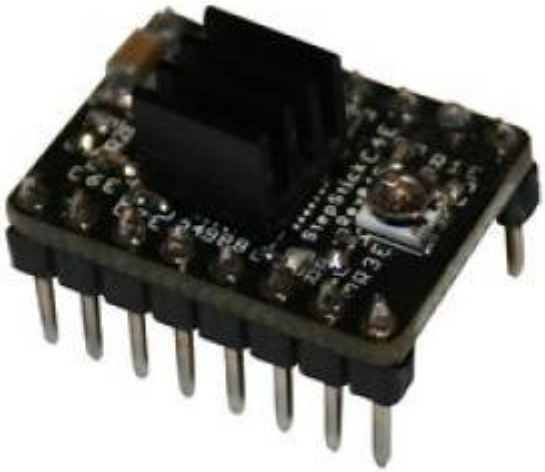
A4998 Doc

Doc A9488 MCHobby

Les moteurs pas à pas Nema ...

Présentation

Cette carte breakout supporte une pilote Microstepping Allegro A4988 (DMOS) incluant une protection contre les sur-courants (overcurrent). C'est le contrôleur de moteur pas-à-pas que l'on retrouve sur les cartes de commande des imprimantes 3D comme OrdBot Hadron.



Ce contrôleur permet de contrôler des moteurs pas-à-pas bipolaires en micro-stepping avec un maximum de 2 ampères par bobine (attention aux conditions de dissipation de chaleur!!!).

Voici quelques caractéristiques clés de ce breakout:

- Interface de contrôle de pas et de direction simplissime.
- 5 résolutions différentes de pas:
- * full-step : Pas complet
- * half-step : 1/2 pas
- * quarter-step : 1/4 de pas
- * eighth-step : 1/8 ième de pas
- * sixteenth-step : 1/16 ième de pas
- Contrôle en courant ajustable - un potentiomètre permet de fixer le courant maximum à l'aide d'un potentiomètre. Cela permet d'utiliser une tension supérieure à celle recommandée par le moteur (puisque le courant est limité) et permet ainsi d'atteindre un débit de pas plus élevé (et donc une vitesse plus élevée)
- Une commande de hachage intelligente qui sélectionne automatiquement le bon mode de

décroissance de courant (fast decay = baisse/chute rapide du courant OU slow decay = chute lente)
- Protection (arrêt) en cas de surchauffe, sous tension, surcharge/sur-courant/court-circuit. Utile en cas de surcharge/blocage moteur (car le courant va monté en flèche, ce qui doit provoquer la mise en protection).

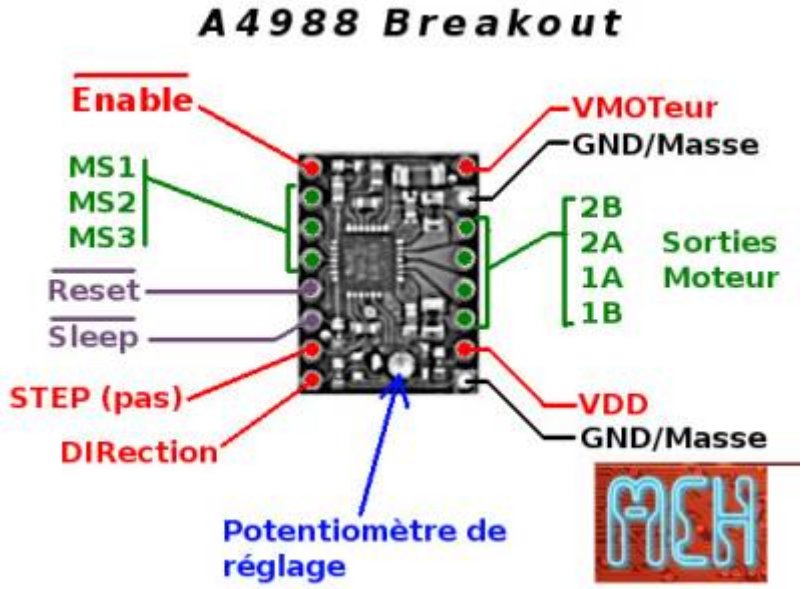
Détail techniques

Pensez à prendre connaissance de la fiche technique du A4988. En effet, s'il est capable de piloter un courant important, cela ne se fait pas dans n'importe quel condition, il fait être attentif à la dissipation de chaleur.

