



MATRIX | ELECTRICAL MACHINES

Electrical Machines & MATLAB v2.0

EM 2025
RELEASE



MATRIX

CP8198

www.matrixsl.com

Copyright © 2023 Matrix Technology Solutions Limited

Santé et sécurité	3
Feuilles de travail	4
Feuille de travail 1 - Pour commencer	5
Feuille d'activité 2 - Utilisation de MATLAB pour la création de graphiques	6
Feuille de travail 3 - Contrôle en boucle fermée	7
Feuille de travail 4 - Stratégies de contrôle de la vitesse	8
Informations sur les machines	10
Informations sur le logiciel et l'API MATLAB	17

Sécurité

Lors de la conception de ce produit, nous avons accordé une attention considérable aux risques potentiels de l'étude des moteurs électriques. Nous pensons avoir mis au point la conception la plus sûre possible. Cependant, il existe encore certains risques vous devez être conscient. Cette page montre comment nous avons pris en compte chaque danger. Vous devez la lire et vous assurer que vos élèves sont protégés lorsqu'ils utilisent l'équipement.

Choc électrique

C'est un minimum : la sortie du boîtier de commande est limitée à 36 V CA ou CC.

Le dynamomètre est capable de générer des tensions continues. À la vitesse maximale, environ 3 000 tr/min, tension générée est inférieure à 30 VDC.

L'unité de contrôle ne génère pas de courant tant qu'un moteur n'est pas branché sur le dynamomètre. Cela empêche l'utilisation de moteurs de tiers avec le système et rend l'utilisation du système sûre.

Choc physique

L'équipement est lourd. Comme pour tout autre équipement lourd de laboratoire, si un élève fait tomber un appareil sur son pied, cela peut causer des dommages considérables. Vous devez décider du niveau de responsabilité des élèves dans ce domaine.

Vous pouvez réduire les risques en demandant à un technicien de disposer l'équipement sur des bancs et en veillant à ce les élèves soient assis à des pupitres lorsqu'ils utilisent l'équipement.

Les pièces rotatives exposées présentent des risques car cheveux et les vêtements peuvent s'y coincer. L'utilisation de moteurs de relativement faible puissance réduisent le risque. La protection en plastique entre le dynamomètre et le moteur testé signifie qu'aucune pièce rotative n'est exposée.

S'il vous plaît :

Si un accident survient pendant l'usage de l'équipement, veuillez nous le signaler à l'adresse ci-dessous, afin que nous puissions réfléchir à la manière de rendre l'équipement encore plus sûr.

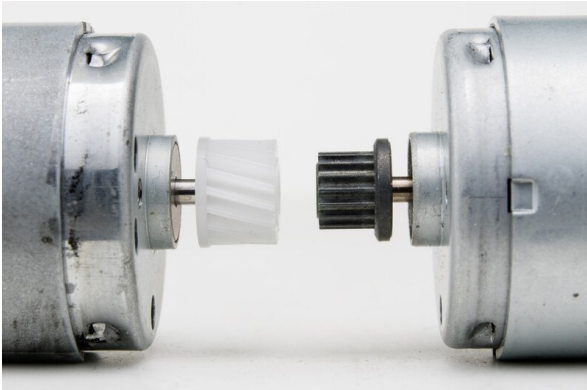
Équipe de
conception Matrix
TSL
33, rue Gibbet HX1
5BA
Angleterre

Attention :
N'utilisez pas d'oscilloscope basé sur un PC avec l'équipement.
Des courants de boucle de terre peuvent circuler entre sa connexion à la terre et le contrôle.
la connexion à la terre de la boîte.

Feuilles de travail

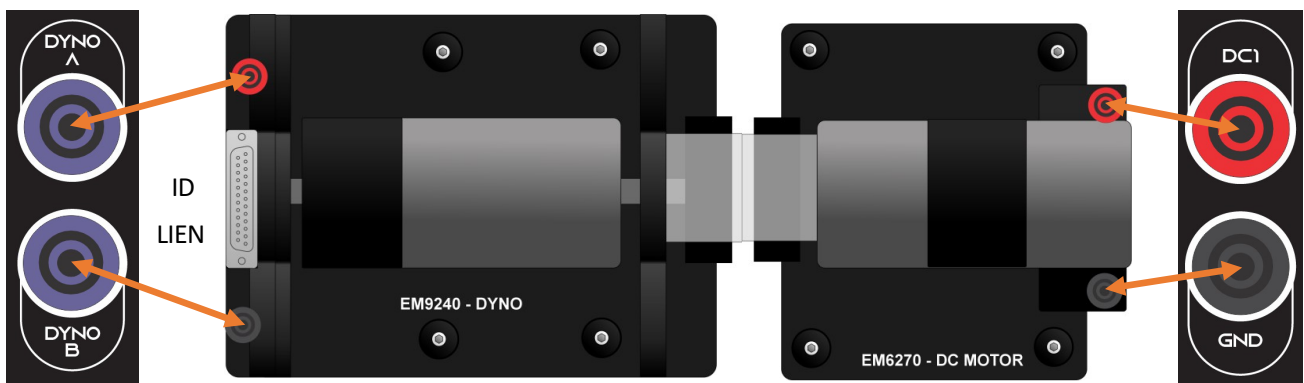
Fiche de travail 1

Pour commencer



Les différents types de moteurs ont des caractéristiques différentes, telles que la vitesse maximale, le couple à différentes vitesses, la tension et l'intensité nominales, la puissance et le rendement.

