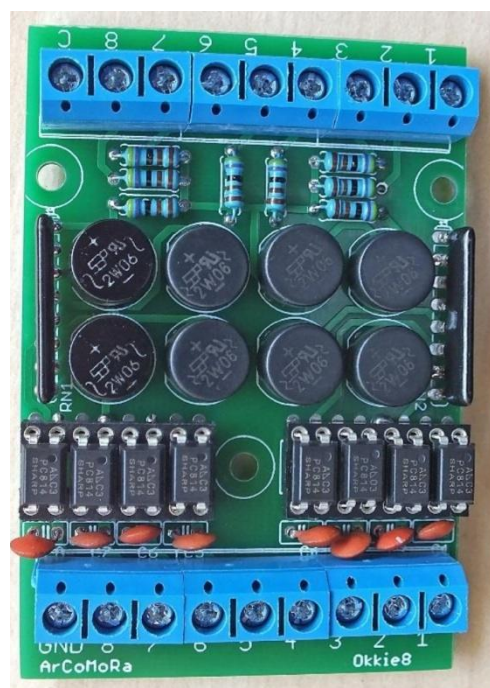
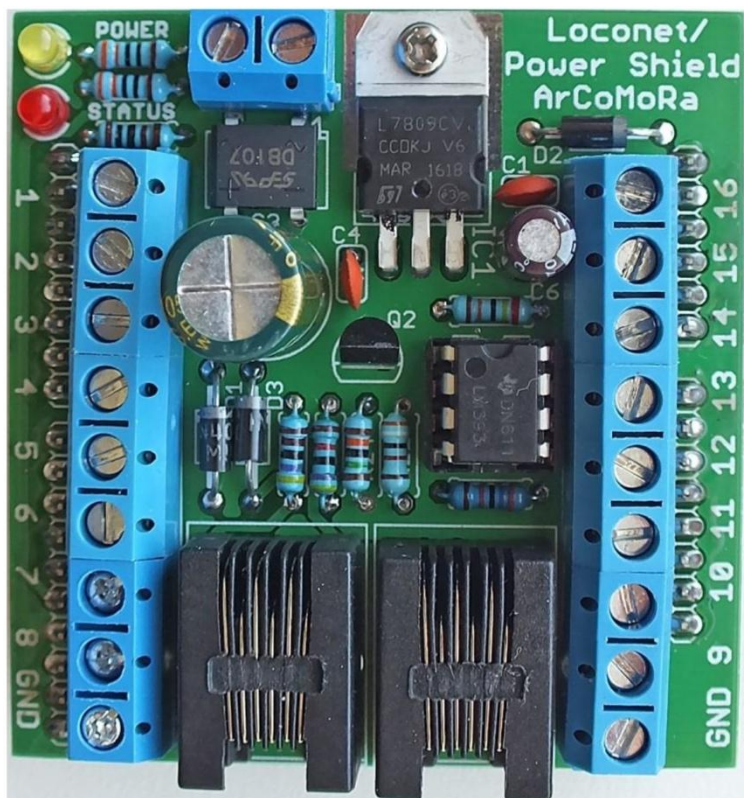


ARCOMORA

ARDUINO CONTROLLED MODEL RAILWAY

ARduinoLOCO_{net}

L'interface LocoNet[®] de rétrosignalisation universelle à base d'Arduino



Auteur : Nico Teering

Traduction : Jean-Michel Bapt

Juin 2020

ArLoco version 3.1

Version du document : 3.1 FR

Info : info@arcomora.com

Web : https://www.arcomora.com/fr_arcomora

LocoNet[®] est une marque déposée de DigitraX, Inc



[TÉLÉCHARGEZ ICI LE PROGRAMME ARLOCO](#)

[VOUS POUVEZ COMMANDER LE SHIELD ARLOCO ET L'OKKIE ICI](#)

Table des matières

Introduction.....	4
ArCoMoRa	4
Pourquoi un Arduino ?	4
Toutes les caractéristiques d'ArLoco d'un seul coup d'oeil :	5
De quoi avez-vous besoin ?	5
Carte fille (Shield) ArLoco	5
Okkie.....	5
Méthodes de détection	6
Détection continue	6
Détection par impulsion.....	6
Détection mixte	6
Adressage	7
Configuration de l'ArLoco.....	8
Les commandes	8
Commande <I> (réglages Initiaux).....	9
Commande (adresse de Base).....	9
Commande <R> (Réinitialisation).....	9
Commande <E> (Exit-Sortie)	9
Commande <V> (Voir)	10
Commande <D> (Débogage)	10
Commande <S> (Sensor-Détecteur).....	11
Commande <L> (adresse LocoNet).....	11
Commande <M> (Méthode).....	11
Commande <I> (Inversion)	11
Commande <V> (Voir)	11
Commande <A> (Activation/Désactivation).....	11
Diagramme de connexion avec un UNO	12
Schéma ArLoco	12
Schéma OKKIE.....	12
Construire la carte fille (Shield) ArLoco	13
Composants :	13
Méthode de montage conseillée de la carte fille ArLoco :.....	14
Construire l'Okkie	16
Composants :	16
Méthode de montage recommandée de l'Okkie :	16

Test de l'Okkie	17
Test de la chaîne complète : Okkie ⇒ ArLoco ⇒ PC.....	17
Schéma simplifié de la carte fille (shield) ArLoco	18
Informations complémentaires.....	18
Démarrage.....	18
Sortie du programme	18
Fichier log	18
Configuration à partir du deuxième décodeur et configuration avec un MEGA2560	19
Modifier le port COM	19
Méthodes de détection	20
1) Détection par impulsions avec ILS (Interrupteur à Lame souple / Reed)	20
2) Détection de courant.....	20
3) Détection de masse	20
4) Détection par réflexion photoélectrique	20
5) Détection par rupture de faisceau lumineux	20

Introduction

L'usage de carte à microcontrôleur Arduino dans le modélisme ferroviaire est connu depuis longtemps. Cependant, pour beaucoup, sa mise en œuvre réelle est un peu trop ardue et son aspect programmation est souvent considéré comme étant un problème majeur.

En conséquence, l'Arduino reste souvent le domaine d'un groupe limité de spécialistes ayant des connaissances techniques.

C'est dommage, car avec l'Arduino, vous pouvez automatiser l'ensemble de votre réseau à moindre coût.

Afin de rendre l'Arduino accessible à tous, j'ai créé trois programmes qui sont suffisants pour l'automatisation complète d'un réseau. Ils forment ensemble le concept ArCoMoRa.

ArCoMoRa

ArCoMoRa signifie **Ar**duino **Co**ntrôled **Mo**del **Ra**ilway. Sur un réseau digital basé sur le concept ArCoMoRa, l'automatisation est entièrement contrôlée par des micro-ordinateurs Arduino.

ArCoMoRa utilise la norme DCC et le protocole LocoNet®.

Avec ArCoMoRa, vous n'avez aucune ligne de code à écrire, ni même à connaître la programmation.