- Tension de fonctionnement minimum : 8 V
- Tension maximale de fonctionnement : 35 V
- Courant par phase (en continu) : 1,2 Amp
- Courant maximum par phase : 2.3 Amp
- Logique de contrôle
- * Tension minimale: 3 V
- * Tension Maximale : 5.5
- Résolution Microstep: Pas complet, 1/2, 1/4, 1/8 et 1/16 de pas
- Protection contre inversion de tension: NON

Fiche technique du A4988

Brochage



- `'Enable'`: Logique Inversée, permet d'activer ou désactivé le moteur. Etat Haut=`High`=Moteur actif... et axe bloqué entre les pas. Etat bas=`Low`=Axe totalement libre
- `'MS1, MS2, MS3'`: permet de sélectionner la configuration Step/MicroStep. Ces broches disposent de résistances Pull-Down ramenant le potentiel à 0v lorsque rien n'est connecté sur ces broches. Voir la section ci-dessous pour plus d'information.

- 'Reset': Logique inversée. Permet de faire une réinitialisation du module. Généralement connecté sur la broche “sleep”.
- 'Sleep': Logique inversée. Généralement connecté sur la broche “Reset” du module.
- 'Step': Envoyer un signal d'horloge (Niveau Haut puis Niveau bas, High puis Low) pour avancer le moteur d'un pas.
- 'DIR': Permet d'indiquer la direction de rotation du moteur. Etat Haut=High pour tourner dans un sens, Etat bas='Low' pour tourner dans l'autre sens.
- 'VMot': Tension d'alimentation du moteur. Habituellement 12V pour les moteurs pas à pas. Tension entre 8 et 12v.
- 'GND': Sous “VMOT”, masse pour l'alimentation moteur. Habituellement mise en commun avec la masse de logique de commande (celle sous “VDD”).
- '2B 2A': Première bobine du moteur pas à pas bipolaire (voir exemple ci-dessous)
- '1A 1B': Deuxième bobine du moteur pas à pas bipolaire (voir exemple ci-dessous)
- 'VDD': Alimentation de la logique de commande entre 3 et 5.5v. Habituellement 5V.
- 'GND': Sous “VDD”, masse de la logique de commande. Souvent mise en commun avec la masse d'alimentation du moteur.

Entrées de contrôle

Chaque impulsion sur l'entrée STEP correspond à un microstep du moteur pas-à-pas dans la direction sélectionnée sur la broche DIR.

Notez que les broches STEP et DIR ne disposent pas de résistances pull-down ou pull-up interne. Vous devez contrôler les deux états Haut (High) et bas (Low) avec votre application. La tension sur ces broches ne peut pas être flottante. Ce qui n'est pas un problème pour les plateformes Arduino

Si vous ne devez faire tourner le moteur que dans un seul sens, vous pouvez placer la broche DIR directement sur VCC ou GND. Le circuit dispose de 3 entrées différentes pour contôler son étage de puissance: RST (Reset), SLP (sleep) et EN (enabled). Voyez la fiche technique pour plus d'information sur ces états.

'Notez que la broche RST (reset) est flottante'; Si vous ne comptez pas utiliser cette broche, vous pouvez la connecter sur la broche SLP (sleep) adjacente pour placer RST au niveau logique haut (high) et activer la carte.

Configuration Step/MicroStepping

La configuration se fait à l'aide des broches MS1, MS2 et MS3.

Les moteurs pas-à-pas dispose de leur propre spécification physique de “pas” (step specification en anglais) connu comme un “pas complet” (full step en anglais). Un moteur 1.8° ou 200 pas par révolution fait parti des moteurs les plus répandus. Un pilote 'microstepping' tel que le A4988 permet d'obtenir une plus grande résolution en autorisant des positions intermédiaires dans un pas. Cela est rendu possible en modulant intelligemment la quantité de courant dans les bobines du moteur pas-à-pas. Par exemple, piloter un moteur en mode “1/4 de pas” permet d'obtenir 800 microsteps (micro-pas) sur un moteur prévu pour 200 pas ar révolution et cela en utilisant 4 niveau de courants différents pour chacun des microsteps.

