

LE BLOG D'ESKIMON

Tutoriels Arduino (ou pas) et articles divers

[Arduino 603] A petits pas, le moteur pas-à-pas

Posté par Eskimon le 12 décembre 2013, 12:22

Pour en terminer avec les différents types de moteurs qui existent, nous allons parler d'un moteur un peu particulier (encore plus que le servomoteur !) et qui est cependant très utilisé dans le domaine de la robotique et tout ce qui touche à la précision d'un mouvement. Comme à l'habitude, nous allons d'abord voir le fonctionnement de ces moteurs, pour ensuite apprendre à les utiliser.

Ce moteur utilise des éléments que nous avons vus dans des chapitres précédents (sur les moteurs à courant continu). Si vous ne vous rappelez pas du L298 je vous conseille de retourner prendre quelques informations à ce sujet 😊

Sommaire

- Les différents moteurs pas-à-pas et leur fonctionnement
- Se servir du moteur
- Utilisation avec Arduino

Les différents moteurs pas-à-pas et leur fonctionnement

Les moteurs pas-à-pas... encore un nouveau type de moteur. Une question vous taraude sûrement l'esprit :

Pourquoi il existe tant de moteur différent !?

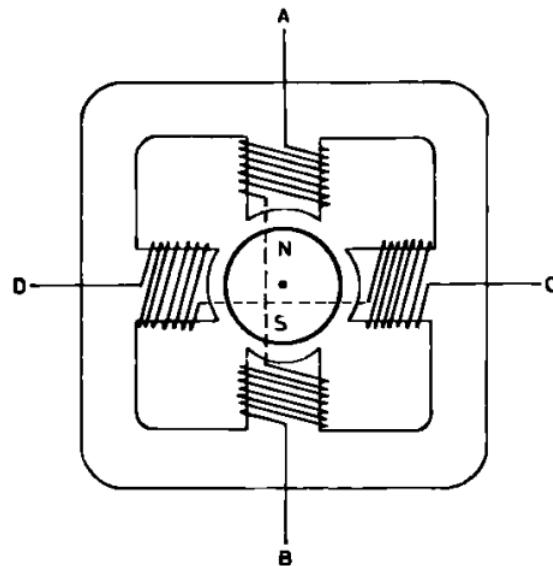
Et bien je vous répondrais par une autre question : Pourquoi il existe autant de langages de programmation différent !! La réponse est pourtant simple : Car ils ont tous leurs avantages et leurs inconvénients. Par exemple, un servomoteur pourra facilement maintenir la position de son axe, tandis que le moteur à courant continu sera plus facile à faire tourner à différentes vitesses. Eh bien le but du moteur pas-à-pas (que j'abrégerais moteur pàp) est un peu une réunion de ces deux avantages. Vous pourrez le faire tourner à des vitesses variables et la position parcourue sera aussi facile à déterminer. En contrepartie ces moteurs ne peuvent pas tourner à des vitesses hallucinantes et sont plus délicat à mettre en œuvre que les moteurs CC par exemple (mais rien d'insurmontable je vous rassure). En parlant de précision, savez-vous dans quel objet du quotidien on retrouve beaucoup de moteur pàp ? Dans l'imprimante (éventuellement scanner aussi) qui traîne sur votre bureau ! En effet, l'aspect « précision » du moteur est utilisé dans cette situation sans avoir besoin d'aller vraiment vite. Vous pourrez donc en trouver un pour faire avancer les feuilles et un autre pour déplacer le chariot avec les cartouches d'encre (et encore un autre pour déplacer le capteur du scanner). Donc si vous avez une vieille imprimante destinée à la poubelle, vous savez ce qu'il vous reste à faire 😊 ! Les moteurs que vous pourrez trouver posséderont 4, 5 voire 6 fils. Le premier (4 fils) est appelé **moteur bipolaire**, les deux autres sont des moteurs **unipolaires** ou à **reluctance variable**. Tout cela doit-être encore un peu confus. Voyons donc plus clairement comment cela marche !

Fonctionnement des moteurs

Comme précédemment avancé, ce moteur possède une certaine complexité pour être mis en œuvre. Et ce, plus que les précédents. Vous souvenez-vous du moteur CC (j'espère bien ! 😊). Il était composé d'un ensemble d'aimants sur l'extérieur (le stator) et d'une partie bobinée où le champ magnétique était créée dynamiquement avec un ensemble collecteur/balais qui transmettait l'électricité aux bobines au centre (rotor). Dans le cas du moteur pàp, c'est sur le rotor (au centre) que l'on retrouve l'aimant permanent, et les bobines sont sur le stator (autour du rotor). Comme pour les moteurs à courant continu, le but du jeu, en quelque sorte, est de « faire tourner un champ magnétique » (à prendre avec des pincettes) pour faire tourner l'aimant fixé au rotor. Il existe cependant différents types de moteurs pàp dont le placement des bobinages diffère les uns des autres et la façon de les alimenter n'est pas non plus identique (d'où une complexité supplémentaire lorsque l'on veut changer le type de moteur pàp à utiliser...). Nous allons maintenant les étudier l'un après l'autre en commençant par celui qui semble avoir le fonctionnement le plus simple à assimiler.

MOTEUR PÀP BIPOLAIRE À AIMANTS PERMANENTS

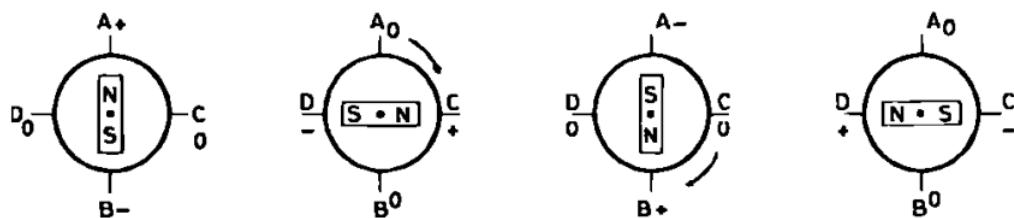
Ce moteur possède quatre fils d'alimentation pour piloter des bobines par paire. Comme un schéma vaut mieux qu'un long discours, voici comment il est constitué :



Vous l'aurez compris, les bobines sont reliées deux à deux en série et sont donc pilotés ensemble. Il n'y a donc finalement que deux enroulement à commander puisque deux bobines montées en série n'en font plus qu'une. Leur placement de part et d'autre de l'aimant permanent du rotor permet de piloter ce dernier. Voyons comment.

- Lorsqu'il n'y a aucun courant traversant les bobines, le rotor (où l'axe de sortie est lié) est libre de tourner, rien ne cherche à le retenir dans sa course
- Maintenant, si nous décidons de faire passer du courant entre les points C et D pour alimenter la bobine de gauche et celle de droite. Un courant va s'établir et deux champs électromagnétiques vont apparaître de part et d'autres du rotor. Que va-t-il alors se passer ? L'aimant du rotor va tourner sur lui même pour se placer de façon à ce que son pôle Nord soit en face du pôle Sud du champ magnétique créé dans la première bobine et que son pôle Sud soit en face du pôle Nord créé dans la deuxième bobine
- Si ensuite on alimente non plus les bobines entre C et D mais plutôt celles entre A et B, le rotor va alors tourner pour s'aligner à nouveau vers les pôles qui l'intéressent (Nord/Sud, Sud/Nord)
- Et c'est reparti, on va alors alimenter de nouveau les bobines entre D et C, donc avec un courant de signe opposé à la fois où l'on les a alimenté entre C et D (par exemple C était relié au « + » de l'alimentation tout à l'heure et là on le fait passer au « - », idem pour D que l'on fait passer du « - » au « + ») et le moteur va encore faire un quart de tour
- On peut continuer ainsi de suite pour faire tourner le moteur en faisant attention de ne pas se tromper dans les phases d'alimentation

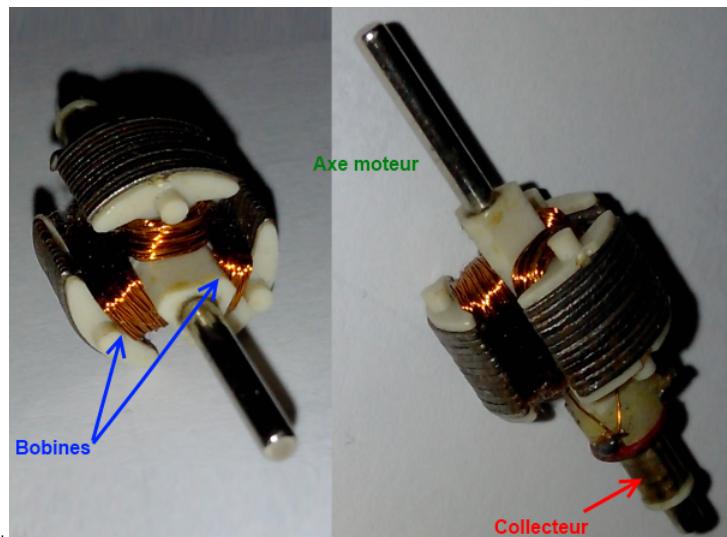
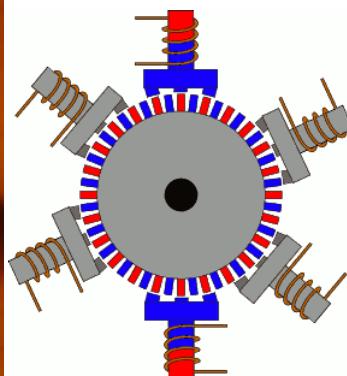
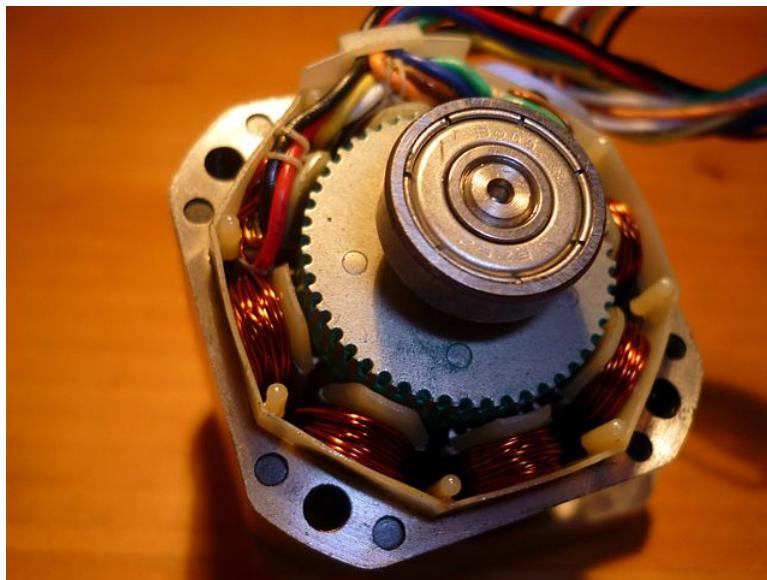
A chaque phase on va donc faire tourner le moteur d'un quart de tour :



Ce quart de rotation s'appelle un **pas**. Et comme il faut plusieurs pas pour faire tourner le moteur sur 360° , on l'a donc appelé ainsi, le moteur **pas-à-pas**.

Dans le cas illustré ci-dessus, on dit que le moteur fait 4 pas par tour. Il existe bien des moteurs qui font ce nombre de pas, mais il en existe qui ont un nombre de pas plus conséquent (24, 48, etc.). Leur constitution mécanique est différente, ce qui leur confère ce pouvoir, bien que le fonctionnement reste identique, puisque l'on cherche toujours à attirer un aimant grâce à des champs magnétiques créés par des bobines parcourues par un courant. Pour avoir plus de pas, on multiplie les aimants au centre. Sur l'image ci-dessous, on peut bien voir les bobines (en cuivre à l'extérieur) et tout les aimants au centre (les petites dents). Il existe aussi deux autres modes de fonctionnement que nous verrons dans la partie suivante : **le pilotage avec couple maximal** et **le pilotage par demi-pas**.



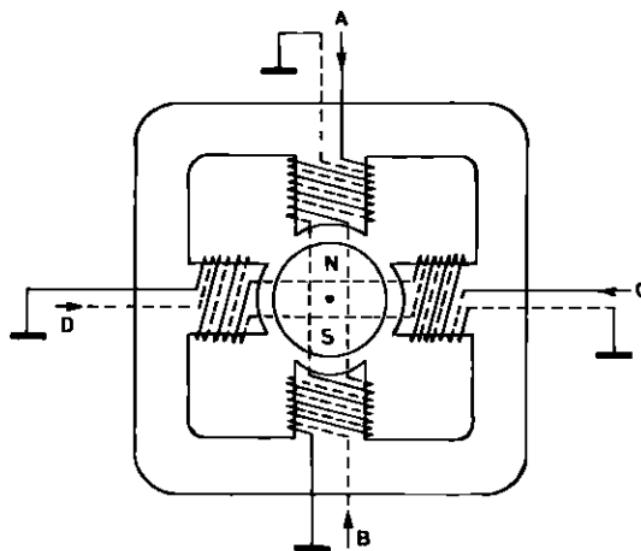


(Pour rappel, voici la vue d'un moteur à courant continu :

LE MOTEUR UNIPOLAIRE

Le moment est enfin venu de vous révéler la véritable signification des noms du moteur vu précédemment et de celui-ci même... non il ne faut pas lire ça sur un ton tragique. 😊 Le moteur bipolaire est nommé ainsi car il présente la faculté d'être commandé en inversant simplement la polarité de ces bobinages. Quant au moteur unipolaire, pas besoin de faire cette inversion, chaque bobinage est commandé séparément et ne requiert qu'une alimentation présente ou absente selon que l'on veuille ou non créer un champ magnétique en son sein. La commande sera donc plus simple à mettre en place qu'avec le moteur bipolaire. Cependant, le nombre de bobine étant plus important, la quantité de cuivre également et le prix s'en ressent ! En effet, il s'agit bien de 4 bobine bien distinctes, alors que le moteur bipolaire à aimant permanent en possède finalement quatre moitiés de bobines, donc deux bobine complètes.



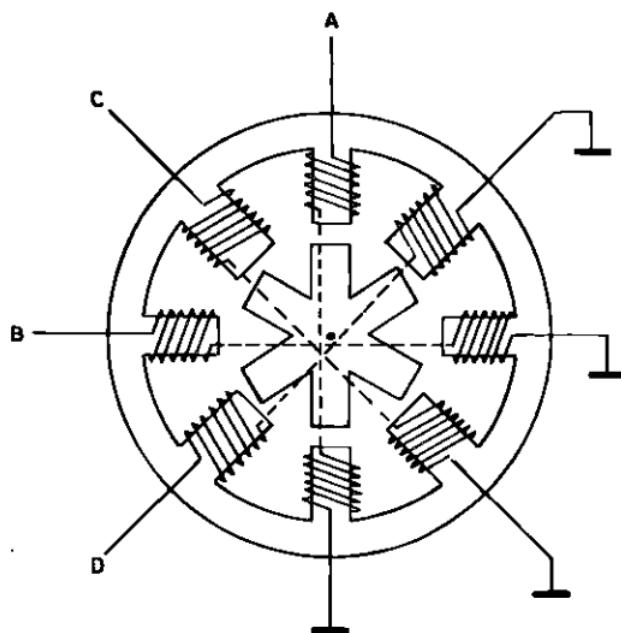


On retrouve nos quatre fils A, B, C et D ainsi qu'un fil de masse commun (bon ben imaginez-le puisqu'il n'est pas dessiné comme tel sur le schéma). Soit 5 fils (v'là pas qu'on sait compter maintenant ! 😊). Le fonctionnement est rigoureusement identique que le précédent moteur. On cherche à créer un champ magnétique « rotatif » pour faire passer l'aimant alternativement devant chacune des bobines. On va donc alimenter la bobine A, puis la C, puis la B, puis la D selon le schéma ci-dessus. Et voilà, le moteur aura fait tout un tour assez simplement. Il suffit d'avoir quatre transistors (un par enroulement) sans avoir besoin de les disposer en H et de les piloter deux à deux. Ici il suffit de les alimenter un par un, chacun leur tour. Facile, non ? 😊

LE MOTEUR À RÉLUCTANCE VARIABLE

gné 😊 ? c'est quoi ce nom barbare ?

Ce moteur est un peu plus compliqué, mais c'est aussi le dernier que nous verrons et le plus fascinant ! Contrairement aux deux précédents, ce moteur ne possède pas d'aimants permanents, ni même d'aimant tous court ! Non, en son centre on trouve un simple morceau de **fer doux**. Ce dernier à la particularité de très bien conduire les champs magnétiques. Du coup, si un champ magnétique le traverse, il voudra absolument s'aligner dans son sens. C'est cette propriété qui est exploitée. Commençons par voir tout de suite le schéma de ce moteur :



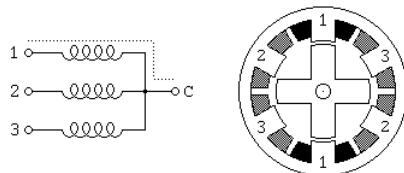
Comme vous pouvez le voir, il possède 4 enroulements (formant 4 paires) et le morceau de fer doux au milieu à une forme d'étoile à 6 branches. Et ce n'est pas un hasard ! Le ratio de 6 pour 8 a une raison très précise car cela introduit un léger décalage (15°) entre une branche et une bobine. En effet, si on a 8 bobines (4 paires) on a un décalage entre chaque bobine de : $\frac{360^\circ}{8} = 45^\circ$. Donc tous les 45° le long du cylindre qu'est le moteur, on trouve un bobinage. En revanche il n'y a que 60° entre chaque extrémité de l'étoile du rotor : $\frac{360^\circ}{6} = 60^\circ$. Mais pourquoi exactement ? Eh bien c'est

avec un peu d'imagination (et quelques dessins). Si l'on commence par alimenter le premier enroulement, le A, le rotor va s'aligner avec.

...enant que se passera t-il si l'on alimente B ? Le rotor, qui était alors positionné avec une de ses branches bien en face de A, va bouger pour s'aligner correctement vers B. Ensuite si l'on alimente C il va se passer de même, le rotor va encore tourner de 15° pour s'aligner. etc etc... Si l'on

effectue cette opération 24 fois, on fera un tour complet car : $24 \times 15^\circ = 360^\circ$

Vous remarquerez que dans cet exemple le rotor tourne dans le sens horaire alors que l'alimentation des bobines se fera dans le sens anti-horaire.



Ces moteurs ont certains avantages. Parmi ces derniers, il n'y a pas besoin de polariser les bobines (peu importe le sens du champ magnétique, l'entrefer n'étant pas polarisé essaiera de s'aligner sans chipoter). Le fait que l'on utilise un simple entrefer en fer doux le rend aussi moins cher qu'un modèle avec des aimants permanents. Vous savez maintenant tout sur les trois types de moteurs pas à pas que l'on peut trouver, place maintenant à la pratique !

De manière générale, n'essayez pas d'ouvrir vos moteur pas-à-pas pour regarder comment c'est fait et espérer les remonter après. Le simple démontage à tendance à faire diminuer la qualité des aimants permanents à l'intérieur et donc votre moteur ne sera plus aussi bon après remontage.

Se servir du moteur

Continuons à parler de notre super moteur. Si vous avez suivi ce que j'ai dit plus tôt, j'ai expliqué qu'il y avait des bobines qui généraient un champ magnétique. Lorsqu'elles sont alimentées, ces bobines ont besoin de courant pour pouvoir générer un champ magnétique suffisant. Vous ne serez donc pas surpris si je vous dis qu'il faudra utiliser un composant pour faire passer la puissance dans ces dernières. Et là, comme les choses sont bien faites, nous allons retrouver le pont en H et le L298. 😊

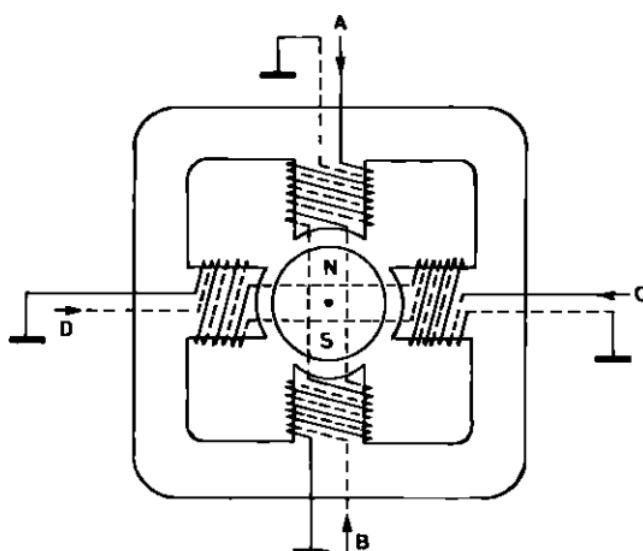
Afin de limiter la redondance d'informations, je vais me contenter de vous expliquer le pilotage du moteur unipolaire et bipolaire. Si vous comprenez correctement ces derniers vous n'aurez aucun problème avec le moteur restant 😊

Les schémas qui vont suivre ont pour source d'énergie une pile +9V. Cependant il est déconseillé de les faire avec car la consommation des moteurs est assez importante et la pile risque de se fatiguer très vite. Utilisez plutôt une source d'alimentation prévue à cet effet (une alimentation de laboratoire).