Le concept ArCoMoRa se compose donc de trois programmes Arduino différents, chacun avec ses propres fonctions :

- **Mardec**, le programme qui permet de gérer un décodeur Arduino DCC multifonctionnel.
Il permet de commander jusqu'à 12 servomoteurs et de nombreux autres accessoires tels que des moteurs à solénoïde, des LEDs ou des relais.
Le Mardec dispose d'un maximum de 16 sorties mais aussi d'"entrées". Cela vous permet de contrôler les accessoires connectés même sans signal DCC, avec ou sans temporisation réglable.
- **ArSigDec**, (Arduino Signal Decoder), le programme qui permet de contrôler les signaux comme vous l'entendez.
L'ArSigDec peut contrôler jusqu'à 8 signaux ou 16 LEDs avec un UNO (30 signaux ou 59 LEDs avec un MEGA2560) et permet 10 aspects différents par signal.
L'ArSigDec contient également une base de données des principaux signaux utilisés en Europe.
- **ArLoco**, contraction d'Arduino et de LocoNet®, est le programme qui permet la rétrosignalisation en utilisant le protocole LocoNet®.
Indicateurs d'occupation à 16 canaux avec un UNO et 58 avec un MEGA2560.
Optimisé pour la détection continue (par détection de courant par exemple) et la détection par impulsion (par exemple avec des ILS ou des détecteurs infrarouges).

Une caractéristique essentielle de ces trois programmes est qu'ils peuvent être configurés facilement sans aucune connaissance du monde Arduino ni de la programmation. Vous configurez simplement les programmes en fonction de vos besoins par un simple jeu de questions-réponses sur votre PC. Donc aucune nécessité d'écrire une seule lettre de code. Mais, comme pour tout autre décodeur DCC, les composants ArCoMoRa doivent être configurés avant leur utilisation.

Cette configuration est souvent appelé de façon incorrecte « programmation ». Pour éviter cette confusion, il convient de réserver terme « programmation » au fait d'écrire du code Arduino. Avec ArCoMoRa, vous ne faites que configurer les programmes.

Tous les programmes ArCoMoRa ont donc un état « configuration » et un état « exploitation ». Dans le mode configuration, vous pouvez tout régler à votre guise. Dans le mode exploitation, les Arduino sont opérationnels et vous pouvez suivre sur votre PC, si vous le désirez, toutes les actions commandées pas le signal DCC.

L'installation de la suite logicielle est très simple. Tous les logiciels nécessaires sont installés en une seule fois. Le téléchargement du programme vers le décodeur est également pratiquement entièrement automatisé.

Pourquoi un Arduino ?

La raison principale est le coût, ce qui est un élément important : Un [clone Arduino UNO](#) ne doit pas coûter plus de 3 €.

Une carte fille (Shield) est nécessaire avec les cartes Arduino.

Pour Mardec et ArSigDec, la carte fille coûte 4,25 €, mais vous pouvez aussi facilement le faire vous-même.

Vous disposez ainsi d'un décodeur DCC 16 voies ou d'un décodeur de signaux pour 2 à 8 signaux pour moins de 7 €. Pour l'ArLoco, une carte fille LocoNet est nécessaire, elle coûte 4,25 €. En option, vous pouvez y connecter deux cartes de détection de présence à 8 canaux (Okkie) au prix unitaire de 5,50 €.

Toutes les caractéristiques d'ArLoco d'un seul coup d'oeil :

- Convient pour Arduino MEGA2560 et Arduino UNO
- Indicateur d'occupation 16 canaux (UNO) ou 58 canaux (MEGA2560)
- 256 Adresses LocoNet possibles
- L'adresse peut être réglée individuellement pour chaque canal.
- Utilise le protocole LocoNet® pour communiquer avec la centrale ou l'ordinateur.
- Fonctionnalités de débogage étendues.
- Optimisé pour la détection par impulsion (ILS, infrarouge, etc.) et la détection continue (détection de courant, détection de masse, etc...).
- Facile à configurer avec votre PC.
- Aucun CV compliqué à paramétrer.
- L'Arduino peut être alimenté par une alimentation externe ou par le câble LocoNet.
- Utilisable avec LocoNet-T et LocoNet-B
- Chaque canal* est configurable individuellement pour la détection par impulsion ou pour la détection continue.
- Détection continue :
 - Prévention des fausses détections dues à la diaphonie ou aux pics de bruits parasites
 - Prévention des interruptions momentanées de détection pour mauvais contacts rail/roues
 - Parfaitement adapté à la carte de détection de présence OKKIE
- Détection par impulsion :
 - Génère une impulsion d'occupation constante de 1 seconde.
 - Supprime les effets de rebond.
 - Fonction anti-duplication en cas de vitesse lente avec des ILS
 - Adapté aux interrupteurs à lames souples (ILS), aux détecteurs infrarouges, etc....

De quoi avez-vous besoin ?

- Une carte Arduino UNO ou MEGA2560
- Une carte fille (Shield) ArLoco
- Une centrale digitale avec LocoNet ayant une connexion USB ou réseau
ou
Une centrale digitale avec LocoNet sans USB ni connexion réseau mais avec une interface LocoNet-USB
- Un PC avec un logiciel d'exploitation de réseau de train
- En option : des modules de détection de présence tel que OKKIE (détection de présence par détection de courant)

Carte fille (Shield) ArLoco

- 2 Connexions LocoNet permettant de chaîner plusieurs ArLoco.
- Alimentation intégrée pour la carte Arduino (AC/DC)
- 16 borniers à vis pour les entrées
- 2 borniers à vis pour les connexions GND
- Une LED d'alimentation
- Une LED d'état



Okkie

- Carte de détection de présence pour 8 cantons
- 8 borniers à vis pour les entrées
- 8 borniers à vis pour les sorties
- Connexion GND
- Connexion commune à une des deux voies du BUS traction



Un ensemble carte fille ArLoco et carte de détection de présence Okkie peut être commandé sur cette page : www.arcomora.com/order

* Sur le canal 1, détection par impulsion uniquement.

Méthodes de détection

L'ArLoco a deux méthodes de détection :

- La détection continue
- La détection par impulsion

Détection continue

Avec la détection continue, lorsque le détecteur passe au niveau bas, ArLoco envoie une impulsion LocoNet avec l'adresse du détecteur à la centrale digitale. Le canton est alors considéré comme "occupé".

Lorsque le détecteur repasse au niveau haut, ArLoco envoie de nouveau une impulsion LocoNet. Le canton est alors considéré comme "libre".

Ce fonctionnement est spécifique à la détection de présence continue. Une entrée de l'ArLoco est reliée à une sortie d'un circuit de détection de présence.

Pour éviter les fausses détections causées par des contacts intermittents, ArLoco dispose de deux temporisations intégrées.

Délai avant de signaler l'état « occupé » :

Le canton est considéré comme occupé lorsque l'entrée est au niveau bas pendant une période plus longue que le paramètre «délai avant occupation».

Ce délai est par défaut de 100 ms, mais il peut être modifié par la commande <O>. Ceci évite les fausses détections causées par la diaphonie ou des pics de bruits parasites. Le pic en T1 (voir figure) n'est pas détecté car sa durée est plus courte que le paramètre « délai avant occupation ». Une information LocoNet est envoyée (T2) après la fin de ce délai de temporisation.