La résolution (`la taille du pas`) est sélectionné à l'aide des entrées MS1, MS2 et MS3. Grâce à eux, vous disposez de 5 résolutions différentes reprisent dans la table ci-dessous. * MS1 et MS3 dispose d'une résistance pull-down interne de 100kΩ.
Une résistance pull-down signifie que `'si'` vous ne placer pas le potentiel de ces broches au niveau logique haut=`high`=VDD (la tension choisie pour la la logique de commande) `'alors'` elle seront automatiquement ramenée au niveau logique bas=Low. * MS2 dispose d'une résistance pull-down de 50kΩ.

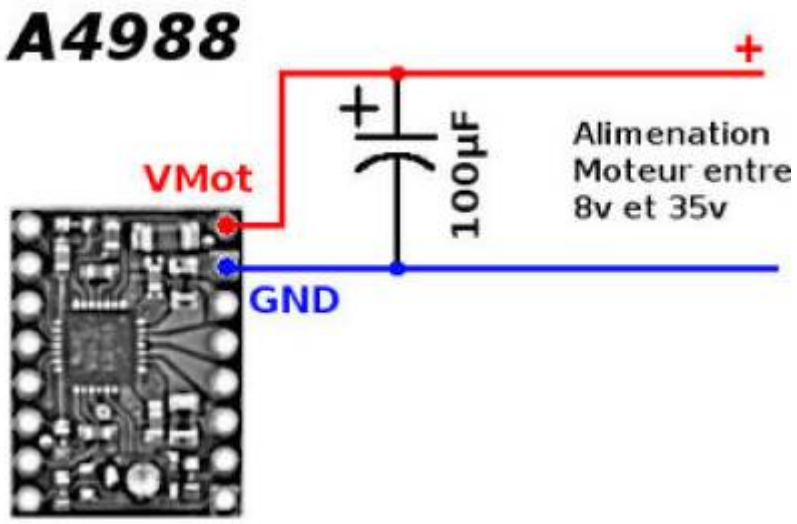
Si vous ne raccordez aucune de ces broches MS1, MS2 et MS3 alors votre breakout A4988 fonctionnera en mode “pas complet” (`full step`). Pour que le microstepping fonctionne correctement, il faut que la limite de courant soit assez bas (voir ci-dessous) de façon à ne pas activer la protection en sur-courant. Sinon, les niveaux de courant intermédiaire ne seront pas correctement maintenu et ne moteur pourrait sauter des microsteps.

MS1	MS2	MS3	Résolution Microstepping
Low	Low	Low	Pas complet (`full step`)
High	Low	Low	1/2 pas
Low	High	Low	1/4 de pas
High	High	Low	1/8 ième de pas
High	High	High	1/16 ième de pas

Les sources d'alimentation

Le pilote requière une tension d'alimentation entre 3 et 5.5 V pour la logique qui doit être connecté entre les broches VDD et GND (adjacente).

La tension d'alimentation moteur est comprise entre 8 et 35 V qui doit être connecté entre les broches VMOT et GND. Cette alimentation devrait être accompagnée d'une capacité de découplage proche de la carte, et devrait être capable de délivrer le courant attendu (des pointes jusqu'à 4 A pour l'alimentation moteur).



Cette carte utilise des capacités céramiques à faible résistance série équivalente ([http://www.jmax-hardware.com/forum/index.php?topic=5171.0;wap2 low-ESR]),ce qui la sensible aux destructions par les pointes de surtensions induites par les circuits

LC, plus spécialement sur vous utilisez des câbles d'alimentations d'une longueur supérieure à quelques centimètres. Sous de mauvaises conditions, ces pointes de tension peuvent excéder les 35 V de tension maximale du A4988 et endommager la carte de façon permanente (même pour une alimentation moteur à 12 V). Une façon de protéger les pilotes est de placer, au plus près de la carte, une capacité électrolytique importante (au moins 47 µF) entre VMOT et la masse.

Branchement moteur

Connecter ou déconnecter les moteurs pas-à-pas pendant que le pilote est sous tension peut le détruire.

D'une façon générale, changer le câblage en cours de fonctionnement sera la source de problèmes.