Le moteur unipolaire

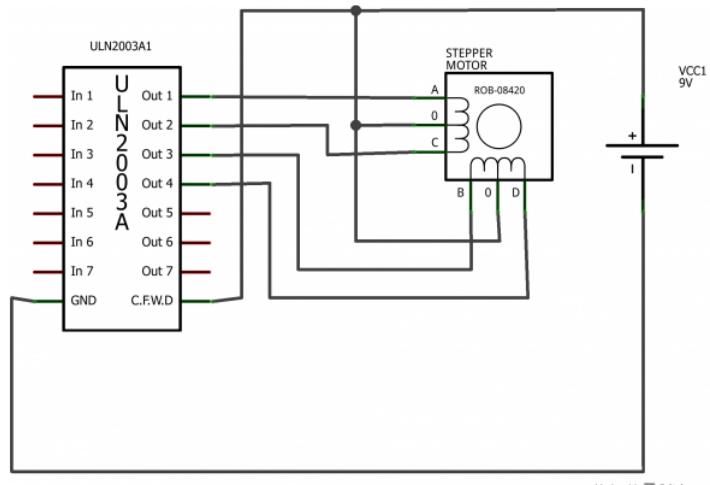
CONNECTER UN MOTEUR UNIPOLAIRE

Pour rappel, voici la structure du moteur unipolaire :

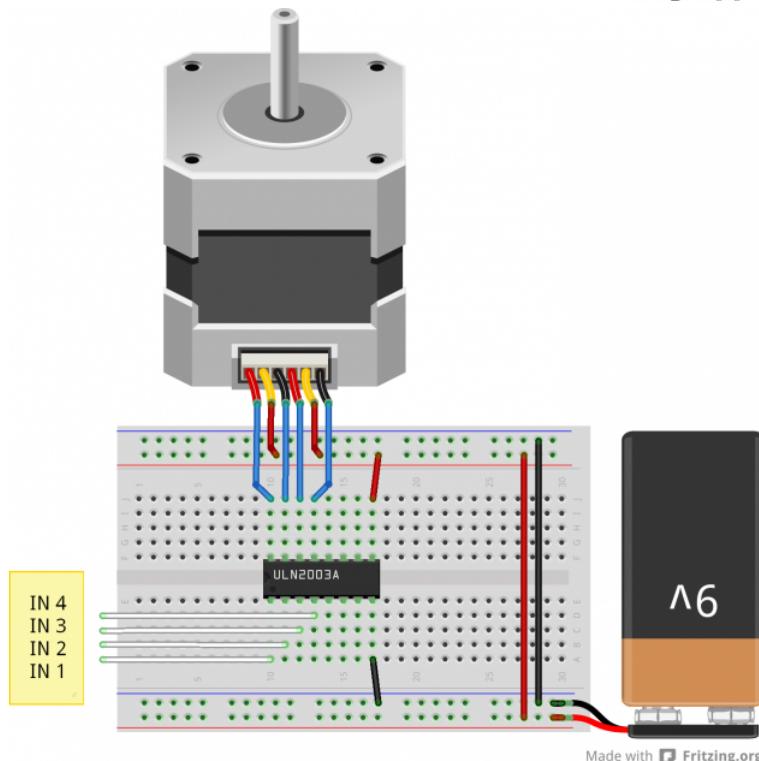


Si vous avez bien lu la partie précédente, vous devez avoir retenu que ce moteur est assez simple à piloter. En effet, il suffit d'alimenter une à une les phases pour que le moteur tourne. Et c'est tout ! Il nous faut juste utiliser le bon composant pour alimenter les bobines et en avant ! A priori, le bon composant serait un bon transistor qui pourrait supporter 50V et 500mA pour débuter. A cela il faudrait ensuite rajouter une diode de roue libre pour ne pas l'abîmer lors des phases de roue libre (tout cela multiplié par 4 puisqu'il y a 4 bobines). Plutôt que de s'embêter à câbler tout ça, je vous

propose l'intervention d'un nouveau composant : le ULN2003A. Ce dernier regroupe les transistors pour faire passer la puissance et aussi la diode. Il est très simple à câbler puisqu'il faut simplement amener l'alimentation et les broches de commandes. Chacune de ces dernières possède respectivement une sortie où la tension sera celle de l'alimentation. Voici une illustration de ce câblage :



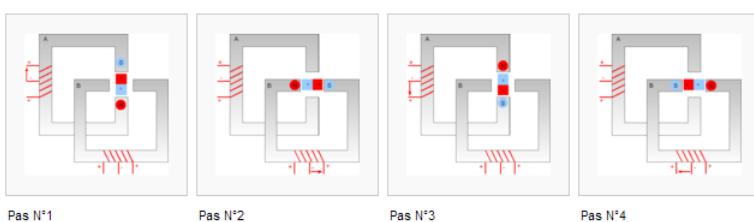
Made with Fritzing.org



Made with Fritzing.org

UTILISER UN MOTEUR UNIPOLAIRE

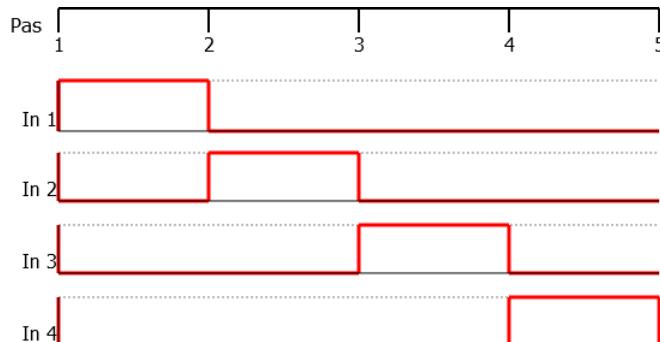
Je l'ai maintenant dit et répété plusieurs fois, pour ce moteur il suffit de piloter chaque bobine une à une, chacune leur tour. Je vais donc vous résumer tout cela de manière plus schématique et on sera bon pour ce moteur. 😊 A la fin, si vous avez bien compris vous devriez être capable de le faire tourner tout doucement en plaçant alternativement les fils In1 à 4 (abrévégé In1..4) au 5V ou à la masse.



Pas N°1 Pas N°2 Pas N°3 Pas N°4

↑	In 1	In 2	In 3	In 4
pas n°1	HIGH	LOW	LOW	LOW

Pas n°2	LOW	HIGH	LOW	LOW
Pas n°3	LOW	LOW	HIGH	LOW
Pas n°4	LOW	LOW	LOW	HIGH

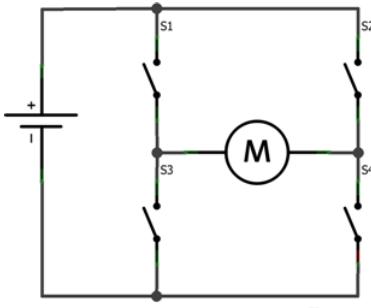


Si vous placez les fils de commande à la masse ou au 5V convenablement, votre moteur devrait tourner 😊 Vous n'avez plus qu'à créer le programme qui va bien pour piloter tout ce bazar... ben vous vous attendiez à quoi ? Une solution peut-être ? Non. 🤪 Bon bon, d'accord... vous vous souvenez de vos premiers pas avec le chenillard ? Tout est dit. 😊 (et si vous ne voulez pas vous fatiguer, attendez la suite du tuto 😊)

Le moteur bipolaire

LE CÂBLER, LA THÉORIE

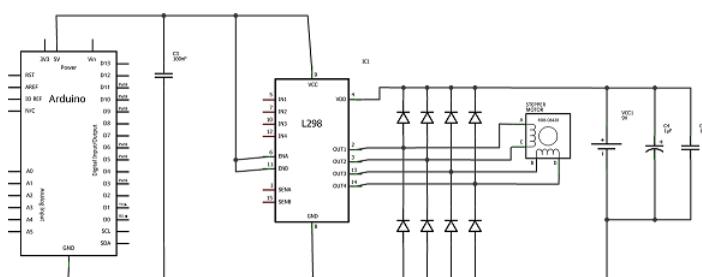
Nous avons vu dans le chapitre précédent que le moteur bipolaire a besoin, à un moment, d'une inversion de courant si l'on veut pouvoir utiliser les bobines à bon escient. Vous vous souvenez probablement que nous avons vu précédemment un composant capable de faire cela : le pont en H. L'idée va donc être de câbler correctement les bobines pour pouvoir faire passer le courant dans un sens, ou dans l'autre. Je vous rappelle la structure du pont en H :



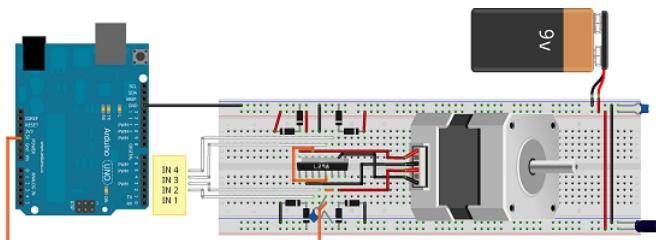
Problème, un pont en H ne peut piloter qu'une « paire de bobine » (celle qui sont dans le même axe et relié entre elles), hors nous en avons deux à piloter. Heureusement le L298 est un composant très bien pensé et ne possède non pas un mais bien deux ponts en H ! Du coup, nous pourrons en utiliser un par couple de bobine 😊 . Plutôt sympa non ?

LA PRATIQUE AVEC LE L298

Dans le chapitre précédent, nous avons justement vu un composant qui possède deux ponts en H : le L298. Le reste du travail semble presque trop simple ! Pour la partie « sortie/puissance », vous devrez relier les sorties Out1 (broche 2) et Out2 (broche 3) à la première bobine et ensuite Out3 et Out4 (13 et 14) à la seconde. N'oubliez pas les diodes de roue libre (8 au total et qui supporte la puissance) ! Enfin, il ne reste qu'à connecter les entrées In1..4 à 4 entrée/sortie numérique (pas besoin de PWM). Pour l'instant, pas besoin de les relier à l'Arduino, contentez-vous de les mettre à la masse, nous allons faire un test ensemble. Comme nous voulons activer les deux ponts, mettez les deux entrés « enable » au +5V. Nous verrons dans la prochaine partie comment l'utiliser avec Arduino 😊 . Voici un petit schéma récapitulatif :

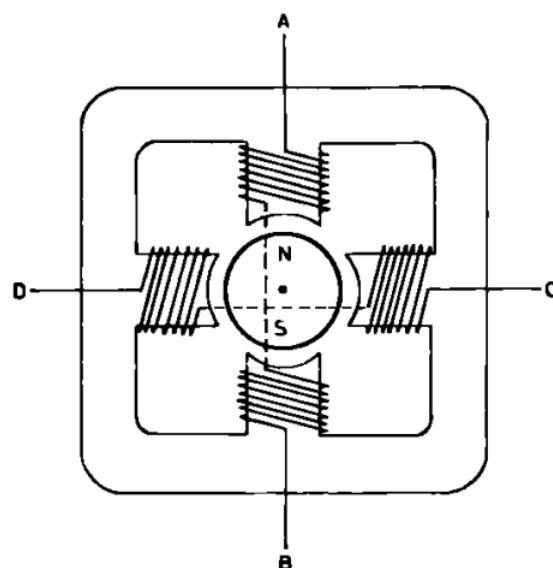


Le schéma de montage avec quelques condensateurs de filtrage qui viennent aider l'alimentation en cas de forte demande de courant et des condensateurs qui filtres les parasites engendrés par les bobines du moteur.



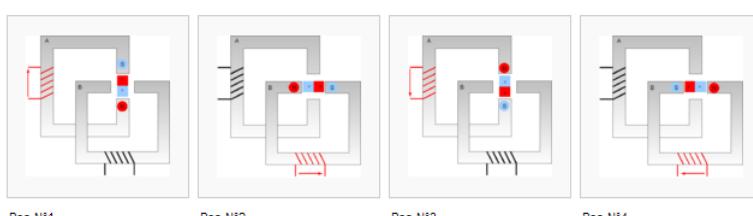
Piloter le moteur bipolaire

Une fois que le moteur est branché, nous allons pouvoir le faire tourner. Comme son nom l'indique, il s'appelle pas à pas et donc nous allons le faire pivoter étape par étape et non de manière continue comme le moteur à courant continu. Cependant, en répétant les étapes de rotation rapidement et successivement, le moteur donnera l'impression de tourner sans s'arrêter entre chaque étape. Il existe trois méthodes distinctes pour piloter les moteurs bipolaires. Nous allons maintenant les voir une par une. Dans les cas qui vont suivre, on va considérer un moteur de 4 pas par tour (ce qui est ridiculement faible). Voici ce à quoi il va ressembler (comment sont placés ses bobines) :



ROTATION PAR PAS COMPLET

Ce mode de fonctionnement est le plus simple, c'est pourquoi nous allons commencer par lui. Grâce à ce dernier, vous allez pouvoir faire des rotations « pas par pas ». Pour cela, nous allons alternativement alimenter les bobines de droites et de gauche et inverser le sens du courant pour pouvoir faire une rotation complète du champ magnétique. Voici l'illustration Wikipédia très bien faite à ce sujet :



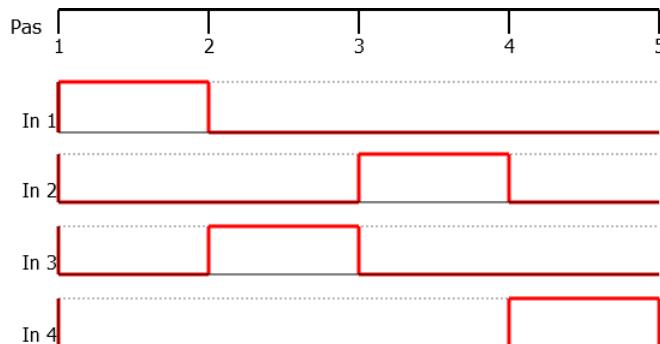
En rouge la bobine alimentée ainsi que le sens du courant symbolisé par une flèche et en noir la bobine qui n'est pas alimentée.

En considérant que la bobine A est connectée aux entrées IN1 et IN2 et que la bobine B est connectée aux commandes IN3 et IN4, on peut donc écrire la séquence de commande suivante :

Etape	IN 1	IN 2	IN 3	IN 4
Pas n°1	HIGH	LOW	—	—
Pas n°2	—	—	HIGH	LOW
Pas n°3	LOW	HIGH	—	—
↑ n°4	—	—	LOW	HIGH

(un état ‘-’ signifie « non nécessaire », placez-le à 0V pour que la bobine soit bien inactive).

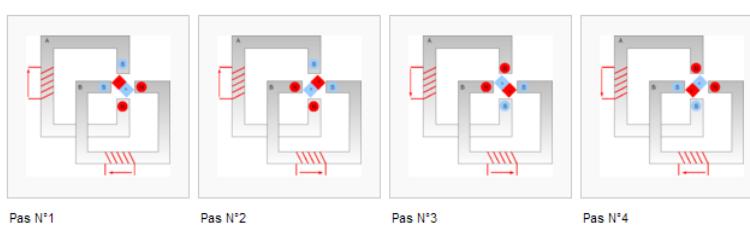
On peut traduire cette activité par le chronogramme suivant :



Comme vous pouvez le voir à travers ces différents moyen d'explication, c'est somme toute assez simple. On va chercher à déplacer l'aimant central en le faisant tourner petit à petit. Pour cela on cherchera à l'attirer dans différentes positions.

ROTATION À COUPLE MAXIMAL

Un autre mode de fonctionnement est celui dit à **couple maximal**. Cette méthode de pilotage utilise toutes les bobines à la fois pour pouvoir immobiliser au maximum l'aimant central. En effet, en utilisant plus de champs magnétique on obtient une force supplémentaire. Par contre on consomme évidemment d'avantage de courant. Pour comprendre ce fonctionnement, voyons les différentes étapes par un dessin puis par un chronogramme. Vous verrez, ce n'est pas très compliqué, le fonctionnement est très similaire, seule les activations de bobines changent un peu :

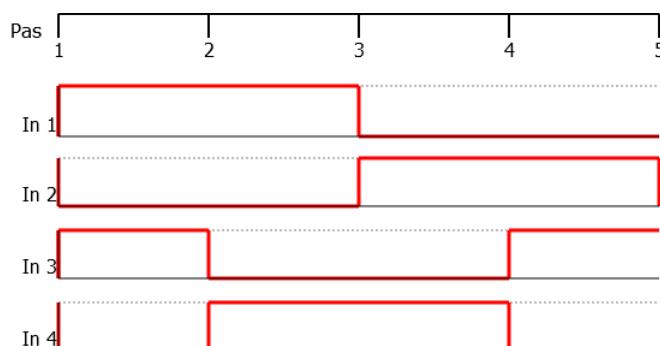


Avez-vous remarqué quelque chose de particulier ? Dans cette utilisation, l'aimant ne fait plus face aux bobines mais se place *entre* les deux. Par contre, il effectue toujours des pas entiers, ces derniers ont juste un décalage constant par rapport à avant.

Voici le tableau correspondant au pilotage des bobines :

Etape	In 1	In 2	In 3	In 4
Pas n°1	HIGH	LOW	HIGH	LOW
Pas n°2	HIGH	LOW	LOW	HIGH
Pas n°3	LOW	HIGH	LOW	HIGH
Pas n°4	LOW	HIGH	HIGH	LOW

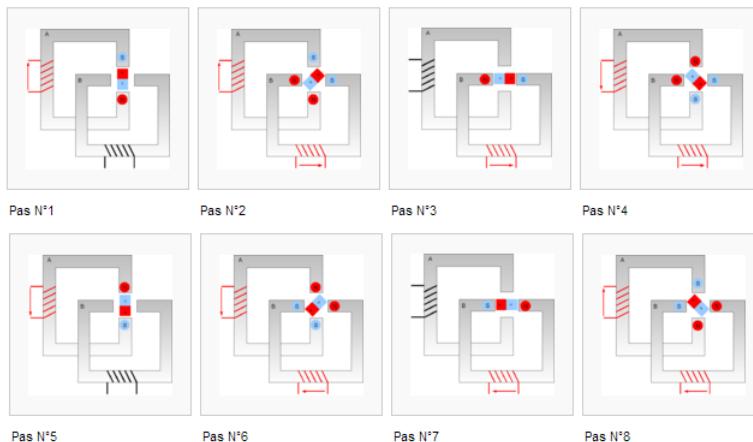
(un état ‘.’ signifie « non nécessaire », placez le à OV pour que la bobine soit bien inactive).



ROTATION PAR DEMI-PAS

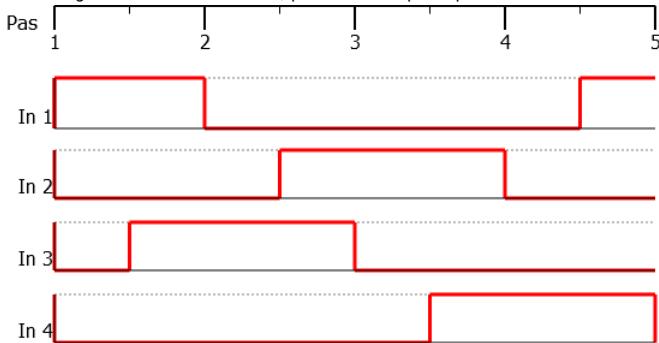
Enfin, le dernier mode de fonctionnement est celui dit à **demi-pas**. Ce mode mélange les deux précédents puisqu'on va alterner les étapes du mode à pas complet et les étapes du mode à couple maximal. En effet, comme nous avons pu le voir dans les explications précédentes, les deux modes tournent l'aimant central de manière différente. L'un est « en face des bobines » alors qu'avec l'autre est plutôt « entre les bobines ». Ainsi, en se mélangant alternativement « en face » puis « entre » les bobines on va effectuer deux fois plus de pas que précédemment puisqu'on intercalera des étapes supplémentaires. Attention, lorsque je dis « deux fois plus de pas » je veux surtout dire que l'on aura des étapes intermédiaires qui

augmentent la précision du déplacement. Ce mode de pilotage est un peu plus compliqué que les précédents puisqu'il est « plus long » (8 étapes au lieu de 4) mais rien d'insurmontable vous allez voir !



Etape	In 1	In 2	In 3	In 4
Pas n°1	HIGH	LOW	—	—
Pas n°1 ½	HIGH	LOW	HIGH	LOW
Pas n°2	—	—	HIGH	LOW
Pas n°2 ½	LOW	HIGH	HIGH	LOW
Pas n°3	LOW	HIGH	—	—
Pas n°3 ½	LOW	HIGH	LOW	HIGH
Pas n°4	—	—	LOW	HIGH
Pas n°4 ½	HIGH	LOW	LOW	HIGH

(un état ‘-’ signifie « non nécessaire », placez le à OV pour que la bobine soit bien inactive).



Maintenant que vous connaissez les différents modes de fonctionnement, vous pouvez essayer de faire tourner le moteur en branchant les entrées IN1..4 à la masse ou au 5V. Si vous le faites dans le bon ordre, votre moteur devrait tourner tout doucement, en allant d'une étape vers l'autre 😊.

