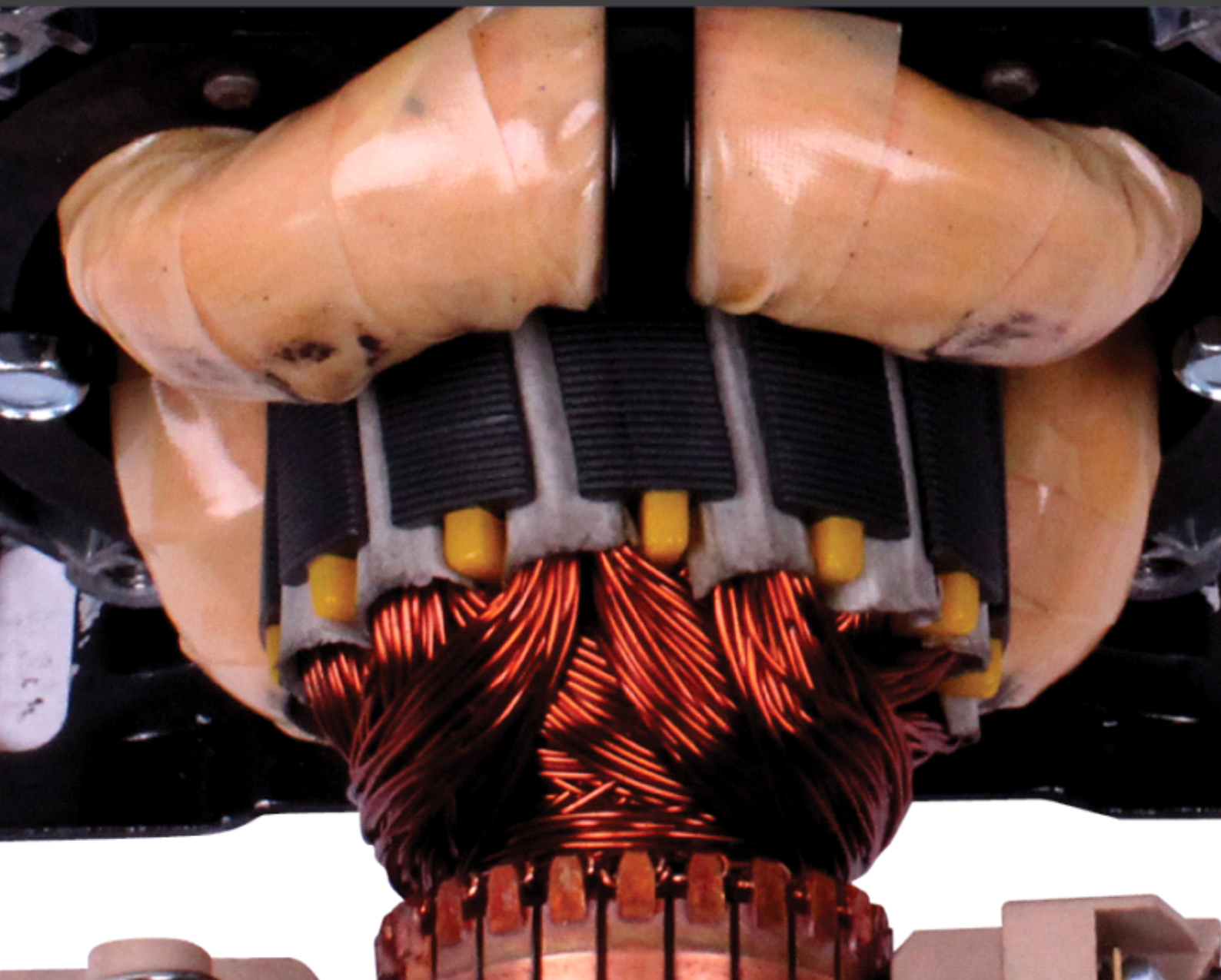




MATRIX | ELECTRICAL
MACHINES

Electrical machines & MATLAB control



MATRIX

CP8385

www.matrixsl.com

Copyright 2024 Matrix Technology Solutions Limited

Santé et sécurité	3
Fiche de travail	4
Fiche de travail 1 - Pour commencer	5
Fiche d'activité 2 - Utilisation de MATLAB pour la création de graphiques	6
Fiche de travail 3 - Contrôle en boucle fermée	7
Fiche 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 liés à l'étude des moteurs électriques. Nous pensons être parvenus à la conception la plus sûre possible. Cependant, il existe encore certains risques dont 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 24 V CA ou CC.

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

L'unité de contrôle ne génère pas de puissance tant qu'un moteur n'est pas branché sur le dynamomètre. Cela permet d'éviter l'utilisation de moteurs de tiers avec le système.

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 que les stu-

Les élèves sont assis à des bureaux pendant qu'ils utilisent l'équipement.

Les pièces rotatives exposées présentent des risques car les cheveux et les vêtements peuvent s'y accrocher. L'utilisation de moteurs de puissance relativement faible réduit 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'utilisation 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.

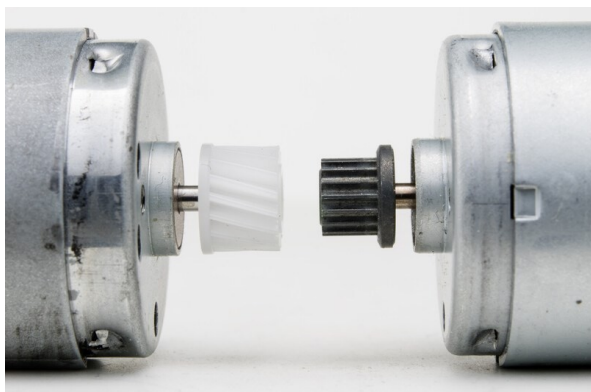
Équipe de
conception
Matrix TSL
33, rue Gibbet
HX1 5BA
Angleterre

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 la connexion à la terre du boîtier de commande.

