

Multifunctional **AR**duino dcc **DEC**oder

Le décodeur DCC Multifonctionnel pour servomoteurs et accessoires avec Arduino pour Tous

(avec une centrale DCC)

Auteur : Nico Teering

Version : 3.4

Avril 2018

Info : Mardec@Ziggo.nl ou Arcomora.wordpress.com/contact

Arcomora.wordpress.com/Mardec

[DCC library](#) and sample coding provided by [Mynabay](#).

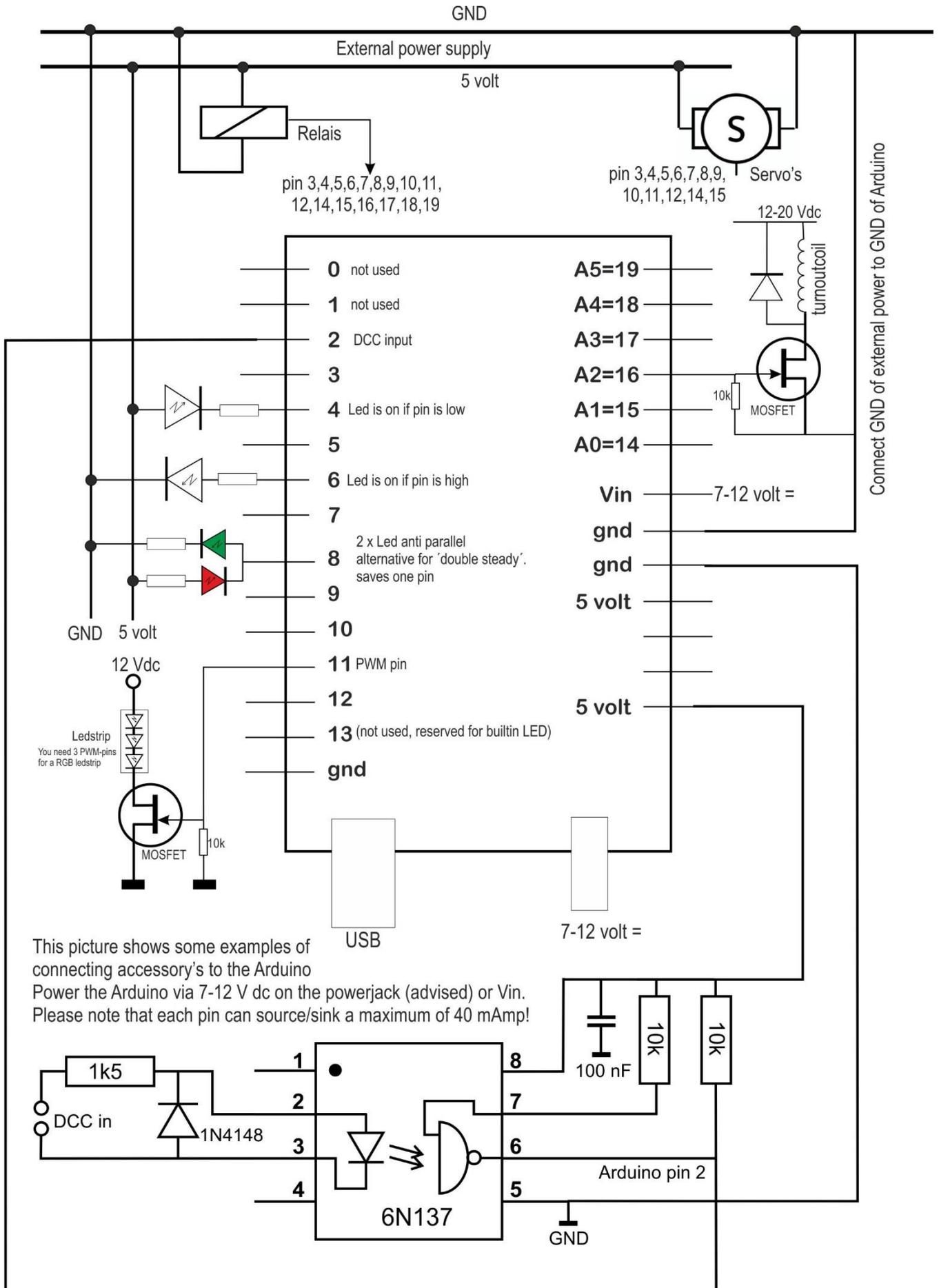
The source code, or any part of it, and the Mardec.hex file may NOT be used in any commercial product.

Nor it may be modified in any private project without informing the author.

Please mail MARDEC@ziggo.nl if you have any comment, questions or suggestions for improvement on this program.

[Telecharger ici le software ARCOMORA](#)

Connection de l'Arduino



Configuration de Mardec

Si tout le logiciel est installé correctement et l'Arduino est détecté par votre PC vous pouvez maintenant configurer Mardec. Cliquez sur le raccourci «Configure Mardec».

Si c'est la toute première fois, le driver USB Arduino sera activé, suivi du chargement de Mardec puis du démarrage de Putty.

Putty est un émulateur de terminal pour la communication entre l'Arduino et votre PC. (voir aussi le manuel d'installation).

Mardec utilise deux modes différents. Le mode de Configuration et le mode opération. Par défaut il démarre en mode configuration. Mais s'il est arrêté en mode opération, il redémarrera en mode opération.

Notez :

- Chaque saisie numérique doit être suivie de <enter>.
- Seules les commandes sur un caractère ne nécessitent pas un <enter>
- Pour la plupart des saisies, la valeur actuelle ne change pas avec un simple <enter>. La valeur actuelle est affichée entre parenthèses.
- En mode configuration la led jaune sur le shield DCC (ou la led intégrée sur la pin13) est toujours allumée.
- En mode opération cette led est éteinte.
- Verrouillez le clavier numérique.
- Vous devez utiliser des minuscules et majuscules pour les commandes.
- N'utilisez que le backspace pour corriger une saisie numérique.
- La première utilisation en mode configuration ou après un reset complet (Commande R) vous pouvez saisir un n° administrateur pour votre Mardec. Ce n° identifie votre Mardec dans le cas où de multiple Mardec sont utilisés. Mardec vous demandera aussi si vous utilisez une Roco z/Z21 ou une MultiMaus. Dans ce cas Mardec appliquera des corrections d'adressage.
- A chaque démarrage Mardec affiche les réglages courants.

```
Mardec, the Arduino DCC Decoder on COM12
MARDEC, the Multifunctional ARduino dcc DECoder
USB version 3.3
Specify number for this MARDEC
Enter value from 1 to 100: 1
This is MARDEC #1
Do you use a ROCO MultiMaus or z/Z21 Y/N ?
configuration mode of MARDEC #1
```