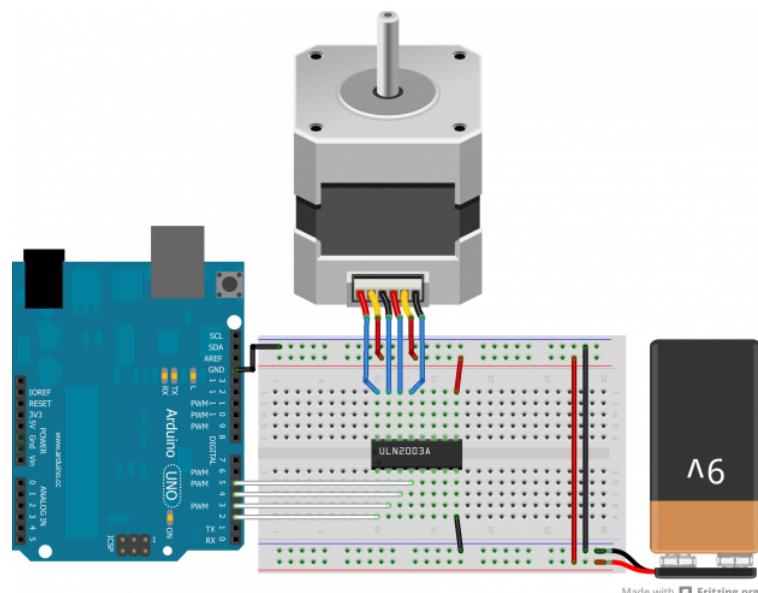
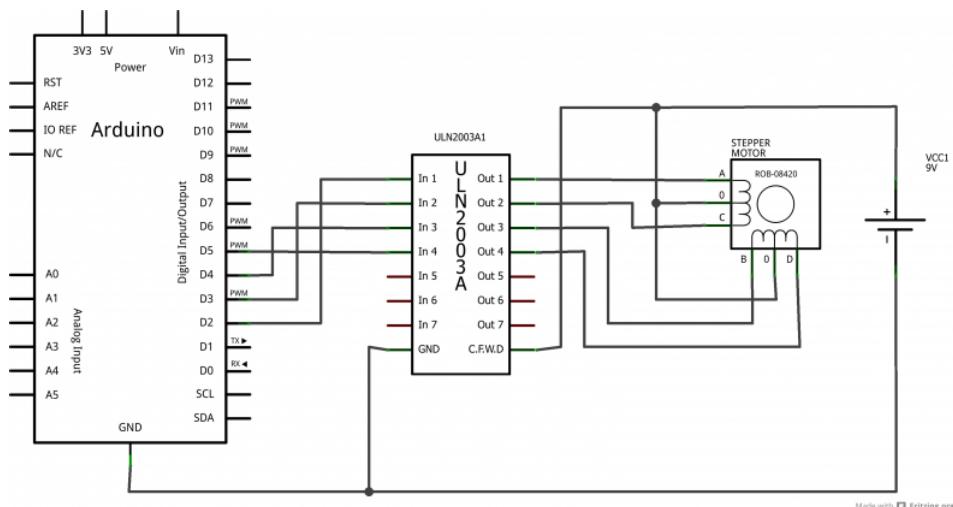
Si vous avez une charge qui demande trop de couple (par exemple un poids à faire monter), il peut arriver que le moteur « saute » un/des pas. Cette donnée est à prendre en compte si vous servez du nombre de pas effectué logiciellement comme moyen de calcul de distance.

Utilisation avec Arduino

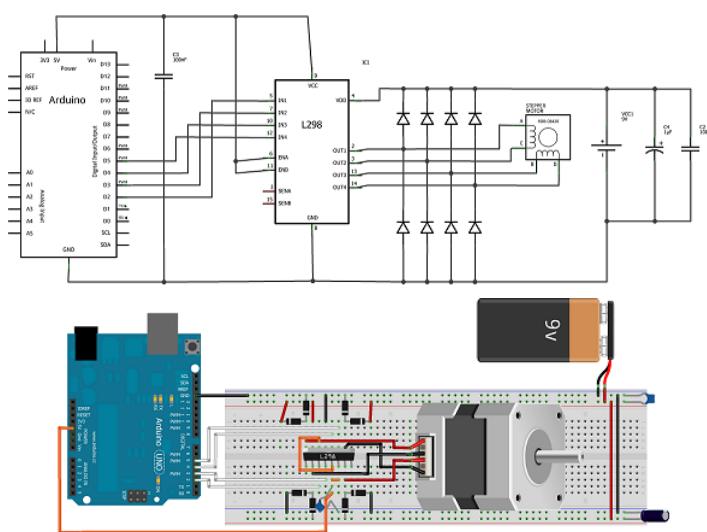
Câbler les moteurs

Avec les chapitres précédents vous avez vu comment on devait utiliser les moteurs avec les composants gérant la puissance. Cependant, nous n'avons pas vu à quoi relier les broches de commande... Et c'est très simple ! En effet, tous les signaux sont tout ou rien et n'ont même pas besoin d'être des PWM ! Ce qui veut dire que les 4 broches de pilotage ont juste besoin d'être reliées à 4 broches numériques de la carte Arduino (2, 3, 4, 5 r ↗ simple). Voyons ce que cela donne en schéma (qui sont exactement les mêmes que ceux de la partie précédente, mais avec une carte Arduino ↗ , ☺)

LE MOTEUR UNIPOLAIRE



LE MOTEUR BIPOLAIRE



Jusque là rien de vraiment compliqué, on passe à la suite !

Piloter les moteurs avec Arduino

LE PRINCIPE

Le principe est toute simple, il suffit de générer la bonne séquence pour piloter les moteurs à la bonne vitesse, vous vous en doutez sûrement. La principale difficulté réside dans la génération des signaux dans le bon ordre afin que le moteur se déplace correctement. Bien entendu, c'est plus

facile à dire qu'à faire. En effet, pour que le mouvement soit fluide, il faut que les changements dans la séquence soient fait de manière régulière et pour cela il faut une gestion du temps correcte. Ça peut sembler simple au premier abord, mais quand il s'agira de mixer le comportement du moteur avec celui du programme principal (qui devra peut-être faire des traitements assez lourd) cela deviendra probablement beaucoup moins trivial. Une bonne méthode consisterait à utiliser un **timer** sur lequel on réglerait la période à avoir qui refléterait ainsi la vitesse à obtenir. Mais avec Arduino vous allez voir que tout devient plus simple...

L'OBJET STEPPER

Sur Arduino les choses sont bien faites pour rester simples et accessibles, un objet a déjà été créé pour vous aider à piloter un moteur pas à pas. Attention cependant, il ne fonctionne que pour les moteurs unipolaire et bipolaire. Il tire partie du fait que ces deux types de moteur peuvent fonctionner avec une séquence commune. Ainsi, tout est généralisé et utilisable le plus simplement possible ! Ce nouveau composant s'appelle « Stepper ». A sa création il prend en argument le nombre de pas total que fait le moteur pour faire un tour (information trouvable dans la documentation constructeur ou empiriquement). Cette information sert à déterminer la vitesse de rotation par minute que vous pourrez ensuite régler à loisir pour faire des déplacements lents ou rapides. Il prend aussi en arguments les quatre broches servant à contrôler l'engin. Son constructeur est donc : `Stepper(steps, pin1, pin2, pin3, pin4)`. Pour initialiser le moteur, nous pouvons donc écrire la ligne suivante :

```
1 //pour un moteur de 200 pas par tour et brancher sur les broches 2, 3, 4, 5
2 Stepper moteur(200, 2, 3, 4, 5);
```

Pour l'utiliser, deux fonctions sont utilisables. La première sert à définir la vitesse de rotation, exprimée en tours par minute (**trs/min**). Pour cela, on utilise la fonction **step(steps)** qui prend en paramètre le nombre de pas à effectuer. Si ce nombre est négatif, le moteur tournera en sens inverse du nombre de pas spécifié. Voici un petit exemple qui va faire faire un aller-retour de 200 pas toute les 2 secondes à votre moteur :

```
1 #include <Stepper.h>
2
3 //pour un moteur de 200 pas par tour et brancher sur les broches 2, 3, 4, 5
4 Stepper moteur(200, 2, 3, 4, 5);
5
6 void setup()
7 {
8     moteur.setSpeed(30); //30 tours par minute
9     //rappel : ici le moteur fait 200 pas par tour, on fera donc 6000 pas par minute)
10 }
11
12 void loop()
13 {
14     moteur.step(1000);
15     delay(1000);
16     moteur.step(-1000);
17     delay(2000);
18 }
```

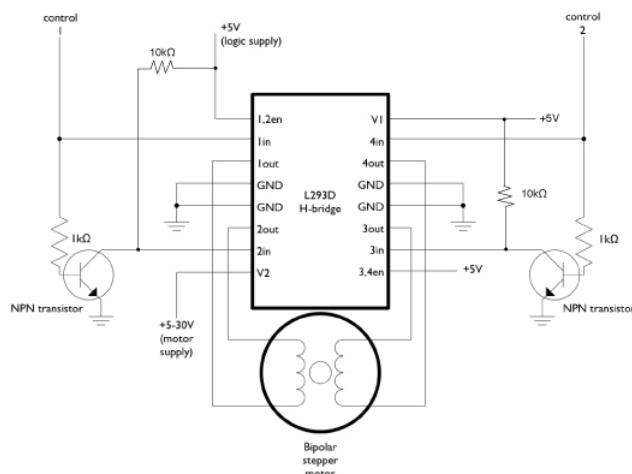
La fonction `step(x)` est bloquante. Cela signifie qu'elle agit comme un délai. Tant que le moteur n'a pas fait les `x` pas demandés, le reste du programme est en attente.

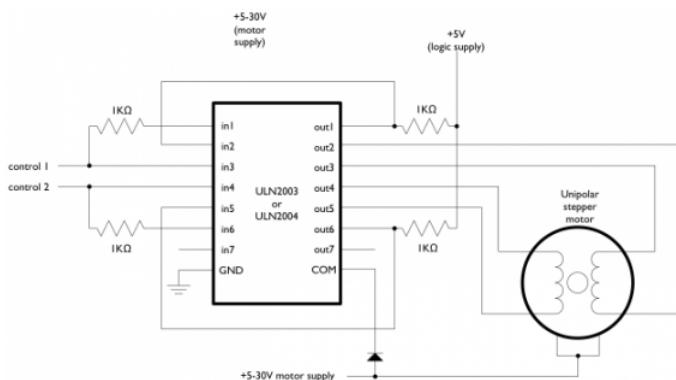
Aller plus loin

Vous êtes devenus incollables sur les moteurs pas à pas ? Vous en voulez encore plus ? Suffit de demander, voilà du bonus d'informations rien que pour toi chère lecteur !

2 FILS AU LIEU DE 4 !

On a toujours utilisé 4 fils pour commander les moteurs. C'est bien, mais que diriez-vous de sauver deux broches et de passer à seulement 2 fils au lieu de 4 ? Pas mal comme amélioration non ? Petite anecdote : un jour, un utilisateur des moteurs pàp s'est rendu compte d'un truc, dans une paire de fils pour piloter un moteur (dans la séquence utilisée par Arduino), l'information est toujours antagoniste. Si un fil est à HIGH, sa paire sera à LOW et vice versa. Du coup il suffit d'un peu d'électronique pour pouvoir inverser un des deux signaux et se retrouver ainsi avec seulement deux fils au lieu de 4 sortant d'Arduino. 😊





(source des images : tigoe.net) Cette solution est plutôt intéressante du fait que les entrées/sorties sont parfois une denrée rare sur Arduino ! 😊

LE L297

Lorsque vous utilisez votre Arduino, vous ne pouvez utiliser qu'une seule séquence. Par exemple pour un moteur bipolaire vous n'avez pas le choix entre le mode pas entier, demi-pas ou couple max. Une des solutions serait de générer vous-même les séquences. Mais c'est assez fastidieux. Une autre solution est électronique et compensera le développement informatique à faire. Un composant, nommé L297 (de la famille du L298 vous vous en doutez) est justement fait pour générer les séquences de moteur pas à pas. Il possède 4 broches de sorties pour générer la séquence et plusieurs en entrée pour « paramétrier » le fonctionnement voulu. Parmi elles on en retrouve trois principales :

- CW/CCW : (ClockWise ou Counter ClockWise) qui décidera du sens de rotation du moteur (horaire ou anti-horaire).
- Half/Full : Qui décide si on est en mode pas entier ou demi-pas.
- CLK : (Clock) qui est l'horloge pour la vitesse. A chaque front descendant le moteur fera un pas.

Je vous laisse un peu chercher sur le net, vous trouverez de plus amples informations à ce sujet. Avant même de regarder sur le net, en fait, regardez plutôt sa datasheet !! 😊 Un des avantages de délester le travail des séquences au L297 est que vous n'aurez plus besoin de l'objet Stepper et de sa fonction step() bloquante. Il faudra cependant toujours utiliser un composant de puissance pour laisser passer les forts courants nécessaires au moteur (comme le L298 par exemple).

Partager :



WordPress :



Soyez le premier à aimer cet article.

Archivé sous Partie 6 - Les moteurs | Taggé Arduino, Moteur, Tutoriel | 271 commentaires | Permalink

271 commentaires

 **12Bastien**
24/02/2014 at 11:10

Bonjour;

je viens de lire ton article et cela m'a aidé pour comprendre le fonctionnement d'un moteur pas à pas mais je suis confronté à un problème je dispose d'une carte arduino uno et d'une Platine « motor shield Rev3 » pour Arduino je n'arrive pas à comprendre le fonctionnement de la platine pour faire les branchements pour un moteur à pas 4 fils

Si tu peux m'aider merci beaucoup 😊



**Eskimon**

24/02/2014 at 11:19

Salut !

Ton moteur est bipolaire ou unipolaire ?

Il faudra connecter ses fils sur les plots B et A, de façon à connecter un enroulement sur A et l'autre sur B (test à l'ohmètre quelle paire de fils convient pour chaque enroulement).

Voici une page (en anglais) qui devrait pouvoir t'aider 😊 <http://www.instructables.com/id/Arduino-Motor-Shield-Tutorial/step6/Stepper-Motor/>**Mauvage**

28/09/2015 at 21:33

Bonjour,

Très bon article.

Mais je suis devant un problème.

Config Funduino et kit moteur PàP 28BYJ + U2003

j'arrive à le faire tourner mais dans un seul sens.

Que ce soit un positif ou un négatif à la place de valeur dans

```
moteur.step( valeur);
```

Pourquoi?

**Emile Caubet**

25/10/2015 at 13:59

Bonjour,

Je suis confronté au même problème que toi, as-tu trouvé une solution ?

Merci d'avance.

**Mauvage**

25/10/2015 at 14:16

Salut,

Voilà le programme que j'ai utilisé et cela fonctionne.

C'est une question de paramètres 2048 * nbre de tours

```
#include
```

```
const int stepsPerRevolution = 2048; //2048 pour la 28BYJ-48 = 1 tour
int Nbre_tour ;
// initialiser la bibliothèque stepper sur les broches D8 à D11:
Stepper myStepper(stepsPerRevolution, 8,10,9,11); //Attribution des broches
```

```
void setup() {
// réglage de la vitesse (28BYJ-48):
myStepper.setSpeed(4); // de 4 à 14 vitesse de rotation 4=lent
// initialize the serial port:
Serial.begin(9600);
}
```

```
void loop() {
```

```
Nbre_tour = 1; //Faire 10 tours dans un sens:
Serial.println(" clockwise ");
myStepper.step(stepsPerRevolution*Nbre_tour);
delay(1000);
```

▲ Nbre_tour = 1; //Faire 5 tours dans l'autre sens:
Serial.println(" counterclockwise ");
myStepper.step(-stepsPerRevolution*Nbre_tour);
delay(1000);

{

En esperant que cela t'aidera.

**Dyole**

25/10/2015 at 16:03

Bonjour à tous

J'étais confronté au même problème et le 8/7/2015 PHaroz avait proposé cette solution qui pour moi avait fonctionnée :

« Pour ceux qui aurait un pb à faire tourner dans les 2 sens un 28BYJ sachez que certain modèle on un brochage erroné, c'était le cas sur le mien.

Au lieu de déclarer
Stepper moteur(64, 8, 9, 10, 11);
j'ai du faire
Stepper moteur(64, 8, 10, 6, 11);

Si cela peut vous aider !!

Didier (Dijon)

**Mauvage**

25/10/2015 at 18:29

Merci Dyole pour la remarque

C'est Exact pour moi, (2048,8,10,9,11)

avec in1 sur D8
in2 sur D9
in3 sur D10
in4 sur D11

Cdlmnt

**Bohdanowicz**

24/10/2016 at 13:39

Bonjour,

J'ai résolu le problème en inversant les fils OUT2 et OUT3 sur la prise du moteur (facile avec un petit tournevis). Comme ça les entrées IN1, IN2, IN3 et IN4 sont connectées à Arduino en 2, 3, 4, et 5.

Jurek B.

**Monak**

12/11/2016 at 16:24

Merci pour l'info.

Le moteur fonctionne très bien maintenant

**Botticelli**

07/08/2016 at 11:48

Bonjour,

Je cherche à bloquer un moteur pas à pas en rotation par rapport à un autre comme si il étais équiper d'un frein que l'on peux contrôler par l'intermédiaire d'une information inscrite dans un programme relier à une carte arduino et vis versa avec l'autre moteur pas à pas et je ne trouve pas de réponse... je voudrais juste savoir si c'est possible à exécuter et si oui quel type de moteur pas à pas faut-il utiliser ? bipolaire ? unipolaire ? ou autres.....

 **12Bastien**
24/02/2014 at 11:34

Je ne sais pas du tout le moteur dont je dispose m'a aucune référence ... ;/
merci pour le site 😊

 **Eskimon**
28/02/2014 at 09:56

Alors il va falloir prendre un multimètre et mesurer la résistance entre les fils pour déterminer qu'elles sont les paires qui vont entre elles.
Si il y a une résistance c'est qu'il y a une bobine (donc forme une paire) sinon c'est qu'il n'y a rien.

 **Maxou**
18/03/2014 at 13:43

salut !
Je suis lycéen et pas très fort en programmation ;
J'ai testé ton programme mais mon moteur (bipolaire) fais du « sur place » ... il bouge mais au bout de 4 ticks il se remet en position initiale
pourrait tu m'aider à comprendre pourquoi est ce il ne marche pas stp ? 😊

 **Eskimon**
18/03/2014 at 17:18

Sans plus d'infos ça va être difficile... 4 ticks ça représente beaucoup sur une rotation ? Tu es sûr que les fils sont bien branchés ? Tu utilises un shield ou quelque chose comme ça ?

 **thomas**
18/03/2014 at 14:16

Bonjour,

J'ai pour projet de piloter deux moteurs pas à pas pour déplacer un capteur laser pour effectuer des mesures. J'aurais aimé savoir s'il était possible d'alimenter deux contrôleurs L293D pour piloter mes moteurs à partir d'un adaptateur branché sur une prise standard et non des piles comme on peut le voir dans de nombreuses vidéos ? Les moteurs que je pense utiliser et adaptateur auquel je pense :

<http://www.gotronic.fr/art-moteur-14hs17-0504s-18353.htm>
<http://www.gotronic.fr/art-adaptateur-ps1220-7996.htm>

Le pilotage se fera avec une arduino uno que j'ai acheté récemment pour réaliser l'ensemble des tutos de ce blog qui est très bien fait au passage 😊
!

Merci

 **Eskimon**
18/03/2014 at 17:26

A priori pas de problèmes. Ton alimentation se branche sur l'Arduino et tu pourras récupérer sa tension sur la borne Vin de l'Arduino pour l'envoyer sur ton L293.
Donc « électroniquement » ça passe. Par contre faudra ruser au niveau du code car la fonction `Step()` de l'objet Stepper est bloquante. Du coup quand tu en fais bouger un l'autre est immobile. Mais des gens ont déjà eu ce problème et donc la solution peut se trouver sur google sans trop de souci a priori (cherche « Arduino 2 Stepper » si tu veux 😊 j'ai regardé vite fait ça à l'air de se faire 😊).

 **thomas**
18/03/2014 at 19:40

Merci pour ta réponse rapide. Ce n'est pas un soucis pour la fonction bloquante car en fait les moteurs ne tourneront pas simultanément mais je vais quand même regardé au niveau du code au cas où. Sinon si je branche à partir de l'alimentation, je n'ai plus besoin de l'usb pour le 5V ?
(je suis un novice niveau électronique et électricité) et concernant l'intensité pas de risque pour la carte arduino ? les 1A ne seront qu'en sorti .u L29X ? Est que prendre des L298N au lieu de L293D ou E ça vaut le coup ou pas ? sachant que les moteurs choisis nécessitent uniquement 0.5A qui peuvent être fournis par les L293 et qu'à priori je ne vais pas devoir changer mes moteurs ?

**Eskimon**

18/03/2014 at 20:45

Ah ah si tu es patient jusqu'à Samedi mon prochain article est justement à propos de l'alimentation de l'Arduino 😊 MAJ : C'est en ligne !
Le seul endroit où vont passer les 1A seront entre le connecteur rond ou tu branches ton alimentation et la broche VIN (puis ensuite le L293 évidemment). Je t'avouerais ne jamais avoir testé mais ça doit passer (d'après une petite recherche rapide, il y a une diode qui va bloquer à 1A).
Pour le choix des composants, si ça passe avec un L293D alors reste avec ça, l'avantage étant qu'il ne faut pas s'embêter avec les diodes de roue libre vu qu'elles sont intégrées.