Limiter le courant

Pour pouvoir atteindre une vitesse plus élevée (un plus haut débit de pas), l'alimentation moteur est typiquement plus élevée que ce qui serait permit sans limitation active de courant. Par exemple, un moteur pas-à-pas typique pourrait avoir un courant maximum évalué à 1 Amp avec une résistance de bobine de 5Ω, ce qui nous indique que la tension maximale du moteur est de 5 Volts (U=R * I). Utiliser un tel moteur sous 12 V permettrait d'atteindre un débit de pas nettement plus important, mais le courant doit être activement limité à moins de 1 Ampt pour éviter d'endommager le moteur.

Le A4988 supporte cette fonctionnalité de “limitation active de courant” et le petit potentiomètre disponible sur la carte est utilisé pour fixer en fixer la valeur.

Ajuster la limite de courant

Une façon d'ajuster la limite de courant est de placer le pilote en pas complet (`full step`) et de mesurer le courant passant dans une bobine du moteur `'sans envoyer de signal'` sur la broche d'entrée STEP. Le courant mesuré sera équivalent à 0.7 fois le courant limite (puisque en mode “full step” les deux bobines sont toujours actives et le courant limité à 70% du courant limite configuré/sélectionné).

Notez que la configuration du courant limite change lorsque vous modifiez la tension de la logique de commande (VDD); la tension appliqué sur la broche “ref” du A4988 par le potentiomètre dépend directement de la valeur de VDD.

Autre méthode

Si vous disposez de matériel de précision, vous pouvez localiser la broche “ref” et en mesurer la tension. Cela sera un tâche très délicate que nous recommandons pas mais documentons néanmoins (car plus facilement accessible sur les breakout Pololu).

Une autre façon d'ajuster le courant limite est de mesurer la tension sur la broche “ref” et de calculer le courant limite correspondant (La résistance d'évaluation du courant `current sense resistors` fait 0.05Ω). Sur les cartes pololu, cette broche de tension “ref” est accessible via une sérigraphie (en cercle) en bas de la carte.

La limite de courant est calculé comme suit par rapport à la tension de référence:

Limite de courant = Vref × 2.5

Par exemple: si la tension de référence est de 0.3 V, le courant limite est de 0.75 A.

Comme mentionné ci-avant, en mode “pas complet” (`full step`), le courant dans les bobines sera limité à 70% du courant limite. Donc, pour obtenir un courant de 1 Amp (en mode `full step`), le courant limite devrait être de 1 Amp/0.7=1.4 Amp, ce qui correspond a un Vref de 1.4 A/2.5=0.56 V.

Pour des informations plus détaillées, vous pouvez vous référer à la [http://df.mchobby.be/datasheet/A4988.pdf fiche technique du A4988] (pdf).

où mesurer le courant

Le courant dans la bobine du moteur peut être vraiment différent du courant délivré par l'alimentation. Par conséquent, il ne faut pas mesurer le courant à la sortir de votre alimentation.

L'emplacement approprié pour relever le courant `'c'est en série avec la bobine du moteur'` donc entre la pilote et le moteur.

Notez que mesurer le courant à la sortie de l'alimentation ne fournit pas une mesure fidèle du courant dans les bobines. La mesure du courant sur l'alimentation peut être plus faible que le courant de bobine car la tension d'entrée du pilote peut être significativement plus élevée que la tension de la bobine. Ainsi, si la tension d'alimentation est nettement plus élevée que celle nécessaire pour le moteur, le cycle utile (`duty cycle`) sera très court pour atteindre la limite de courant configuré, ce qui conduit à ne différence notable entre les courants moyens et les courants RMS.

Dissipation de chaleur

Le circuit intégré A4988 support un courant maximum de 2 Amp par bobine, mais le courant que vous pouvez vraiment délivré dépend principalement de l'efficacité du refroidissement du circuit intégré. Le carte est conçue pour évacuer la chaleur aussi efficacement que possible mais si vous voulez fournir un courant d'environ 1 Amp par bobine il faudra prévoir un dissipateur de chaleur et/ou autre méthode de refroidissement.