Fiches de travail

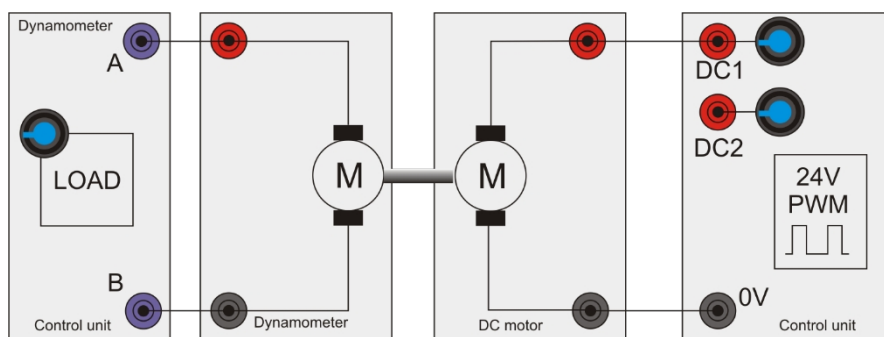
Fiche de travail 1

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 tels que 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 : débranchez le câble USB et utilisez les commandes manuelles pour actionner la tension de sortie du DC1 pour faire varier la vitesse du moteur. Assurez-vous qu'il tourne dans le sens des aiguilles d'une montre et qu'il met pression sur la cellule de charge. S'assurer que le dynamomètre est correctement branché 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) Rebranchez le câble USB.
- 5) Reportez-vous à la section de référence du logiciel MATLAB à la fin de ce document pour comprendre les fonctions MATLAB à votre disposition.
- 6) Créez un programme MATLAB simple qui contrôle la vitesse du moteur à courant continu.
- 7) Étendez votre programme pour vous permettre de contrôler la charge du dynamomètre.

Fiche de travail 2

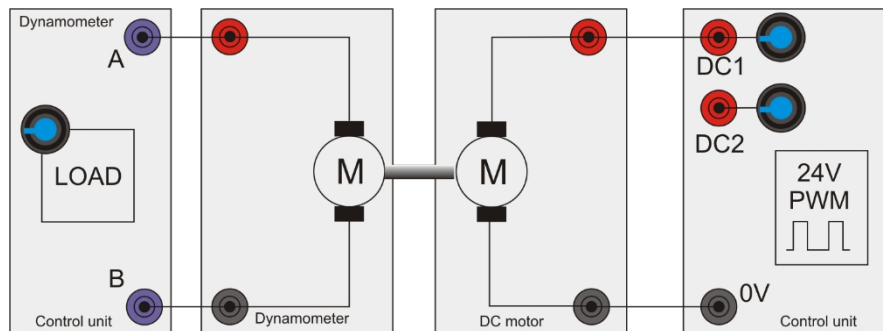
Utilisation de MATLAB pour la création de graphiques



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 photo 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 façon à ce qu'il augmente la charge du dynamomètre de 0 % à 100 % par paliers de 4 %.
- 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.
- 6) Créez des routines supplémentaires qui tracent des courbes de vitesse en fonction du courant et de couple en fonction du courant.

Fiche de travail 3

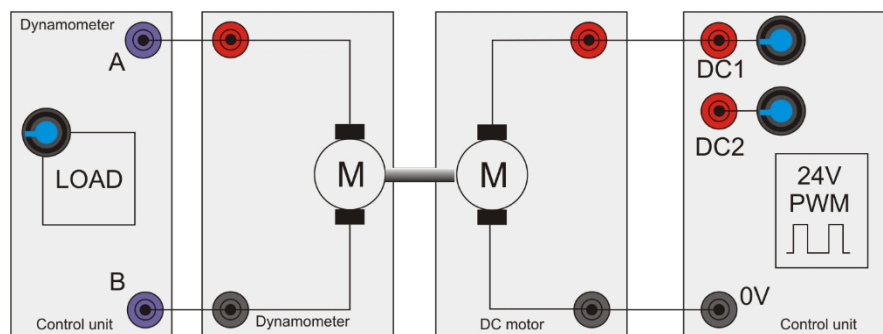
Contrôle en boucle fermée



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 est généralement simple sur...
de l'eau.



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 à l'aide de la fonction API SetDynoLoad et vérifiez que la vitesse reste constante.
- 3) La vitesse peut-elle être rendue constante pour toutes les valeurs de la charge ?

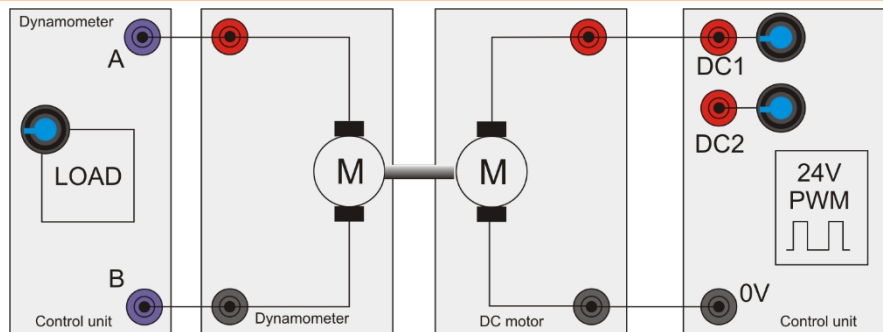
Fiche de travail 4

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 pourquoi 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 contrôler 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 dans la limite de 0 % de la charge et de 70 %.
de charge tout en conservant la vitesse nominale de 1500 tr/min.
- 6) Utilisez les tracés de notre graphique pour prouver que les performances du système sont conformes aux spécifications.

