines & MATLAB v2.0

EM 2025  
RELEASE



Salud y seguridad	3
Hojas de trabajo	4
Ficha 1 - Primeros pasos	5
Hoja de ejercicios 2 - Uso de MATLAB para gráficos	6
Ficha 3 - Control en bucle cerrado	7
Ficha 4 - Estrategias de control de velocidad	8
Información sobre la máquina	10
Información sobre software y API de MATLAB	17

## Seguridad

Durante el diseño de este producto hemos prestado considerable atención a los riesgos potenciales de estudiando los motores eléctricos. Creemos que hemos dado con el diseño más seguro posible. Sin embargo, aún existen algunos riesgos que debe conocer. Esta página muestra cómo hemos considerado cada peligro. Debes leerla y asegurarte de que tus alumnos están protegidos mientras utilizan el equipo.

### Choque eléctrico

Es mínima: la salida de la caja de control está limitada a 36 V CA o CC.

El dinamómetro es capaz de generar tensiones CC. A velocidad máxima, en torno a 3.000 RPM, la tensión generada es inferior a 30 VCC.

La unidad de control no generará energía hasta que se conecte un motor al dinamómetro. Esto evita el uso de motores de terceros con el sistema, así como hace que el sistema sea seguro de usar.

### Choque físico

El equipo es pesado. Como ocurre con otros equipos pesados de laboratorio, si a un alumno se le cae un aparato al pie puede causar daños considerables. Usted debe decidir el nivel de responsabilidad que los alumnos asumen aquí. Puede reducir el riesgo haciendo que un técnico coloque el equipo en bancos y asegurándose de que los alumnos estén sentados en pupitres mientras utilizan el equipo.

Las piezas giratorias expuestas crean riesgos, ya que el pelo y la ropa pueden quedar atrapados en ellas. El uso de motores de potencia relativamente baja reduce el riesgo. La protección de plástico entre el dinamómetro y el motor sometido a prueba evita la exposición de piezas giratorias.

### Por favor:

Si se produjera un accidente durante uso del equipo, le rogamos que nos lo comunique a la dirección indicada más abajo, para que podamos estudiar cómo hacer que el equipo sea aún más seguro.

Equipo de diseño

Matrix TSL

33 Gibbet Street HX1

5BA

Inglaterra

### Precaución:

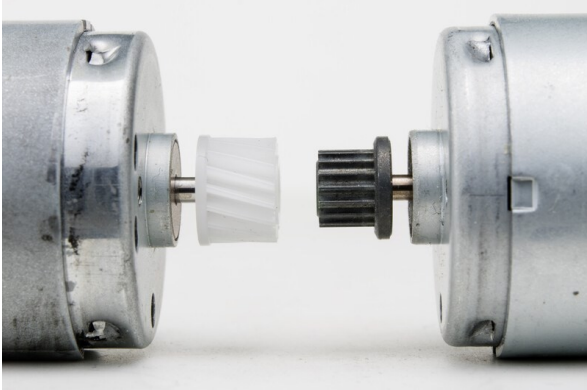
**No utilice un osciloscopio basado en PC con el equipo.**

**Las corrientes de bucle a tierra pueden fluir entre su conexión a tierra y el control conexión a tierra de la caja.**

# Hojas de trabajo

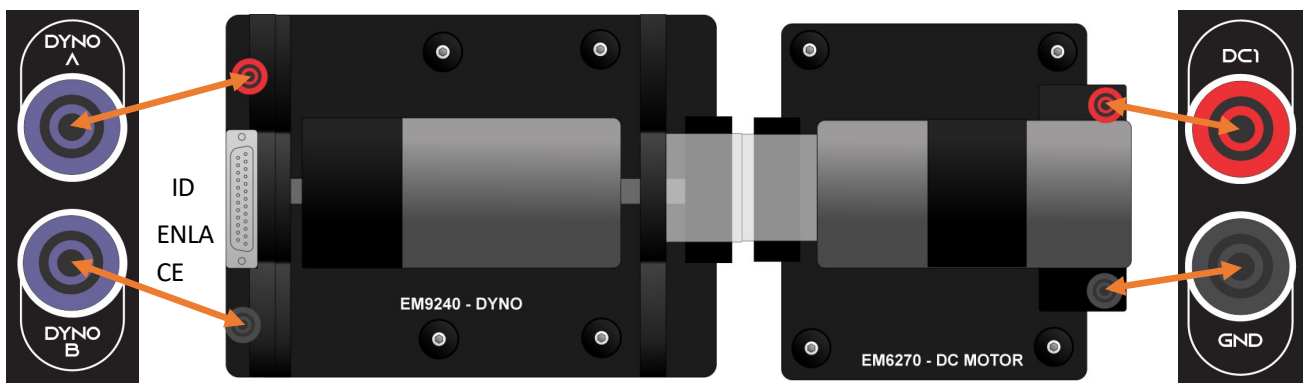
# Ficha 1

## Cómo empezar



Los distintos tipos de motor tienen características diferentes, como la velocidad máxima, el par a distintas velocidades, la tensión y la corriente nominales, la potencia de salida y el rendimiento.