Ce produit peut devenir suffisamment chaud pour vous bruler longtemps avant d'entrer lui-même en état de surchauffe. Restez prudent lorsque vous manipulez ce produit ou d'autres composants connectés sur celui-ci.

Exemple Arduino

Comment raccorder et utiliser votre jouet avec un Arduino Uno

Montage

Vous trouverez ci-dessous le montage avec les deux codification de couleurs des moteurs pas-à-pas les plus répandues

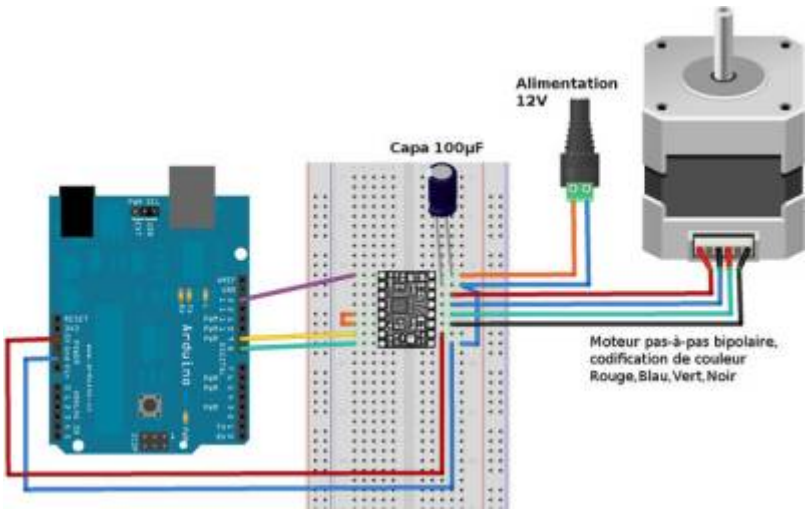
Pour commencer, nous alimentons le StepStick avec 5 volts. * Brancher le +5V d'Arduino sur la broche VDD * Brancher le GND d'Arduino sur la broche GND (en dessous de VDD)

Ensuite, nous branchons l'alimentation moteur en 12V * Brancher le +12V volts sur la broche VMOT * Brancher le GMD/Masse de l'alimentation 12V sur la broche GND (sous la broche VMOT). * Mettre les deux masse en commun. Brancher un fil entre le GND moteur (sous VMOT) et le GND de la logique (sous VDD). * Brancher une capacité électrolytique de min 47µF entre les deux broches GND et VMOT

Raccorder Arduino sur le StepStick: * Brancher la broche 13 d'Arduino sur l'entrée ENABLE du StepStick * Brancher la broche 9 d'Arduino sur l'entrée STEP du StepStick * Brancher la broche 8 d'Arduino sur l'entrée DIR du StepStick

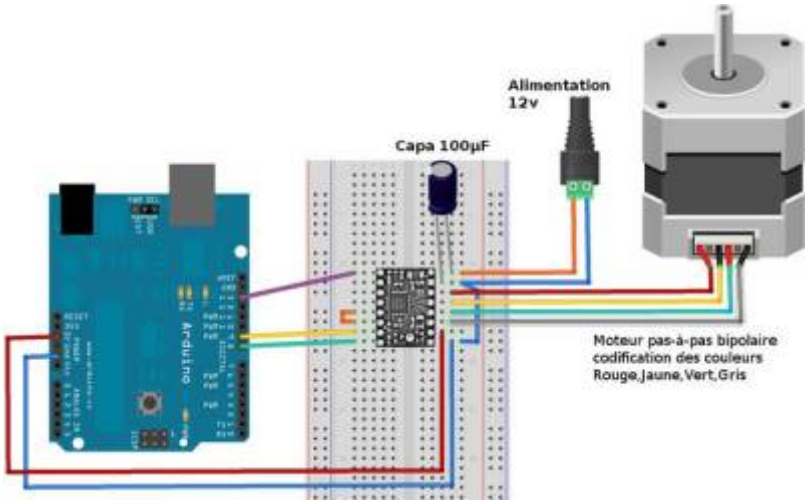
Configurer StepStick + MicroStepping: * Pour cet exemple, nous allons utiliser le mode `Full Step` (sans micro-stepping).
Ne brancher pas MS1, MS2 et MS3 qui seront donc LOW. * Brancher ensemble les broches SLEEP et RESET du stepstick.

