

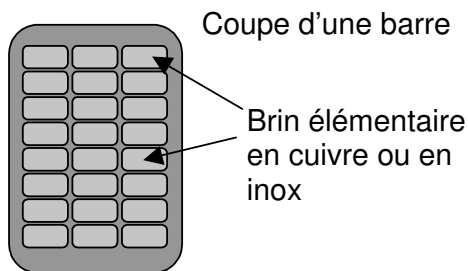
## Dispositif de Contrôle de l'isolation des brins des barres de stator d'alternateur

### Avertissement

Les sujets concernant des applications industrielles sont toujours très longs, mais les algorithmes sont souvent simples. En général, le sujet est en fait un cahier des charges avec des solutions suggérées.

### Description

Dans les alternateurs, les stators comportent des barres conductrices dans lesquelles sera induit le courant. Dans le cas de courant alternatif, et c'est toujours le cas des alternateurs, le courant a tendance à se concentrer en périphérie du conducteur, ce qui diminue en fait la section efficace. Pour palier ce défaut, on utilise une barre comportant plusieurs conducteurs isolés (brins) tressés de manière à ce que chaque brin occupe différents emplacements dans le volume de la barre. Chaque brin est alors équilibré, du point de vue densité de courant par rapport à l'autre. En bout de barre, les brins sont à nouveau réunis par soudage.

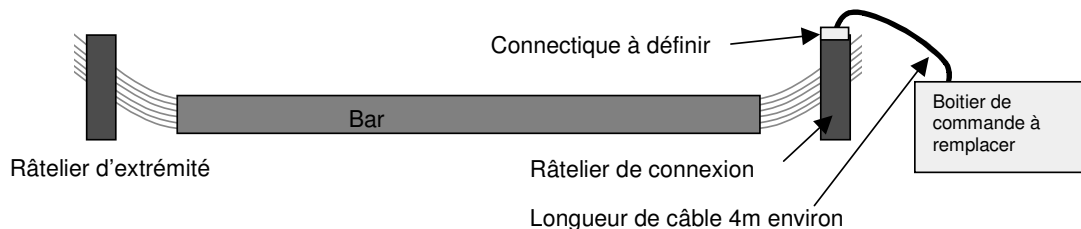


Section d'une barre



Connexion des brins au connecteur du dispositif de contrôle

Lors du tressage, il arrive que l'isolant se fissure et que des brins voisins soient en contact électrique, ce qui aurait pour effet de dégrader les performances électriques de la barre. On effectue, après l'opération de tressage, un contrôle de l'isolation entre les différents brins. Si un court-circuit entre deux brins est détecté, il faut procéder à sa localisation, et à la réparation lorsque c'est possible.



Le tressage est tel que chaque brin change régulièrement de brin voisin. Il est donc nécessaire de rechercher les courts-circuits de manière logique.

Le contrôle de la qualité de l'isolation des brins d'une barre est réalisé avant la réunification des brins en extrémité. Pour réaliser ce test, la barre est insérée entre deux râteliers, dont l'un a pour fonction de connecter électriquement le dispositif de test simultanément à chaque brin.

Chaque brin est alors connecté au commun d'un relais. Au repos, le relais relie le brin attribué à la masse. Lorsqu'il est excité, il relie le brin à une source haute tension (220V, haute impédance, pour éviter les accidents en cas de court-circuit). Il est alors possible de mesurer le courant qui circule vers le brin. Ce courant reste faible (réactif, car dû à la capacité entre les brins) lorsque le brin est normalement isolé des autres. En cas de court-circuit, le courant est important, supérieur à une valeur que nous appellerons SEUIL\_COURTCIRCUIT. On peut, de même, détecter un défaut de connexion au niveau du râtelier, car dans ce cas, la capacité parasite naturelle entre brins n'est plus détectée et le courant devient très faible, dû seulement aux capacités internes du câble de liaison.

Le nombre de brins dans une barre dépend du type d'alternateur auquel elle est destinée. On peut procéder à la détermination du nombre de brins par des mesures d'impédance sur chaque connexion. Le dispositif de mesure est, bien entendu, supposé comporter un nombre de relais supérieur au nombre de brins à contrôler.

On souhaite écrire des algorithmes concernant des procédures et fonctions utilitaires destinées à mener à bien des contrôles de barres.

### **1. Fonction de détermination du nombre de brins connectés au râtelier.**

Les brins sont connectés dans l'ordre au râtelier. Ainsi, le relais numéro  $j$  est connecté au brin numéro  $j$  (avec indigage par rapport à 0). Pour estimer le nombre de brins connectés au râtelier, on décide de mesurer les impédances au niveau de chaque relais, en commençant par le relais de rang le plus élevé, et en effectuant les mesures sur les relais successifs, jusqu'à obtention d'une valeur d'impédance significative (suffisamment faible) : ainsi, tous les relais non connectés présenteront des impédances anormalement élevées.

On dispose de :

- la variable globale entière  $nbRelais$  qui indique le nombre de relais de mesure disponibles (la capacité de mesure peut être augmentée en ajoutant des cartes relais supplémentaires),
- la procédure  $SelectRelais(n)$ , qui permet d'activer le relais numéro  $n$  afin de mesurer l'impédance,
- la fonction  $GetImpedance()$  qui retourne un entier indiquant l'impédance mesurée au niveau du relais activé,
- la constante  $IMPEDANCE\_SEUIL$ , valeur en dessous de laquelle on estime qu'un brin est connecté au râtelier.