Photo : ces minuscules moteurs à courant continu sont utilisés dans les petits appareils portatifs comme les jouets.



- 1) Installez le moteur à courant continu et le dynamomètre comme indiqué dans le diagramme ci-dessus.
- 2) Assurez-vous que l'appareil fonctionne correctement : utilisez les commandes de l'écran tactile pour actionner la tension de sortie DC1 afin de faire varier la vitesse du moteur. Assurez-vous que le dynamomètre est correctement connecté et que l'augmentation de la charge du dynamomètre ralentit le moteur.
- 3) Assurez-vous que les pilotes de l'unité de contrôle sont installés sur votre ordinateur. Si ce n'est pas le cas, suivez les instructions de la section de référence du logiciel MATLAB ci-dessous.
- 4) Branchez le câble USB entre le boîtier du lecteur numérique et l'ordinateur.
- 5) Reportez-vous à la section de référence du logiciel MATLAB à la fin de ce document pour comprendre fonctions MATLAB à votre disposition.
- 6) Créez un programme MATLAB simple qui contrôle la tension d'entraînement du moteur à courant continu.
- 7) Étendez votre programme pour vous permettre de contrôler la charge du dynamomètre.

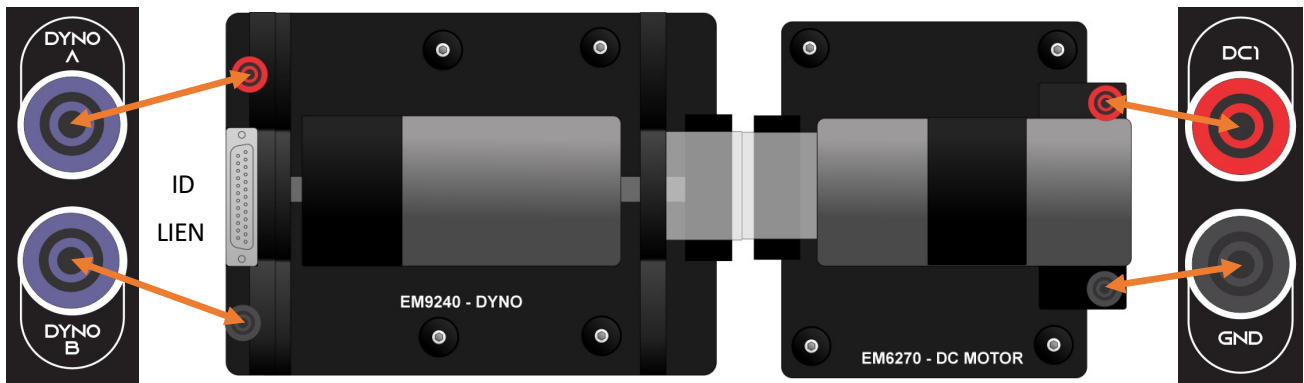
Utilisation de MATLAB pour la création de graphiques



Les moteurs à courant continu à stator bobiné sont souvent configurés comme des moteurs à bobinage **shunt** ou à bobinage **en série**.

Les connexions séparées pour les enroulements de champ du stator et du rotor/armature nous permettent de contrôler le courant dans les deux parties du moteur séparément, afin d'examiner le comportement du moteur.

La photographie montre un ancien moteur à enroulement shunt.



Avec la même configuration que la feuille de travail précédente :

- 1) Créez un programme qui fait fonctionner le moteur à courant continu à 50 % de sa puissance de sortie maximale.
- 2) Modifier le programme de manière à ce qu'il augmente la charge du dynamomètre de 0 à 100 % par paliers de 10 %.
- 3) Créer des routines qui mesurent le couple sur la cellule de charge.
- 4) Créer des routines qui mesurent la vitesse du moteur en tours par minute (RPM).
- 5) Tracez un graphique du nombre de tours par minute en fonction du couple. Il s'agit d'une courbe vitesse-couple pour le moteur à courant continu. Les courbes vitesse-couple constituent le moyen le plus élémentaire de caractériser les machines électriques.