Codification rouge, bleu, vert, noir



Codification rouge, jaune, vert, gris

Cette codification est utilisé pour les moteurs pas-à-pas que nous proposons ici chez MC Hobby



Le code

Le seul point a relever dans ce code est le comportement de la broche ENABLE branché sur la sortie 13 d'Arduino.

Enable fonctionne en logique inverse!

Cela signifie que:

- Le module est activé lorsque la sortie 13 est LOW (niveau bas) - L'axe du moteur est donc maintenu bloqué en attente du prochain pas.
- Le module est désactivé lorsque la sortie 13 est HIGH (niveau haut) - L'axe du moteur est donc totalement libre de tout mouvement puisque le moteur est désactivé.

Commande_d_un_StepStick_Driver_A4988.ino

```
// A4988_Test.ino
//
// Commande d'un moteur pas-à-pas à l'aide d'un pilote A4988 avec
//     Arduino.
//
// Un projet www.mchobby.be (vente de kit et composant)
// Meurisse D. - Licence CC-SA-BY
//
// Un tutoriel http://mchobby.be/wiki/index.php?title=A4988
// Ou Acheter un StepStick A4988
//    http://shop.mchobby.be/product.php?id_product=349
//

#define pinEnable 13 // Activation du driver/pilote
#define pinStep    9 // Signal de PAS (avancement)
#define pinDir     8 // Direction


void setup(){
  Serial.begin(9600);
  Serial.println("Test A4988");
```

```
  pinMode( pinEnable, OUTPUT );
  pinMode( pinDir   , OUTPUT );
  pinMode( pinStep  , OUTPUT );
}

void loop(){
  int i = 0;

  digitalWrite( pinDir   , HIGH); // Direction avant
  digitalWrite( pinStep  , LOW);  // Initialisation de la broche step

  // Avance de 200 pas
  for( i=0; i<200; i++){
    Serial.println( i );
    digitalWrite( pinStep, HIGH );
    delay( 10 );
    digitalWrite( pinStep, LOW );
    delay( 10 );
  }

  // Changer de direction
  digitalWrite( pinDir   , LOW); // Direction avant

  // Refaire 200 pas dans l'autre sens
  for( i=0; i<200; i++){
    Serial.println( i );
    digitalWrite( pinStep, HIGH );
    delay( 1 );
    digitalWrite( pinStep, LOW );
    delay( 1 );
  }

  // Pas de step et pas d'ordre...
  //   l'axe du moteur est donc bloqué
  Serial.println("Axe bloqué + attendre 5 sec");
  delay( 5000 );

  // déblocage de l'axe moteur
  Serial.println("Deblocage axe");
  digitalWrite( pinEnable, HIGH ); // logique inversée

  // Fin et blocage du programme
  // Presser reset pour recommander
  Serial.println("Fin de programme");
  while( true );
}
```

Exemple Raspberry

Selon toute vraisemblance, ce module devrait fonctionner avec un Raspberry Pi puisqu'il fonctionne avec une tension d'alimentation VDD à partir de 3v.

- Brancher le 3.3v de votre GPIO Raspberry sur l'entrée VDD (pour commander le stepstick en 3.3v)
- Brancher les broches du GPIO sur les broches du StepStick.

Si vous voulez tester un StepStick avec un Raspberry-Pi, alimentez le StepStick en 3.3v sur sa broche VDD. **N'ALIMENTEZ PAS** LE STEPSTICK EN 5V SI VOUS UTILISEZ UN Raspberry-Pi

Nous ne l'avons pas encore testé et ne pouvons donc rien garantir.

Ressources

* En apprendre plus sur l'effet destructeur des pointes de surtension dans un circuit LC.Voir l'article publié par polulu.com (''anglais'')

Où acheter

* Controleur moteur pas-à-pas A4988 * Moteur pas-à-pas bipolaire et son équivalent avec un couple plus élevé

From:
https://chanterie37.fr/fablab37110/ - **Castel'Lab le Fablab MJC de Château-Renault**

Permanent link:
https://chanterie37.fr/fablab37110/doku.php?id=start:arduino:a4998:doc

Last update: **2024/02/03 10:11**