**thomas**

22/03/2014 at 21:39

Ok merci, je vais aller jeter un coup d'œil ! J'ai fini mon montage et tout fonctionne, par contre mes moteurs chauffent beaucoup (je leur envoie du 12V alors qu'ils veulent du 7.5V) ! Il y a t'il un moyen (petit montage) pour être sûr d'amener la bonne tension aux bornes de mes moteurs ?

Merci

**Eskimon**

22/03/2014 at 22:27

Ouai, utiliser un régulateur (à découpage c'est le must, ou linéaire comme sur l'Arduino mais il risque de chauffer pas mal aussi) pour diminuer la tension 😊

**djiwo555**

21/05/2015 at 09:23

je recherche l'article dont tu fais référence ici.

**Maxou**

24/03/2014 at 11:13

ça représente un pas en avant et un pas en arrière j'utilise un shield Rev3 arduino et j'ai branché les fils sur les Channels A et B mais je pense que cela viens du programme enfin des pins à lettre dans la fonction « Stepper » j'ai mis ça :Stepper mon_moteur(nbr_pas, 3, 12, 13, 11)

**Eskimon**

24/03/2014 at 16:23

Il va probablement falloir trouver le bon ordre à mettre pour les broches dans ce cas... Avec un multimètre tu pourrais trouver quelles sont les paires de fils pour les bobines et ensuite les mettre correctement dans le Stepper.

**clement**

09/04/2014 at 08:37

Bonjour thomas, je vais devoir contrôler de deux à 3 stepper moteurs pour mon projet, j'aimerais pouvoir te contacter. Bien cordialement.

**Eskimon**

09/04/2014 at 15:29

Salut Clement !
(c'est pas moi Thomas !!)

Le plus simple est probablement de faire un sujet sur le site OpenClassrooms section électronique, j'y passe assez souvent et je n'ai pas de section forum ici...

Après je ne fais pas d'aide particulière, je n'en ai pas le temps ou alors il faudrait que j'ouvre un bureau d'étude rien que pour ça 😊
Sinon pour quand même te répondre, il te faudra sûrement un shield par moteur, en effet, un shield ne peut piloter qu'un seul moteur pas à pas.

**Polooooow**

12/05/2014 at 15:19

Bonjour,

J'ai comme projet de bloquer le démarrage d'un moteur avec une carte arduino a partir de ma tablette.

En fait mon projet est un ewee (un gyropode français de type segway), je dois faire un tableau de bord avec une tablette, ça c'est fait, il ne me reste qu'à bloquer le démarrage des deux petits moteur des roues tant qu'un mot de passe n'est pas activé. Je vais faire ça avec un module bluetooth, mais le problème c'est que je ne sais pas comment contrôler le moteur avec la carte arduino.

**Eskimon**

12/05/2014 at 18:44

Mais si ton arduino doit faire juste une sorte de « verrou » pour les moteurs, pourquoi pas simplement mettre un relais sur l'alimentation ?

**Poloooow**

13/05/2014 at 13:57

Comment ça ?

**Poloooow**

13/05/2014 at 14:00

J'avais pensé à mettre la carte arduino entre le moteur et le bouton d'alimentation, je ne sais pas si c'est possible et si oui comment faire.

**Eskimon**

13/05/2014 at 17:23

Là je suis désolé mais on va sortir du cadre d'une zone de commentaire... Je te conseille d'aller demander de l'aide sur un forum (arduino.cc, openclassrooms rubrique électronique) pour qu'il y ait plus de place/d'outils pour te répondre correctement 😊

Bon courage !

**JoJodu57**

15/05/2014 at 16:23

Bonjour, au sujet des puissances pour chaque élément du circuit, il faut une alimentation qui peut donner plus de courant que ce que demande le moteur, mais qui doit être inférieur au maximum supporté par le transistor ?

**Eskimon**

15/05/2014 at 16:30

Normalement l'alimentation ne devrait jamais délivrer tout ce qu'elle a dans le ventre. Du coup les transistors sont censé être bien dimensionné pour les moteurs et leur application

**JoJodu57**

15/05/2014 at 17:38

Merci, aussi, sachant qu'un moteur qui démarre, demande le double de courant, faut-il se contenter d'une alimentation avec des gros condensateurs pour un moteur stepper bipolaire de 3A ?, vous pouvez donner un exemple pour un ensemble bien équilibré ?

**Eskimon**

15/05/2014 at 18:17

Pour le démarrage d'un moteur il faut être malin...

Soit on surdimensionne l'alim, mais ça coutera cher.

Soit on met des gros condensateurs mais ça résout pas forcément tout.

Soit on fait un soft-start (démarrage progressif) de manière logiciel (possible matériellement aussi), ce qui me semble être une bonne solution souple.

**Poloooow**

16/05/2014 at 15:07

J'ai déjà été sur le forum d'arduino, mais ils ne m'ont jamais aidé.

**Eskimon**

16/05/2014 at 21:47

Viens sur OpenClassrooms rubrique « électronique », on mord pas 😊

**BOB**

27/05/2014 at 09:50

Salut ! 😊

je suis encore et toujours la ^^.

sur ce, je suis en plein doute pour le composant ULN2003. Commençons par le commencement. Alors je travaille avec de moteur PAP en 12V. J'ai vue sur farnell que le ULN2003 s'alimente en 5V et a (au maximum) 50V en sortie. alors le doute que j'ai maintenant, comment ce règle la tension de ce composant ou sinon avec quelle entrée.

Alors merci de toujours être là pour aider et grand merci (même si c'est le 15 eme XD)

**Geschikt**

05/06/2014 at 14:39

Bonjour,

Je travaille avec un moteur pap, et es-ce que c'est possible de brancher directement l'arduino a ce moteur ?

**Eskimon**

05/06/2014 at 15:06

Salut !

Comme pour tout les moteurs, la réponse est non. Les broches de l'Arduino ne peuvent pas fournir le courant nécessaire au fonctionnement du moteur. D'ailleurs il est fréquent que les moteurs pas-à-pas fonctionnent à des tensions supérieures à 5V (par exemple 12V), et donc il faut trouver une source de tension externe à l'Arduino (qui ne possède que du 5V)

**jackie&michel**

26/05/2016 at 15:57

ah merde.

**david**

10/06/2014 at 14:49

bonjour,

si on veux que le moteur pas à pas démarre toujours au même endroit (angle) avant d'effectuer le début d'une ou plusieurs rotation comment faut il faire?

**Eskimon**

10/06/2014 at 14:57

Il faudra rajouter un moyen externe pour déterminer la position du moteur. Un encodeur absolu/potentiomètre sur l'axe de sortie pourrait sûrement faire l'affaire (pour détecter un angle).

Mais sans capteur, impossible de savoir à la mise sous tension où est le moteur.



 **Houphouet kouassi**
20/06/2014 at 10:57

Document très bien réalisé. On comprend bien les idées de l'auteur et aussi le fonctionnement de ces moteurs.
Grand merci à l'auteur.

 **alex**
15/07/2014 at 14:25

peut t-on établir un quota de tour ? c'est à dire demander au moteur (quand une condition est remplie (nombre de tour)) de s'arrêter puis par une deuxième condition lui faire faire le chemin inverse ? si oui comment ?
sinon merci pour ce cours !

 **Eskimon**
15/07/2014 at 15:21

En soi oui c'est possible, après il faut le coder ^^
Soit tu le fais « au feeling » : Tu connais le nombre de pas par tour et le nombre de pas que tu envoies donc tu peux déterminer le nombre de tour que tu fais
Soit tu le fais avec un capteur (encodeur quadrature probablement) qui te donne des impulsions quand ton moteur tourne.

Pingback: ARDUINO | Pearltrees

 **bw_syl**
04/09/2014 at 15:38

Bonjour,

J'essaie de faire un montage avec un moteur pas à pas et le L293D, tes explications sont vraiment très bien faites 😊
Et j'aimerais savoir sur le schéma « Le moteur bipolaire » quelles sont les valeurs des condensateurs que tu utilises ?

Merci d'avance.

 **Eskimon**
04/09/2014 at 15:48

Ah oui c'est vrai que c'est pas très lisible tout ça... Le schéma Fritzing est disponible ici si tu veux : <https://github.com/Eskimon/Tuto-Arduino-SDZ/blob/master/Partie%206%20-%20Le%20mouvement%20gr%C3%A2ce%20aux%20moteurs/Chapitre%203%20-%20Moteur%20pas%20%C3%A0%20pas/cablage%20bipolaire.fzz>

Sinon de gauche à droite on a : 100nF, 1uF, 100nF

 **Eduardo castellani**
19/07/2016 at 01:36

Bonjour, je vous remercie pour le tutoriel, demandez qui est le programme qui ouvre ces fichiers? l'Fritzem J'ai, mais parler d'autres fichiers

 **bw_syl**
04/09/2014 at 16:05

Nicel, merci 😊

Au passage, si je peux me permettre as-tu un endroit où je peux trouver comment on peut choisir ces valeurs, ainsi que la tension (j'ai cherché ce composant sur différents sites, et on en trouve 1uF 50V ou 100V etc).
Dans la mesure où je vais utiliser une alimentation 12V, est-ce que n'importe quelle valeur au-dessus de 12V est OK ?

J'aimerais me rendre sur le forum ZdS pour présenter mon petit projet et poser mes nombreuses autres questions 😊

Merci encore

**Eskimon**

04/09/2014 at 22:04

Dans la mesure ou je vais utiliser une alim 12v, est-ce que n'importe quelle valeur au dessus de 12v est ok ?

Absolument ! Pour le choix d'un condensateur, la tension marquée dessus est la tension minimale. Si ton montage est en 5V, il est prudent de prendre un condo avec une valeur supérieur à 5V (comme 12V, 24V, 63V...). Plus tu seras proche de la tension (prendre 6V pour 5V par exemple) et moins tu as de marge de sécurité. Bien sur tout ne vas pas péter, mais la durée de vie du condo sera compromise... Rien ne sert pour autant de prendre du 100V pour un montage 5V bien sur 😊. Qui plus est, pour une capacité donnée, plus la tension sera élevée et plus le volume du condensateur sera important.

Je vais me rendre sur le forum ZdS pour présenter mon petit projet et poser mes nombreuses autres questions 😊

Super, a bientôt sur ZdS alors 😊

**Rodolphe**

11/09/2014 at 09:02

Hello

Merci pour cette explication qui m'éclaire l'esprit, je commence avec l'arduino et la prog et j'avoue être un peu perdu par moment avec les moteur pas à pas.

Voilà si vous pouvez m'aider 😊

J'ai un arduino uno avec un easydriver alimenté en 19v et un moteur pas à pas unipolaire (200 tr/min) (4 fils) connecté dessus

Donc deux fils 3 fils branchés sur mon arduino (le gnd, le rouge et le noir)

J'aimerai savoir comment donner une angle maximum au moteur, exemple une ouverture de 180 degrés max qu'il ne fasse pas un tour complet

exemple j'ai un soft qui lui envoie des données négative ou positive tant que c'est positif le moteur va tourner dans un sens mais j'aimerai dire au moteur tu as un angle vers la gauche max de 90 en partant de zero et idem pour la droite

idem je ne comprends pas trop la variable « stepper.monmoteur1=map(valor_servo,?,?,?,?)

Si quelqu'un peut m'aider 😊

Merci

**Eskimon**

11/09/2014 at 10:10

Pour ce genre d'application (contrôle angulaire), le servo moteur (parti précédente) serait sûrement plus adapté et plus simple à prendre en main.

Si tu tiens tout de même à utiliser un moteur pas à pas, je t'invite à venir poster ton sujet sur un forum où il sera plus facile de répondre que cette zone de commentaire 😊 (Je passe très souvent dans la rubrique « Systèmes et Matériel » du site Zeste De Savoir par exemple 😊)

**Rodolphe**

11/09/2014 at 12:20

Merci Eskimon, je vais essayer de trouver mon bonheur dans les forums 😊

**Ames**

30/09/2014 at 08:55

Bonjour. J'ai un projet de piloter un moteur pas à pas pour un slider de reflex. Les moteurs qui répondent à mes critères font au minimum 0.68A et vont jusqu'à 1.7A. Est-ce qu'il y aurait une puce capable d'accepter ces courants ?



**Eskimon**

30/09/2014 at 09:26

Je n'ai pas de référence en tête pour le moment, mais si les composants « classiques » sont trop faibles, il est toujours possible de passer à l'utilisation de transistor (+ diode de roue libre) pour accepter les courants que l'on veux ! (il suffit de bien choisir son transistor avec des caractéristiques acceptant les cas max. du moteur)

**Ames**

03/10/2014 at 10:41

Merci 😊

J'allais partir sur cette solution également c'était juste pour faire le paresseux de partir sur ce genre de puce 😊

**Nono**

28/10/2014 at 11:18

Salut

Merci pour le tuto !

J'ai une petite question, comment gérer avec le code l'arrêt du moteur ?

En gros, arrêter le moteur lors d'un événement précis et le redémarrer une fois l'événement fini.

Si je comprends bien ton code, on ne peut que commander le moteur en lui indiquant un nombre de tours à faire ... (la fonction step utilisée nous empêche de faire autre chose en attendant)

Mon idée est de faire tourner le moteur pendant un temps inconnu et l'arrêter lors d'un événement quelconque ..

Merci

**Eskimon**

28/10/2014 at 11:48

Effectivement, la fonction step ne peut être arrêtée une fois démarrée. Une solution pourrait sûrement être d'avancer juste quelques pas puis quelques pas au lieu de faire une longue avancée d'une traite... Il va falloir ruser pour éviter que la fonction bloquante bloque pendant trop longtemps 😊

C'est vrai qu'avec ce genre de fonctions on tombe vite dans les limites de l'Arduino.

**Nono**

28/10/2014 at 12:05

Oui j'y ai pensé, entre temps j'ai vu aussi que d'autres se passaient de cette fonction et à l'aide des états HIGH/LOW des sorties reproduisaient les 4 cycles nécessaires pour faire un tour, ils en font une fonction avec une boucle j'imagine et le moteur tourne jusqu'à ce que l'on décide de l'arrêter.

C'est un poil plus lourd niveau code mais permet peut-être plus de liberté ...

Je vais creuser cette option

merci en tout cas 😊

**Darkoss**

03/03/2015 at 11:55

Bonjour as-tu réussi à arrêter le moteur au moment souhaité du coup ?

**Rémi**

05/11/2014 at 00:26

Merci pour le tuto, je suis également ton cours sur le sdz, excellent !
J'essaye d'utiliser 2 moteurs pas à pas avec mon arduino uno.
Pour ça j'ai le shield v2 de chez Adafruit et deux moteurs bipolaires.

Le problème quand je lance une séquence test (dans les exemples) mes moteurs vibrent plus qu'ils ne tournent. La plupart du temps ils font du sur place, et sur certain passages tournent timidement avec des méga saccades.

J'ai essayé d'inverser les fils, diminuer les rpm mais rien n'y fait.
Je ne sais pas trop quoi faire d'autre pour essayer de solutionner le problème... (je débute)
tu aurais une idée?
Est ce que je dois mettre des composants entre le shield et le moteur genre resistances/condensateurs?

Il faut peut être que j'adapte mieux mon alimentation mais j'ai un doute, au risque de passer pour un gros amateur, hum, mes deux moteurs ont les mêmes caractéristiques : 12V 0.4A 300Ohms, pour calculer l'alimentation à leur envoyer je dois considérer qu'ils sont en série ou en dérivation? Hors sujet?

 **Eskimon**
05/11/2014 at 09:24

Tu utilises quoi actuellement comme alimentation ? Car effectivement je t'aurais bien conseiller de commencer avec des vitesses pas trop élevée et vérifier les branchements mais si tu l'as déjà fait... 😊

 **Rémi**
05/11/2014 at 09:59

J'ai mis une alim de 12V en 1A, je ne sais pas trop comment calculer le besoin en énergie du système.

 **Eskimon**
05/11/2014 at 10:20

L'alim délivre la bonne tension et avec un courant suffisant apparemment, donc c'est un bon point et je pense que l'on peut l'écartier si tu as bien tout relié. Pour continuer à « debugger » ton problème hardware, pourrait-on discuter sur un forum plutôt que dans la zone de commentaire du tutoriel ? (je suis souvent (avec plein d'autres gens plus compétents !) dans la rubrique « Système et Matériels » de Zeste de Savoir 😊)

 **Clément**
16/11/2014 at 18:55

Petite question : peut-on affecter une variable dans la définition de la vitesse du moteur pà ?
Comme ça :

```
//pour un moteur de 200 pas par tour et brancher sur les broches 2, 3, 4, 5
Stepper moteur(200, 2, 3, 4, 5);
int vitesse =100
void setup()
{
moteur.setSpeed(vitesse); //30 tours par minute
//(rappel : ici le moteur fait 200 pas par tour, on fera donc 6000 pas par minute)
}

void loop()
{
moteur.step(1000);
delay(100);
moteur.step(-1000);
delay(2000);
}
```

Ce qui permettrait de pouvoir la modifier dans le programme.
Merci de votre aide 😊

 **Eskimon**
17/11/2014 at 15:42

Bien sur, on peut !



Eskimon

25/11/2014 at 14:36

Mais il me semble que la précision du moteur est moindre à petite vitesse (je ralentis quand je pense trouver bientôt le point du laser).

Ce ne serait pas cohérent puisqu'un moteur pas à pas peut justement être piloté « au pas près » (donc dans ton cas à 0.025mm près)



Gwénolé

25/11/2014 at 16:55

Bonjour, et oui encore moi !

Je voulais te demander comment avancer seulement d'un seul et unique pas ? Et non pas de 4 par 4....

Merci d'avance.



Eskimon

26/11/2014 at 08:21

Normalement `moteur.step(1);` (en considérant que `moteur` est bien déclaré comme un `Stepper`) fait avancer uniquement d'un pas



Elo-Mon

26/11/2014 at 07:39

Moi aussi je veux bien savoir comment avancer d'un pas 😊



pierrot93300

04/12/2014 at 11:23

Merci pour ce tuto super clair !

Du coup j'ai réussi à me faire un programme avec lequel je contrôle le moteur pas à pas (28BYJ-48) grâce à deux boutons poussoirs : un pour avancer, l'autre pour reculer.

Config : Arduino Mega 2560

Je met mon code pour ceux que ça intéresse :

Bon bricolage !

pierrot93300

EDIT Eskimon: Je me suis permis de « cacher » le code dans un bloc repliable pour éviter d'avoir un message trop long 😊



babasdu24

12/01/2015 at 19:37

Désidément, ce tuto est devenu mon .html de chevet 😊

Une question qui ne tarde : est-il possible de « bloquer » électriquement un moteur PaP à un pas précis ? (éviter la « roue-libre » une fois sa rotation effectuée)



Eskimon

13/01/2015 at 08:54

Un moteur pas à pas n'est pas « auto-verrouillé ». Par magnétisme il y a un petit couple résistant, mais si tu met une charge dessus alors il bougera (test simple, on voit très bien qu'on peut faire tourner le moteur à la main).

Et je me rends compte que je ne répond pas à la question 😊

Pour « verrouiller » le moteur, il faut du couple. Pour avoir du couple, il faut du courant. Donc, pour verrouiller le moteur dans une position il faut ⌈ tenir les bonnes bobines pour « bloquer » le pas actuel (mais là comme ça j'ai pas d'idée de manière programmatoire pour faire ça ⌉

 **babasdu24**
13/01/2015 at 10:45

ok, donc théoriquement faisable, c'est l'essentiel 😊 Merci beaucoup !

 **Guicha29**
21/01/2015 at 08:59

J'ai une question : Peut on brancher un moteur pas a pas 2 A 4.4 V unipolaire avec le motoshield 3 ?

 **Yvan**
25/01/2015 at 23:33

Salut !

Bravo pour cet article, tout est très bien expliqué (aussi bien que celui avec le moteur CC)

J'aurais une petite question concernant le vocabulaire que l'on peut trouver sur les fiches constructeurs.

J'ai fais mes petits calculs de couple nécessaire à mon application et trouve environ 10kg/cm. Je trouve donc un moteur pap qui développe ce moment et regarde le courant utilisé.

Il y a marqué : « Courant/phase: 2,8 A »

Je me demande donc si cette valeur de courant représente l'intensité requise pour effectuer un pas ou bien l'intensité d'une bobine pour faire un pas (et donc nécessiter 5,6A pour actionner les deux bobines).

Avoir un courant de fonctionnement de 2,8 m'arrangerait grandement car je pourrais utiliser le L298 plutôt qu'acheter un shield supplémentaire.

Par avance merci 😊

 **Eskimon**
26/01/2015 at 09:02

Si je dis pas de bêtise, en électrotechnique une phase = une bobine. Donc si tu veux en piloter deux en même temps alors effectivement on passe à 2x2.8. Mais je m'autorise le bénéfice du doute sur cette question tout de même 😊 .

 **Yvan**
26/01/2015 at 21:26

Ok ok,

Et en mode rotation par pas complet, une seule bobine est utilisée donc ?