Foto: estos pequeños motores de corriente continua se utilizan en pequeños dispositivos portátiles como juguetes.



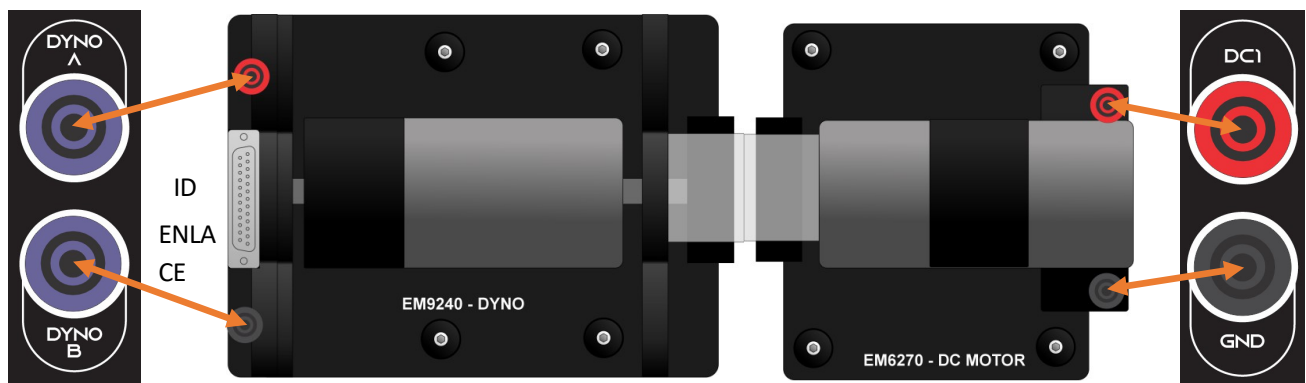
- 1) Coloca el motor de corriente continua y el dinamómetro como se muestra en el diagrama anterior.
- 2) Asegúrese de que la unidad funciona correctamente: utilice los controles de la pantalla táctil para accionar la tensión de salida DC1 para variar la velocidad del motor. Asegúrese de que el dinamómetro está conectado correctamente y de que el aumento de la carga del dinamómetro ralentiza el motor.
- 3) Asegúrese de que tiene instalados en su los controladores para la unidad de control. Si no es así siga las instrucciones de la sección de referencia del software MATLAB más abajo.
- 4) Conecte el cable USB entre la caja de accionamiento digital y el ordenador.
- 5) Consulte la sección de referencia del software MATLAB al final de este documento para comprender las funciones de MATLAB que tiene a su disposición.
- 6) Cree un programa MATLAB sencillo que controle la tensión de accionamiento del motor de CC.
- 7) Amplíe su programa para poder controlar la carga del dinamómetro.

## Ficha 2

### Uso de MATLAB para gráficos



Los motores de CC de estator bobinado suelen configurarse como bobinados **en derivación** o **serie**. Las conexiones separadas para los devanados de campo del estator y el rotor / inducido nos permiten controlar la corriente en ambas partes del motor por separado, con el fin de examinar el comportamiento del motor. La fotografía muestra un antiguo motor bobinado en derivación.



Con la misma configuración que la hoja de cálculo anterior:

- 1) Crea un programa que haga funcionar el motor de corriente continua al 50% de su potencia máxima de salida.
- 2) Modifique el programa para que aumente la carga del dinamómetro de 0% a 100% en pasos de 10%.
- 3) Crear rutinas que midan el par en la célula de carga.
- 4) Crear rutinas que midan la velocidad del motor en Revoluciones Por Minuto (RPM)
- 5) Represente gráficamente las RPM frente al par. Se trata de la curva velocidad-par del motor de CC. Las curvas velocidad-par son la forma más básica de caracterizar las máquinas eléctricas.