Délai avant de signaler l'état « non occupé » :

Le canton est défini comme « libéré » à partir du moment où l'entrée est de nouveau à l'état haut après un délai minimum fixé par la variable « délai avant libération » (par défaut 1000 ms, peut être modifié par la commande <F>). Ceci permet de compenser de brèves interruptions de détection dues, par exemple, à un mauvais contact avec le rail. La non détection en T3 est ignorée par ArLoco car elle est plus courte que le paramètre « délai avant libération ».

A T4 + « délai avant libération », le canton est libéré par l'envoi d'une information LocoNet.

Sur le réseau idéal, ces deux paramètres devraient être à zéro.

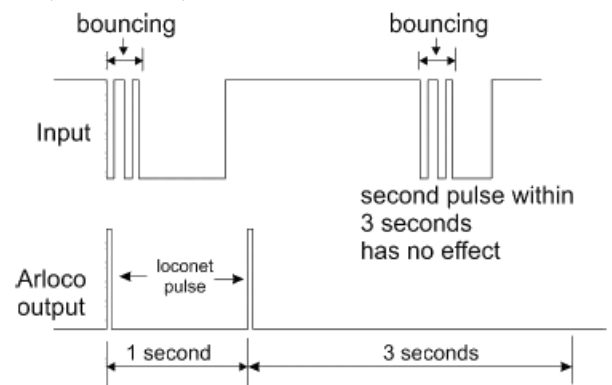
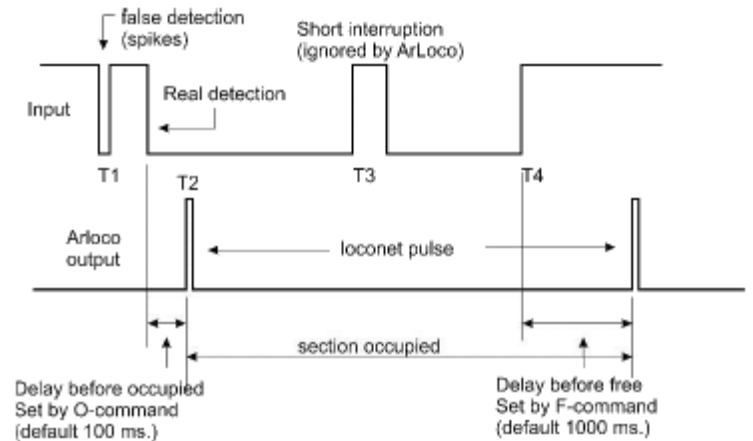
Attention : Sur l'entrée 1 du Shield ArLoco, seule la détection par impulsion est possible !

Détection par impulsion

Cette méthode est destinée à être utilisée avec des interrupteurs à lame souple (ILS ou Reed). Si l'entrée passe au niveau bas, ArLoco enverra une information LocoNet « zone occupée » à la centrale digitale. Un temps fixe de 1 seconde plus tard ArLoco envoie une information LocoNet « zone libre ». Ceci évite les fausses détections par rebond du contact. De plus, la vitesse du train n'a aucune influence, même si un train est arrêté au-dessus de l'ILS. Parce qu'il n'y a que de petits courants impliqués, il n'y a pas de danger en cas de pic de tension. Par conséquent, aucune précaution particulière n'a été prise contre ces pics de tension. Après la deuxième impulsion LocoNet, ArLoco attendra trois secondes avant de vérifier à nouveau l'entrée. Ce délai évite qu'une deuxième information LocoNet soit émise en cas de conduite lente. Ce délai d'attente peut être réglé avec la commande <W>.

Détection mixte

Par défaut, toutes les entrées sont configurées soit en « détection continue » (sauf la 1 qui restera toujours configurée en « détection par impulsion ») soit en « détection par impulsion ». Cependant, avec la commande <S>, vous pouvez définir le mode de détection individuellement pour chaque entrée. Il est ainsi possible d'utiliser les deux méthodes de détection sur un seul ArLoco.



Adressage

En fonction de l'Arduino utilisé, ArLoco a 16 ou 58 entrées. Ceci correspond à 16 ou 58 adresses. Chaque entrée a sa propre adresse et la plage des adresses d'ArLoco va de l'adresse 1 à l'adresse 256.

Chaque entrée de l'ArLoco peut être désactivée individuellement (commande <A>). De cette façon, elle n'utilisera pas d'adresse. Cette faculté est intéressante dans le cas d'un grand réseau composé de modules indépendants qui nécessiteraient chacun un ArLoco avec un nombre limité de zones à détecter. Il est donc possible d'utiliser plus de 16 ArLoco.

Chaque ArLoco doit recevoir une adresse de base (commande). Ainsi, par exemple, si vous ne voulez utiliser que 10 adresses à partir de l'adresse 25, vous indiquerez comme adresse de base 25 puis vous définissez les entrées 11 à 16 comme étant "inactives". La première entrée active aura l'adresse de base, la seconde l'adresse de base + 1, la troisième l'adresse de base + 2, etc.

Dans l'exemple ci-dessous, l'adresse de base est définie sur 45 et les entrées 1 et 10 jusqu'à 16 sont désactivées.

```
Specify action (A/B/M/E/N/O/S/F/D/W/V/R/?): B
Set Base address for this ArLoco
Enter value from 1 to 255 (12): 45
Base address set to 45

Specify action (A/B/M/E/N/O/S/F/D/W/V/R/?): v
Sensor settings:
Sensor 1 set to Pulse and NOT active, on address 0
Sensor 2 set to Continuous and Active, on address 45
Sensor 3 set to Continuous and Active, on address 46
Sensor 4 set to Continuous and Active, on address 47
Sensor 5 set to Continuous and Active, on address 48
Sensor 6 set to Continuous and Active, on address 49
Sensor 7 set to Continuous and Active, on address 50
Sensor 8 set to Continuous and Active, on address 51
Sensor 9 set to Continuous and Active, on address 52
Sensor 10 set to Continuous and NOT active, on address 0
Sensor 11 set to Continuous and NOT active, on address 0
Sensor 12 set to Continuous and NOT active, on address 0
Sensor 13 set to Continuous and NOT active, on address 0
Sensor 14 set to Continuous and NOT active, on address 0
Sensor 15 set to Continuous and NOT active, on address 0
Sensor 16 set to Continuous and NOT active, on address 0
```

Grâce à cette méthode, l'ArLoco peut utiliser de manière très flexible la plage d'adressage disponible.

Il est également possible de donner à chaque entrée sa propre adresse, mais veillez à ce que cette adresse ne soit pas utilisée par un autre ArLoco et que celle-ci soit en dehors de la plage d'adresses de cet ArLoco.

Le tableau sur la droite indique la correspondance entre les entrées de l'ArLoco et les broches respectives des cartes Arduino.

Les broches 1, 8, 12 et 13 ne sont pas utilisées avec le UNO. Les numéros des entrées sont indiqués sur la carte fille.