 **Eskimon**
27/01/2015 at 08:16

Oui 😊 (si tu ne recherches pas à être en condition de couple maximal)

 **fitsi71**
27/01/2015 at 17:01

Bonjour,

Tout d'abord très bel article mais il me reste un problème pour mon projet qui est une « course » de robots donc pour moi ce sera 2 moteurs et en connaissant exactement le circuit je me demandais si je pouvais utiliser ce mode PAP mais il faudrait pour ça (d'après ce que j'ai lu) un hacheur par moteur est-ce obligatoire ?

 **Eskimon**
28/01/2015 at 08:51

↗ pas le choix. Mais il existe des shields/composants dédié au pilotage de deux moteurs pas a pas à la fois.

 **vsahler**
01/02/2015 at 13:26

merci beaucoup je ne savait rien dessus maintenant
je sait beaucoup!!

 **Yvan**
01/02/2015 at 19:02

Re-bonjour,

J'aurais une autre question sur ces moteur pap.

Pour dimensionner le moteur, il nous faut réaliser un calcul de couple.

Jusqu'à là, mes cours de mécaniques me vont très bien...mais une fois calculé le moment d'inertie (en $\text{kg} \cdot \text{m}^2$), je dois le multiplier par l'accélération (en $\text{rad} \cdot \text{s}^{-1}$) pour obtenir le couple en N.m.

Cependant, le moteur est piloté par des pas. Cela signifie-t-il que l'accélération du système tournant se fait en seulement 1 pas ?

Par avance merci 😊

 **Gelabert**
07/02/2015 at 21:25

Bonsoir,

Très intéressant tout cela.

Je me présente, Jean-Pierre, passionné par l'optique et la construction de télescope et d'observation du ciel. Bref, je voudrai bien savoir piloter deux moteurs pas à pas. Depuis cet hiver je me suis mis sérieusement à la programmation en C++. J'ai une culture zéro en électronique et mes questions seraient :

– quelle « carte » arduino me permettrait-elle de commander 2 moteurs pap avec un couple assez important, à savoir que je suis équipé pour usiner des couples roue/vis sans fin pour la réduction). Pour donner une idée cet axe est contraint entre deux roulement conique en opposition de 80 mm intérieur. Sur cet axe sera appliquée un poids pouvant aller jusqu'à 50 kg (bref, une mécanique assez lourde mais parfaitement équilibrés).

– un des axes dit d'ascension droite doit être capable de compenser la rotation de la Terre c-à-d 15 ° par heure après réduction, 3 autres vitesses doivent être possibles 0.5 X, 2 X, 4 X la vitesse de rotation initiale

– quel type de moteur acheter ?

– puis-je éviter un maximum de conception électronique (un truc où « y a qu'à brancher »)

– tout cela pilotable depuis via port usb ou autre port pourvu que ça marche ?

Merci pour votre patience

Jean-Pierre

 **Eskimon**
09/02/2015 at 11:47

Salut !

Eh bien, joli projet !

Une Arduino UNO devrait faire l'affaire sans trop de problème. Couplée avec le bon shield (qu'il faut trouver) il y aura moyen de limiter l'électronique à pas grand chose. Après il sera inévitable de faire un peu de programmation, mais c'est jouable. Pour la taille des moteurs je ne saurais pas trop te dire, n'étant pas trop un spécialiste du domaine. Du coup je te propose de créer un sujet sur Zeste de Savoir rubrique « Systèmes et Matériels » pour en parler plus en détails avec d'autres connaisseurs 😊

 **Didier**
07/02/2015 at 22:43

Bonsoir, merci pour ce tuto

J'ai entré le code (Copier coller) et le moteur (28 BYJ – 48) tourne correctement, mais il ne change pas de sens lorsque la valeur est négative ?

Je ne vois pas pourquoi ? merci de votre avis

**Eskimon**

09/02/2015 at 08:37

Ce moteur est un moteur bipolaire. Il faut s'assurer que les branchements sont bien fait et que les broches sont bien déclarées dans le bon ordre pour l'objet Stepper. Sorti de cela je ne peux pas faire grand chose de plus...

**Simon**

16/02/2015 at 09:48

Bonjour,

Je souhaite utiliser un moteur pas à pas pour un indicateur.

Je comptais partir sur ce moteur : <http://www.gotronic.fr/art-moteur-14hs11-1004s-18351.htm>

Je compte réaliser le montage décrit dans l'article avec une carte arduino UNO pour le traitement.

Ma question est la suivante : Le moteur étant alimenté en 3,5V, puis-je prendre directement le 3,3V en sorti de la carte Arduino en lieu et place du 9V de votre exemple?

D'avance merci.

**Didier**

16/02/2015 at 11:21

Bonjour,

Ne pas prendre le 3.3v de l'Arduino, il ne peut délivrer le courant pour les moteurs et de plus, il faut que ce 3.3v soit préserver de source de parasites, qui ne manqueraient pas d'apparaître lors de l'alimentation des bobines du moteur.

En règle générale, toute alimentation pour des organes de puissance (Moteurs, servo-moteurs, électro- aimants ...) doit être externe.

J'espère que cela répond à ta question

@+

**Simon**

16/02/2015 at 11:28

Bonjour,

Merci pour le retour. Je vais devoir développer une carte perso alimenté en 28V. J'en profiterai pour générer un 3.3V exclusivement utilisé pour le moteur en espérant que ce soit ok.

Cordialement

**Eskimon**

16/02/2015 at 12:18

Malheureusement non. Le 3.3V de l'Arduino n'est pas pour des tâches de « références » et ne peut délivrer que 50mA au maximum, bien insuffisant pour faire tourner un moteur.

**Simon**

24/02/2015 at 11:06

↑
bonjour,

J'ai acheté le moteur suivant : 14HS11-1004S alimenté en 3V5 par une source externe. Après avoir réalisé le schéma que vous proposez avec le

L298 et le code que vous fournissez je constate que le moteur ne répond pas. Il n'y a aucune rotation de celui-ci. Lorsque que je mesure la tension sur les 4 « fils » du moteurs j'obtiens toujours la même tension de 3V5. Je ne vois pas d'où le problème peut venir, avez-vous une idée?

(J'ai pris comme diode des 1N4002).

J'ai vérifié le schéma plusieurs fois le câblage semble ok.

Cordialement

**Eskimon**

24/02/2015 at 12:03

Le L298 a besoin d'une tension de puissance (pour les moteurs) supérieure (ou égale, à confirmer avec la datasheet) à celle de commande (venant de l'Arduino) et dans la plage de 5-46V. Du coup j'ai bien peur que les 3.5V soient trop faibles.

**2bgiles**

25/02/2015 at 00:02

Bonjour je suis novice avec Arduino, merci pour ce tuto, je dispose d'un L297 mais vous parler aussi de L298 je ne comprend pas, est ce que je peux piloter mon moteur avec un L297

Merci

**Eskimon**

25/02/2015 at 08:38

Les deux sont différents. Le L298 est un pont en H, il sert d'interface de puissance pour faire passer le courant dans le moteur. Le L297 est un « séquenceur », il va servir à « faire la séquence » pour le moteur pas à pas, il suffit simplement de lui envoyer la direction et des impulsions. Ils sont de la même famille dans le sens que « ce sont des composants dédiés au pilotage de moteur » mais ils ne remplissent pas les mêmes fonctions.

**2bgiles**

25/02/2015 at 10:32

le tuto propose un schéma qui utilise le L298, est ce que le montage est identique avec un L297 ?

**2bgiles**

25/02/2015 at 10:41

Au cours de mes recherches sur Internet j'ai vu qu'il fallait un L297 + un L298

**Eskimon**

25/02/2015 at 10:54

Oui, comme j'écriveais plutôt, le L297 génère la séquence, le L298 prend le relais pour la puissance.

**Eskimon**

25/02/2015 at 10:53

Le L297 n'est pas obligatoire, si on ne l'utilise pas il faut cependant gérer la séquence du moteur via l'Arduino (but de l'objet Stepper)

**2bgiles**

25/02/2015 at 12:29

Merci beaucoup et une dernière chose, je voudrai acheter un L298 mais il y en a beaucoup et je ne sais pas trop comment ça se présente, pouvez-vous m'indiquer une référence svp

**Eskimon**

25/02/2015 at 12:58

L298 tout seul par exemple : http://www.rs-particuliers.com/WebCatalog/Circuit_de_commande_pour_moteur_en_pont_complet_L298N_Double_driver_de_moteur_Full_Bridge_3A_MULTIWATT_15-7147711.aspx

 **dis pilulier**
05/03/2015 at 12:11

bonjour
je suis débutante en programmation arduino
je désire commander un moteur pas à pas qui fera 16,36 °
voici le programme que j'utilise
j'arrive pas à modifier les paramètres du moteur pour le faire tourner 16,36°

```
1 #include
2
3 //pour un moteur de 200 pas par tour et brancher sur les broches 2, 3, 4, 5
4 Stepper moteur(200, 2, 3, 4, 5);
5
6 void setup()
7 {
8     moteur.setSpeed(30); //30 tours par minute
9     // (rappel : ici le moteur fait 200 pas par tour, on fera donc 6000 pas par minute)
10 }
11
12 void loop()
13 {
14     moteur.step(1000);
15     delay(100);
16     moteur.step(-1000);
17     delay(2000);
18 }
```

 **Eskimon**
05/03/2015 at 12:53

Je ne suis pas sur de comprendre d'où vient le problème. Pour faire tourner un moteur d'un certain angle, il faut savoir combien de pas il faut faire. Pour cela il faut savoir combien de pas le moteur fait par tour puis faire un produit en croix pour obtenir le nombre de pas à faire.

 **dis pilulier**
07/03/2015 at 21:42

dnc il faut seulement modifier la fct moteur.step ??

 **Eskimon**
09/03/2015 at 15:14

Non, il faut juste l'appeler en précisant le bon nombre de pas à faire. Par exemple, si un moteur fait 200 pas par tour, cela signifie qu'il faut 200 pas pour faire 360°. Donc pour faire 90° il faudrait juste faire $360/90 = 4$ fois moins de tour, soit 50 pas.

Le même raisonnement appliqué à tes angles et les caractéristiques de ton moteur permettront de savoir combien de pas il faut faire.

 **Dark-snox**
06/03/2015 at 08:51

Bonjour

je souhaite commander un moteur Pas à Pas avec un capteur infrarouge et faire en sorte que le moteur s'arrête lorsque le capteur détecte un objet.

```
int sensor = 2; // pin 2
int motor_pin_1 = 3;
int motor_pin_2 = 13;
int motor_pin_3 = 11;
int motor_pin_4 = 2;
int val = 0;
int p;
// motor functions
` ^ _motor()
{
switch(p){
```

case 0 :

```
digitalWrite(motor_pin_1, HIGH); //1100
digitalWrite(motor_pin_2, LOW);
digitalWrite(motor_pin_3, HIGH);
digitalWrite(motor_pin_4, LOW);

delayMicroseconds(800);

digitalWrite(motor_pin_1, HIGH); //1100
digitalWrite(motor_pin_2, LOW);
digitalWrite(motor_pin_3, HIGH);
digitalWrite(motor_pin_4, LOW);

delayMicroseconds(800);

digitalWrite(motor_pin_1, LOW); //0110
digitalWrite(motor_pin_2, HIGH);
digitalWrite(motor_pin_3, HIGH);
digitalWrite(motor_pin_4, LOW);

delayMicroseconds(800);

digitalWrite(motor_pin_1, LOW); //0110
digitalWrite(motor_pin_2, HIGH);
digitalWrite(motor_pin_3, HIGH);
digitalWrite(motor_pin_4, LOW);

delayMicroseconds(800);

digitalWrite(motor_pin_1, LOW); //0011
digitalWrite(motor_pin_2, HIGH);
digitalWrite(motor_pin_3, LOW);
digitalWrite(motor_pin_4, HIGH);

delayMicroseconds(800);

digitalWrite(motor_pin_1, LOW); //0011
digitalWrite(motor_pin_2, HIGH);
digitalWrite(motor_pin_3, LOW);
digitalWrite(motor_pin_4, HIGH);

delayMicroseconds(800);

digitalWrite(motor_pin_1, HIGH); //1001
digitalWrite(motor_pin_2, LOW);
digitalWrite(motor_pin_3, LOW);
digitalWrite(motor_pin_4, HIGH);

delayMicroseconds(800);

digitalWrite(motor_pin_1, HIGH); //1001
digitalWrite(motor_pin_2, LOW);
digitalWrite(motor_pin_3, LOW);
digitalWrite(motor_pin_4, HIGH);

delayMicroseconds(800);
break;
}
}
```

```
void stop_motor ()
{
  ^ (p) {
    digitalWrite(motor_pin_1, LOW); //1001
```

```

digitalWrite(motor_pin_2, LOW);
digitalWrite(motor_pin_3, LOW);
digitalWrite(motor_pin_4, LOW);

delayMicroseconds(800);
break;

}

}

void setup(void)
{
pinMode(sensor, INPUT); // declare sensor as input

pinMode(motor_pin_1, OUTPUT);
pinMode(motor_pin_2, OUTPUT);
pinMode(motor_pin_3, OUTPUT);
pinMode(motor_pin_4, OUTPUT);

Serial.begin(9600); //Set Baud Rate
}
void loop(void)
{
val = digitalRead(sensor);
if (val == HIGH)
{
stop_motor();
}
else
{
run_motor();
}
}

```

Mon seul problème est au niveau de ma fonction stop_motor() , je n'arrive pas à arrêter le moteur.

Le moteur tourne en continu.



Eskimon

09/03/2015 at 14:58

A quoi sert la variable 'p' et les switch correspondants ? As tu essaye sans tout cela pour garder un code simple ?



Dark-snox

10/03/2015 at 08:21

j'ai essayé avec ceci :

```

[code]int motor_pin_1 = 3;
int motor_pin_2 = 13;
int motor_pin_3 = 11;
int motor_pin_4 = 2;

void setup() {
pinMode (motor_pin_1, OUTPUT);
pinMode (motor_pin_2, OUTPUT);
pinMode (motor_pin_3, OUTPUT);
pinMode (motor_pin_4, OUTPUT);
}

void loop() {
digitalWrite(motor_pin_1, HIGH); //1100
digitalWrite(motor_pin_2, LOW);
}

```

```
digitalWrite(motor_pin_3, HIGH);
digitalWrite(motor_pin_4, LOW);

delayMicroseconds(800);

digitalWrite(motor_pin_1, HIGH); //1100
digitalWrite(motor_pin_2, LOW);
digitalWrite(motor_pin_3, HIGH);
digitalWrite(motor_pin_4, LOW);

delayMicroseconds(800);

digitalWrite(motor_pin_1, LOW); //0110
digitalWrite(motor_pin_2, HIGH);
digitalWrite(motor_pin_3, HIGH);
digitalWrite(motor_pin_4, LOW);

delayMicroseconds(800);

digitalWrite(motor_pin_1, LOW); //0110
digitalWrite(motor_pin_2, HIGH);
digitalWrite(motor_pin_3, HIGH);
digitalWrite(motor_pin_4, LOW);

delayMicroseconds(800);

digitalWrite(motor_pin_1, LOW); //0011
digitalWrite(motor_pin_2, HIGH);
digitalWrite(motor_pin_3, LOW);
digitalWrite(motor_pin_4, HIGH);

delayMicroseconds(800);

digitalWrite(motor_pin_1, LOW); //0011
digitalWrite(motor_pin_2, HIGH);
digitalWrite(motor_pin_3, LOW);
digitalWrite(motor_pin_4, HIGH);

delayMicroseconds(800);

digitalWrite(motor_pin_1, HIGH); //1001
digitalWrite(motor_pin_2, LOW);
digitalWrite(motor_pin_3, LOW);
digitalWrite(motor_pin_4, HIGH);

delayMicroseconds(800);

digitalWrite(motor_pin_1, HIGH); //1001
digitalWrite(motor_pin_2, LOW);
digitalWrite(motor_pin_3, LOW);
digitalWrite(motor_pin_4, HIGH);

delayMicroseconds(800);
}[/code]
```

Et mon moteur tourne parfaitement mon seul problème est de trouver un moyen de l'arrêter, je suis débutant en programmation et j'ai donc voulu arrêter le moteur en mettant tous les pins en » LOW »

Les switchs je voulais les utilisés pour essayer d'arrêter le moteur grâce aux breaks.

 **Eskimon**
16/03/2015 at 13:12

▲ Pour l'arrêter il suffit de ne pas être dans la boucle effectivement (pourquoi pas un switch oui). Cependant, pourquoi ne pas directement utiliser l'objet **Stepper** dédié à ce genre de chose ?



MarcS

10/03/2015 at 01:01

Bonjour,

Alors actuellement je pilote deux moteurs pas a pas de façon manuel

Chaque moteur possède son driver, les deux driver, son piloter par un générateur de pulsation et variateur de vitesse.

J'aurais donc voulus remplacer le générateur par la carte.

je te met en lien le matériel que j'utilise actuellement:

le générateur:

<http://www.ebay.com/itm/Pulse-Signal-Generator-Stepper-Motor-Speed-Regulator-Controller-Motor-Driver-/330692980897>

Les drivers:

<http://www.ebay.com/itm/1-Axis-Controller-Stepper-Motor-Drivers-TB6560-3A-Driver-Board-CNC-Router-Single-/171219013554>

Le schema de montage:

<http://i38.servimg.com/u/f38/11/24/04/90/cablag10.jpg>

Merci d'avance.



Eskimon

16/03/2015 at 13:11

Ok. Et ? Il y a un point qui bloque ? En utilisant des alternances de `digitalWrite()` High et Low tu devrais pouvoir faire un signal carre pour faire comme le générateur de pulsation non ?



MarcS

17/03/2015 at 10:50

Je viens d'essayé et sa ne fonctionne pas le moteur grogne et tourne pas correctement.



Eskimon

19/03/2015 at 08:19

Peut-être les alternances HIGH/LOW ne sont pas assez rapide (ou au contraire trop rapide), que se passe t-il en mettant un `delay()` entre les deux ?



lycéendanslamerde

11/03/2015 at 12:53

Bonjour, je réalise un projet pour mon épreuve bac de terminale SI avec mes amis. Dans ce projet nous devons mettre en rotation un moteur pas à pas de 1.8° bipolaire avec 4 fils, en passant par un driver ainsi que la carte et le logiciel Arduino. Le problème est que notre moteur ne tourne pas et nous avons beau chercher nous n'arrivons pas à déterminer s'il s'agit de notre branchement ou bien de notre programme.

Voici ci-dessous notre programme:

```
#include
```

```
Stepper moteur(300, 2, 3, 4, 5);
```

```
void setup()
{
moteur.setSpeed(5);
}
void loop()
{
moteur.step(295);
r  .setSpeed(295);
i  .step(-295);
moteur.setSpeed(5);
}
```

Si notre programme est bon pouvez vous nous expliquer comment réaliser le branchement.

Merci d'avance pour votre réponse.



Eskimon

16/03/2015 at 13:24

Salut,

Le code semble bon, donc les soucis sont sûrement du côté du branchement. Pour cela je conseil d'aller faire un sujet sur le forum de Zeste de Savoir (rubrique « Systèmes et Matériels ») et d'y poster quelques photos pour mieux voir. J'y passe très souvent ainsi que de nombreux autres passionnés qui sauront aussi t'aider 😊 !



seb

12/05/2015 at 16:53

Hello, je pensais que pour les progs avec des moteurs il fallait la librairie #include stepper.h..? ou de ce genre...



Eskimon

13/05/2015 at 07:36

Oui il le faut, mais le module de commentaires de wordpress a la mauvaise tendance à dégager les « include » 😞



Christian

13/03/2015 at 11:22

Bonjour,

Je viens de lire votre article que je trouve très bien expliqué, cependant j'ai une question... Je voudrais réaliser un appareil me permettant de pouvoir calibrer des drones utilisant un module APM 2.6, il faudrait donc que je pilote plusieurs moteur pas à pas, Je laisse les liens vidéos pour montrer ce que je souhaite faire... par la suite je voudrais savoir quel serait vos conseils afin de réaliser ce projet ainsi que le matériel nécessaire:

https://www.youtube.com/watch?v=_mjfPIHL9o
<https://www.youtube.com/watch?v=hgiTbf2Hw5I>

Je vous remercie pour votre aide

Cordialement



Eskimon

16/03/2015 at 13:21

Bonjour,

Je ne fais pas d'aide sur le blog pour les gros projets (comme les drones). Cependant, pour cela je conseil d'aller faire un sujet sur le forum de Zeste de Savoir (rubrique « Systèmes et Matériels »). J'y passe très souvent ainsi que de nombreux autres passionnés qui sauront aussi t'aider 😊 !



Amaury

25/03/2015 at 16:06

Bonjour,

Votre article est très intéressant ! J'ai cependant une petite question : je possède une carte qui délivre directement un signal PWM et me permettant de contrôler un moteur pas à pas (electrovanne IAC). Malheureusement je ne peux connecter ma carte qu'avec deux fils... Comment faire pour relier mon moteur pas à pas à 4 fils jusqu'à l'entrée de ma carte à 2 fils ?

ement

**Eskimon**

25/03/2015 at 16:12

Il y a t-il une référence écrite quelque part sur la carte ou quoi que ce soit qui pourrait amener à une documentation ?