Informations sur le paquet

Comprendre le système

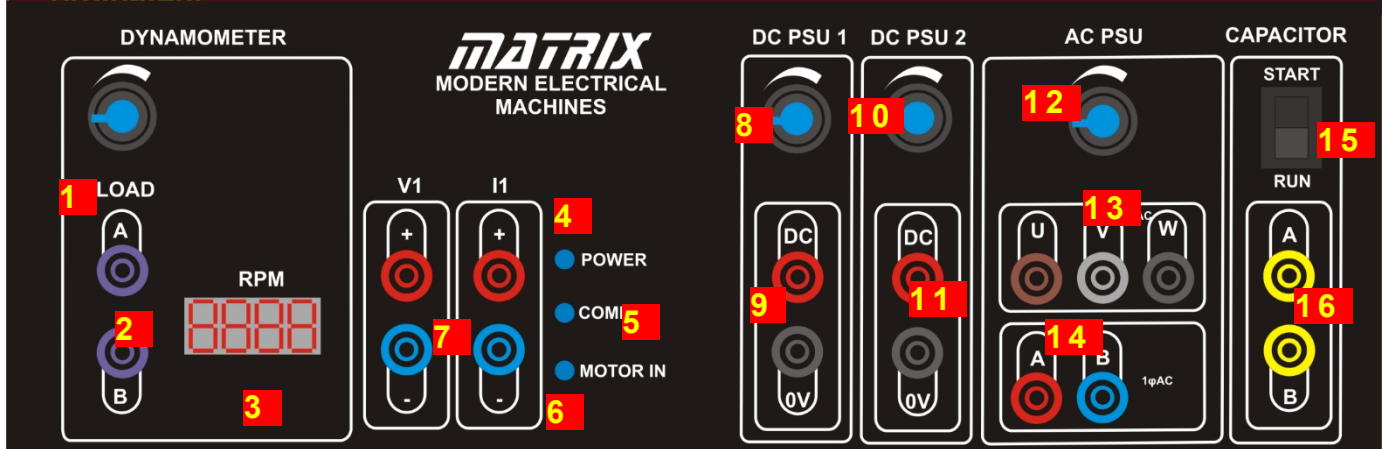


Le système se compose d'un certain nombre de machines électriques de 24 V, d'une unité de contrôle, d'applications logicielles pour piloter l'unité de contrôle et d'un ensemble de feuilles de travail. La photographie ci-dessus montre ces éléments. Il s'agit de

- 1) Le dynamomètre et le berceau qui se connecte à l'unité de contrôle à l'aide d'un câble de type D à 25 voies.
- 2) Solde
- 3) Moteur à l'essai - dans ce cas, un moteur à enroulement shunt
- 4) L'unité de contrôle qui se connecte à un PC à l'aide d'un câble USB.
- 5) Un moteur à enroulement en série
- 6) Moteur à induction CA monophasé
- 7) Moteur CC sans balais
- 8) Moteur à courant continu
- 9) Moteur à induction triphasé à courant alternatif
- 10) L'application logicielle fonctionnant sur un PC

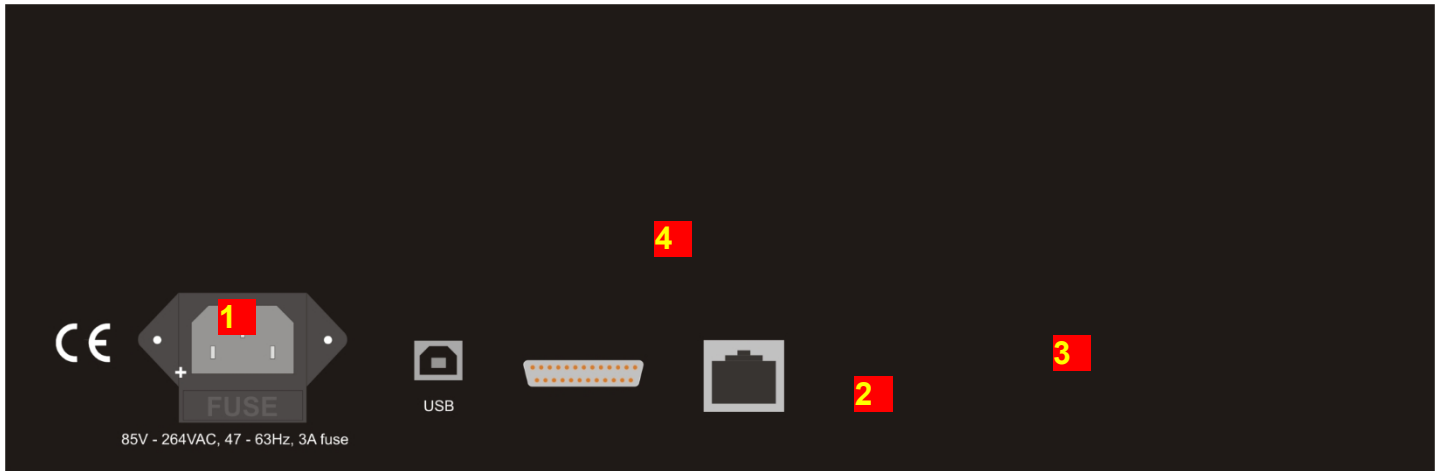
Comprendre l'unité de contrôle

Note - les commandes manuelles sont annulées dès que l'appareil est branché sur un ordinateur.