Les broches 1, 8, 12, 13 et 48 ne sont pas utilisées avec le MEGA. Le numéro des entrées 1 à 16 sont indiqués sur la carte fille.

Le numéro des entrées 17 à 58 sont indiqués sur la carte MEGA.

Attention : à partir de la broche 48, les numéros des broches et des entrées ne sont plus en concordance !

L'adresse LocoNet d'une zone de détection est donnée par la simple formule suivante :

numéro d'une entrée ArLoco + l'adresse de base du ArLoco - 1

Entrées	Broches UNO	Broches MEGA
1	0	0
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	9	9
9	10	10
10	11	11
11	14	54
12	15	55
13	16	56
14	17	57
15	18	58
16	19	59
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		49
49		50
50		51
51		52
52		53
53		A10 (64)
54		A11 (65)
55		A12 (66)
56		A13 (67)
57		A14 (68)
58		A15 (69)

Configuration de l'ArLoco

Une fois les logiciels correctement installés et l'ArLoco connecté, vous pouvez commencer à configurer ArLoco. Cliquez sur le raccourci *ArLoco* créé sur le bureau.

La toute première fois que lancerez ArLoco, les drivers nécessaires seront installés pour utiliser le port USB sur lequel votre Arduino est connecté, et le programme sera téléversé dans l'Arduino.

Ensuite, la communication avec le programme démarre immédiatement (voir également le manuel d'installation).

Attention :

- Chaque saisie numérique doit être validée par la touche <Entrée>.
- Seules les commandes alphabétiques à un seul caractère ne nécessitent pas la touche <Entrée>
- Dans la plupart des cas, les valeurs actives ne changent pas par la seule frappe de la touche <Entrée>. Les valeurs du réglage actuel sont indiquées entre parenthèses.
- En mode configuration, la LED « STATUS » sur la carte LocoNet/PowerShield est allumée fixe.
- En mode exploitation, cette LED « STATUS » est éteinte mais elle clignote lors de l'émission d'un signal LocoNet.
- Activez la touche Numlock du pavé numérique de votre clavier pour le verrouiller.
- Vous pouvez utiliser sans distinction les minuscules ou les majuscules pour chaque commande.
- La touche <RetourArrière> ne peut être utilisée que pour corriger une saisie numérique.

Toutes les captures d'écran qui suivent sont basées sur l'Arduino UNO. Mais les affichages sont identiques avec un MEGA2560.

Vous pouvez maintenant entrer une commande.

La première fois que vous démarrez l'ArLoco, ou après une réinitialisation complète, la commande <I> s'active automatiquement. (Voir plus bas la commande <I>). Diverses valeurs par défaut doivent être définies.

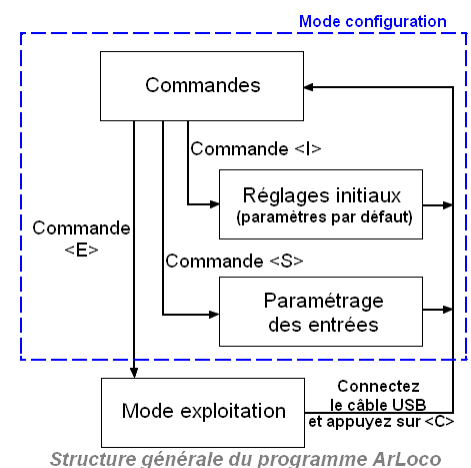
Une vue d'ensemble de la configuration mémorisée s'affiche automatiquement à chaque démarrage du mode configuration.

Les commandes

L'ArLoco, comme le Mardec et l'ArSigDec, a deux états : le mode configuration et le mode exploitation. Le mode configuration comporte des commandes générales, des commandes spécifiques aux entrées et des réglages initiaux.

La saisie d'un ? (point d'interrogation) donne un aperçu des commandes disponibles.

```
Specify action (B/D/E/I/R/S/V/?): ?
B = Set Base Address
S = Set Sensor settings
I = Set Initial(default) settings
D = Set Debug mode
V = View all sensors
R = Reset ArLoco
E = Exit to normal mode
Specify action (B/D/E/I/R/S/V/?):
```



Chaque commande se compose d'une seule lettre et n'a PAS besoin de se terminer par la touche <Entrée>. Les lettres sont basées sur la description anglaise de la fonction de chaque commande.

ATTENTION : ArLoco 2.x ne peut pas être mis à jour vers la version 3.0 ou ultérieure car la méthode de stockage des données dans l'EEPROM a été radicalement modifiée. Ceci était nécessaire pour donner à chaque entrée sa propre adresse.

Commande <I> (réglages Initiaux)

La commande <I> permet d'ajuster les paramètres par défaut. Il n'y a pas de sous-menu et tous les réglages de base seront demandés les uns après les autres.

Ces réglages de base sont les suivants :

- Le numéro administratif pour cet ArLoco qui identifiera celui-ci.
- Le mode standard de détection (par impulsion ou continue).
Toutes les entrées seront réglées sur ce mode ! (à l'exception du capteur 1 qui est toujours en mode impulsion).
Modifiable ensuite individuellement par entrée avec la commande <M>.
- L'adresse de base.
Paramètre modifiable ultérieurement avec la commande .
- Inversion générale des entrées.
Par défaut, un niveau bas (=0 Volt) sur une entrée est synonyme "d'occupation". Cependant, lorsqu'une cellule photo-électrique est utilisée, celle-ci enverra un signal haut (soit 5 Volt) lorsque le faisceau lumineux est coupé par un train.
L'inversion permet à ArLoco d'accepter ce signal haut lorsque le canton est occupé.
Modifiable ensuite individuellement par entrée avec la commande <I>.

- En cas de détection par impulsion, durée du temps de latence pendant lequel une entrée n'est pas "testée " après l'envoi d'une impulsion d'occupation.
(Pour la temporisation, voir également la section sur la détection par impulsions).
- Délai avant libération du canton : par défaut, 1000 ms. La plage de réglage possible va de 1 à 3000 ms.
- Délai avant déclaration d'occupation du canton : par défaut : 100 ms. La plage de réglage possible va de 1 à 3000 ms.

```
Specify action (B/D/E/I/R/S/V/?): i
Specify number for this ARLOCO
Enter value from 1 to 100 (1): 1
This is your ARLOCO #1

Specify default detection method; 1=Continuous, 2=Pulse
Enter value from 1 to 2: 1
Detection method set to Continuous

Set Base address for this ArLoco
Enter value from 1 to 250 (1): 32
Base address set to 32

Default Inversion set to NOT Inverted
Change inversion ? Y/N (N=default) :

Specify Waittime for pulse detection (msec)
Enter value from 1 to 3000 (1001):
Waittime for pulse detection 1001 milliseconds

Specify delay before setting free (msec)
Enter value from 0 to 3000 (1000):
Delay before setting free 1000 milliseconds

Specify delay before setting occupied (msec)
Enter value from 0 to 1000 (100):
Delay before setting occupied 100 milliseconds

Specify action (B/D/E/I/R/S/V/?): I
```

Commande (adresse de Base)

Avec la commande , vous définissez l'adresse de base de l'ArLoco.