**Amaury**

25/03/2015 at 16:17

La carte en question est une carte cosworth pectel T2 déstinée à l'automobile et qui permet d'envoyer un signal PWM pour gérer une électrovanne. J'ai en ma possession une électrovanne du type moteur pas à pas à 4 fils donc je ne sais pas comment m'y prendre pour le relier à ma carte pectel qui ne demande que 2 fils pour envoyer le signal PWM.

**Eskimon**

26/03/2015 at 09:20

Donc la carte n'est pas compatible avec le moteur. Un moteur pas a pas ne demande pas une PWM mais plutôt une séquence d'état.

**nawfel**

26/03/2015 at 09:03

bonjour

je peut ajouter un codeur optique avec le moteur pas a pas ?
comment le programmer ?

**Eskimon**

26/03/2015 at 09:26

Oui on peut. Un codeur est indépendant de la technologie du moteur.

Pour la programmation cependant, cela mérirerait un article à lui tout seul, donc je te conseille d'aller te promener sur google car je ne pourrais en faire,

**BILel**

26/03/2015 at 22:03

Bonsoir, je réalise un projet PFE je besoin le programme mikro C d'un moteur pas à pas liée avec ULN2003A et PIC 16F887

**Eskimon**

27/03/2015 at 08:35

Bonjour,

Je ne fais pas de mikroC et ca fait bien longtemps que je n'ai pas utilise des PIC... bon courage dans vos recherches.

**MarcS**

29/03/2015 at 16:06

Bonjour me voila de retour.

Alors je suis passé sur un Shield Motor R3 arduino et mon Arduino UNO pour commandé un moteur pas à pas.

Voici mon programme :

```
#include<br><math>const int stepsPerRevolution = 200;</math><math>Stepper myStepper(stepsPerRevolution, 12,13);</math><math>^ it pwmA = 3;</math><math>const int pwmB = 11;</math><math>const int dirA = 12;</math>
```

```
const int dirB = 13;

void setup() {
    pinMode(pwmA, OUTPUT);
    pinMode(pwmB, OUTPUT);

    digitalWrite(pwmA, HIGH);
    digitalWrite(pwmB, HIGH);

}

void loop() {
    myStepper.setSpeed(100);
    myStepper.step(900);
    delay(1000);
}
```

Donc le moteur tourne correctement jusque la pas de soucis.

Mais dés que je souhaite augmenté la vitesse la tout déconne ... le moteur grogne et tourne mal et je comprend pas trop pourquoi... Si je dépasse 100 sa passe pas en dessous pas de soucis.

Si je passe au-dessus de 100 et que je diminue le nombre de pas du moteur par exemple a 96 48 24 sa passe je comprend pas le rapport entre les deux si on pouvais éclairé ma lanterne 😊 Merci d'avance.

 **Eskimon**
30/03/2015 at 08:07

Le moteur est correctement alimenté (bonne tension et intensité suffisante) ?

 **MarcS**
03/04/2015 at 23:11

Oui pas de soucis la dessus j'ai vérifier a plusieurs reprise l'alimentation pose aucun soucis. Sur un autre montage manuel j'arrive a piloter 2 moteur sans aucun soucis avec la meme alimentation . D'ou mon interrogation sur le sujet

 **Nitch**
22/06/2015 at 15:47

Bonjour,
Est ce que ton problème est réglé? Pourrais tu m'envoyer la photo de ton montage?
Merci d'avance

 **MarcS**
23/06/2015 at 18:31

Mon problème est réglé mais j'ai fait complètement différemment enfaite. J'avais un système manuel du coup j'ai automatisé le système manuel

 **2bgiles**
29/03/2015 at 16:19

Bonjour j'ai réaliser le montage avec le moteur pas a pas 6 fils. Au debut tout fonctionnait normalement. ensuite j'ai mis 360 a « motor step », la valeur est petite, le moteur a tourné puis c'est bloqué.. Depuis le moteur a des problème j'ai refis le montage, changer la valeur (j'ai mis 1000 et à l'autre -1000) le moteur tourne dans un sens mais pas dans l'autre, l'arbre moteur ne tourne plus et le moteur vibre... Pouvez expliquer cela ? Ai-je r
c
e moteur ?

**Eskimon**

30/03/2015 at 08:08

le moteur a tourné puis c'est bloqué

C'est à dire ? Il y a eu mécaniquement un problème ? J'avoue ne pas bien voir comment l'enchaînement des pas pourrait casser le moteur...

**Benouville**

29/03/2015 at 22:55

Salut Eskimon,
j'ai lu avec beaucoup d'attention tes différents articles, ils, m'ont été très utiles merci beaucoup!
Je travaille actuellement sur un projet de cnc à découpe fil chaud et souhaiterai utiliser une carte arduino pour piloter mes moteurs pas à pas. Hélas je ne m'y connais pas énormément en électronique.
De tout ce que j'ai peu voir sur internet, les shields arduino sont compatibles avec deux moteurs pas à pas seulement, or moi j'ai besoin de 4 moteurs, si ce n'est 5, et en plus de tout cela j'ai besoin de garder certaines broches libres sur mon arduino afin de réaliser certain asservissements!
Je ne sais pas si de tels shields existent, as tu des recommandations?
Aussi je pense devoir utiliser une arduino due car je crois comprendre que je vais vite manquer de broches!

merci de ta réponse!

**Benouville**

29/03/2015 at 23:03

petite précision! je parle ici de moteurs bipolaires (j'envisage des NUM)
merci beaucoup

**Eskimon**

30/03/2015 at 08:17

La DUE ne doit pas être choisi pour son nombre de broches, c'est une approche complètement différente de la programmation. Plutôt choisir une Mega dans ce cas là. Pour ce qui est des shields pouvant piloter 4 pas à pas, je sais qu'il en existe mais je ne connais pas de référence précise.

**Benouville**

31/03/2015 at 17:23

très bien, merci beaucoup de ta réponse!

**JBB**

01/04/2015 at 11:27

Salut Eskimon,
Je suis actuellement en terminale S SI et je doit faire un projet pour le bac.
Dans notre projet on utilise un moteur pas à pas et on essaye de le programmer afin qu'il fasse 590 pas en 59 min soit 1 pas toute les 6 secondes et une fois ces 590 pas fait il faut qu'il refasse 590 pas en a peu près 1 min, soit environ 1 pas toutes les 0.1 sec, dans le sens inverse. J'ai fais un programme mais j'ai une question par rapport aux deux boucle « for ». Est-ce que il y a un risque que mes deux boucles « for » s'effectuent simultanément?
Je vous joint mon programme ci-dessous:

```
#include

int n = 0;
Stepper moteur(200, 2, 3, 4, 5);

void setup()
{
  moteur.setSpeed(50);
  int n = 0;
}

void loop()
{
```

```
for(int n = 0; n < 590; n++)
{
moteur.step(-1);
delay (6000);
}
for(int n = 590; n < 1180; n++)
{
moteur.step(1);
delay (100);
}
```

Merci de votre réponse.

**Eskimon**

01/04/2015 at 11:33

Ca devrait aller, as tu essaye le code ?

**JBB**

01/04/2015 at 11:55

Beh pas encore parce que niveau branchement on est en galère. On dispose d'un arduino UNO, d'un moteur pas à pas 1.8° Sanyo Denki, un driver L298N, un autre Samotronic101 et aussi un easy step 1000 mais même avec tout ça on reste bloqué sur les branchements...

PS: Merci d'avoir répondu aussi rapidement.

**kamel**

01/04/2015 at 14:34

Bonjour, je possède un moteur pas à pas « Unipolaire » de marque shinano kenshi, modèle STH-54D500, avec une tension de 6.5 V et une intensité de 1.3 A. J'aimerai savoir si j'ai besoin des mêmes composants et branchements fait dans votre tutoriel ? A savoir, le ULN2003 et une carte arduino Uno bien sur.

Je tiens à préciser que je dois utiliser ce moteur sur un dispositif qui nécessite relativement un certain couple, donc j'aimerai savoir, si tout les composants de contrôle et de pilotage n'ont rien à voir là dedans ? Merci 😊

**Eskimon**

01/04/2015 at 15:09

Il faudra trouver autre chose que le ULN2003, ce dernier ne pouvant que fournir 500mA ça sera vite contraignant pour ce moteur.

**kamel**

01/04/2015 at 16:25

Rebonjour, franchement je suis perdu, car je n'ai pas trouvé de driver pour mon moteur et je ne sais pas quoi mettre à la place, pouvez vous me conseiller sur ce fait, quel composant devrais je utiliser à la place de l'ULN2003 ?

**Eskimon**

02/04/2015 at 07:40

Ce sujet du forum officiel Arduino (anglais) devrait pouvoir t'éclairer un peu. Je ne connais pas tous les composants possibles et imaginables, donc à part te conseiller de chercher un « driver de moteur » par exemple je n'ai pas de meilleures réponses à donner.

**TERMINALESI**

02/04/2015 at 11:04

Bonjour nous sommes un groupe d'élèves en terminale S SI et nous avons fait un programme pour faire tourner un moteur pas à pas (rélié à un bras avec un bouton sur le bout du bras plus deux autres boutons permettant de commander la rotation du bras dans un sens ou l'autre). Par exemple quand on appuie sur le bouton de droite le moteur fait tourner le bras à droite et tant que le bouton sur le bras n'a pas appuyé sur un obstacle le moteur continue de tourner). On utilise un moteur pas à pas de 1.8° bipolaire avec 4 fils passant par un moteur shield, un système Grove pour les boutons et une carte arduino UNO. On alimente arduino par secteur avec un adaptateur de 5V en sortie relié par USB à la carte Arduino.

Voilà notre programme, pensez-vous qu'il marchera ?

```

1 #include "Stepper.h"
2
3 Stepper moteur(200, 2, 3, 4, 5);
4
5
6 const int btn_bras = 6;
7
8 const int btn_droite = 7;
9
10 const int btn_gauche = 8;
11
12
13
14 void setup()
15 {
16     moteur.setSpeed(15); //15 tours par minute
17     pinMode(btn_bras, INPUT);
18     pinMode(btn_droite, INPUT);
19     pinMode(btn_droite, INPUT);
20 }
21
22
23
24
25 void loop()
26 {
27     int droite = digitalRead(btn_droite);
28
29     if(droite == LOW){
30
31         int bras = digitalRead(btn_bras);
32
33         while(bras == HIGH){ //bouton pousoir pas appuyer
34
35             moteur.step(5); //9°
36         }
37         delay(10);
38         moteur.step(-80); //recule de 144° pour lacher la feuille
39     }
40
41     //faire pareil pour l'autre bras.....
42
43     int gauche = digitalRead(btn_gauche);
44
45     if(gauche == LOW){
46
47         int bras = digitalRead(btn_bras);
48
49         while(bras == HIGH){ //bouton pousoir pas appuyer
50
51             moteur.step(-5); //avance de 9° pour attraper la feuille
52         }
53         delay(10);
54         moteur.step(80); //recule de 144° pour lacher la feuille
55     }
56
57 }

```



Eskimon

02/04/2015 at 11:55

Ptet bien que oui, ptet bien que non 😊 . Pourquoi ne pas directement l'essayer *in-situ* ?



hamdi

02/04/2015 at 13:05

Bonjour à tous

<http://cjoint.com/?ECmrCL34Ffv>

voilà une image pour bien comprendre mon problème ... en fait je veux savoir comment je pourrais faire la commande (asservissement) d'un moteur pas à pas bipolaire Nema 42 par une carte arduino uno car j'ai envie d'asservir le système du broche de ma perceuse sensitive... le moteur doit être à vitesse variable et qui suit les consignes reçus par les capteurs de position alors si vous pourriez m'aider à faire le code de mon moteur et les capteurs de présence (capacitifs)

Merci d'avance pour votre aide
Hamdi



Eskimon

02/04/2015 at 13:23

Bonjour,

Pour les demandes d'aide pour des projets je te propose d'aller ouvrir un sujet sur le forum « Systèmes et Matériels » de Zeste de Savoir. Plein de

gens compétents devraient aussi pouvoir t'aider 😊

**hamdi**

02/04/2015 at 14:30

Merci, mais vous ne pourrez jamais m'aider? 😊

**Eskimon**

02/04/2015 at 14:48

Si, sur ton sujet sur Zeste de Savoir 😊

**hamdi**

03/04/2015 at 12:44

Rebonjour personne n'a répondu :(:(c'est urgent s'il vous plaît :(

**Eskimon**

03/04/2015 at 13:35

Comme je te l'indiquais juste au-dessus : Pour les demandes d'aide pour des projets je te propose d'aller ouvrir un sujet sur le forum "Systèmes et Matériels" de Zeste de Savoir. Plein de gens compétents devraient aussi pouvoir t'y aider 😊

**Geoffrey**

03/04/2015 at 16:21

Bonjour j ai un moteur pas à pas et je voudrais le faire tourner d'un tour complet puis l arrêter et ensuite le faire tourner dans l'autre sens pour qu'il revienne dans sa position initiale. Aurais-tu une idée du code à compiler ?

**Eskimon**

03/04/2015 at 16:24

Bonjour, as tu lu le tutoriel ? Cette question est répondu exactement par un des codes de la page ou il suffit de changer la valeur à envoyer par celle du nombre de pas pour faire un tour.

**Geoffrey**

05/04/2015 at 23:06

Oui j'ai lu et quand je rentre le code comme dans l'exemple. Le moteur tourne en continue. Il ne s'arrête pas et ne change pas de direction

**hamdi**

05/04/2015 at 01:48

bonsoir 😊

pourriez vous me donner fonction qui active la rotation (sans faire entré le nombre des as) du moteur pas à pas...j'en ai besoin pour que le moteur se met à tourner (descendre) en appuyant sur l'interrupteur qui est branché à la carte arduino et pour qu'il avance vers les capteurs lesquelles j'ai fais des conditions dans mon sketch (à réaliser)..merci beaucoup

**hamdi**

05/04/2015 at 02:31

(sans faire entré le nombre des pas*) désolé



**Eskimon**

07/04/2015 at 08:05

Il n'y a pas de telles fonctions car ce n'est pas vraiment le but d'un pap mais plutôt d'un moteur CC.

**Tournie**

10/04/2015 at 14:54

Bonjour Eskimon est-ce possible de réaliser un montage bipolaire (4 fils sur un arduino uno et un L293D.

**Eskimon**

11/04/2015 at 08:55

Normalement oui, la partie « LA PRATIQUE AVEC LE L298 » explique comment le brancher (le L298 étant similaire au L293) et un peu plus loin dans l'article la partie pilotage est mise en oeuvre.

**hamdi**

11/04/2015 at 02:53

bonsoir Eskimon...en bref j'ai vraiment essayé de trouver des solutions pour mon problème mais sérieusement c'était pas évident...je voulais savoir si vous pourriez m'aider à trouver la solution pour désactiver le capteur b1 lorsque la broche est au niveau b2 et le réactiver lorsque la broche est au niveau h

Merci pour votre compréhension

<http://www.cjoint.com/15av/EDlefM1tD3.htm> pour l'état des capteurs

<http://cjoint.com/?ECmrCL34Ffv> pour la description du système

**Eskimon**

11/04/2015 at 08:50

Bonjour,

J'ai bien vu ton sujet sur ZdS, j'y répondrais là bas si je peux 😊

**hamdi**

11/04/2015 at 11:29

Bonjour,

je vous attends avec impatienc 😊 Merci infiniment

Hamdi

**Daniel F**

11/04/2015 at 18:51

Bonjour,

pas moyen de trouver un code.

je débute et je cherche à commander la vitesse d'un moteur pas à pas via un potentiomètre avec une carte uno et le shield adafruit. entre autre pas moyen de faire la jonction entre la lecture du potentiomètre et la vitesse du moteur....

help

merci d'avance

**Eskimon**

13/04/2015 at 07:59

Le problème des librairies pour les moteurs pas à pas c'est qu'elles sont bloquantes. Autrement dit, une fois que la vitesse est réglée on peut envoyer un nombre de pas à faire à cette vitesse et on ne peut rien toucher tant que le moteur tourne. Du coup la solution de « faire juste un seul pas à la fois » n'est pas dénuée de sens (mais je suis d'accord, on aimerait mieux ^^)

 Daniel F

11/04/2015 at 20:28

j »ai trouvé une solution mais ca me semble boiteux et inhesthetique malgré le fait que ca marche.....

un conseil?

Dan

 Daniel F

11/04/2015 at 20:29

ouupss.

voici le code:

```
1 #include
2 #include
3 #include "utility/Adafruit_PWM_Servo_Driver.h"
4
5 // Create the motor shield object with the default I2C address
6 Adafruit_MotorShield AFMS = Adafruit_MotorShield();
7 // Or, create it with a different I2C address (say for stacking)
8 // Adafruit_MotorShield AFMS = Adafruit_MotorShield(0x61);
9
10 // Connect a stepper motor with 200 steps per revolution (1.8 degree)
11 // to motor port #2 (M3 and M4)
12 Adafruit_StepperMotor *myMotor = AFMS.getStepper(200, 1);
13 int analogPin=A0;
14 float VMax=10;
15 int potar;
16 int memo_potor=0;
17 float constante=VMax/1024;
18 int vitesse;
19
20
21
22 void setup() {
23   Serial.begin(9600);           // set up Serial library at 9600 bps
24   Serial.println("Stepper test!");
25
26   AFMS.begin(); // create with the default frequency 1.6KHz
27   //AFMS.begin(1000); // OR with a different frequency, say 1KHz
28
29 }
30
31 void loop() {
32
33   potar = analogRead(analogPin);
34   if (potar != memo_potor){
35     vitesse=potor*constante;
36     memo_potor=potor;
37   }
38   myMotor->setSpeed(vitesse); // 10 rpm
39
40   myMotor->step(1, FORWARD, SINGLE);
41 }
```

 Farouk

14/04/2015 at 09:13

Bonjour,

Je vois que beaucoup de personne cherche a arréter le moteur par code, du coup je me pose la question suivante :

Est-il possible de l'arréter tout simplement par une interruptions du timer? et donc sa impliquerai de le relancer a la suite d'une action (capteur, bouton poussoir...). Après je suis tout nouveau en arduino je sais même pas faire tourner le moteur ^^. Bref d'apres le datasheet de l'arduino sa devrait etre possible.

 Daniel F

14/04/2015 at 10:23

Bonjour Farouk,

Pour te faire la main et avoir des exemples de code pour gérer un moteur utilise la librairie « exemples », elle est assez complète.

pour ce qui est des interruptions, le seul truc que j'ai trouvé, c'est une boucle sur un pas interrompu par l'ouverture ou la fermeture d'un contact mis sur anal1.

il doit y avoir certainement mieux





Nans P

14/04/2015 at 12:27

bonjour je suis actuellement en terminale s si et pour nos PPE nous devons fabriquer une commande d'axe d'une imprimante 3d delta (rostock MAX). Nous avons donc commandé le moteur pas à pas bipolaire (<http://www.gotronic.fr/art-moteur-42byghw811-12154.htm>), et nous avons aussi une carte arduino méga 2560 et un motor SHIELD rev3.

la question est :

quel sont les branchements à faire pour pouvoir faire fonctionner le moteur ?

Est-ce que le SHIELD rev3 fait office de L298 ?

Ou faut-il que nous achetions aussi un L298 ?

Merci d'avance pour votre réponse 😊



Eskimon

14/04/2015 at 13:45

| Est-ce que le SHIELD rev3 fait office de L298 ?

Oui

| quel sont les branchements à faire pour pouvoir faire fonctionner le moteur ?

A vous de faire un peu de recherche, ça fait partie de votre travail de PPE je suppose 😊



farouk

15/04/2015 at 07:21

Salut,

je suis en stage, donc pour l'instant je gratte des programmes jusqu'à avoir l'autorisation de tester, je me pencherai ensuite sur l'idée de l'interruption.

D'autres part, je dois réaliser une rampe de vitesse, pour un moteur pas à pas, dans le cadre de mon projet, est-ce que quelqu'un a déjà eu l'occasion de le faire ?

Je pensais à incrémenter une variable mais il y a encore cette fonction bloquante qui me gêne. alors si je met mon interruption « overflow » est-ce que sa pourrait fonctionner ? Est-ce que je sortirais de cette fonction « step » ?

Merci à l'avance



Eskimon

15/04/2015 at 08:40

Pour la rampe c'est l'idée, on incrémente au cours du temps une variable représentant la vitesse.

Pour le caractère bloquant essaie d'aller faire un tour la dessus : <http://interface.khm.de/index.php/lab-log/arduino-stepperlab3-motor-library/>. Je n'ai jamais essayé mais ça peut valoir le coup d'œil !



farouk

15/04/2015 at 14:04

Ah ben merci, ça a l'air d'être ce que je cherche je teste sa dès que j'ai l'équipement et je vous tiens au courant.



farouk

16/04/2015 at 09:46

J'ai gratter sa, je devrais bientôt le tester mais je voulais savoir ce que vous en pensiez histoire que je fasse pas un flop.

j'ai finalament opter pour la solution de contrôle pas après pas avec un delay pour gérer la fréquence de pas. Sa devrait marcher... non ?