Les Commandes

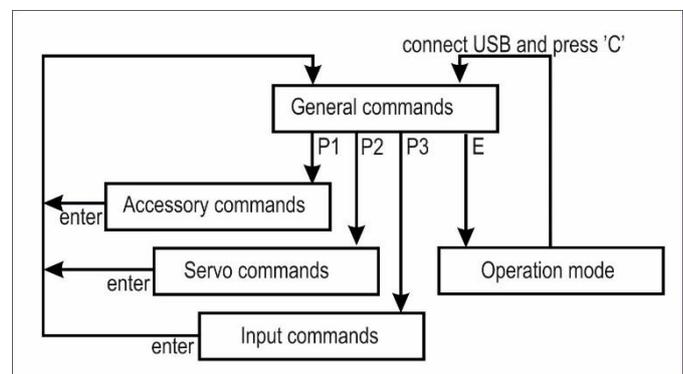
Il y a quatre types de commandes:

1. Commandes Générales
2. Commandes de réglage Servomoteurs
3. Commandes de réglage Accessoires
4. Commandes de réglage entrées

Le mode de configuration a donc quatre états. En saisissant '?' vous verrez la liste des commandes disponibles dans le mode courant.

Au démarrage le mode commandes générales est automatiquement actif. Toutes les commandes sont sur un seul caractère sans <enter>.

La commande la plus importante est 'P' (pour pin). La commande 'P' vous permet de définir une pin Arduino.



Commandes Générales

Commande P (pin)

Après la commande **P** on vous demandera un n° de pin. Si cette pin n'est pas utilisée vous devrez d'abord indiquer l'adresse DCC de la fonction souhaitée (servomoteur, accessoire ou entrée)
Si la pin est déjà utilisée, sa configuration courante s'affiche et son état de configuration changera pour correspondre au type de pin.
Pour les accessoires et les entrées vous pouvez utiliser le n° de pin 3 à 12 et 14 à 19.

Pour les servomoteurs, seules les pins 3 à 12, 14 et 15 peuvent être utilisées. Les pins Arduino A0 à A5 correspondent aux pin 14 à 19.

Un accessoire utilisant le PWM (mode 7 et 8) utilisera les pins 3, 5, 6 et 11.

Suivant votre choix Mardec sera configuré pour les commandes correspondantes.

```
Specify action (P/A/R/T/D/S/E/?): p
Specify pin number (3-12,14-19): 7
Pin is undefined. First specify DCC address

Set DCC address for pin 7
Enter value from 1 to 2000: 34
Address in use for servo/accessory on pin 4 (34)
Is that OK ? (Y/N, N=default):

Set DCC address for pin 7
Enter value from 1 to 2000: 56
DCC Address set to 56
```

Commandes A (Address)

If a pin is already configured you can also modify it by entering its DCC address. Mardec will find the correct pin number

```
Specify action (P/A/R/T/D/S/E/?): a
Specify DCC address
Enter value from 1 to 2000: 23
DCC 23, Servo #0, Angles 75/105, Frog pin 19/18, no Inv., Speed 30
Angle set to LOW: 75
```

Commandes T (test)

Après avoir configurés tous les servomoteurs et accessoires, vous pouvez tester ceux qui sont connectés. Ils fonctionneront l'un après l'autre, pour un court moment. La commande Test est aussi disponible pour commander servomoteurs et accessoires. Elle ne testera alors que la pin sélectionnée.

Commande S (Speed)

La commande **S** vous permet de configurer la vitesse par défaut d'un servomoteur. Elle peut varier de 5 (rapide) à 50 (lent) ms/degré. Cette vitesse est automatiquement assignée aux nouveaux servomoteurs.

La commande **M** est aussi utilisable pour commander un servomoteur. Elle vous permet de contrôler la vitesse du servomoteur sélectionné.

```
Specify action for Servo on pin 3 (23)
A/N/9/+/-/C/F/D/T/S/I/?/enter: s

Set speed for pin 3
Enter value from 5 to 50 (25): 30
Speed set to 30 ms/step
```

Commande D (Document)

La commande **D** affiche la configuration de toutes les pins.
Vous pouvez faire une copie d'écran ou un copier coller dans un fichier.
Ainsi vous documentez la configuration de votre Mardec.

Commande R (Reset)

La commande **R** réinitialise votre Arduino.

```
Please specify action
P/R/T/D/C/S/M/E/? : R

Are you sure you want to clear all values? Y/N: y
Are you absolutely sure? Y/N: n
```

Commande E (Exit)

La commande E passe en mode exécution. Maintenant vous pouvez contrôler les accessoires configurés par de commandes DCC émises par votre centrale.

La led d'état est éteinte. Elle clignotera quand une adresse DCC configurée sera identifiée.

En même temps une information sera affichée sur l'écran Putty.

Vous pouvez cependant fermer Putty et débrancher votre câble USB.

Pour revenir en mode configuration, connectez à nouveau le câble USB, redémarrez Putty (Raccourci: Configure Mardec) et appuyez sur 'C' au clavier.

```
Specify action (P/R/T/D/C/S/M/E/?): E
Are you sure you want to start normal mode? Y/N 
```

```
Mardec, the Arduino DCC Decoder on COM12
MARDEC, the Multifunctional ARduino dcc DECoder
USB version 3.3
Normal mode of MARDEC #1
To configuration: connect USB and press 'C'
```

Commandes Entrées

Avec une pin configurée en entrée vous pouvez simuler un signal DCC pour un servomoteur ou un accessoire ayant la même adresse que la pin d'entrée.

Par exemple:

Sur la pin 12 une led clignotante est configurée avec l'adresse DCC 34.

Sur la pin 5 est configurée une entrée avec la même adresse DCC 34.

Par défaut une pin d'entrée est à l'état haut sur l'Arduino. Si elle passe à l'état 'bas' par exemple avec un bouton poussoir ou un détecteur de consommation, Mardec interprétera ceci comme une commande DCC pour l'adresse 34 et la led clignotera sur la pin 12 (ou non). S'il y a aussi un servomoteur à l'adresse 34 il bougera. **Rien ne se passera quand il repassera à l'état 'haut'.**

Bien entendu, vous pouvez arrêter ou démarrer le clignotement en envoyant une vraie commande DCC à l'adresse 34.