Voir également le paragraphe sur l'adressage.

```
Specify action (B/D/E/I/R/S/V/?): B
Set Base address for this ArLoco
Enter value from 1 to 250 (54): 25
Base address set to 25
```

Commande <R> (Réinitialisation)

La commande <R> réinitialise tous les paramètres de l'ArLoco. Une double confirmation vous sera demandée.

Ensuite, le programme redémarre.

```
Specify action (B/D/E/I/R/S/V/?): r
Are you sure you want to clear all settings? Y/N [N]: y
ABSOLUTELY sure? Y/N [N]:
```

Commande <E> (Exit-Sortie)

Avec la commande <E>, vous passez en mode Exploitation. ArLoco enverra maintenant une information à la centrale digitale pour chaque train détecté.

Pour revenir en mode Configuration, connectez le câble USB et appuyez sur la touche <C> de votre clavier.

Commande <V> (Voir)

La commande <V> affiche les paramètres de toutes les entrées.

```
Specify action (B/D/E/I/R/S/V/?): v
Default detection method set to Continuous
Delay before setting occupied (msec): 150
Delay before setting free (msec): 800
Waittime for pulse detection (msec): 500

Base address: 25

Sensor settings:
Sensor 1 set to Pulse and Active, on address 25, with input not inverted
Sensor 2 set to Continuous and Active, on address 26, with input not inverted
Sensor 3 set to Continuous and Active, on address 27, with input not inverted
Sensor 4 set to Continuous and Active, on address 28, with input not inverted
Sensor 5 set to Continuous and Active, on address 29, with input not inverted
Sensor 6 set to Continuous and Active, on address 30, with input not inverted
```

Commande <D> (Débogage)

Avec la commande <D>, vous pouvez activer et désactiver le mode débogage. En mode débogage, chaque changement d'état des détecteurs est signalé lorsque vous êtes en mode exploitation.

```
Specify action (A/M/E/N/O/S/F/D/?): D
Do you want to show all sensor changes? (Y/N, N=default) y
Debug mode set to ON
Specify action (A/M/E/N/O/S/F/D/?): █
```

```
ARLOCO, the ARduino LocoNet feedback encoder
USB version 1.0

Normal mode of ARLOCO #4

To configuration: connect USB and press 'C'

Default detection method set to Continuous
Delay before setting occupied (msec): 100
Delay before setting free (msec): 500

This Arloco uses addresses 49-64:

Set address 59 from sensor 11 as occupied
Set address 60 from sensor 12 as occupied
Train detected on address 53 from sensor 3
Set address 58 from sensor 10 as free
Train detected on address 54 from sensor 6
Set address 61 from sensor 13 as free
Set address 59 from sensor 11 as free
Train detected on address 55 from sensor 7
Set address 62 from sensor 14 as occupied
Set address 61 from sensor 13 as occupied
Set address 60 from sensor 12 as free
Set address 61 from sensor 13 as free
Train detected on address 53 from sensor 3
Train detected on address 54 from sensor 6
Set address 63 from sensor 15 as occupied
Train detected on address 56 from sensor 8
Set address 62 from sensor 14 as free
Train detected on address 54 from sensor 6
Set address 58 from sensor 10 as occupied
Train detected on address 53 from sensor 3
Set address 63 from sensor 15 as free
Train detected on address 54 from sensor 6
```

Mode d'opération normal

Attention : lorsque vous êtes en mode exploitation, si l'Arduino est connecté à votre PC et que le programme Putty est démarré, vous pouvez communiquer via le clavier avec l'Arduino. Cette communication utilise la broche 0 (Tx) qui est aussi l'entrée 1 de ArLoco. Une frappe involontaire sur le clavier peut donc provoquer un message d'occupation via le détecteur 1. Pour éviter ceci, assurez-vous que le programme Putty n'est pas la fenêtre active de Windows.

```
Specify action (A/M/E/N/O/S/F/D/?): d
Do you want to show all sensor changes? (Y/N, N=default) y
Debug mode set to ON

Specify action (A/M/E/N/O/S/F/D/?): e
Do you want to start normal mode? Y/N (N=default): y
ARLOCO, the ARduino LocoNet feedback encoder
USB version 1.0

Normal mode of ARLOCO #4

To configuration: connect USB and press 'C'

Default detection method set to Continuous
Delay before setting occupied (msec): 100
Delay before setting free (msec): 500

This Arloco uses addresses 49-64

365691 No change on sensor 1, still free
365735 End of wait on sensor 5
365769 End of wait on sensor 6
365802 End of wait on sensor 8
365836 Sensor 10 changed to 0
365868 Sensor 14 changed to 1
365939 Set address 58 from sensor 10 as occupied
366371 Set address 62 from sensor 14 as free
366616 Sensor 13 changed to 0
366660 Train detected on address 55 from sensor 7
366717 No change on sensor 13, still free
366732 Sensor 13 changed to 0
366814 Train detected on address 54 from sensor 6
366833 No change on sensor 13, still free
366851 Sensor 13 changed to 0
366947 Train detected on address 53 from sensor 5
366952 No change on sensor 13, still free
366978 Sensor 13 changed to 0
367081 Set address 61 from sensor 13 as occupied
367661 End of pulse on sensor 7
367733 Sensor 13 changed to 1
367815 End of pulse on sensor 6
367890 Sensor 10 changed to 1
367948 End of pulse on sensor 5
368234 No change on sensor 13, still occupied
368393 Set address 58 from sensor 10 as free
368424 Sensor 11 changed to 0
368527 Set address 59 from sensor 11 as occupied
369044 Sensor 13 changed to 1
369547 Set address 61 from sensor 13 as free
369852 Sensor 12 changed to 0
369955 Set address 60 from sensor 12 as occupied
```

Mode debug. Le nombre indique les millisecondes écoulées depuis le lancement du programme.

RAPPEL : vous ne pouvez pas connecter une sortie d'un détecteur OKKIE ou d'un autre détecteur de présence sur l'entrée 1 de ArLoco.