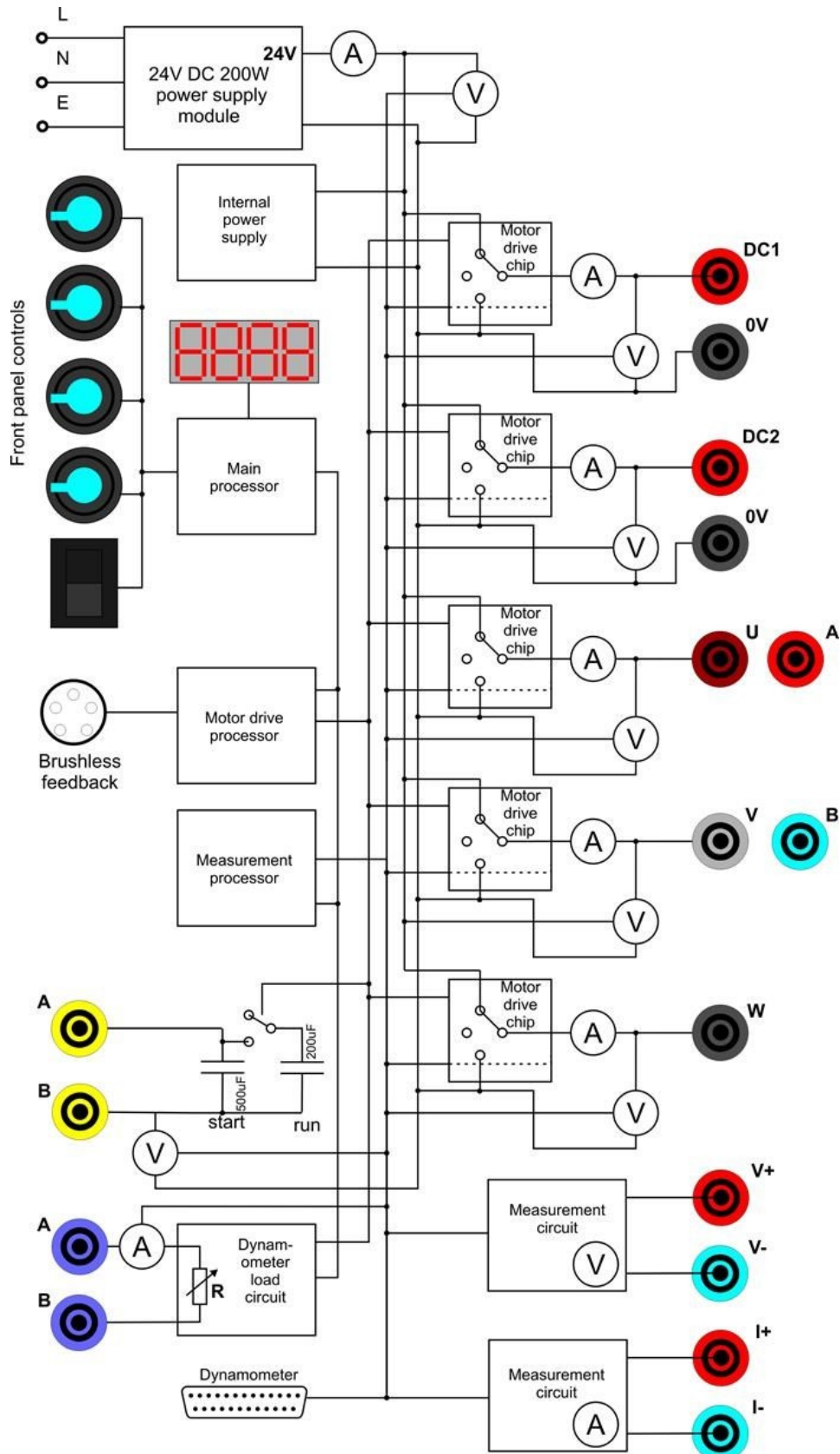
- 1) La résistance du dynamomètre : elle permet de contrôler la résistance effective placée à travers le dynamomètre : une faible résistance électrique signifie une grande résistance mécanique, et une résistance électrique élevée signifie une grande résistance mécanique, et une résistance électrique faible signifie une grande résistance mécanique, et une résistance électrique élevée signifie une grande résistance mécanique.
- sistance signifie une faible résistance mécanique
- 2) Les connexions du dynamomètre
- 3) La vitesse du moteur en tours par minute (RPM).
- 4) Le voyant d'alimentation qui indique que l'unité de contrôle est sous tension.
- 5) La LED COMMS qui s'allume lorsque le logiciel PC est en communication avec l'unité de contrôle.
- 6) La LED "Motor In" qui indique qu'une machine est physiquement connectée au dynamomètre.
- 7) Les connexions internes de l'ampèremètre et du voltmètre
- 8) Contrôle de la sortie DC 1 Supply.
- 9) Les connexions DC 1.
- 10) Contrôle de la sortie DC 2 Supply.
- 11) Les connexions DC2.
- 12) Le contrôle de la fréquence AC
- 13) Les connexions d'alimentation triphasée
- 14) Les connexions de l'alimentation monophasée
- 15) Le sélecteur de mode du condensateur : il contrôle la valeur interne du condensateur connecté à l'entrée de l'appareil les bornes A et B. Il existe deux valeurs : START, RUN
- 16) Bornes à capacité variable.

Comprendre l'unité de contrôle



- 1) La fiche d'entrée secteur - l'appareil est alimenté par 240V ou 110V
- 2) Le connecteur USB
- 3) Un connecteur de type D à 25 voies qui est utilisé pour connecter l'unité de contrôle au frein dynamométrique.
- 4) Trous d'aération pour le ventilateur interne - ne pas couvrir.

Schéma de l'unité de contrôle



Description de l'unité de contrôle

Veillez vous référer au schéma de principe de la page précédente.

L'unité de contrôle livrée avec le système Matrix Modern Electrical Machines est l'une des plus modernes au monde. L'unité comporte trois processeurs distincts : un processeur principal qui gère toutes les tâches d'interface utilisateur et de communication, un processeur d'entraînement du moteur qui gère toutes les sorties de courant élevé et tous les systèmes de forme d'onde et de synchronisation, et un processeur de mesure. Ces trois processeurs utilisent des bus pour communiquer entre eux et entre les différents circuits électroniques de l'appareil.

L'une des principales caractéristiques de l'unité de contrôle est qu'il est possible de mesurer presque toutes les quantités qu'elle contient. Sur le schéma, vous pouvez voir qu'il y a environ 13 ampèremètres et voltmètres distincts dans le système.

Cela donne au logiciel d'instrumentation et à l'utilisateur de nombreuses options pour afficher ce qui se passe dans un système de machine électrique. La mesure de chaque quantité a lieu des milliers de fois par seconde et peut être traitée par des applications logicielles PC pour afficher une quantité ou une forme d'onde.

Les puces d'entraînement du moteur sont toutes des unités à base de FET à faible tension. Comme vous pouvez le voir sur le schéma, toutes les sorties sont des sorties numériques de 24V. Des algorithmes simples de modulation de largeur d'impulsion sont utilisés pour faire varier la tension de sortie.

puissance de sortie effective sur les sorties DC. Sur les sorties AC, la modulation de largeur d'impulsion est à nouveau utilisée.