La grande différence est que en l'utilisant rien ne se passe quand elle revient à l'état 'haut' alors que le changement d'état DCC inverse l'état des accessoires et servomoteurs.

Commandes 2 et 3 (seconde et troisième adresse)

Il est aussi possible de définir une seconde ou une troisième adresse connectée à une pin en entrée. Pour cela vous pouvez utiliser les commandes 2 et 3. En activant la pin entrée les accessoires sur ces adresses seront aussi commandés.

Pour les secondes et troisième adresses vous pouvez aussi spécifier un retard jusqu'à 25 secondes par pas de 0.1 secondes. Les accessoires sur ces 2nd et 3^{eme} adresses ne seront alors activés qu'une fois passée ce retard.

Sur la figure de droite le premier accessoire à adresses 20 ('own' adresses) sera activé lors de l'activation de la pin 5. Il n'y a pas de retard possible.

Après 5 sec. l'accessoire à l'adresse 25 (2nd adresse) est activé, et après 7 sec. l'accessoire à l'adresse 26 (3^{eme} adresse). Toutes les adresses doivent, bien entendu, être configurées sur le même Arduino.

Si vous souhaitez seulement 'retarder' une action vous pouvez configurer une adresse fantôme 'own' adresse de la pin d'entrée. Vous pouvez aussi spécifier l'adresse d'une autre pin comme 2nd ou 3rd adresse! Cela vous permettra d'effectuer une suite d'actions.

De même, avec une adresse DCC.

Une entrée peut être activée par une transition 1 ->0 du signal DCC.

Mais un accessoire sur la même adresse s'inversera sur les transitions 1->0 et 0->1

Par exemple sur la figure suivante les résultats sont les suivants:

DCC adresse 1->0: Accessoire à l'adresse 20 est activé. Après 5 sec. l'accessoire adresse 25 et après 7 sec. l'accessoire à l'adresse 26.

DCC adresse 0->1 Seul l'accessoire à l'adresse 20 s'inverse.

Entrée pin 5 1->0: Accessoire à l'adresse 20 est activé. Après 5 sec. l'accessoire à l'adresse 25 puis après 7 sec. l'accessoire à l'adresse 26.

Entrée pin 5 0->1: Rien ne se passe.

Pour les commandes A, N et D voir les commandes servomoteurs

```
Specify action (P/R/T/D/M/E/?): p
Specify pin number (3-12,14-19): 5
Pin is undefined. First specify DCC address

Set DCC address for pin 5
Enter value from 1 to 2048: 20
DCC Address set to 20

Specify Accessory(1), Servo(2) or Input(3)
Enter value from 1 to 3: 3
Pin 5 set as Input pin.

Specify action for Input on pin 5
A/N/?/D/2/3/enter: 2
Specify Second control address.
Enter value from 1 to 2048: 25
Specify delay time in 0.1 seconds.
Enter value from 1 to 250: 50
Address 25 set as Second control address.
With a delay of 5.0 seconds

Specify action for Input on pin 5
A/N/?/D/2/3/enter: 3
Specify Third control address.
Enter value from 1 to 2048: 26
Specify delay time in 0.1 seconds.
Enter value from 1 to 250: 70
Address 26 set as Third control address.
With a delay of 7.0 seconds

Specify action for Input on pin 5
A/N/?/D/2/3/enter:
Pin settings are saved!
```

Commande Servomoteurs

Le dialogue de configuration d'un servomoteur ressemble à ceci:

Un nouveau servomoteur est configuré avec des angles de 75(bas) et 105(haut) degrés. Un servomoteur existant conserve ces angles .

Dans les deux cas, le servomoteur est positionné en position 'bas'.

Commande -(moins)

La commande '-' diminue l'angle de 1 degré.

L'angle haut doit au minimum être supérieur de 5 degrés que l'angle bas.

Commande + (plus)

La commande '+' augmente l'angle de 1 degré.

Commande 9

La commande 9 positionne un servomoteur à 90 degrés.

Cela est pratique pour positionner la tringle d'un servomoteur au point milieu. Effectuez cette opération préalablement au montage d'un servomoteur sous les voies.

Commande C (Change)

La commande C passe d'un angle à l'autre. Vous permettant ainsi de régler chacun d'eux à l'aide des touches '-' et '+'.

Commande I (Invert)

Suivant comment les servomoteurs sont montés sous la voie l'alignement sera droite ou déviée.

La commande I inverse le sens de rotation du servomoteur.

Commande T (Test)

La commande T permet de tester un servomoteur.

Il effectuera 3 aller retour.

Commande S (Speed)

La commande permet d'ajuster individuellement la vitesse de chaque servomoteur de 5 (rapide) à 50 (lent) ms/degré.

La valeur par défaut est 25 ms, elle peut être affectée par la commande générale M.

```
Specify action (P/R/T/D/M/E/?): p
Specify pin number (3-12,14-19): 4
Pin is undefined. First specify DCC address

Set DCC address for pin 4
Enter value from 1 to 2048: 32
DCC Address set to 32

Specify Accessory(1), Servo(2) or Input(3)
Enter value from 1 to 3: 2
Pin 4 set as Servo pin.
Servo on address 32 set to 75 degrees

Specify action for Servo on pin 4
A/N/9/+/-/C/F/D/T/M/I/?/enter: -
LOW angle set to 74 degrees

Specify action for Servo on pin 4
A/N/9/+/-/C/F/D/T/M/I/?/enter: c
Servo on address 32 set to 105 degrees
Angle set to HIGH: 105

Specify action for Servo on pin 4
A/N/9/+/-/C/F/D/T/M/I/?/enter: +
HIGH angle set to 106 degrees

Specify action for Servo on pin 4
A/N/9/+/-/C/F/D/T/M/I/?/enter: m

Set speed for pin 4
Enter value from 5 to 50 (25): 30
Speed set to 30 ms/step

Specify action for Servo on pin 4
A/N/9/+/-/C/F/D/T/M/I/?/enter: n

Specify accessory/servo/input number on pin 4
Enter value from 0 to 2000: 2
Number set to 2

Specify action for Servo on pin 4
A/N/9/+/-/C/F/D/T/M/I/?/enter: i

Invert servo? Y/N (N=default) y

Servo set to 'Inverted'

Specify action for Servo on pin 4
A/N/9/+/-/C/F/D/T/M/I/?/enter:

Pin settings are saved!
```

Commande F (Frog point)

Avec la commande F, Mardec assignera automatiquement un n° de pin à laquelle vous pourrez connecter un relais pour alimenter le cœur d'un aiguillage (frog point).