Commande <S> (Sensor-Détecteur)

La commande <S> permet de régler un certain nombre de paramètres d'une entrée de détection. En saisissant un point d'interrogation, les sous-commandes sont rappelées.

```
Specify action (B/D/E/I/R/S/V/?): S
```

```
Which sensor do you want to set?  
Enter value from 1 to 58: 12
```

```
Specify action for Sensor 12 (L/A/I/M/V/?):
```

```
Specify action for Sensor 12 (L/A/I/M/V/?): ?
```

```
L = Set Loconet address  
M = Set detection Method  
A = Set Active on/off  
I = Set Inversion on/off  
V = View settings
```

```
Specify action for Sensor 12 (L/A/I/M/V/?):
```

Les sous-commandes suivantes sont disponibles :

Commande <L> (adresse LocoNet)

Avec la commande <L>, vous pouvez affecter à un détecteur une adresse personnalisée de 1 à 256. Attention à ce que cette adresse ne soit pas déjà utilisée sur un autre ArLoco et que celle-ci soit en dehors de la plage d'adresses de cet ArLoco. Si un détecteur avait déjà antérieurement une adresse, en saisissant 0 (zéro) vous lui remettez son adresse automatiquement définie par l'adresse de base de l'ArLoco.

```
Specify action for Sensor 12 (L/A/I/M/V/?): L
```

```
Set LocoNet address for sensor 12  
Enter value from 0 to 256 (36): 200  
LocoNet address for sensor 12 set to 200
```

```
Specify action for Sensor 12 (L/A/I/M/V/?): L
```

```
Set LocoNet address for sensor (0=back to base) 12  
Enter value from 0 to 256 (200): 0  
LocoNet address for sensor 12 set to 36
```

Commande <M> (Méthode)

Avec la commande <M>, vous modifiez la méthode de détection de l'entrée sélectionnée.

```
Specify action for Sensor 2 (L/A/I/M/V/?): m
```

```
Method for sensor 2 set to Pulse
```

```
Specify action for Sensor 2 (L/A/I/M/V/?): m
```

```
Method for sensor 2 set to Continuous
```

Commande <I> (Inversion)

Avec la commande <I>, vous pouvez inverser la méthode de fonctionnement de l'entrée sélectionnée (niveau haut <-> niveau bas).

```
Specify action for Sensor 12 (L/A/I/M/V/?): I
```

```
Sensor 12 set to Inverted
```

```
Specify action for Sensor 12 (L/A/I/M/V/?): I
```

```
Sensor 12 set to not Inverted
```

Commande <V> (Voir)

La commande <V> affiche les paramètres de l'entrée sélectionnée.

```
Specify action for Sensor 12 (L/A/I/M/V/?): V
```

```
Sensor 12 set to Continuous and Active, on address 36, with input not inverted
```

Commande <A> (Activation/Désactivation)

La commande <A> permet d'activer ou de désactiver une entrée. Une entrée inactive ne sera pas "testée". En déclarant inactives un certain nombre d'entrées, un nombre équivalent d'adresses LocoNet deviennent disponibles pour d'autres ArLoco.

```
Which sensor do you want to set?  
Enter value from 1 to 58: 50
```

```
Specify action for Sensor 50 (L/A/I/M/V/?): A
```

```
Sensor 50 set to not active  
Set sensors 51 to sensor 58 also InActive?
```

Si une entrée est inactivée, il sera demandé si les entrées suivantes doivent également être inactivées. Ceci est utile si vous n'avez pas prévu de connecter d'autres détecteurs sur cet ArLoco. Lorsqu'une entrée est réactivée, elle retrouvera l'adresse qu'elle avait auparavant. Il sera demandé de la même façon si les entrées suivantes doivent également être réactivées.

```
Which sensor do you want to set?  
Enter value from 1 to 58: 15
```

```
Specify action for Sensor 15 (L/A/I/M/V/?): A
```

```
Sensor 15 set to Active  
Set all other inactive sensors active again? (Y/N)
```