[code]
↑ ↑ e

```
int Freq=48;  
float tps=21.00;
```

```
int ENA=2; // Arduino Pin 2
int IN1=3; // Arduino Pin 3
int IN2=4; // Arduino Pin 4
int ENB=5; // Arduino Pin 5
int IN3=6; // Arduino Pin 6
int IN4=7; // Arduino Pin 7
int sens=8; //Arduino PIN 8
int go=12; // Arduino Pin12
digitalWrite(13,HIGH);
void setup()
{
pinMode(ENA,OUTPUT); //Les 6 pins configurées en sorties
pinMode(ENB,OUTPUT);
pinMode(IN1,OUTPUT);
pinMode(IN2,OUTPUT);
pinMode(IN3,OUTPUT);
pinMode(IN4,OUTPUT);
pinMode(sens,INPUT); // Interruuteur Marche
pinMode(go,INPUT); //Interupteur de sens
digitalWrite(sens,HIGH);
digitalWrite(go,HIGH);
}

void loop()
{
if (digitalRead(sens)==HIGH)
{
delay(50) //Supression des rebonds
while(digitalRead(go)==HIGH) //Bouton Go pour alimenter les bobines
{
digitalWrite(IN1,LOW); //0110 A-B
digitalWrite(IN2,HIGH);
digitalWrite(IN3,HIGH);
digitalWrite(IN4,LOW);
delay(tps);
digitalWrite(IN1,LOW); //0101 A-B
digitalWrite(IN2,HIGH);
digitalWrite(IN3,LOW);
digitalWrite(IN4,HIGH);
delay(tps);
digitalWrite(IN1,HIGH); //1001 AB-
digitalWrite(IN2,LOW);
digitalWrite(IN3,LOW);
digitalWrite(IN4,HIGH);
delay(tps);
digitalWrite(IN1,HIGH); //1010 AB
digitalWrite(IN2,LOW);
digitalWrite(IN3,HIGH);
digitalWrite(IN4,LOW);
delay(tps);
while(tps>2.5){tps-=0.5;} // Pallier à 400Hz
}
}

if (digitalRead(sens)==LOW) // Deuxieme Sens
{
delay(50)
while(digitalRead(go)==HIGH)
{
digitalWrite(IN1,HIGH); //1010 AB
digitalWrite(IN2,LOW);
digitalWrite(IN3,HIGH);
digitalWrite(IN4,LOW);
// write(IN4,LOW);
// .ps);
digitalWrite(IN1,HIGH); //1001 AB-
digitalWrite(IN2,LOW);
}
```

```
digitalWrite(IN3,LOW);
digitalWrite(IN4,HIGH);
delay(tps);
digitalWrite(IN1,LOW); //0101 A-B
digitalWrite(IN2,HIGH);
digitalWrite(IN3,LOW);
digitalWrite(IN4,HIGH);
delay(tps);
digitalWrite(IN1,LOW); //0110 A-B
digitalWrite(IN2,HIGH);
digitalWrite(IN3,HIGH);
digitalWrite(IN4,LOW);
delay(tps);
while(tps>2.5){tps-=0.5;} //pallier à 400Hz
}
}
}

[/code]
```

 **abidi dalenda**
20/04/2015 at 12:18

slt mon projet un suiveur photovoltaïque a l'aide d'un moteur pas a pas unipolaire 12 v 48 pas mon progleme programme arduino d'un moteur pas a pas unipolaire tourne 12h chaque jour , chaque 1 heure 15° (07:00 a 19:00) et mercii stp j'att l reponse

 **Eskimon**
20/04/2015 at 14:41

Et alors ? Ou est le problème ? Ce blog n'est pas là pour faire les projets d'années d'étudiants 😊 Si tu veux te l'aide je te propose d'ouvrir un sujet sur le forum « Systèmes et Matériels » du site Zeste de Savoir et d'y expliquer ce que tu cherches à faire et ce que tu as essayé.

 **abidi dalenda**
20/04/2015 at 17:08

ui je ce mais il y a un defc pour l programma arduino c le probleme comment faire le probleme

 **Eskimon**
21/04/2015 at 07:34

Bon là il va vraiment falloir faire un effort sur l'écriture, je n'ai absolument rien compris.

 **214**
04/05/2015 at 12:21

Bonjour,
Merci pour cet article, j'aime beaucoup ta pédagogie!
Un seul regret, l'absence d'un exemple concret avec dimensionnement électrique mais ce n'est peut-être pas la vocation de l'article.

 **David**
05/05/2015 at 14:42

Bonjour,
Je possède un moteur pas à pas unipolaire et un L293D et j'ai vu sur un blog (<http://electroniqueamateur.blogspot.ca/2013/02/controle-dun-moteur-pas-pas-avec-un.html>) qu'on pouvait utiliser les deux sans problèmes, est-ce vrai ou on est obligé d'utiliser le ULN2003A ?
Et merci pour le tuto.

 **Eskimon**
07/05/2015 at 08:17

C'est probablement possible si l'auteur de ce site y arrive...

 **Didier**
11/05/2015 at 15:53

Bonjour Eskimon,

je suis totalement perdu , je voulais acheter une carte Arduino et un vendeur m'a conseillé une carte : Rumba .l'avantage de cette carte c'est qu'elle a déjà les shields intégrés.(A4988) . mais pas moyen de faire tourner un moteur pas a pas .
je laisse la carte au port USB et je branche ma batterie 12V qui est chargée
Quand je copie des programmes via arduino , faut il changer certaines appellations ?
J'espère avoir été clair , je suis totalement novice la dedans.

Merci

 **Eskimon**
12/05/2015 at 07:12

Je ne connais pas du tout ce type de carte (jamais entendu parler), je crains de ne pas pouvoir être d'une grande aide 😞 .

 **PaTLeC**
12/05/2015 at 17:45

yo
merci pour vos réponses

ma question est celle-ci : une fois le montage réalisé afin d'obtenir des déplacements dans 2 ou 3 axes ; en supposant que sur l'ordinateur se trouve le logiciel le plus adapté pour le modèle : comment se fait la traduction DXF -> Arduino

Peut etre la question d'un candide mais merci de me répondre

 **Eskimon**
13/05/2015 at 07:45

Hello !

DXF ? Pour ce genre de questions je te conseille de poster sur les forums de Zeste de Savoir (rubrique « Systèmes et Matériels ») pour en parler plus en détails (et aussi afin de garder cette section pour les commentaires sur le tutoriel lui-même). Des gens connaîtront peut-être (surement) la réponse !

 **sabrine**
15/05/2015 at 09:16

salut je besoin d'aider ,je recherche un programme arduino d'un moteur pas a pas pour augmenter la vitesse de ce dernier

 **Eskimon**
15/05/2015 at 11:58

Salut. Lit le tuto c'est expliqué. ([speed...](#))

 **Xray59**
20/05/2015 at 16:16

Bonjour,

Tout d'abord super tuto, j'y ai appris pas mal de choses et c'est très clair, même si à mon grand regret je m'intéresse plus à celui à réductance variable :p Je désirerai juste savoir si la commande d'un pâp à réductance variable est possible à l'aide d'arduino et de ULN2003A comme tu l'indiques ci-dessus pour un unipolaire. Même si, comme tu le précises, je devrai trouver un autre code que celui que tu proposes.
J'avance

**Eskimon**

21/05/2015 at 07:27

En non, pour pouvoir le piloter complètement, il faut pouvoir inverser le sens du courant, ce que ne permet pas l'ULN.

**Xray59**

21/05/2015 at 12:36

Argh bon d'accord, je vais me renseigner, merci pour cette réponse aussi rapide 😊
Et une fois de plus super boulot!

**Amal Emely**

02/06/2015 at 20:32

Salut

Merci pour le tuto !

J'ai une petite question cvp
quels sont les différences entre les deux circuits L293D et ULN2003A pour moteur DC et carte arduino uno ????

**Eskimon**

03/06/2015 at 12:48

Le L293 est un pont en H (donc des transistors sont reliés entre eux) alors que le ULN2003 est juste un amplificateur de signal.

**GarennePb**

03/06/2015 at 09:17

Salut à tous ! Je travaille en ce moment sur une caméra de surveillance fonctionnant avec des moteurs pap mais je voudrais les remplacer par des brushless pour mon projet. Mon problème c'est que je voudrais redresser le signal de commande alternatif que je capte en sortie des moteurs, j'ai essayé avec une diode mais cela ne marche pas. Il faudra ensuite que je modifie ce signal en PWM pour faire marcher mes brushless mais ça c'est un autre problème !

Merci d'avance!

**ELOY Didier**

03/06/2015 at 11:51

Bonjour,

Ce n'est pas très clair, il faudrait au moins un schéma avec le type d'alimentation.
De plus si les brushless sont en direct la caméra va faire la toupie !!

Didier (Dijon)

**GarennePb**

03/06/2015 at 17:13

La caméra est alimentée par son cordon. En gros j'essaie juste pour l'instant de récupérer le signal émis par les moteurs. Mais le signal carré que j'obtiens passe dans les négatifs (+4,5V à -7,5V), or je veux le connecter à l'arduino AtMega328 et celui-ci n'accepte que des valeurs comprises entre -0,5V et 5V. Je ne sais pas comment faire pour redresser le signal, j'ai essayé avec une diode mais cela ne marche pas. A terme je remplacerais les moteurs pap par des brushless mais ce n'est pas mon problème majeur.

**Eskimon**

05/06/2015 at 08:05

▲ Dans quel but récupérer le signal émis par les moteurs ? Savoir la rotation faite ? Si oui dans ce cas il faudrait plutôt se tourner vers des composants fait pour cela (plus fiable) comme les encodeurs

 **Meulenberghs**
03/06/2015 at 12:45

Bonjour,

Tout d'abord je trouve que c'est un très bon tuto ! Cependant j'ai je me demandais quelles est l'utilité des deux condensateur se trouvant en // sur l'alimentation continue du moteur bipolaire. Est-ce pour « lisser » le courant ? Pour éviter les chutes de tension ? Et comment déterminer la valeur de ceux-ci ?

Merci d'avance de votre réponse !

 **Eskimon**
05/06/2015 at 08:07

Oui les composants sont là à la fois dans un but de filtrage (car les moteurs vont générer un paquet de parasites) et aussi pour « amortir » les périodes transitoires qui risquerait de trop « tirer » sur l'alimentation tout d'un coup.

 **Lucas**
05/06/2015 at 23:29

Bonjour,

J'ai un petit problème, j'essaye de faire fonctionner mon moteur pas à pas avec un potentiomètre qui me permettrait de faire varier la position de mon moteur (j'utiliserais si possible la fonction de mise à l'échelle -> map). Pour cela, je dispose d'un shield mais ça ne fonctionne pas. A mon avis, j'ai une erreur dans mon programme (StepperTest). Pourriez vous m'aider svp ?

 **Eskimon**
06/06/2015 at 08:20

Pour bien répondre il faudrait un schéma du montage. Histoire de garder cette zone de commentaire pour le tuto en lui-même, je te propose que l'on se retrouve sur les forums de Zeste de Savoir ou tu pourras y créer un sujet expliquant le souci 😊

 **Lucas**
06/06/2015 at 11:30

Oui je comprend alors juste une dernière question : pourquoi l'Arduino ne comprend pas les termes BACKWARD, FORWARD et SINGLE ?

 **Meulenberghs**
07/06/2015 at 12:11

Merci de votre réponse, pourriez-vous m'indiquer comment calculer ses derniers ou bien si ce sont des valeurs qui ne varient pas suivant la puissance des moteurs pas à pas ?

 **Eskimon**
08/06/2015 at 15:37

En fait leur calcul dépend de la fréquence de la PWM, la puissance du moteur (car en fonction du courant utilisé au démarrage etc). Généralement pour le gros condensateur chimique on prend quelques microfarad et pour les plus petits de filtrage quelques dizaines/centaines de nano-farad

 **Lucas**
05/06/2015 at 23:34

Voici mon programme :

```
1 /*  
2 This is a test sketch for the Adafruit assembled Motor Shield for Arduino v2  
3   won't work with v1.x motor shields! Only for the v2's with built in PWM  
4   itrol  
5  
6 For use with the Adafruit Motor Shield v2  
7 ----> http://www.adafruit.com/products/1438  
8 */
```

```

9
10
11 #include
12 #include
13 #include "utility/Adafruit_PWM_Servo_Driver.h"
14
15 // Create the motor shield object with the default I2C address
16 Adafruit_MotorShield AFMS = Adafruit_MotorShield();
17 // Or, create it with a different I2C address (say for stacking)
18 // Adafruit_MotorShield AFMS = Adafruit_MotorShield(0x61);
19
20 // Connect a stepper motor with 200 steps per revolution (1.8 degree)
21 // to motor port #2 (M3 and M4)
22 Adafruit_StepperMotor *myMotor = AFMS.getStepper(48, 2);
23
24
25 void setup() {
26   Serial.begin(9600);           // set up Serial library at 9600 bps
27   Serial.println("Stepper test!");
28
29   AFMS.begin(); // create with the default frequency 1.6KHz
30   //AFMS.begin(1000); // OR with a different frequency, say 1KHz
31
32   myMotor->setSpeed(5); // 10 rpm
33 }
34
35 void loop() {
36   Serial.println("Single coil steps");
37   myMotor->step(48, FORWARD, SINGLE);
38   myMotor->step(48, BACKWARD, SINGLE);
39 }

```

 Phillippe

07/06/2015 at 16:12

Bonjour je réalise un projet dans lequel je dois faire tourner un plateau avec un moteur pas a pas et un engrenage à contact intérieur (pignon moteur + couronne). Je dois avoir une vitesse rotation du plateau précise, avec le rapport de transmission j'en déduis la vitesse de rotation du moteur en tr/min. je reprend le même codage que vous j'ai donc la vitesse de rotation. Si je veux faire tourner le plateau de x ° comment dois faire pour trouver le nombre de pas, dois je utiliser le nombre de pas par minute ?

 Sullivan Serfati

09/06/2015 at 10:06

Bonjour esquimon :

nous sommes une classe de sti2d et nous faisons une imprimante 3d a notre sauce en utilisant une RAMPS on as un probleme avec le programme arduino car on y arrive pas peut tu nous montrer un modele qui fonctionnerai stp ?
ps : le moteur c'est un bipolaire hein 200 pas

 Eskimon

15/06/2015 at 09:28

Pour bien répondre il faudrait un schéma du montage. Histoire de garder cette zone de commentaire pour le tuto en lui-même, je te propose que l'on se retrouve sur les forums de Zeste de Savoir ou tu pourras y créer un sujet expliquant le souci 😊

 Anne

10/06/2015 at 16:27

bonjour

J'ai une petite commande numérique Charlyrobot dont le moteur des Y vient de me lâcher et ces moteurs ne se font plus
Comment puis-je trouver l'équivalent sachant que l'étiquette signalétique est TYPE 23KM CO.33 18V
N°T2Y 18.01 MINEBEA
Patrick

 Ons

22/06/2015 at 10:03

salut Eskimo

j'ai un moteur pas à pas bipolaire 4 fils , arduino uno et l289n.
je veux les connecter avec un potentiomètre qui varie la vitesse de moteur et un interrupteur qui guider le sens du moteur et à gauche , c pour ça
↑ e programme et je veux que vous le corriger, Bientôt.

#include

```
const int moteur = 200;

// pour un moteur de 200 pas par tour et brancher sur les broches 2, 3, 4, 5, 6
Stepper mymoteur(moteur, 2, 3, 4, 6);
const int bouton = 9;
int etatBouton;// variable qui enregistre l'état du bouton
const char potar = 0; // une entrée analogique (A0) pour régler la vitesse manuellement

void arret(int a) //déclare une procédure qui s'appelle « arret »
{
delay(a);
}
// la PWM pour la vitesse
const int enable = 10;

void marche(int a)
{ analogWrite(enable,a);
digitalWrite(enable,HIGH);
}

void setup()
{
pinMode(bouton, INPUT);
etatBouton = HIGH; // on initialise l'état du bouton comme « relâché »
Serial.begin(115200); // la vitesse définie par le potentiomètre
}

void loop()
{
while(1)
{
// on lit la valeur du potentiomètre
int vitesse = analogRead(potar);
// division de la valeur lue par 4
vitesse /= 4;
// envoie la nouvelle vitesse sur le moteur
analogWrite(enable, vitesse);
// on affiche la vitesse sur le moniteur série
Serial.println(vitesse);
delay(50);

{etatBouton = digitalRead(bouton); // Rappel : bouton = 9
int valinter= 0; // on crée une variable valinter pour lire l'état de l'interrupteur
valinter= digitalRead(bouton); // on lit la valeur de l'interrupteur (LOW ou HIGH)
if (etatBouton == HIGH) // test si le bouton a un niveau logique HAUT
delay(2000);
{
marche(255);
delay(5000);
arret(5000); // arret pendant 5 secondes
}
(etatBouton == LOW); // test si le bouton a un niveau logique différent de HAUT (donc BAS)
delay(2000);
{
marche(-255);
delay(5000);
arret(5000); // arret pendant 5 secondes
}
}
}
}
```

 **Nitch**
30/06/2015 at 12:52

Bonjour eskimon,
Premièrement félicitation pour ton site!

Je voudrais savoir s'il est possible de commander deux moteurs pas à pas séparément avec deux motor shield arduino empilés?

J'ai réussi à les faire tourner mais que de la même manière.

Merci d'avance

Rommon

**Eskimon**

01/07/2015 at 11:56

Il faudra passer par un shield dédié à ce genre de chose. En effet, en empilant deux fois le même shield tu utiliseras deux fois les mêmes broches et aura donc deux fois le même comportement 😊

**PHaroZ**

08/07/2015 at 07:36

merci Eskimon pour ton travail sur ces tutos !

Pour ceux qui auraient un pb à faire tourner dans les 2 sens un 28BYJ sachez que certains modèles ont un brochage erroné, c'était le cas sur le mien.

Au lieu de déclarer

Stepper moteur(64, 8, 9, 10, 11);

j'ai du faire

Stepper moteur(64, 8, 10, 6, 11);

**Eskimon**

08/07/2015 at 14:04

Merci de l'info 😊

**Didier (Dijon)**

09/07/2015 at 10:55

J'avais le même problème, c'est la solution à celui-ci Merci

**Badjojo**

20/07/2015 at 10:56

Bonjour Eskimon, je travaille actuellement sur le contrôle de moteur pas à pas à l'aide d'une carte ARduino Mega v2 en passant par Xcos-Scilab. L'étape connexion et communication est bonne, c'est pour la suite que j'ai un peu plus de problèmes. As tu déjà effectué des travaux sur ce sujet ?

**Eskimon**

20/07/2015 at 18:07

Je ne connais pas du tout Xcos-Scilab malheureusement 😞

**ons**

27/07/2015 at 14:25

salut,

je veux varier le sens d'un moteur pas à pas avec ce programme:

#include

AccelStepper MonStepper ;

```
void setup () {  
  MonStepper.setSpeed(50);  
}  
void loop() {  
  MonStepper.runSpeed();  
}
```

 **mario**
29/07/2015 at 15:09

Hello

Est ce que je peux vous demander si vous connaissez un petit moteur\actionneur (alimenté en 3V max) et qui proposerait que 2 positions ? C'est pour utiliser dans une serrure connectée...

Super merci

 **Eskimon**
01/08/2015 at 17:57

Un petit servomoteur devrait pouvoir faire l'affaire

 **Micka**
12/09/2015 at 18:22

Salut eskimon.

Tes explications m'ont l'air très clair (je suis totalement novice pour le moment).

Mais je t'avoue que je bloque un peu sur le choix du matos.

Je voudrais faire un PAP pour mon telescope (focuser).

Mais je ne sais pas quelle cartes et moteur prendre quand je suis sur aliexpress.

Tu pourrais m'orienté pour tout le matos qu'il faut stp.

Si je me trompe pas il me faut :

Arduino
une carte shield
moteur
courroie
et surment résistance et led.

Je ne sais pas si c'est possible mais je souhaite avoir la possibilité de le piloter a la fois avec l'ordi (lors des sortie astrophoto) mais aussi avec une télécommande.

Pour l'alimentation je posséde un power tank qui peu me délivré ceci : 3V/1A, 6V/1A, 9V/1A.

Je voudrais faire un PAP vraiment sensible avec plusieurs vitesse de rotation.

Comme je t'ai dis je suis totalement novice mais je pense que si je sais quel matos prendre avec un tuto comme le tiens je devrais y arrivé.

Merci d'avance

Micka

Pingback: Arduino | Pearltrees

Pingback: Taxi volant | Pearltrees

 **Bat**
22/09/2015 at 12:49

Bonjour, je ne sais pas si le topic est toujours d'actualité, mais je rencontre un problème assez frustant : J'utilise une L293D pour contrôler un stepper Bipolaire/ Je me suis basé sur ce schéma ci : http://lh6.ggpht.com/_7GPjaQGzbM/TFohE0GIU1I/AAAAAAAAC8/zCXT1frAr6c/L293D_steppermotor_Steckplatine.jpg

Cela fonctionne lorsque je ne branche pas d'alimentation externe, mais le moteur n'est pas assez puissant. Lorsque j'ai branché le 12v, j'ai fais planter mon PC sans dommages heureusement (ce qui fait flipper quand même), mais j'ai aussi bousillé ma carte arduino, qui freeze depuis sur la Ir ↗ et qui ne peut ni transmettre ni recevoir d'information.

Est ce que votre schéma de moteur bipolaire avec une L293D en H bridge marche correctement sur 12v ? Comment puis-je protéger ma (nouvelle) carte arduino du 12v dont ont besoins mes moteurs ?

cordialement

**Eskimon**

22/09/2015 at 16:26

Tu es sur que tu ne t'es pas trompé dans un branchement ? Car à priori le schéma me semble correcte.