Pour un seul relais sélectionnez '1' ou '2' pour deux relais.

Si un cœur est déjà assigné, Mardec demandera si vous souhaitez le remplacer.

Le relais commutera quand le servomoteur sera à mi-course. Ce qui assure une inversion en douceur des polarités.

Le relais sera ferme en position 'haute'.

Si nécessaire, vous pouvez inverser la commutation avec la commande I

```
Specify action for Servo on pin 4 (34)
A/N/9/+/-/C/F/D/T/S/I/?/enter: F
How many relays ? (0/1/2): 1
```

```
Pin 17 is set as first frogpin for servo on pin 4 (34), No inversion
```

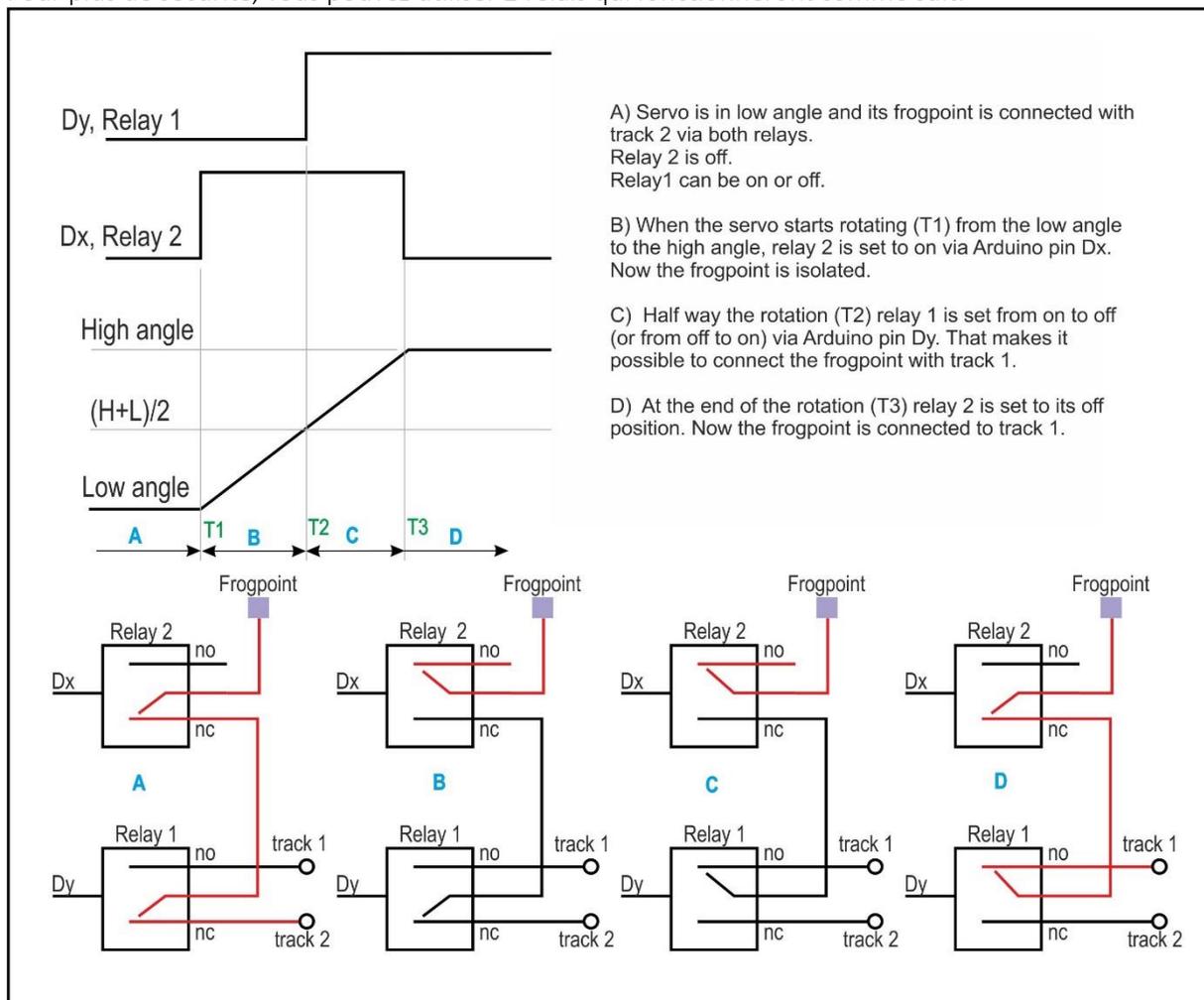
```
Specify action for Servo on pin 4 (34)
A/N/9/+/-/C/F/D/T/S/I/?/enter: F
```

```
Remove frogpoint or toggle relay inversion? R/1/2: 1
First Relay inverted
```

```
Specify action for Servo on pin 4 (34)
A/N/9/+/-/C/F/D/T/S/I/?/enter: f
```

```
Remove frogpoint or toggle relay inversion? R/1/2: 1
First Relay NOT inverted
```

Pour plus de sécurité, vous pouvez utiliser 2 relais qui fonctionneront comme suit:



Certains relais sont fermé à l'état 'bas' (0 Volt au lieu de 5 V)

Mardec vous proposera l'Inversion'.

Ceci assure que le relais n'est fermé quand le servomoteur bouge.

Commande A (Address)

La commande A définit l'adresse DCC d'une pin. Si cette adresse est déjà utilisée un message vous en avertira. En acceptant vous pourrez contrôler de multiples accessoires par une seule adresse. **En spécifiant l'adresse 0 vous pouvez faire un reset de la pin.**

```
Set DCC address for pin 4
Enter value between 0 and 2048 (?): 0
Do you want to reset pin 4? Y/N: y
DCC Address set to 0
Pin settings are saved!
```

Commande N (Number)

La commande N assigne un n° administratif à une pin. Elle n'a aucune signification technique.

Commande D (document)

La commande D affiche un résumé de la configuration de toutes les pins.