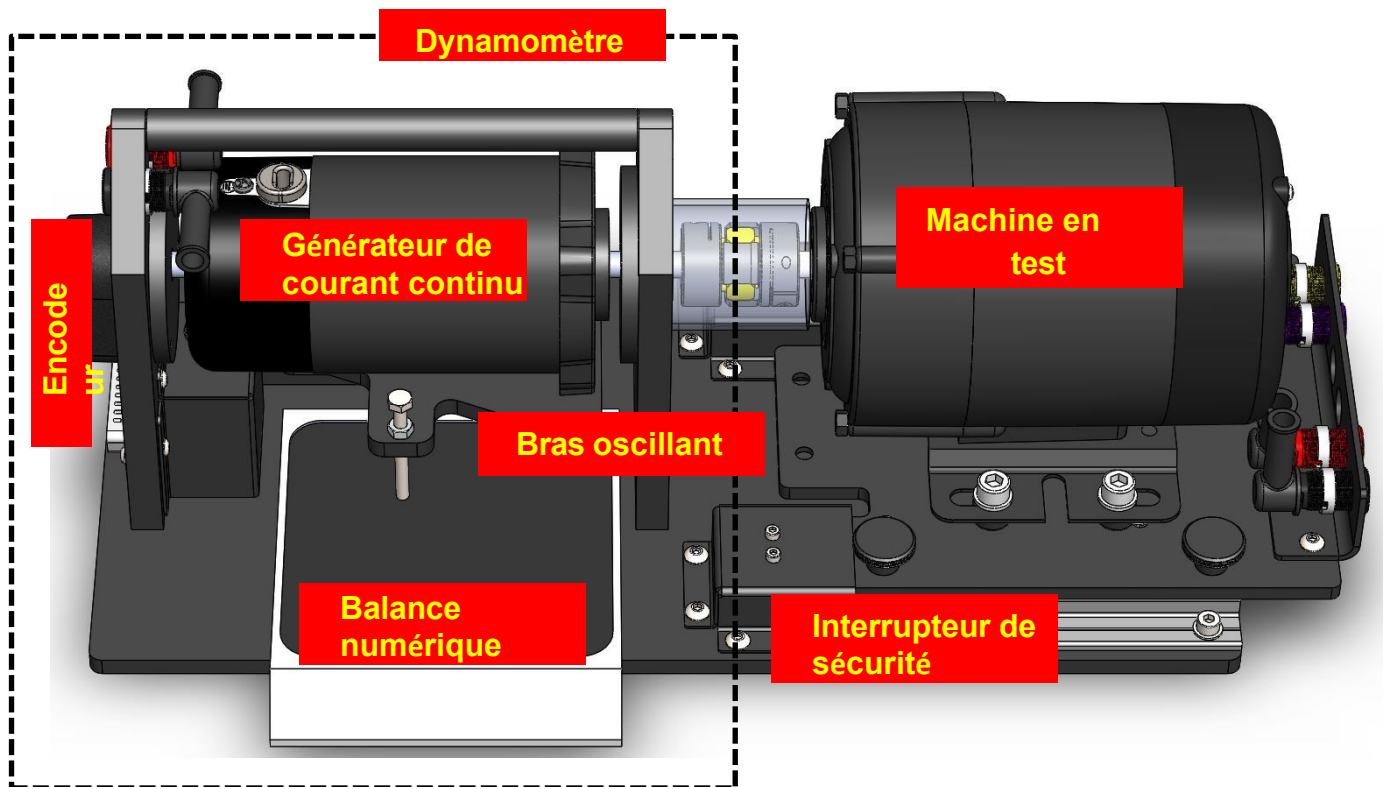
mais avec un algorithme pseudo-sinusoïdal plus avancé qui fait varier la puissance de sortie de manière sinusoïdale

sur la période de la forme d'onde de sortie. Cette technique est utilisée par les contrôleurs de moteur les plus avancés de l'industrie. Les systèmes de commande de moteur plus anciens, et peut-être moins chers, utilisent parfois une simple forme d'onde de sortie numérique pour commander des moteurs à induction triphasés où les trois sorties sont simplement des sorties numériques déphasées de 120 degrés. Cette sortie numérique est également disponible dans certaines applications logicielles livrées avec le système afin que les étudiants puissent étudier l'efficacité de chaque méthode.

Les sorties de la puce du pilote du moteur peuvent être placées dans l'un des trois états suivants : 24V, 0V et circuit ouvert. Cela permet d'appliquer différentes stratégies de commande lors de la génération de formes d'onde pour la commande de moteurs. Dans le cadre de ce module d'apprentissage, les étudiants étudient le PWM pour la commande de moteurs à courant continu et les ondes pseudo-sinusoïdales pour la commande de moteurs à courant alternatif monophasé. L'unité permet également de générer des ondes pseudo-sinusoïdales plus complexes en six étapes pour piloter un moteur triphasé, ce qui peut être examiné à l'aide de l'oscilloscope interne multicanal et d'un oscilloscope externe. Pour faciliter cela, en mode CA, une petite impulsion est donnée sur la sortie DC1 qui permet à un oscilloscope externe d'être déclenché pour capturer les différentes formes d'ondes. L'unité comprend deux valeurs de condensateur avec un simple relais pour l'investigation de démarrage / marche pour les moteurs à induction monophasés.

L'unité comprend également une charge à base de FET pour le dynamomètre dont la résistance effective est contrôlée par logiciel.

Comprendre le dynamomètre



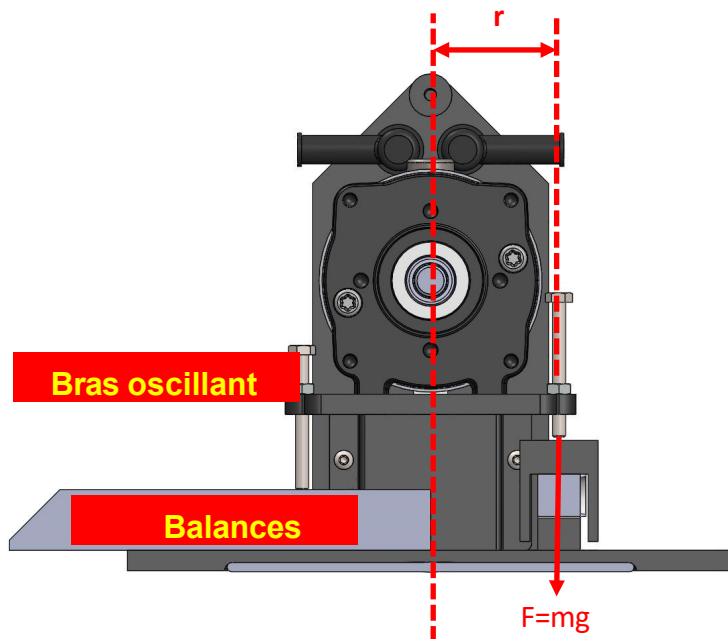
Un dynamomètre est un instrument qui mesure directement la force produite et la vitesse d'un objet en rotation.

machine (la force motrice). Le couple et la puissance peuvent être calculés à partir de la force mesurée et de la force motrice.