## Ficha 3

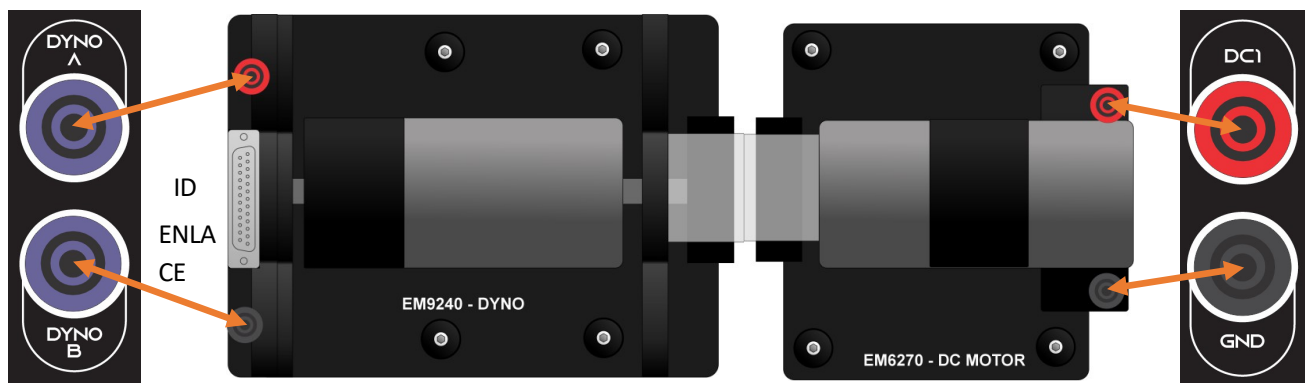
### Control en bucle cerrado



Los coches "dodgem" de las ferias utilizan simples motores eléctricos de corriente continua que funcionan a entre 12 y 48 V.

Los vehículos tienen dos escobillas: una toca el suelo metálico, para 0 V, y la otra toca el techo metálico, para una tensión positiva.

En el caso de los dodgems, el control de velocidad suele ser un simple encendido y apagado.



En un contexto industrial, a menudo es necesario que un motor funcione a una velocidad constante. Independientemente de la carga a la que esté sometido. Utilizando la misma configuración que antes:

- 1) Crea un script de sistema de bucle cerrado para hacer funcionar el motor a una velocidad definida.
- 2) Varíe la carga del motor y confirme que la velocidad se mantiene constante.
- 3) ¿Puede hacerse que la velocidad sea constante para todos los valores de carga?

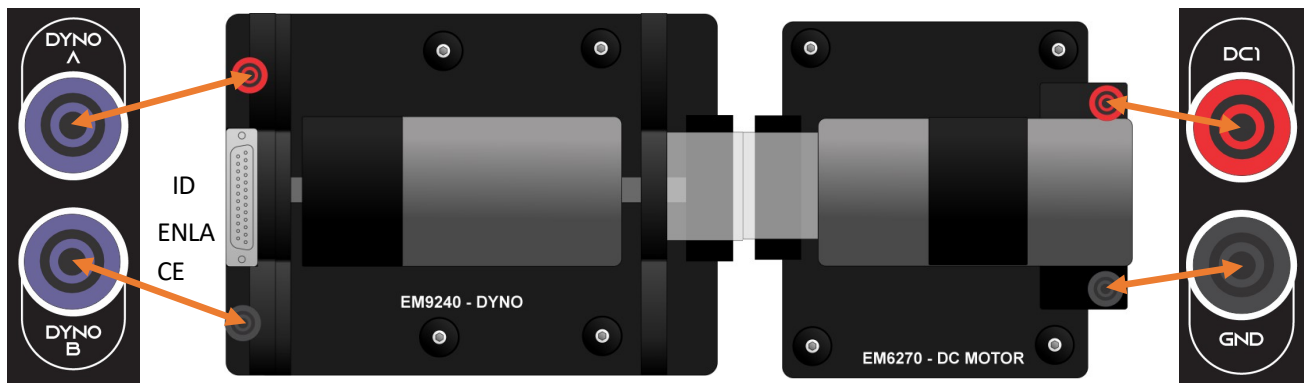
## Ficha 4

### Estrategias de control de velocidad



El control de velocidad es un elemento importante del diseño de sistemas electromecánicos.

Algunos motores son mejores que otros para autorregular la velocidad. El motor shunt es realmente bueno en la regulación de la velocidad y por esta razón se utilizan a menudo en dispositivos como tornos, que necesitan funcionar a una velocidad constante.



Un motor no puede pasar de no moverse a moverse a una velocidad determinada en un instante. Las técnicas de pasar de velocidad cero a velocidad nominal tienen muchas implicaciones en el diseño de sistemas de control de motores.