Commande <enter>

La commande enter clos l'édition de la configuration d'une pin.

La nouvelle configuration ou sa modification est alors enregistrée en mémoire.

Commandes Accessoire

Les commandes DCC n'ont que deux états (0/1, on/off, bas/haut, droit/dévie, rouge/vert) MARDEC effectue plusieurs fonctions avec ces états.

Toutes les fonction 'double' spécifiés ci dessous nécessitent l'utilisation d'une seconde pin nommée 'buddy' pin.

Elle est automatiquement assignée lors de la configuration des fonctions doubles.

Pour une nouvelle pin MARDEC demandera la fonction souhaitée (mode); pour une pin existante la configuration courante sera affichée.

Les modes suivantes sont disponibles:

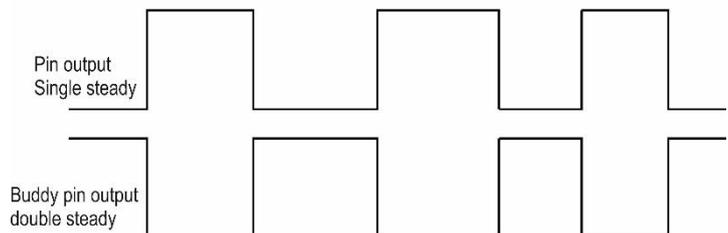
1. Single Steady

La pin Arduino copie le signal DCC.



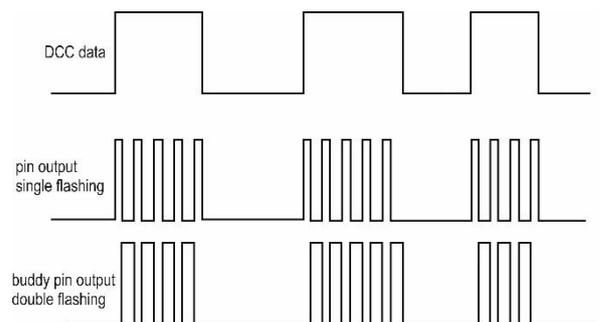
2. Double steady

La pin Arduino copie le signal DCC et une seconde buddy pin inverse ce signal.



3. Single Flashing

La pin Arduino pin clignote temps que le signal DCC est haut. Les temps On/Off peuvent être réglés séparément de 50 à 5000 ms.



4. Double Flashing

Identique au single flashing, mais une buddy pin inverse le signal. Les deux pins sont à l'état bas quand le signal DCC

est bas

Voir aussi les commandes L (Low) / H (High)

5. Single One Shot

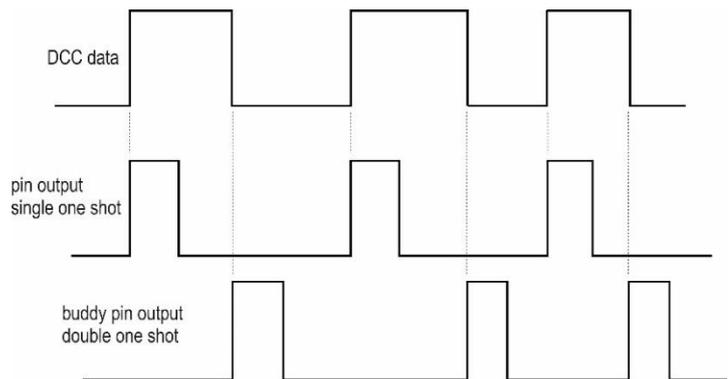
En single one shot la pin Arduino est positionnée à l'état haut pour une durée configurable de 50 à 5000ms quand le signal DCC passe de bas à haut.

Si l'adresse d'un single one shot est identique à celle d'une autre pin alors les deux seront activées; à la fin de l'impulsion

l'autre pin est de-activée.

Par exemple: sur l'adresse 12 vous configurez un one shot sur la pin 5 et un single flashing sur la pin 7.

Si l'adresse 12 est envoyée à MARDEC, le one shot allume sur la pin 5 et une led clignote sur la pin 7. au front descendant, la pin 7 arrêtera de clignoter.



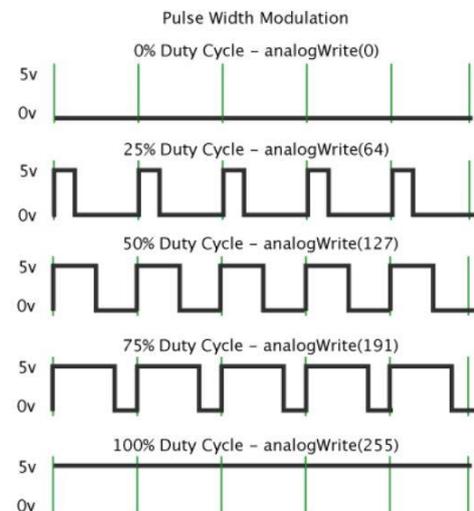
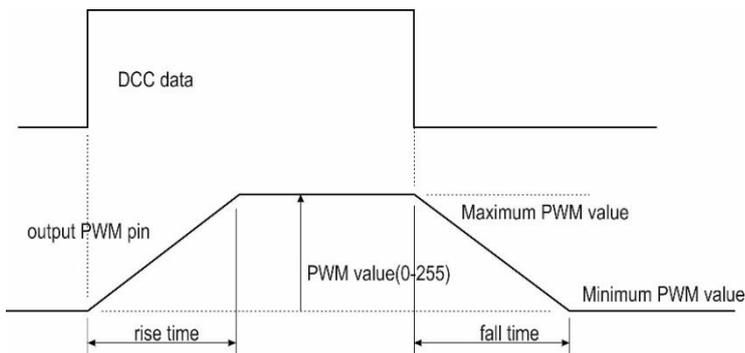
6. Double One Shot

Identique au Single One Shot mais une seconde buddy pin sera mise en haut sur le front descendant du signal DCC. Ceci peut être utilisé pour les aiguillages a solénoïdes. Mais attention! Un aiguillage a solénoïde ne peut pas être connecté directement à une pin Arduino : il consomme trop de courant et le voltage est trop faible. Vous devez utiliser un MOSFET pour commander le solénoïde.

7. Analog PWM control

Un nombre limité de pins (3, 5, 6, and 11) sur un Arduino peuvent fournir un signal PWM sur lequel le rapport cyclique peut varier de 0 à 100%. En connectant une led sur une telle pin, le niveau lumineux pourra varier.