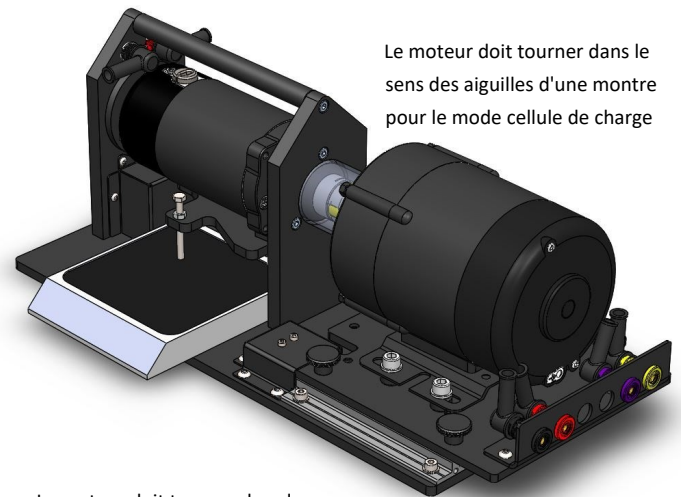
vitesse. Dans notre cas, la rotation mécanique est produite par l'un des moteurs électriques à courant continu ou alternatif. Ce dynamomètre est un dynamomètre à bras oscillant / à balancier de type absorption, dans lequel le moteur est utilisée pour entraîner un générateur de courant continu auquel est attachée une charge à résistance variable. La force résultante peut être mesurée de deux manières :

- Une balance numérique (illustrée dans l'image ci-dessus) peut être utilisée pour mesurer la force verticale effective sur la balance et donne une lecture en kilogrammes. Cette valeur peut être convertie en newtons et newton mètres - voir ci-dessous.
- Une cellule de charge (située derrière le dynamomètre et non représentée sur l'image ci-dessus) connectée à la boîte de contrôle donne la même information - à quelques pour cent près.

Il est à noter que la rotation du dynamomètre détermine si c'est la balance ou la cellule de charge qui est utilisée. La détermination du sens du mouvement de chaque machine est abordée ci-dessous. L'encodeur situé sur l'arbre du dynamomètre permet à l'unité de contrôle de détecter la vitesse de rotation. Celle-ci est affichée sur l'unité de contrôle en tours par minute (tr/min) et sur l'application logicielle. Les machines testées sont couplées au dynamomètre à l'aide d'un accouplement logé dans un tube en plastique pour éviter que les vêtements et les cheveux ne se prennent dans le mécanisme. Le dynamomètre est équipé de deux micro-interrupteurs. Si la machine testée n'est pas en place, l'unité de contrôle ne s'activera pas.



$$\text{Couple} = mgr$$



Le moteur doit tourner dans le sens des aiguilles d'une montre pour le mode cellule de charge

Le moteur doit tourner dans le sens inverse des aiguilles d'une montre pour le mode échelle numérique.

Le dynamomètre tourne dans le même sens que la machine testée. Il fournit une résistance mécanique à la machine testée. Il génère également une tension et, si une résistance est placée entre les bornes du dynamomètre, il produit un courant.

Si la tension et le courant circulent, une puissance est générée - un travail est effectué. Plus la puissance générée par le dynamomètre est importante, plus la machine testée doit générer de puissance mécanique.

Nous testons les machines électriques pour comprendre leurs propriétés et savoir quand utiliser les différents types de machines. Nous les testons en faisant varier leur vitesse et la puissance mécanique qu'elles doivent générer. Le dynamomètre et l'unité de contrôle nous permettent de le faire.

Le dynamomètre est composé d'un moteur à courant continu et d'un berceau. Le moteur à courant continu fait également office de générateur de courant continu. Le berceau suspend la machine à courant continu à deux roulements qui permettent au corps de la machine de tourner.

La force exercée sur le capteur (ou la balance numérique lorsque le moteur tourne dans le sens inverse des aiguilles d'une montre) crée un poids sur le capteur ou la balance (qui représente la masse). Si le bras oscillant est parfaitement équilibré et horizontal lorsqu'il est au repos, la force produite sur le capteur de pesage ou la balance numérique est $\text{Force} = \text{masse} \times 9,81 \text{ m/s}^2$. Le couple peut être calculé à partir de la formule $\text{Couple} = \text{Force} \times \text{Distance}$. Dans ce cas, la distance est celle qui sépare le centre de rotation du générateur de l'axe d'action du bras oscillant. **Pour le dynamomètre Matrix, $r = 38,12 \text{ mm}$.**