- 1) Crea un script que permita aumentar y reducir la velocidad del motor para alcanzar la velocidad deseada y reducir el consumo excesivo de corriente.
- 2) Investigar el uso de medidas Proporcionales, Integrales y Derivativas (PID) para controlar el velocidad de los motores.
- 3) Crea una variación de tu rutina gráfica que te permita trazar la velocidad del motor durante un tiempo determinado al arrancar un motor.
- 4) Utiliza técnicas PID para crear un programa en MATLAB que aumente de forma óptima la velocidad del desde la parada hasta 1500RPM sin sobrepasar esta velocidad.
- 5) Modifique su programa para que permita al dinamómetro variar la carga entre el 0% de la carga y el 70% de la carga y aún así mantener la velocidad nominal de 1500 RPM.
- 6) Utiliza los trazados de y nuestro gráfico para demostrar que el rendimiento del sistema se ajusta a las especificaciones.



# Software

## Información y API de MATLAB

# Referencia

## Software: Integración de MATLAB



El software de control de máquinas eléctricas puede descargarse de la web de Matrix.

Dentro del software hay una carpeta llamada Matlab que contiene una serie de scripts de ejemplo diseñados para ayudarle a empezar.

La funcionalidad de Máquinas Eléctricas Modernas se controla mediante una conexión de puerto USB COM al ordenador.

Hemos proporcionado scripts de ejemplo para permitir que los diversos aspectos del sistema de Máquinas Eléctricas sean investigados y controlados. Los scripts requieren que se declare un puerto COM serie en su proyecto para establecer el número de puerto COM como se muestra a continuación:

*Busque el puerto COM de su ordenador en el que está conectada la unidad de control Matrix EM. y utilízalo aquí en lugar de "COM5".*

```
EM_USB= serialport('COM5',115200); fopen(EM_USB);
```

*El programa principal reside aquí*

```
clear EM_USB; % Cerrar el puerto COM
```

La caja de accionamiento digital se comunica utilizando datos codificados JSON con los que Matlab puede interactuar fácilmente. La función `jsonencode` toma una variable Matlab y la convierte en una cadena que puede ser enviada al puerto serie. La función `jsondecode` toma una cadena del puerto serie y la convierte de nuevo en un Variable Matlab.

A continuación se muestra un ejemplo para leer el ID del banco de potencia y del motor, así como para obtener la tensión disponible para el motor.

```
status.status= 1; % Montar comando de estado
json= jsonencode(status); % Convertir claves de estado a cadena JSON
fwrite (EM_USB, json); % Transmitir cadena JSON a Digital Drive Box rx
= readline (EM_USB); % Leer respuesta de comando
rxdata = jsondecode(rx); % vuelve a convertir la respuesta JSON en valor
de datos modulevoltage_max = rxdata.v1; % guarda el voltaje máximo del módulo del
motor
dynoID= rxdata.id0; % Guardar el ID del módulo dyno
motorID= rxdata.id2; % Guardar el ID del módulo del motor
```

A continuación se muestra un ejemplo para configurar el voltaje, la frecuencia, el modo de accionamiento, la carga del banco de potencia y poner en marcha el motor.

```
control.v1= 0.5; % 50% Tensión
control.freq= 50; % 50Hz Frecuencia
control.mo= 1; % Modo AC Drive
control.load= 0; % Establece la carga del dinamómetro %
control.es= 1; % Habilitar la salida del motor
json = jsonencode(control); % Convertir los pares clave, valor del control en
cadena JSON fwrite (EM_USB, json); % Transmitir cadena JSON a Digital Drive Box
```

A continuación se muestra un ejemplo para leer los valores del sensor de RPM y Torque desde la caja de transmisión digital.

```
poll.poll= 1; % Assemble poll command
json= jsonencode(poll); % Convertir las claves del sondeo en una cadena JSON
fwrite (EM_USB, json); % Transmitir cadena JSON a Digital Drive Box rx
= readline (EM_USB); % Leer respuesta de comando
rxdata = jsondecode(rx); % convierte la respuesta JSON de nuevo en un
valor de datos TorqueNM = rxdata.tor; % Copia la lectura de Par en una variable
SpeedRPM = rxdata.spe; % Copia la lectura de Velocidad en una variable
```

El conjunto completo de claves de comando y claves de respuesta de la API JSON disponibles para la caja de transmisión digital puede consultarse en el documento Guía de inicio rápido de EMv2 disponible en la sección Máquinas eléctricas del sitio web de Matrix TSL.

26 02 25 primera edición