En activant cette fonction par un signal DCC, le PWM de la pin Arduino évoluera lentement de la valeur mini a la valeur maxi. En désactivant cette fonction par un signal DCC, le PWM de la pin Arduino évoluera lentement de la valeur maxi a la valeur mini.



Vous pouvez régler :

- Valeur maximum PWM (min. à 255)
- Valeur minimum PWM (0 à max)
- Temps de montée (0 - 500 sec)
- Temps de descente (0 - 500 sec)

Une application typique est le contrôle d'un

ruban led . En utilisant les pins PWM vous pouvez contrôler un ruban de led RGB. Vous aurez recours a un MOSFET entre l'Arduino et le ruban de led.

Voir aussi les commandes R(ise) / F(all) et L(ow) / H(igh)

8. Flickering mode

Cette fonction permet à une led connectée de vaciller avec un temps ajustable. Seules sont disponibles les pins PWM pins 3, 5, 6 et 11. En utilisant des codes de couleur différentes sur plusieurs pins vous pourrez simuler un feu ou un poste de soudure.

9. Random on/off

Une led connectée s'allumera et s'éteindra en permanence temps que le signal DCC sera haut. À chaque cycle, un nouveau temps aléatoire d'allumage sera utilisé de 2 à 500 sec. Par pas de 2 sec.

10. Disco light

A l'identique du random on/off mais 1000 fois plus rapide : on/off de 2 à 500 millisecondes.

Commande M (Mode)

La commande M permet de changer le mode de fonctionnement d'une pin accessoire.

La commande a sa propre fonction d'aide:

```
Specify mode for this accessory
0=Help. Enter value from 0 to 10 (4): 0
Available Modes:
 1 Single steady,   2 Double steady
 3 Single flashing, 4 Double flashing
 5 Single One shot, 6 Double One shot
 7 Analog PWM,     8 Flickering
 9 Random on/off, 10 Disco lights
Enter value from 0 to 10 (4): █
```

Commande H (High)

Les fonctions 3,4,5 et 6 fixent le temps 'on'
Il peut aller de 100 à 500 ms. par pas de 20 ms.
La fonction 7 fixe la plus grande valeur PWM.
Les fonctions 9 et 10 fixent la période maximum.

Commande L (Low)

Les fonctions 3,4 fixent le temps 'off'
Il peut aller de 100 à 500 ms. par pas de 20 ms.
La fonction 7 fixe la plus petite valeur PWM.
Les fonctions 9 et 10 fixent la période minimum.

Commande R (Rise)

Ne s'applique qu'à la fonction 7. Elle fixe le temps de montée.
C'est le temps nécessaire au PWM pour passer du minimum au maximum.

Commande F (Fall)

Ne s'applique qu'à la fonction 7. Elle fixe le temps de descente.
C'est le temps nécessaire au PWM pour passer du maximum au minimum.

Autres commandes

Commande T test d'un accessoire

Commande N d'assignation d'un n° administratif

Commande A d'assignation d'adresse DCC à une pin ou resets d'une pin (0)

Commande D résumé des configuration

```
Specify pin number
3-12,14-19: 6
Pin 6: Accessory with DCC 15, Nr 15, Mode Analog PWM
value, PWM 30/255, Time R/F 14/20

Specify action for Accessory on pin 6
A/N/H/L/M/C/?/D/T/F/R/enter: A

Set DCC address for pin 6
Enter value between 0 and 2048 (15): 20
DCC Address set to 20

Specify action for Accessory on pin 6
A/N/H/L/M/C/?/D/T/F/R/enter: N

Specify accessory/turnout number on pin 6
Enter value between 0 and 2000 (15):
Number set to 15

Specify action for Accessory on pin 6
A/N/H/L/M/C/?/D/T/F/R/enter: H

Set maximum PWM value for pin 6
Enter value between 30 and 255 (255): 240
Maximum PWM value set to 240

Specify action for Accessory on pin 6
A/N/H/L/M/C/?/D/T/F/R/enter: L

Set minimum PWM value for pin 6
Enter value between 0 and 240 (30): 20
Minimal PWM value set to 20

Specify action for Accessory on pin 6
A/N/H/L/M/C/?/D/T/F/R/enter: R

Set RISE time (sec) for PWM pin 6
Enter value between 0 and 500 (14): 20
Rise time set to 20 sec.

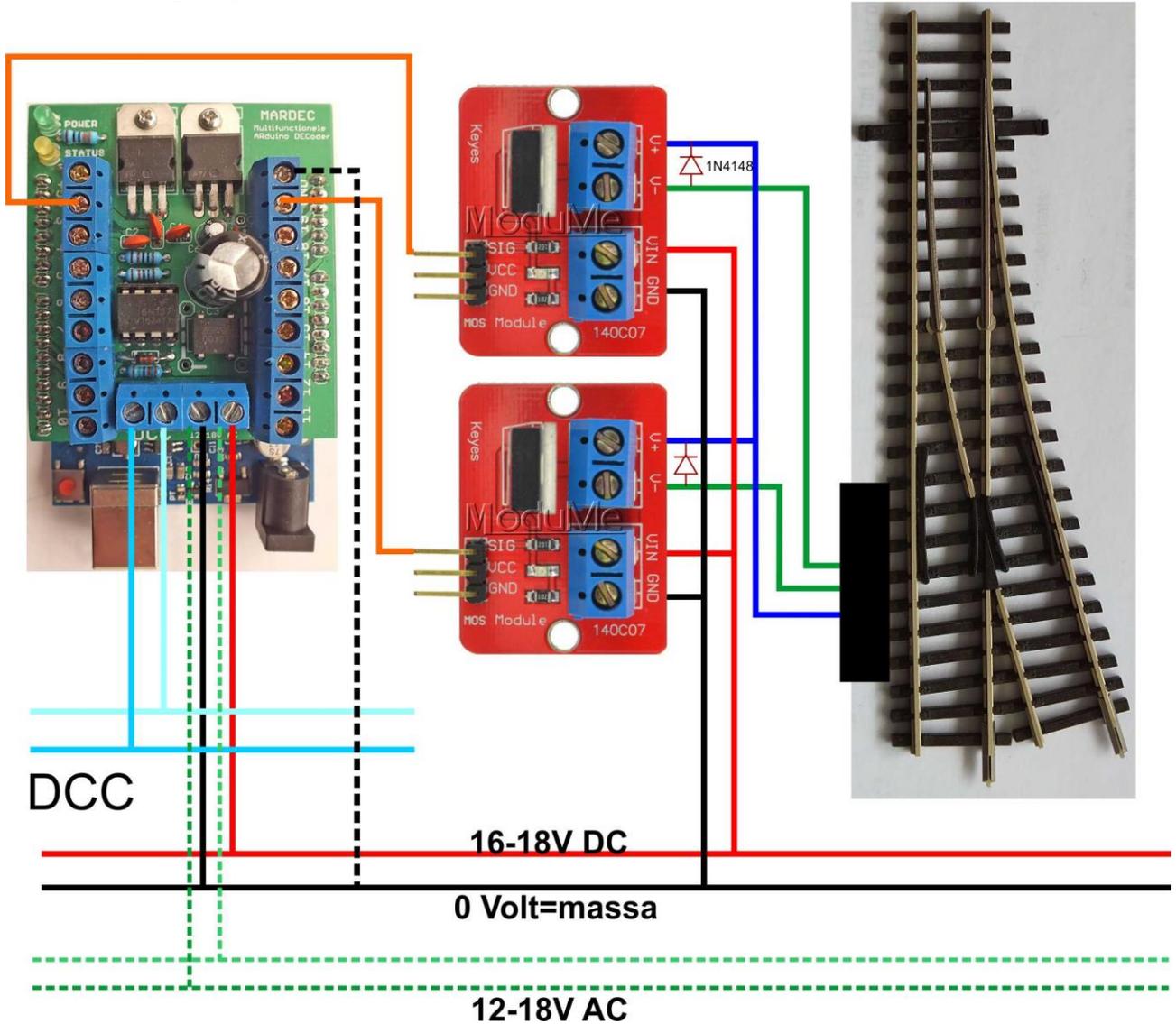
Specify action for Accessory on pin 6
A/N/H/L/M/C/?/D/T/F/R/enter: F

Set FALL time (sec) for PWM pin 6
Enter value between 0 and 500 (20): 16
Fall time set to 16 sec.

Specify action for Accessory on pin 6
A/N/H/L/M/C/?/D/T/F/R/enter: T
Testing accessory 15 on pin 6 in analog mode
Accessory 15 on pin 6 and address 20 starts rising
Accessory 15 on pin 6 and address 20 starts falling
```