Ecrire la fonction  $GetnbStrands()$  qui retourne le numéro du brin de rang le plus élevé trouvé. La numérotation des brins, ainsi que celle des relais, commence à 0.

### **2. Fonction de détection de faux contacts au niveau du râtelier**

Il arrive qu'un brin soit mal connecté au niveau du râtelier. Comme vu dans la question précédente, on détecte une mauvaise connexion ou une connexion inexistante en mesurant l'impédance. Dans le cas présent, on connaît le nombre de brins connectés donné par la variable globale  $nbStrands$ , et on recherche les brins mal connectés en commençant par le

premier (indice 0). On dispose bien entendu des routines *SelectRelais(n)* et *GetImpedance()*, ainsi que de la constante `IMPEDANCE_SEUIL`.

Ecrire la fonction *GetBadConnect(n)* qui retourne le numéro du premier relais suivant le relais de rang *n* (donné en argument de la fonction) dont l'impédance mesurée dépasse la valeur `IMPEDANCE_SEUIL`. L'argument *n* indique le rang du relais à partir duquel on commence le contrôle. Si aucun défaut de connexion n'est détecté, la fonction retourne -1. Les relais sont numérotés à partir de 0.

### 3. Fonction de détection de court-circuit

Ecrire la fonction *TestStrands()* qui retourne un entier, nombre de défauts détectés. Cette fonction va tester tous les brins et mémoriser les brins en défaut. La détection d'un brin en défaut se fait en mesurant l'impédance entre le brin et la masse. Normalement, chaque brin est relié à la masse par le relais qui y est attaché, lorsqu'il est au repos. On dispose, pour la mémorisation des défauts, de deux tableaux d'entiers (variables globales) : *Nb[]* et *Impedance[]* de `MAXDEFAUT` éléments (`MAXDEFAUT` est une constante numérique arbitraire que l'on fixe au moment de la compilation. Elle est donc supposée définie et connue).

La détection de court circuit se déroule comme suit :

On met à 0 tous les éléments des tableaux *Nb[]* et *Impedance[]*

On teste successivement tous les brins et on met à jour les tableaux en plaçant dans *Nb[i]* le rang du **brin en défaut**, et dans *Impedance[i]* l'impédance mesurée pour ce brin. Un brin est en défaut si l'impédance mesurée est inférieure à la valeur constante prédéfinie `IMPEDANCE_MINI`.

On retourne le nombre de défauts relevés.

On dispose des fonctions *SelectRelais(n)* et *GetImpedance()*, de la variable globale *nbStrands* qui contient le nombre de brins connectés à tester.

On remarquera que cette fonction ne détecte que les courts-circuits, et n'indique pas quels sont les brins en relation.

### 4. Fonction de mesure courant

La mesure du courant se fait en relevant la chute de tension aux bornes d'une résistance traversée par ce courant. Cette tension est amplifiée et redressée avant d'être appliquée au convertisseur analogique-numérique. Le facteur d'amplification est fixe et on préfère commuter la résistance de mesure pour des raisons de précision de la mesure.

On dispose de la fonction *Convert()* qui retourne un entier compris entre 0 et 1023. C'est une image de la valeur de la tension aux bornes de la résistance de mesure. Pour une tension de 5V, *Convert()* retourne la valeur 1023, et 0 pour une tension nulle. On peut donc facilement calculer la valeur de la tension mesurée. Les résistances commutables sont organisées en décades, ce qui signifie que les résistances disponibles auront pour valeur 1, 10, 100 et 1000 ohms. Ainsi, si l'on commute vers une décade supérieure, on multiplie la valeur mesurée par 10.

*Remarque concernant les opérations utilisant différents types de variables :*

Nous considèrerons qu'une opération utilisant deux variables de types différents donnera comme résultat le type de rang le plus élevé. Ainsi, classés par ordre de rang croissant, nous trouverons :

- entiers sur 1 octet,
- entiers sur 2 octets,
- entiers sur 4 octets,
- Réels sur 4 octets,

- Réels sur 8 octets.

L'addition d'un entier sur 2 octets avec un réel sur 4 octets donnera un réel sur 4 octets. Lorsqu'il s'agit de constantes, le compilateur considèrera la valeur 5 comme un entier, et la valeur 5.0 comme un réel. Si on multiplie 10.0 par 20, le résultat sera le réel 200.0 ; si l'on multiplie 10 par 20, le résultat sera l'entier 200.

Pour assurer une mesure suffisamment précise, on change de décade tant que cela est possible. Dans notre cas, on cherche à avoir une valeur mesurée supérieure à 100. Si la valeur mesurée est inférieure à 100, on augmente la résistance de mesure si c'est possible, ce qui aura pour effet de multiplier la valeur mesurée par 10, sans pour autant dépasser la capacité du convertisseur.

On dispose de la fonction  $R(i)$  qui commute la résistance d'ordre  $i$ . Pour  $i = 0$ , elle commutera la résistance de 1 Ohm, et pour  $i = 3$ , celle de 1000 ohms. Les valeurs des résistances sont placées dans l'ordre croissant dans le tableau de réels `Res[]`

Après avoir connecté la résistance adéquate, on calcule la tension mesurée, et on détermine la valeur du courant qui est un réel, valeur qui sera retournée.

On demande d'écrire la fonction *MesureCourant()*, qui retourne un réel, valeur du courant mesuré.