Diagramme de connexion avec un UNO

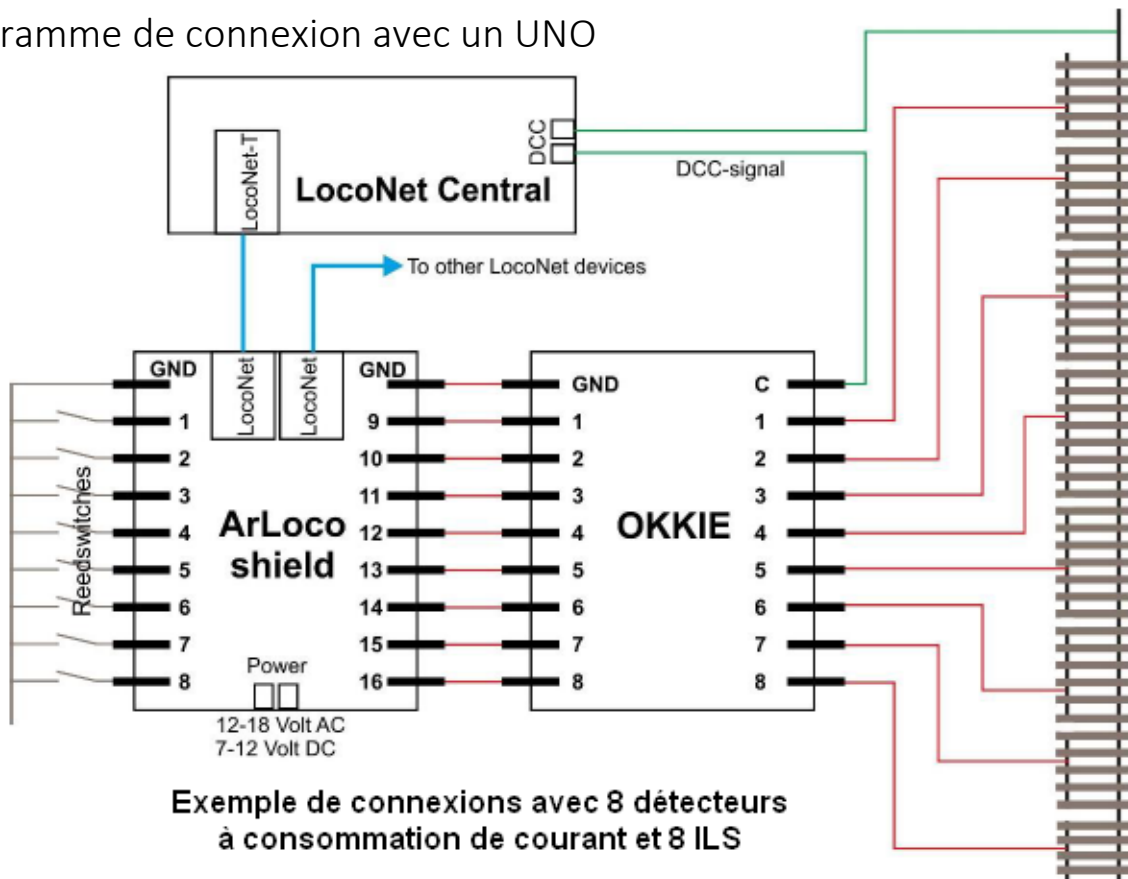


Schéma ArLoco

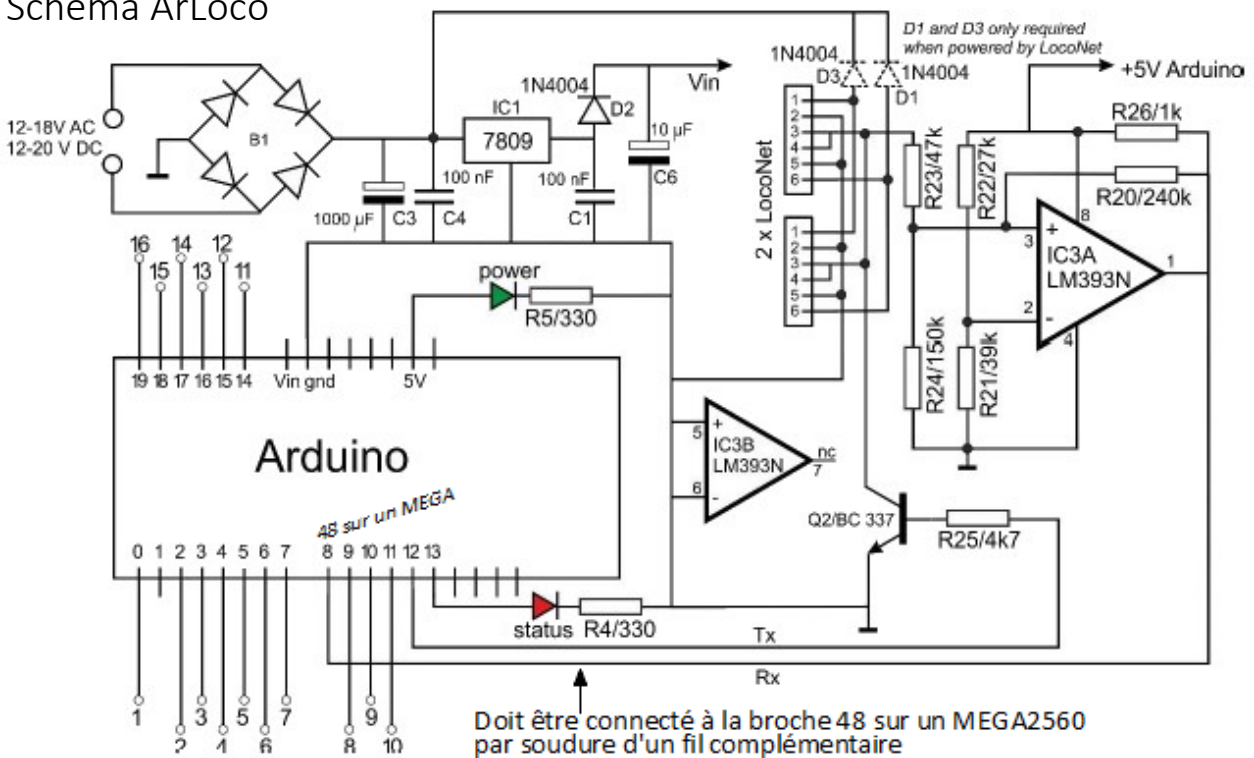


Schéma OKKIE

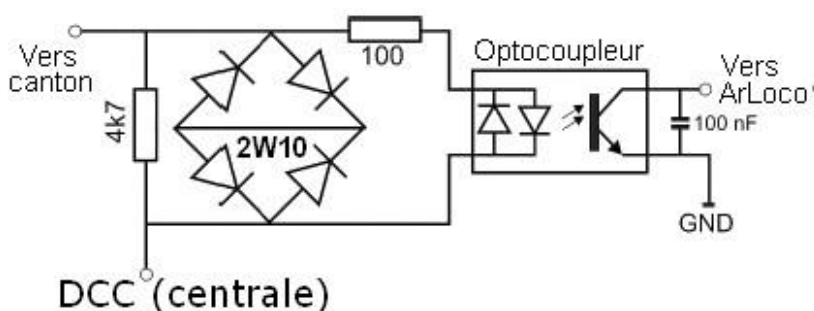
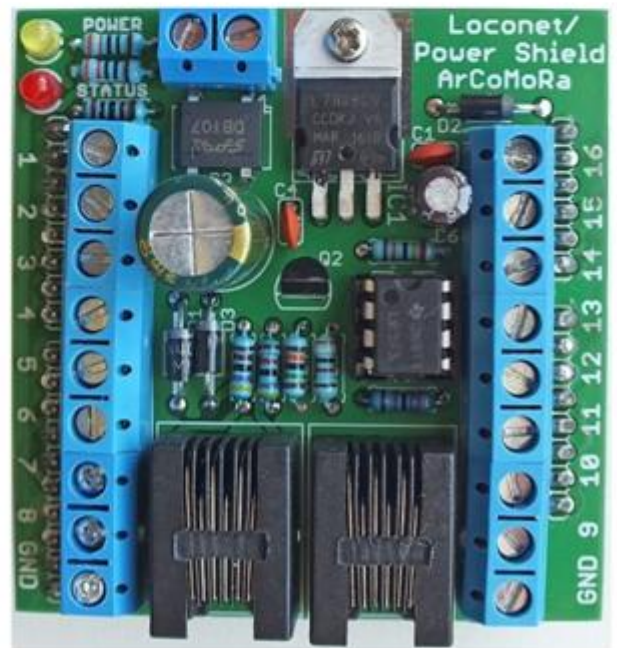
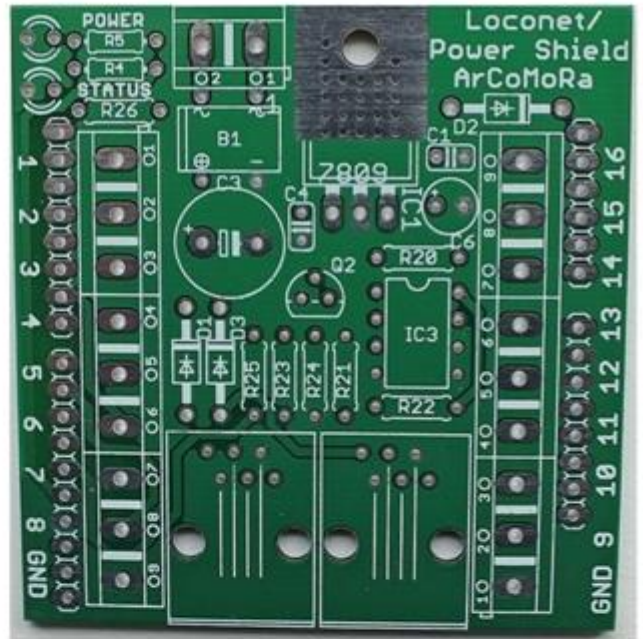


Schéma pour un canton. La carte Okkie contient 8 exemplaires de ce circuit.

Construire la carte fille (Shield) ArLoco

Composants :

Résistance	R4	1k
Résistance	R5	1k
Résistance	R20	240k
Résistance	R21	39k
Résistance	R22	27k
Résistance	R23	47k
Résistance	R24	150k
Résistance	R25	4k7
Résistance	R26	1k
Condensateur	C1	100 nF
Condensateur	C4	100 nF
Condensateur	C6	10 µF
Condensateur	C3	1000 µF
Pont de diodes	B1	DB107
Régulateur	IC1	L7809CV
Comparateur	IC3	LM393N
Diode *	D1	1N400x
Diode	D2	1N400x
Diode *	D3	1N400x
Transistor	Q2	2N3904



Boulons et écrous M2,5

Bornier : 6 x 3 broches

Bornier : 1 x 2 broches

2 x connecteur Loconet RJ12

Embase à 40 broches

1 LED rouge

1 LED verte ou jaune

Support CI à 8 broches

L'Arduino peut être alimenté par le Shield ArLoco de trois manières différentes :

- Par alimentation externe
- Par Loconet
- Par la prise jack (les composants C1, C3, C4, C6, B1, D1, D3, et IC1 ne sont alors pas nécessaires)

* D1 et D3 sont obligatoires en cas d'alimentation par Loconet.