Logiciel

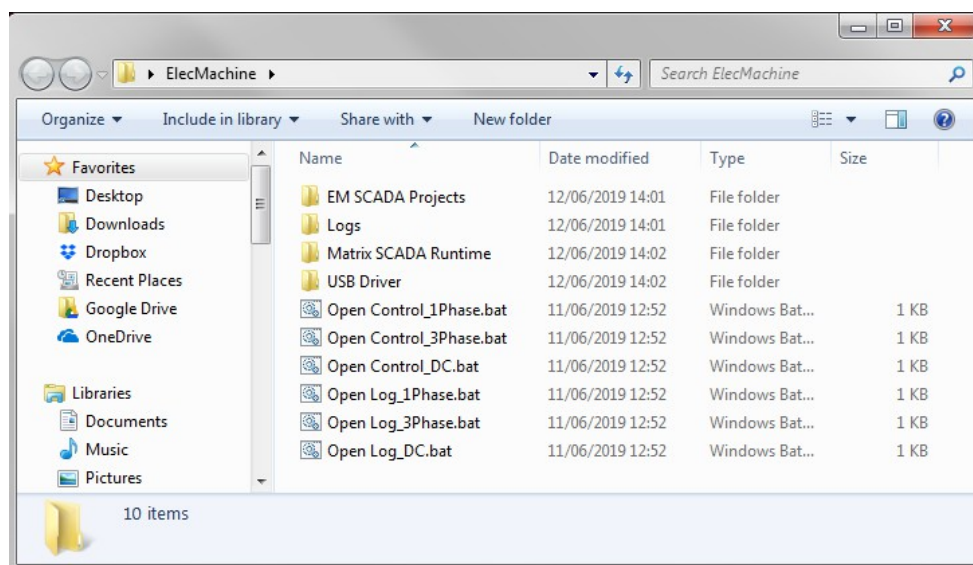
Informations et API MATLAB

Installation du logiciel

Copier le logiciel

Le logiciel pour les machines électriques est disponible en téléchargement sur le site web de Matrix. Lorsque vous décompressez le fichier zip, vous verrez les fichiers suivants :

Il y a 6 applications qui sont utilisées pour l'Electri-



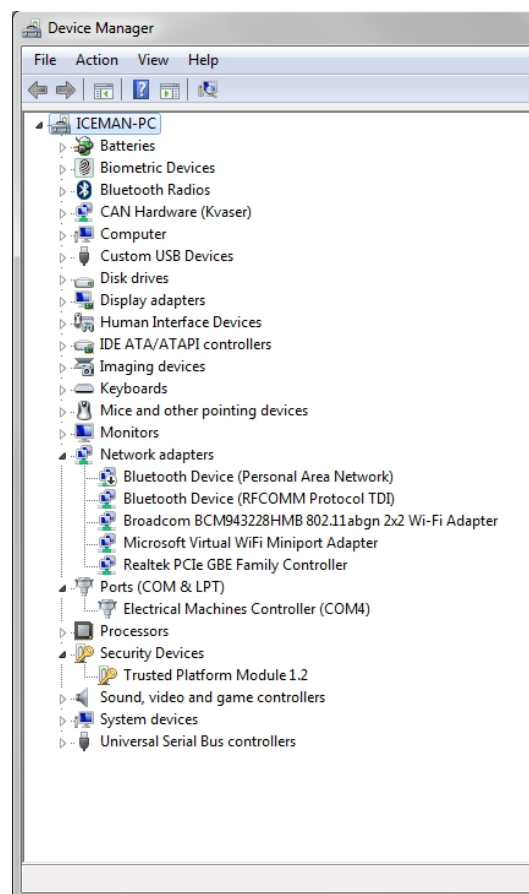
cal Machines range. Ils sont décrits ci-dessous.

Veuillez copier ces programmes et répertoires dans un endroit approprié de votre disque dur. Vous pouvez faire des raccourcis vers les fichiers *.BAT si vous le souhaitez.

Rincer chaque programme en cliquant sur le fichier '*.BAT'.

Installation des pilotes

Allez dans le dossier des pilotes USB et exécutez le pilote 32 bits ou 64 bits selon votre ordinateur. Le pilote du boîtier de commande devrait être installé. Vous pouvez vérifier que le pilote est correctement installé en consultant le Gestionnaire de périphériques et en vérifiant que le contrôleur de machines électriques apparaît sous Ports - voir ci-contre. site.



Logiciel : Intégration de MATLAB



Le logiciel MATLAB peut être téléchargé à partir du site web de Matrix.

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

ordinateur.

Nous avons fourni une bibliothèque de fonctions permettant d'étudier et de contrôler les différents aspects du système des machines électriques. Les différents scripts des fonctions Machines se trouvent dans le dossier Prim-itives. Les scripts nécessitent la déclaration d'une variable globale dans votre projet pour définir le numéro du port COM, comme indiqué ci-contre :

```
global EM_USB ;  
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 "COM1".  
EM_USB = serial('COM1', 'Baudrate',115200, 'Databits',8)  
; fopen(EM_USB) ;  
Le programme principal réside ici  
fclose(EM_USB) % Ferme le port COM
```