Feuille de travail 3

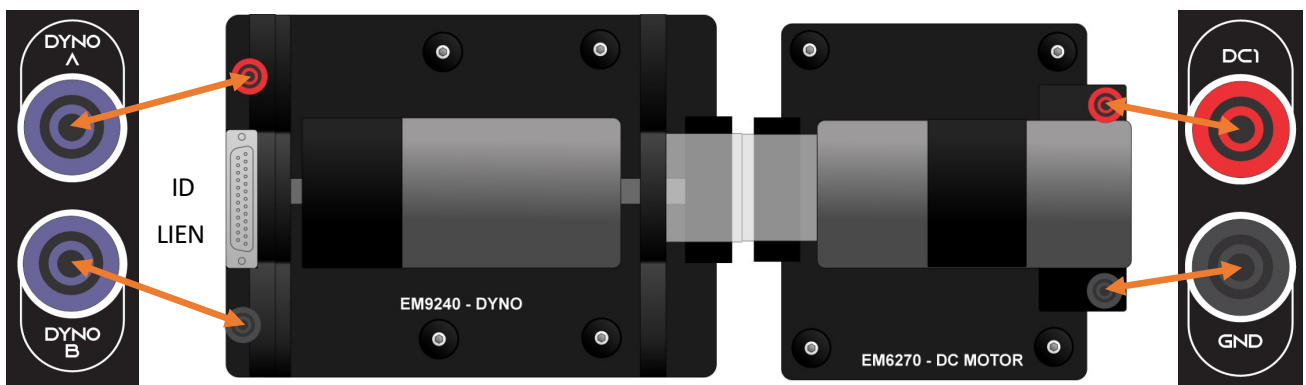
Contrôle en boucle fermée



Les voitures "dodgem" des fêtes foraines utilisent de simples moteurs électriques à courant continu fonctionnant à une tension comprise entre 12 et 48V.

Les véhicules ont deux balais - l'un touchant le plancher métallique, pour une tension de 0V, et l'autre touchant le plafond métallique, pour une tension positive.

Pour les dodgems, le contrôle de la vitesse se fait généralement par un simple "on- off".



Dans un contexte industriel, il est souvent nécessaire qu'un moteur tourne à une vitesse constante. Indépendamment de la charge qui lui est imposée. En utilisant la même configuration que précédemment :

- 1) Créer un script de système en boucle fermée pour faire tourner le moteur à une vitesse définie.
- 2) Faites varier la charge du moteur et vérifiez que la vitesse reste constante.
- 3) La vitesse peut-elle être rendue cohérente pour toutes les valeurs de la charge ?

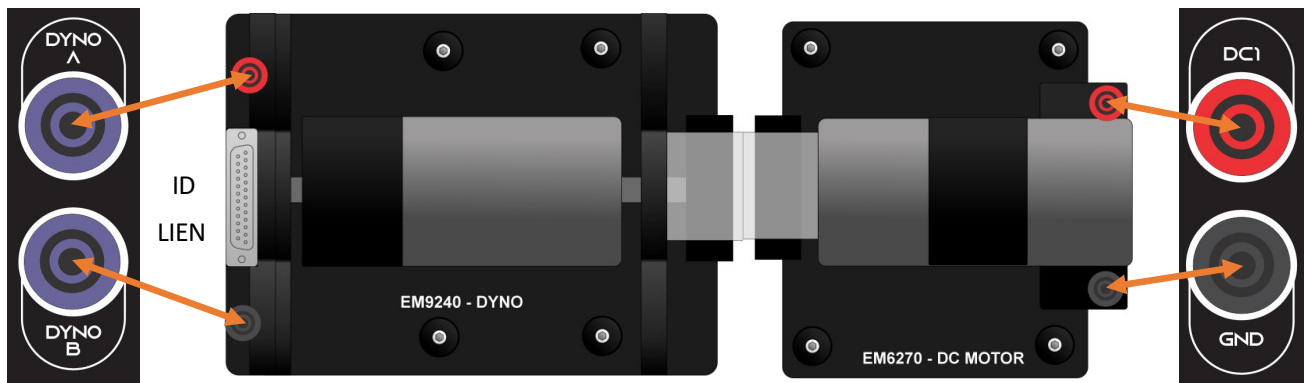
Fiche de travail 4

Stratégies de contrôle de la vitesse



Le contrôle de la vitesse est un élément important de la conception des systèmes électromécaniques.

Certains moteurs sont plus performants que d'autres en matière d'autorégulation de la vitesse. Le moteur shunt est très performant en matière de régulation de la vitesse et c'est il est souvent utilisé dans des appareils tels que les tours, qui doivent fonctionner à une vitesse constante.



Un moteur ne peut pas passer de l'immobilité à une certaine vitesse en un instant. Les techniques de
Le passage de la vitesse zéro à la vitesse nominale a de nombreuses implications dans la conception des systèmes de contrôle des moteurs.