Exemples de configurations

Connecter un aiguillage à solénoïdes avec deux MOSFETS



```
Specify action (P/R/T/D/M/E/?): p
Specify pin number (3-12,14-19): 3
Pin is undefined. First specify DCC address

Set DCC address for pin 3
Enter value from 1 to 2048: 5
DCC Address set to 5

Specify Accessory(1), Servo(2) or Input(3)
Enter value from 1 to 3: 1
Pin 3 set as Accessory pin.

Specify mode for this accessory
0=Help. Enter value from 1 to 10: 6
Mode set to Double One shot.

Pin 19 is assigned as buddy pin to accessory on pin 3
Specify 'On' time
Enter value from 20 to 5000: 300
'On' time set to 300 msec.

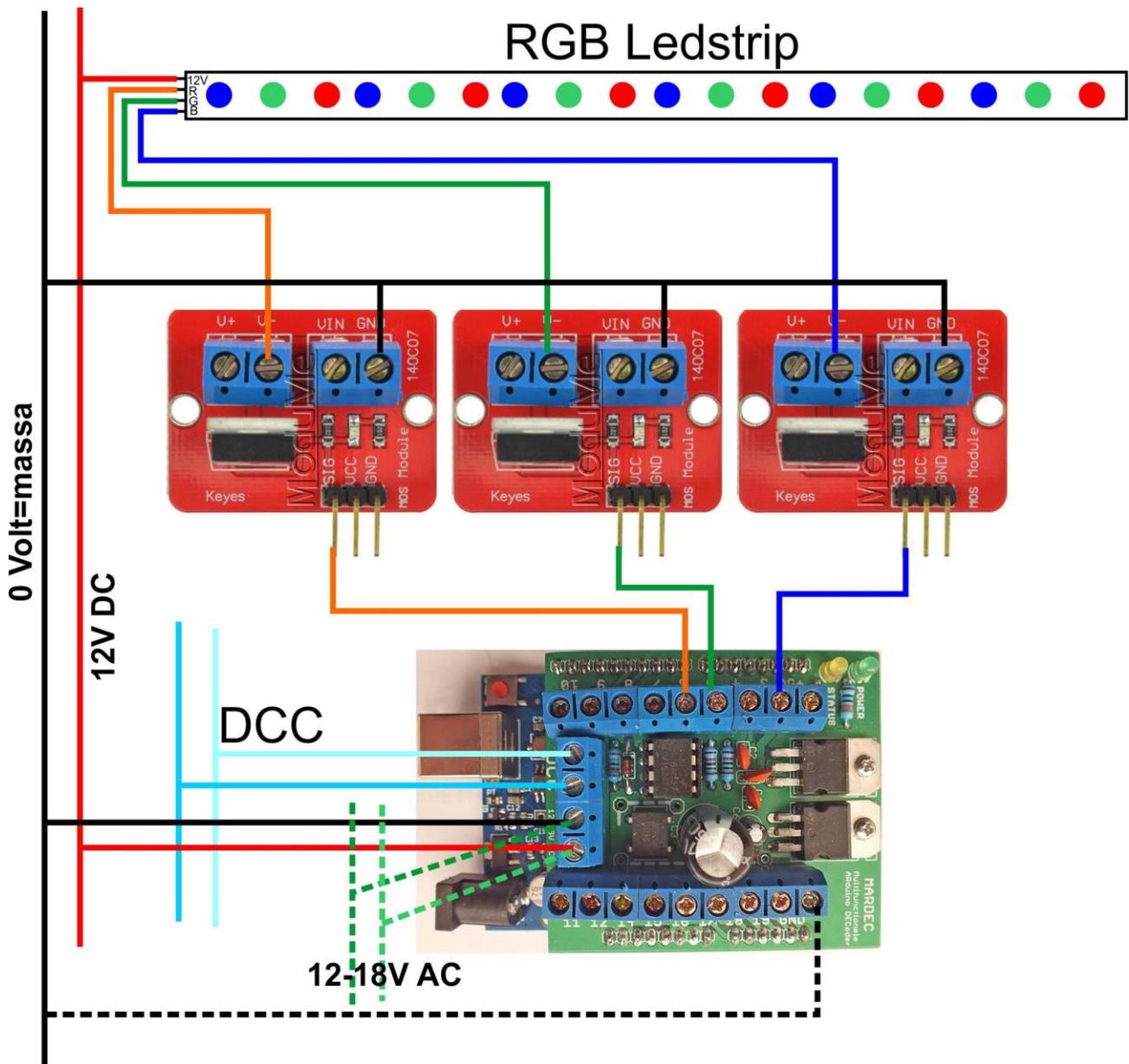
Specify action for Accessory on pin 3
enter/A/N/D/T/M/H/L/?/F/R:

Pin settings are saved!
```

```
Specify action (P/R/T/D/M/E/?): d
Settings of MARDEC #1
Pin 3: DCC 5, Acc.type 6 (Double One shot), #0, Buddy 19, Time(ms) 300
Pin 4: not used
Pin 5: not used
Pin 6: not used
Pin 7: not used
Pin 8: not used
Pin 9: not used
Pin 10: not used
Pin 11: not used
Pin 12: not used
Pin 14: not used
Pin 15: not used
Pin 16: not used
Pin 17: not used
Pin 18: not used
Pin 19: Buddy pin for pin 3

Default Servo speed: 25 ms/step
```