Pingback: Actuateurs | Pearltrees

**alain**

08/10/2015 at 18:16

Bonjour

Le circuit LV8548MC permet de commander facilement un moteur pas à pas, je l'ai testé ici
<http://raspbuino.fr/?cat=32>

Pingback: Divers - Arduino | Pearltrees

**Emile Caubet**

25/10/2015 at 15:13

Bonjour,

J'ai essayé d'utiliser ton programme sur un moteur bipolaire mais il tourne dans le même sens que le nombre de pas dans la fonction step() soit positif ou négatif.

Si tu peux m'aider merci beaucoup 😊

**Mauvage**

25/10/2015 at 18:22

Re

connais tu les caractéristiques du moteur:.

nbre de pas par tour ou rpm

tension d'alim

Moi j'ai utilisé un 28BYJ-48 en 5v

2048 rpm

**Emile Caubet**

25/10/2015 at 19:02

C'est un moteur de 200 pas, alimenté en 12V. J'ai réussi à résoudre le problème grâce à la remarque de Dyole. J'ai appelé le moteur comme ceci : Stepper pap1 = Stepper(200, 2, 4, 3, 5);

au lieu de comme cela : Stepper pap1 = Stepper(200, 2, 3, 4, 5);

Sans changer les branchements sur le driver.

**Mauvage**

25/10/2015 at 20:11

Yes, bravo

A une prochaine
et bonne expérimentation

**Denis**

27/10/2015 at 23:46

Bonjour,

Je m'appelle Denis et j'ai 67 ans, c'est dire si c'est difficile pour moi.
J'ai désossé un disque dur et ai sorti le moteur. Il a 4 entrées. Il est donné pour 7200tpm.
je pense que c'est un pap
puis-je le piloter avec une Arduino UNO et un L298N que j'ai.
Bravo pour ton tuto

 **Eskimon**
29/10/2015 at 15:10

D'après quelques recherches on se rapproche plutôt d'un moteur brushless particulier (dernier type de moteur dont je n'ai pas parlé car plus difficile à piloter). D'autres infos par ici par exemple : <http://forum.arduino.cc/index.php?topic=33284.0>

 **hcp7kue**
03/11/2015 at 21:42

Bonjour,
J'ai récemment commencé à utiliser une carte arduino avec un moteur pas à pas.
Je voudrais qu'il exécute un mouvement de huit pas dans un sens, et un mouvement de 8 pas dans l'autre sens. J'ai été sur divers forums pour voir comment faire ce mouvement, et j'ai correctement réussi à faire tourner mon moteur. Le seul problème : mon moteur exécute un mouvement dans un sens, mais lorsqu'il s'agit de le faire tourner dans l'autre sens, rien ne marche.
Merci pour votre aide,
hcp7kuz

 **Jean Pierre Gelabert**
08/11/2015 at 14:09

Bonjour les bricoleurs,
Je me présente, Jean-Pierre, passionné par la construction de télescopes (ma dernière fabrication : <https://youtu.be/Qlol0lxBZ4>) que je souhaiterai motoriser. Ce type de télescope a une monture dite « Alt-Azimutale ». La plupart des télescopes motorisés ont une monture Équatoriale dont l'axe horaire (l'axe qui pointe vers l'étoile polaire) facilite le suivi stellaire en un seul mouvement de compensation de la Terre de 15 ° par heure. De fait il existe des tables des coordonnées de tous les objets du ciel observable.
Dans mon cas, établir une table est impossible, elle changerait à chaque seconde du fait que ma monture est à l'horizontale au sol et ne pointe pas vers le Nord vrai.
Donc j'ai déjà effectué la partie logicielle sur l'IDE Code:Blocks de la conversion de coordonnées équatoriales en Alt-Azimutale mais je pose quelques questions :

- cela va-t-il être compatible avec l'IDE d'arduino ? J'utilise un main() fichier .cpp puis des classes dans un fichier .h, j'ai besoin des bibliothèques classiques (bien que sur le terrain j'envisage d'utiliser un mini portable sous linux).
- existe-t-il avec un module arduino la possibilité de récupérer ma position GPS ?
- j'ai entendu parler de « STINO » ? (<https://tutoarduino.com/tutoriel-un-meilleur-ide-pour-larduino/>) en quoi serait-il meilleur que l'officiel ?
- Me conseillerez-vous plutôt un Raspberry Pi pour cette application ?

Merci par avance JP

PS : si quelqu'un est intéressé pour obtenir la classe permettant de convertir une coordonnée équatoriale en alt-azimutale je peux la fournir, elle est précise jusqu'au centième de seconde d'arc pour les 2000 ans à venir (elle ne permet pas encore de trouver les planètes de notre système solaire, ni les comètes)

 **Eskimon**
10/11/2015 at 12:04

Salut !

En voilà un projet intéressant !!

À ce qui est de Stino, c'est un plugin de l'IDE SublimeText. C'est intéressant pour ceux qui en ont marre d'utiliser l'officiel qui a quelques lacunes. J'avais fait un article à ce sujet il y a un an et demi : <http://eskimon.fr/2224-arduino-mini-tuto-utiliser-sublime-text-ide>

Pour ta classe, si tu arrives à la rendre indépendante de bibliothèques système et autre trucs trop lourd alors ça peut passer. Le plus simple reste

de créer un projet et d'essayer de le compiler ^^

Pour le GPS il en existe pas mal ouai. google et les sites chinois de matériel (dx, bangood, etc) devraient pouvoir t'aider.

Enfin, pour le choix de passer à une raspberryPi ca dépendra pas mal de tes compétences. Si tu es à l'aise avec un ordinateur sous linux et ce genre de chose ca peut se tenter, mais ca me semble ps nécessaire.

 **barracudaNoir**
22/11/2015 at 09:22

Salut a tous,
Je voudrais savoir si les moteurs capstan sont-ils aussi de la famille des moteurs pas a pas?
Est-ce qu'ils peuvent etre controler comme les PaP?
merci

 **Pascal Roduit**
06/12/2015 at 04:53

J'ai monté dans mon jardin une serre et j'aimerai ouvrir et fermer un clapet avec un moteur pap, commandé par un thermostat

 **Pascal Roduit**
06/12/2015 at 04:55

comment faire ?

 **Lahcen**
27/12/2015 at 22:32

Bonjour Eskimon

Très bon document. Je comprends mieux le fonctionnement des moteurs pas à pas, cependant une question me turlupine encore.

La question est la suivante.

Un moteur bipolaire pourrait il voir son sens de rotation s'inverser s'il rencontre une contrainte trop importante dans son sens initial (et que la contrainte soit moins importante dans le sens contraire) en considérant que son champ tournant se retrouverait suffisamment déphasé pour « correspondre » à un champ tournant allant dans l'autre sens de rotation ... ?

...

.....

Waouahhh, je me rend compte que si j'avais pas posé la question, je me comprendrais pas moi même !!

J'espère que j'ai été assez clair ?

Merci et bonne journée

 **Eskimon**
11/01/2016 at 22:34

Je pense te comprendre 😊

Mécaniquement, rien n'empêche effectivement une charge d'entraîner le moteur. Il y a alors un risque de « sauter des pas », le moteur ayant tendance à s'aligner avec les champs que l'on génère.

 **Gh**
18/01/2016 at 22:09

Bonjour,

j'ai récupéré un moteur pas à pas dans un lecteur DVD, c'est le moteur qui permet de déplacer la lentille de lecture sous le CD.J'aimerais savoir s'il est possible de controler ce moteur avec arduino sans shield ni driver, uniquement avec la carte arduino et un programme qui ressemblerait a cela

↑
,p()

{

```
//pas 1
```

```
digitalWrite(a1,HIGH);
```

```
digitalWrite(a2,LOW);
```

```
digitalWrite(b1,LOW);
```

```
digitalWrite(b2,LOW);
```

```
delay(200);
```

```
//pas 2
```

```
digitalWrite(a1,LOW);
```

```
digitalWrite(a2,LOW);
```

```
digitalWrite(b1,LOW);
```

```
digitalWrite(b2,HIGH);
```

```
delay(200);
```

```
//pas 3
```

```
digitalWrite(a1,LOW);
```

```
digitalWrite(a2,HIGH);
```

```
digitalWrite(b1,LOW);
```

```
digitalWrite(b2,LOW);
```

```
delay(200);
```

```
//pas 4
```

```
digitalWrite(a1,LOW);
```

```
digitalWrite(a2,LOW);
```

```
digitalWrite(b1,HIGH);
```

```
digitalWrite(b2,LOW);
```

```
delay(200);
```

```
}
```



Eskimon

19/01/2016 at 22:23

J'aurais tendance à le déconseiller. Les moteurs demandent du courant pour pouvoir se déplacer, et l'Arduino seule n'est pas apte à délivrer ce courant. Au mieux le moteur ne bougera pas, au pire tu crameras la broche utilisée, et au vraiment pire le microcontrôleur de l'Arduino sera foutue aussi. Il est possible de faire un pont en H avec quelques transistors, mais devant le prix des shields sur certains services chinois comme dx ou bangood c'est dommage de se priver.



mounir

24/01/2016 at 00:35

↑ ↗ `aliser des angles bien priser quel moteur je peut l utiliser ??? 😊



Cosm

27/01/2016 at 18:24

Bonjour Eskimon!

Tout d'abord je tenais à te remercier pour ce très bon document. Je suis en première S et comme tout les premières je dois faire un TPE. Le sujet de mon TPE porte sur les moteurs pas à pas dans l'imprimante 3D.

Ce superbe document m'a permis d'avancer dans mon TPE car je n'y comprenais pas grand chose...Malheureusement il y a une chose que je ne comprends pas bien, qui pourtant doit être assez facile à comprendre – Je me demande comment un moteur pas à pas hybride peut faire 200 pas par tours alors que sur ton exemple le moteur fait seulement 24 pas pour 8 bobines et 6 branches...J'ai entendu parler de rotor avec des dents, mais je ne comprends pas très bien...Si tu pouvais éclairer un jeune lycéen égaré je t'en serais reconnaissant ^^

Merci et bonne soirée



Eskimon

27/01/2016 at 19:02

Il y a peut-être un réducteur *5 ? (voir le chapitre sur les moteurs à courant continu, j'y parle des réducteurs).



Cosm

27/01/2016 at 19:32

Merci pour m'avoir répondu rapidement. J'ai lu ton texte sur les réducteurs, mais honnêtement je doute qu'il y est un rapport avec le problème des 200 pas. Sur ce PDF: <http://oseguin.free.fr/cnc/paps.pdf> on peut voir le schéma du moteur hybride avec des dents, et si j'ai bien compris elles sont chacune aimantées. Il y a aussi une histoire de deux rotor liés par un aimant permanent ou quelque chose comme ça. en tout cas merci pour ta réponse, je crois avoir compris à peu près par moi même ^^ Merci et bonne soirée 😊



Riadh

17/02/2016 at 23:43

je fait un projet de fin etude et mon projet « est realisation une carte elecronicque qui permet de varier l'angle d'inclinaison de paneau » et pour cela il faut commander un moteur avec verin pour faire varier l'angle de paneau une seule fois chaque par exemple en mars l'angle 20° ET EN JUIN 35° ET EN SEP 47° ET SVP Aider moi pour trouver le solution avec pic ou arduino



Sehli

25/02/2016 at 16:18

Je suis une élève ingénieur en terminal et je suis en train de faire mon projet fin étude » pilotage d'un banc de test universel en vu de réaliser un testeur à sondes mobiles pour un circuit imprimé ». Je vais commencer par une seule sonde dès le début.

J'ai un fichier python qui contient la liste des composants électroniques passives à tester ainsi leur positions sur le circuit imprimé et je voudrai commander 3 moteurs pas à pas (un moteur pour chaque axe) en utilisant arduino. Je ne sais pas comment exploiter les coordonnées des composants pour commander les moteurs et faire bouger la sonde à fin de tester les composants.

Je serai reconnaissante si vous pourriez m'aider et merci 😊



Lemercier clémentine

29/02/2016 at 08:54

Bonjour,

Nous sommes en Terminale, et avons un PPE à faire pour la fin de l'année.

Nous avons choisis de créer un dressing automatisé, pour cela nous devons commander 2 moteurs pas à pas.

Nous avons une carte arduino sparkfun et 2 cartes arduino motor shield.

D'abord le premier moteur doit tourner dans un sens (au total nous devons avoir 8 positions), là le deuxième moteur doit translater, puis revenir à sa position (en gros avancer/reculer). Ensuite, le premier moteur, reviens à sa position dans le sens inverse.

Nous arrivons à faire fonctionner chaque moteur individuellement mais cependant nous n'arrivons pas à les faire les 2 réunis...

Voici un programme (qui ne marche pas encore) :

```
int directionA=12 ;  
int directionB=13 ;  
int speedA=3 ;  
int SpeedB=11 ;
```

```
int Direction=6;
int DIRECTION=10;
int Vitesse=5;
int VITESSE=9;

void setup()
{
pinMode (directionA,OUTPUT);
pinMode (directionB,OUTPUT);
pinMode (SpeedA,OUTPUT);
pinMode (SpeedB,OUTPUT);

pinMode (Direction,OUTPUT);
pinMode (DIRECTION,OUTPUT);
pinMode (Vitesse,OUTPUT);
pinMode (VITESSE,OUTPUT);
}

void loop()
{
tourner();
avancer();
revenir();
}

void tourner(int pindirectionA, int pindirectionB, int pinSpeedA, int pinSpeedB)
{
digitalWrite (3,HIGH);
digitalWrite (11,HIGH);
digitalWrite (12,LOW);
digitalWrite (13,LOW);

delay(100);

digitalWrite (12,HIGH);
digitalWrite (13,LOW);

delay(100);

digitalWrite (12,HIGH);
digitalWrite (13,HIGH);

delay(100);

digitalWrite (12,LOW);
digitalWrite (13,HIGH);

delay(500);
}

void avancer(int pinVitesse, int pinVITESSE,int pinDirection, int pinDIRECTION)
{
digitalWrite (5,HIGH);
digitalWrite (9,HIGH);
digitalWrite (6,LOW);
digitalWrite (10,LOW);

delay(100);

digitalWrite (6,HIGH);
digitalWrite (10,LOW);

delay(100);

^
digitalWrite (6,HIGH);
digitalWrite (10,HIGH);
```

```

delay(100);

digitalWrite (6,LOW);
digitalWrite (10,HIGH);

delay(500);
}

void revenir (int pindirectionA, int pindirectionB, int pinSpeedA, int pinSpeedB)
{
digitalWrite (3,HIGH);
digitalWrite (11,HIGH);
digitalWrite (12,LOW);
digitalWrite (13,LOW);

delay(100);

digitalWrite (12,LOW);
digitalWrite (13,HIGH);

delay(100);

digitalWrite (12,HIGH);
digitalWrite (13,HIGH);

delay(100);

digitalWrite (12,HIGH);
digitalWrite (13,LOW);

delay(100);
}

```

On pense ne pas pouvoir compiler une carte en mettant l'autre de côté, mais là, on ne sait pas comment réaliser le câblage.

De plus, les arguments dans les fonctions ne semblent pas être correctes, mais nous n'arrivons pas à les améliorer.

Aussi, sur le côté de la carte motor shield compilé (actuellement donc peut être à modifier), nous avons les pin 3 et 5 pris par le Bluetooth (qui nous servira pour notre future application).

Peut-être n'y a-t-il pas assez d'entrée pour tout fonctionner ?



Yann

02/03/2016 at 19:02

bonjour et félicitation pour ce tuto qui est vraiment super bien fait.

Je suis en train de faire un projet pour le lycée avec un collègue. Nous avons décidé de faire une boîte pour mesurer les pieds. Pour cela nous utilisons un moteur PAP(48), un ULN2003A, une carte UNO, un LCD 2x16, une barrière IR et un inter. La séquence pour la mesure se fait en 6 étapes:

-L'écran demande de « mettre son pied »

-On met son pied, on passe l'inter de LOW à HIGH

-LCD « mesure en cours » s'affiche et arrêt du moteur quand IR est coupé

-L'écran affiche la pointure et le moteur reste dans sa position

-On enlève le pied

-On change l'état de l'inter et le moteur revient à zéro l'écran affiche « reset » puis « mettez votre pied ».

Nous avons 3 petits soucis que nous n'arrivons pas à résoudre:

-Lors de la mesure, le moteur a un temps long entre chaque tour (sup à delay (10)), puis au bout de quelques tours continu dans le bon timing

-L'arrêt se fait bien quand on coupe l'IR, le LCD affiche la pointure, mais si on libère l'IR il continu à reprendre la mesure, et ne reste pas dans sa position.

-Et enfin quand on repasse sur reset, le moteur revient à zéro, donc le compteur fonctionne, mais le moteur chauffe, car les bobines restent alimentées.

Pouvez-vous me donner un petit coup de main

Merci d'avance

Je mets ci-dessous le programme.

```

^
.e infrarouge avec Arduino
LED IR sur D3 (géré par bibliothèque IRremote) + TSOP32138 -> D2*/
#include

```

```
//Inclure la librairie stepper.h
#include

#include << LiquidCrystal.h >> //ajout de la librairie
//Vérifier les broches !

//constante avec broches

//detecteur IR
#define CAPTEUR 2 // récepteur IR -> D2
#define LED_STATUT 3 // émetteur IR -> D3
IRsend irsend; // déclaration de l'objet émetteur (pin D3 obligatoire)

//pour un moteur de 48 pas par tour et brancher sur les broches 4 rose , 5 marron, 6 jaune, 7 bleu
Stepper moteur(48, 4, 5, 6, 7);

//Lcd config
LiquidCrystal lcd(13,12,11,10,9,8); //liaison 4 bits de données

//le bouton est connecté à la broche A0 de la carte Adruino
const int bouton = A0;

//variable qui enregistre l'état du bouton
int etatBouton;

/*Declarer compteur
int compteur;

/* Declarer Pointure
String pointure; /* Tester avec CHAR ?

String messagePointure; /*Tester avec CHAR ?

/*Declarer etatMoteur
int mesureEnCours;

void setup()
{
pinMode(CAPTEUR, INPUT);
pinMode(LED_STATUT, OUTPUT);

irsend.enableIROut(38); // crée une sortie modulée à 38KHz
Serial.begin(9600);

moteur.setSpeed(150); //150 tours par minute
//(rappel : ici le moteur fait 48 pas par tour, on fera donc 7200 pas par minute)

lcd.begin(16,2); //utilisation d'un écran 16 colonnes et 2 lignes
lcd.write(<< Bonjour Mettez votre pied >>); //petit test pour vérifier que tout marche

pinMode(bouton, INPUT); //le bouton est une entrée
etatBouton = HIGH; //on initialise l'état du bouton comme « relaché »

compteur = 0; //On part de 0
mesureEnCours = 0; //Moteur à l'arrêt
}

void stop() //Fonction pour stopper le moteur
{
moteur.step(0);
}

void loop()
{
  ↑
  irsend.mark(0); // envoi du signal pour une durée illimitée

  boolean mesure1 = digitalRead(CAPTEUR); // première mesure
```



```
lcd.clear();
delay(200);
lcd.write("RESET");
delay(200);

compteur = -compteur;

Serial.println("Retour compteur = " + compteur);

moteur.step(compteur); // remise a zero voir si compteur ok

delay(120000); //temps retour 0 (2min)
lcd.clear();
delay(10);
lcd.write("METTEZ VOTRE PIED");

}

}

}

//Fin du loop
```

 Yann
02/03/2016 at 19:08

 djo
14/03/2016 at 08:55

Bonjour,
étant débutant en programmation, j'aimerais savoir si l'utilisation d'un shield est recommandé, ou nécessaire, pour un branchement avec un moteur bipolaire ?

 med amine
19/03/2016 at 02:08

bonjour,
je suis un etudiant en génie mécanique et j'ai un petit problème avec la partie programmation
j'arrive pas a réaliser un programme qui me permet de lire le nombre de tours d'un moteur thermique(a travers un capteur disons) et envoyer ce nombre a une carte arduino uno qui le convertit en nombre de tours d'un moteur pas a pas lié au conduite d'admission (c'est a dire par exemple pour un nombre de tours du moteur thermique égale à 1000 tr/mn le moteur pas a pas doit faire 100 tours et s'arrêter..pour 1500 tr/mn du moteur thermique, le moteur pas a pas doit faire 50 tours..ensuite lorsque le nombre de tours du moteur thermique baisse a 1300 tr/mn le moteur pas a pas doit faire 20 tours au sens inverse de celui précédent !)
s'il vous plaît comment procéder sachant que je suis débutant dans ce domaine et merci d'avance

 **Ibrahim**
24/04/2016 at 01:42

bonjour,
je suis un etudiant en génie mécanique J'ai un projet de commande d'une machine de marquage sur métal avec l'arduino et je trouve un problème du choix de cartes de puissance qui permet de commander deux moteurs pas à pas (5,1V 1A) et le poinçon (12V 6A) de marquage. le programme que je utiliser son grbl

 E.Farid
08/08/2016 at 20:32

Bonjours
En cherchant je viens de decouvrir ce poste qui est tres instructif et bien expliquer meme pour un neophyte comme moi. merci de ce poste.