- 1) Créer un script permettant d'augmenter et de réduire la vitesse du moteur pour atteindre la vitesse souhaitée et réduire l'utilisation de courant excessif.
- 2) Étudier l'utilisation des mesures proportionnelles, intégrales et dérivées (PID) pour le contrôle du système d'alimentation en eau potable.
la vitesse des moteurs.
- 3) Créez une variante de votre routine graphique qui vous permet de tracer la vitesse du moteur pendant un certain temps lors du démarrage d'un moteur.
- 4) Utilisez les techniques PID pour créer un programme dans MATLAB qui augmente de manière optimale la vitesse du système depuis l'arrêt jusqu'à 1500RPM sans dépasser cette vitesse.
- 5) Modifiez votre programme de manière à ce qu'il permette au dynamomètre de faire varier la charge entre 0 % et 70 % de la charge, tout en maintenant la vitesse nominale de 1 500 tr/min.
- 6) Utilisez les tracés de notre graphique pour prouver que les performances du système sont conformes aux spécifications.

Logiciel

Informations et API MATLAB

Logiciel : Intégration de MATLAB



Le logiciel de contrôle des machines électriques est disponible en téléchargement sur le site web de Matrix.

Le logiciel contient un dossier nommé Matlab qui contient un certain nombre d'exemples conçus pour vous aider à démarrer.

La fonctionnalité Machines électriques modernes est contrôlée par une connexion USB COM à l'ordinateur.

Nous avons fourni des exemples de scripts permettant d'étudier et de contrôler les différents aspects du système des machines électriques. Les scripts nécessitent qu'un port COM série soit déclaré dans votre projet pour définir le numéro du port COM comme indiqué ci-dessous :

Trouvez le port COM de votre PC sur lequel l'unité de contrôle Matrix EM est branchée. % et l'utiliser ici à la place de "COM5".

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

Le programme principal réside ici

```
clear EM_USB ;
```

```
Fermer le port COM
```

Le boîtier de commande numérique communique à l'aide de données encodées JSON avec lesquelles Matlab peut facilement s'interfacer. La fonction `jsonencode` prend une variable Matlab et la convertit en une chaîne qui peut être envoyée au port série. La fonction `jsondecode` prend une chaîne du port série et la reconvertit en une variable Matlab.

Variable Matlab.

L'exemple ci-dessous permet de lire l'ID du banc et du moteur et d'obtenir la tension disponible pour le moteur.

```
status.status= 1 ; % Assembler la commande d'état
json= jsonencode(status) ; Conversion des clés d'état en chaîne JSON
fwrite (EM_USB, json) ; % Transmettre la chaîne JSON au boîtier
d'entraînement numérique rx = readline (EM_USB) ; Lecture de la réponse à la
commande
rxdata = jsondecode(rx) ; % convertir la réponse JSON en valeur de données
modulevoltagegmax = rxdata.v1 ; % Sauvegarde la tension maximale du module
moteur
dynoID= rxdata.id0 ; Sauvegarde de l'ID du module dyno
motorID= rxdata.id2 ; Sauvegarde de l'ID du module moteur
```

Vous trouverez ci-dessous un exemple de réglage de la tension, de la fréquence, du mode d'entraînement, de la charge du banc d'essai et du démarrage du moteur.

```
control.v1= 0.5 ; % 50% Tension
control.freq= 50 ; % 50Hz Fréquence
control.mo= 1 ; Mode d'entraînement % AC
control.load= 0 ; % Régler la charge du dynamomètre %
control.fr= 1 ; Activer la sortie du moteur
json = jsonencode(control) ; % Convertit les paires clé/valeur du contrôle en chaîne
JSON fwrite (EM_USB, json) ; Transmettre la chaîne JSON au boîtier d'entraînement
numérique
```

Voici un exemple de lecture des valeurs des capteurs RPM et Torque à partir du boîtier de commande numérique.

```
poll.poll= 1 ; % Assembler la commande poll
json= jsonencode(poll) ; % Convertit les clés du sondage en chaîne JSON
fwrite (EM_USB, json) ; % Transmettre la chaîne JSON au boîtier
d'entraînement numérique rx = readline (EM_USB) ; Lecture de la réponse à la
commande
rxdata = jsondecode(rx) ; % convertir la réponse JSON en valeur de données
TorqueNM = rxdata.tor ; % Copier la valeur du couple dans une variable
SpeedRPM = rxdata.spe ; % Copie la valeur de la vitesse dans une
variable
```

L'ensemble des clés de commande et de réponse JSON API disponibles pour le boîtier de commande numérique se trouve dans le document EMv2 Quick Start Guide disponible dans la section Electrical Machines du site Web de Matrix TSL.

26 02 25 première libération