Connexion d'une bande de led RGB avec des amplificateurs MOSFETS

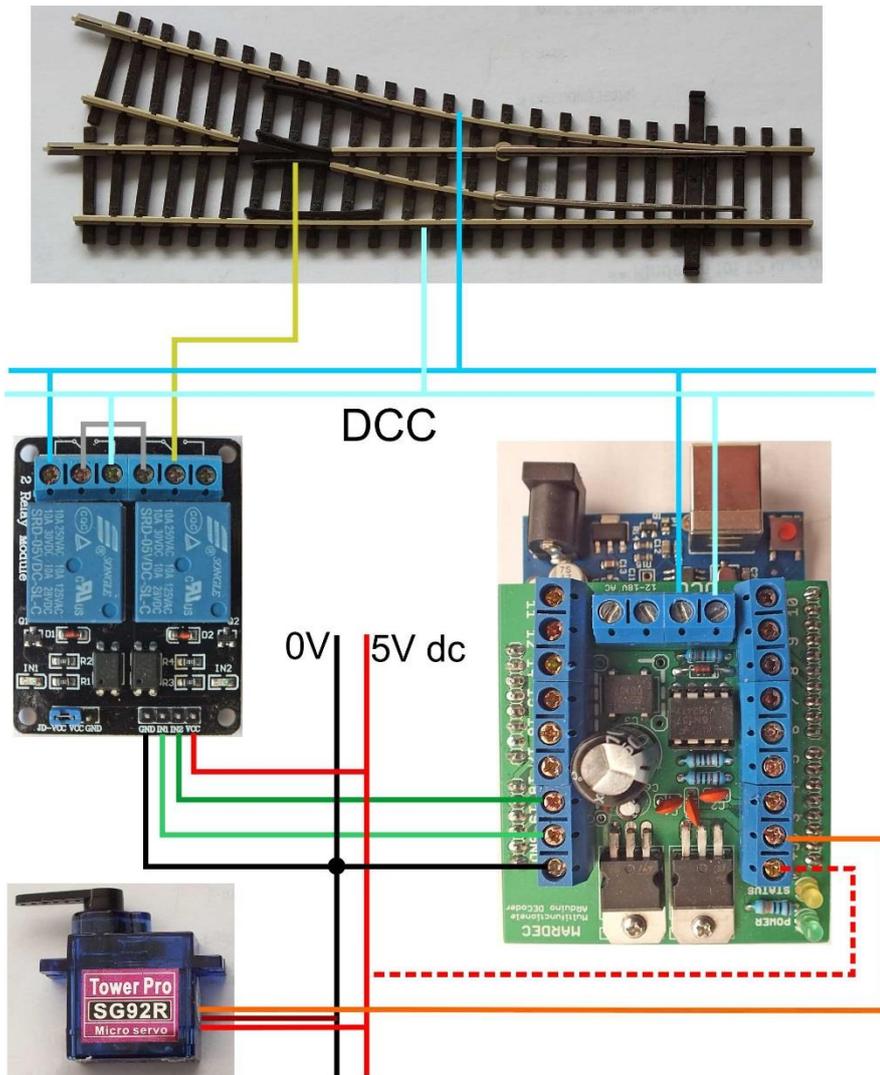


Specify action (P/R/T/D/M/E/?): D

Settings of MARDEC #1

Pin 3: DCC 12, Acc.type 7 (Analog PWM), #0, PwM 20/240, Time R/F(sec) 10/10
 Pin 4: not used
 Pin 5: DCC 12, Acc.type 7 (Analog PWM), #0, PwM 10/240, Time R/F(sec) 20/20
 Pin 6: DCC 12, Acc.type 7 (Analog PWM), #0, PwM 20/240, Time R/F(sec) 30/30
 Pin 7: not used
 Pin 8: not used
 Pin 9: not used
 Pin 10: not used

Connexion d'un servomoteur à double relais sur un aiguillage



```

MAR_1 on port COM12
Specify action (P/R/T/D/M/E/?): p
Specify pin number (3-12,14-19): 17
Pin is undefined. First specify DCC address

Set DCC address for pin 17
Enter value from 1 to 2048: 34
DCC Address set to 34

Specify Accessory(1), Servo(2) or Input(3)
Enter value from 1 to 3: 2
Pin 17 set as Servo pin.
Servo on address 34 set to 75 degrees

Specify action for Servo on pin 17
A/N/9/+/-/C/F/D/T/M/I/?/enter: f
Do you want frogpoint polarisation for servo on pin 17 ?
(<enter>=No; 1,2=Yes) Enter value from 1 to 2: 2
Do you want inversion for first relay ? Y/N y

Pin 19 is set as frogpin for servo on pin 17, with inversion

Do you need inversion for second relay ? Y/N n
Pin 18 is set as second frogpin
    
```

Pin 17: DCC 34, Servo #0, Angles 75/105, Frog pin 19/18, no Inversion, Speed 25
 Pin 18: Second relay (No inv.) for pin 17
 Pin 19: First relay (Inv.) for pin 17