Référence

Fonctions MATLAB

Comp Code	Cmd Code	Fonction	Param1	Param2	Param3	Param4	Retour	Taille de retour	Description
0	0	ObtenirAPIVersion					Numéro de version	Octet	Obtient la version de l'API du micrologiciel - Doit être définie à 1 lors de la sortie de l'application
0	1	SetMotorType	Type (0-3)						Définit le type d'entraînement du moteur - 0=DC, 1=1 phase, 2=3 phases, 3=PIDControl
0	2	Régler la tension	TensionLSB	TensionMSB	Canal (0-1)				Règle la tension de l'une des sorties CC (0-3000)
0	3	Définir la fréquence	FreqLSB	FreqMSB					Règle la fréquence de l'entraînement CA (10 - 100)
0	4	SetActive	Course (0-1)						Contrôle si les sorties sont actives ou non
0	5	SetPhase	Canal (0-2)	PhaseLSB	PhaseMSB				Permet de décaler l'une des phases du courant alternatif par rapport à la valeur par défaut de 180 ou 120
0	6	Poids zéro							Met à zéro la sortie du capteur de pesage
0	7	SetACDriveMode	Mode (0-1)						Définit le mode de la forme d'onde de la sortie AC 0=Sine Wave PWM 1=Trapezoidal
0	8	SetFanSpeed	SpeedLSB	SpeedMSB					Contrôle la vitesse du ventilateur, largement redondant maintenant et le ventilateur devrait être automatique (0-3000)
0	10	SetCapBank	Valeur (0-3)						Contrôle les condensateurs connectés aux bornes C - 0=0F, 1=200uF, 2=300uF, 3=500uF
0	11	SetDynoLoad	Valeur (0-255)						Contrôle la résistance de la charge du dynamomètre (0=environ 145R, 255=environ 5R).
0	12	SetACGenMode	Valeur (0-1)						Contrôle la génération de l'onde sinusoïdale triphasée (0=simple, 1=précise (double bosse)).
0	13	GetLogData						Octet[64]	Collecte un paquet de 64 octets contenant les données du journal
0	14	LogDataReady					Prêt (0-1)	Octet	Vérifie si les données du journal sont prêtes à être envoyées
0	15	IsBrushlessActive					Bactive (0-1)	Octet	Vérifie que le retour d'information sans balais est connecté
0	16	GetLogStartAngle					Angle (0-35)	Octet	Obtient l'angle de phase du courant U au début de l'enregistrement, 0-35 = 0 à 350 degrés
0	17	Étalonnage des capteurs de charge	FloatLSB	Flottant1	Flottant2	FloatMSB			Définit le facteur d'échelle du capteur de charge - généralement 0,0046082949308756
0	20	GetRPM					RPM	Int non signé	Renvoie le nombre de tours/minute avec une résolution de 0,1 tour/minute.
0	21	Position absolue					AbsPosition	Long non signé	Renvoie la position absolue du codeur avec une résolution de 0,25 degré.
0	22	Obtenir la tension	Canal (0-5)				Tension	Int non signé	Renvoie la lecture de la tension avec une résolution de xxx V
0	23	GetCurrent	Canal (0-7)				Actuel	Int non signé	Renvoie la valeur actuelle avec une résolution de xxx V
0	24	GetErrorStatus					ErrorStatus	Octet	Obtient l'état actuel de l'erreur
0	25	GetSwitches					Interrupteur	Octet	Renvoie l'état des interrupteurs de sécurité (0=Stop, 1=OK)
0	26	GetLoadCell	Mode (0-1)				Chargement	Int non signé	Renvoie le poids sur la cellule de charge 0=poids en G 1=couple en mNM
0	27	Angle de phase	Chaîne						Renvoie la différence d'angle de phase mesurée à l'aide des pics de la forme d'onde
0	28	GetMotorDir					Direction	Octet	Renvoie le sens de rotation du moteur
0	29	ResetAbsPosition							Remet la position angulaire absolue à 0
0	50	SetSetpoint	FloatLSB	Flottant1	Flottant2	FloatMSB			Définit le point de consigne de l'angle
0	51	SetKP	FloatLSB	Flottant1	Flottant2	FloatMSB			Règle le gain proportionnel
0	52	SetKI	FloatLSB	Flottant1	Flottant2	FloatMSB			Règle le gain intégral
0	53	SetKD	FloatLSB	Flottant1	Flottant2	FloatMSB			Règle le gain dérivé
0	54	Obtenir un retour d'information					Angle	Flotteur	Obtient la position absolue sous la forme d'un angle à virgule flottante
0	55	GetControlOutputs					OP,OI,OD,Sum	Flottant[4]	Obtient les sorties de contrôle PID sous forme de valeurs à virgule flottante
0	56	SetMaxDriveDuty	FloatLSB	Flottant1	Flottant2	FloatMSB			Définit le service d'entraînement maximal sous la forme d'une valeur à virgule flottante comprise entre 0 et 1.
0	57	GetDataArraySize					Nombre d'échantillons	Octet	Collecter le nombre d'échantillons de contrôle dans la mémoire tampon
0	58	Tableau de données (GetDataArray)						Flottant[5]	Collecter SP, FB, PO, IO, DO
0	59	Get3xDataArray						Flottant[15]	Réseau de collecte de données x 3
0	80	SetIPAddress	IPO	IP1	IP2	IP3			Définit l'adresse IP de la boîte, par défaut 192.168.1.235
0	81	SetIPPort	PortLSB	PortMSB					Définit le port de la boîte, par défaut 88

Le tableau ci-dessus présente les appels de fonctions publiques disponibles dans la bibliothèque MATLAB et leur fonction.

Exemple MATLAB

Le texte ci-contre est un simple script de commande de moteur écrit en Matlab. Ce texte et les séries de commandes API présentées dans les pages précédentes devraient vous aider à créer vos programmes de test pour les machines électriques.

```
global EM_USB ;
Remplacez 'COM1' par le numéro de port correct pour la
boîte EM EM_USB = serial('COM1', 'Baudrate', 115200,
'Databits', 8); fopen(EM_USB);
SetMotorType (0); % DC Motor pause(0.1)
;
motor_voltage = 10; % 10V DC

% Test comms are ok
api_version = GetAPIVersion();
disp(['api_version = ', num2str(api_version)]);

Vérifier les interrupteurs de sécurité du
banc d'essai switch_state = GetSwitches ()
;
if switch_state >= 1
disp('==== DC MOTOR TEST START ====')
SetDynoLoad(0);
index = 1;
SetActive(1); % Moteur en marche
SetVoltage(0, motor_voltage); % Régler la tension DC1
pause(0.3); % Attendre que le moteur se mette en marche
% Effectuer les mesures échantillonnées dans une
boucle for loop = 0:255
load_cell_torque_range(index) = GetLoadCell(1);
encoder_speed_range(index) = GetRPM();
index = index + 1;
SetDynoLoad(loop);
pause(0.1);
fin
% Fin de l'échantillonnage, arrêt du moteur, fermeture des
communications SetActive(0); % Moteur désactivé
fclose(EM_USB)

% Tracer le RPM en fonction du temps
current_torque_plot = figure;
figure(current_torque_plot);
movegui(current_torque_plot, 'northwest');
hold on
p2 = polyfit(encoder_speed_range, load_cell_torque_range, 1); % ligne de
meilleur ajustement y2 = polyval(p2, encoder_speed_range);
plot(encoder_speed_range, load_cell_torque_range, 'ro', encoder_speed_range, y2, '-r');
current_speed_polynomial = poly2sym(p2, sym('T'));
current_speed_equation = sprintf('Speed = %s', current_speed_polynomial);
disp(current_speed_equation);
grille sur
legend('Samples', 'Best fit line', 'Location', 'southeast');
xlabel('Speed, RPM');
ylabel('Couple, mNm');
title(['1PM DC Motor Speed v Torque @ ', num2str(motor_voltage), 'V']);
attendr
e autre
chose
disp('==== Motor not fitted in Dynamometer ====');
end
```