Méthode de montage conseillée de la carte fille ArLoco :

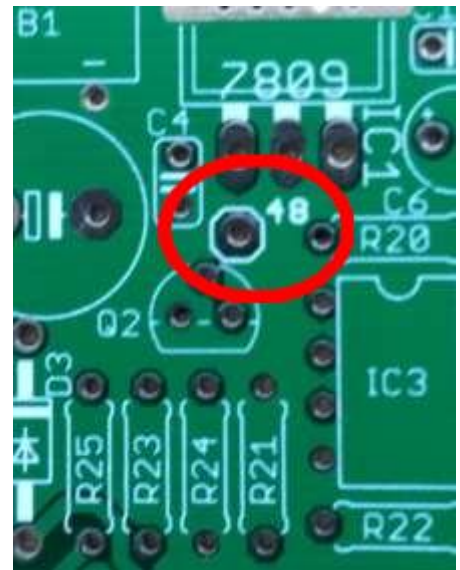
1. Soudez toutes les résistances. **Attention au code de couleur**
2. Soudez la diode D2. **Attention à l'orientation.**
3. Soudez les diodes D1 et D3 si vous voulez alimenter l'Arduino par le LocoNet
4. Soudez le pont de diode DB107. **Attention à l'orientation.**
5. Soudez le support de circuit intégré avec le repère du côté de la résistance R20
6. Soudez les condensateurs C1 et C4 de 100 nF
7. IC1 (L7809CV). Pliez préalablement les trois pattes du composant à 90 degrés à l'aide d'une pince. Insérez ensuite ces trois pattes dans le circuit imprimé de manière à ce que le trou de fixation d'un radiateur coïncide avec le trou correspondant du circuit imprimé. Soudez les trois pattes et bloquez le composant avec la vis et l'écrou M2.5.
8. Soudez les LEDs. **La patte courte est positionnée sur le bord du circuit imprimé.**
9. Soudez le transistor Q2, (2N3904). **Attention à l'orientation.**
10. Placez le composant IC3 (LM393) sur son support. **Le repère du pin 1 est du côté de la résistance R20.**
11. Coupez l'embase à 40 broches en 4 ensembles de 6, 8, 8 et 10 broches et placez-les sur un Arduino. Positionnez la carte fille sur les broches et soudez les deux broches extérieures de chaque ensemble. Assurez-vous que les connecteurs sont bien jointifs avec le circuit imprimé. Pour éviter de chauffer inutilement l'Arduino, retirez à nouveau la carte fille de l'Arduino. Vérifiez encore une fois si les bandes de connecteurs sont bien positionnées contre le circuit imprimé, rectifiez si nécessaire. Soudez ensuite toutes les autres broches.
12. Emboitez en les glissant l'un dans l'autre trois borniers à 3 broches pour obtenir un bornier à 9 broches. Faites de même avec les trois autres.
13. Placez les borniers sur le circuit imprimé ainsi que le bornier à 2 broches (facultatif si souhaitez alimenter l'Arduino à partir de LocoNet). Retournez alors le circuit imprimé. Comme n'y a pas de composants plus hauts que les borniers, la platine repose ainsi à plat sur la table.
14. Soudez les borniers. Conseil : soudez d'abord les broches extérieures des deux blocs. Ils seront alors fixés et ne peuvent plus tomber. Vérifiez leur positionnement. Corrigez si nécessaire et soudez les autres broches.
15. Insérez et soudez les deux connecteurs RJ12.
16. Soudez le condensateur C6 (10 μ F). **Attention à l'orientation.** La patte la plus longue est le (+) et le pole (-) est écrit sur le condensateur.

17. UNIQUEMENT EN CAS D'UTILISATION D'UN MEGA2560

Soudez l'extrémité d'un câble Dupont (M/F ou M/M) dans le trou marqué 48 près du transistor en conservant l'extrémité comportant la prise mâle.

Utilisez ensuite ce câble Dupont pour relier ce contact à la broche 48 du MEGA.

Sans cette connexion, ArLoco ne fonctionnera pas.



18. Soudez le condensateur C3 (1000 μ F). **Attention à l'orientation.** La patte la plus longue est le (+) et le pôle (-) est écrit sur le condensateur.

Mettez maintenant en place la carte fille sur l'Arduino.

Pour tester l'alimentation externe, branchez une alimentation AC ou DC au connecteur d'alimentation.

La DEL d'alimentation devrait maintenant s'allumer. Faites-le sans câble LocoNet ni câble USB !

Pour tester l'alimentation LocoNet, raccordez un câble LocoNet à une centrale LocoNet.

La DEL d'alimentation devrait maintenant s'allumer. Faites-le sans alimentation externe ni câble USB !

Connectez l'Arduino (avec le programme ArLoco chargé) avec un câble USB à votre PC, et lancez *Configure ArLoco*. Configurez l'ArLoco et passez ensuite en mode exploitation.

Testez maintenant successivement chaque entrée de rétrosignalisation en la connectant à la masse (GND) à l'aide d'un petit bout de câble.

```
Set address 21 from sensor 2 as occupied
Set address 21 from sensor 2 as free
Set address 22 from sensor 3 as occupied
Set address 22 from sensor 3 as free
Set address 31 from sensor 12 as occupied
Set address 31 from sensor 12 as free
Set address 53 from sensor 34 as occupied
Set address 53 from sensor 34 as free
```

Construire l'Okkie

Composants :

8 x résistances 4k7

8 x ponts de diodes 2W10

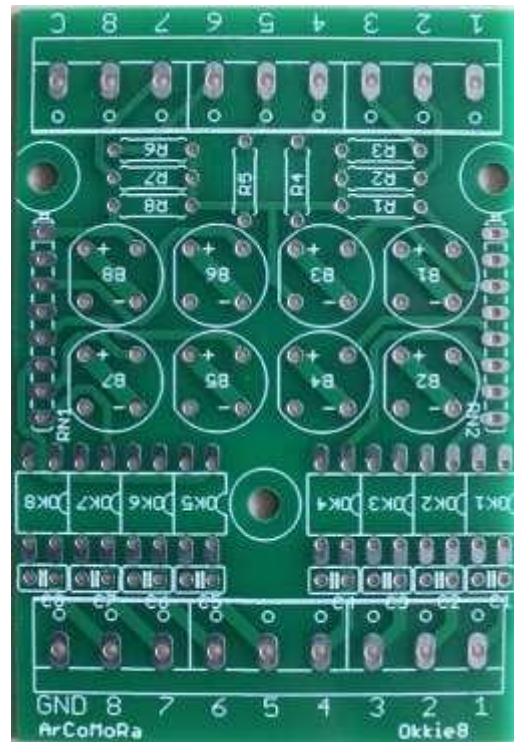
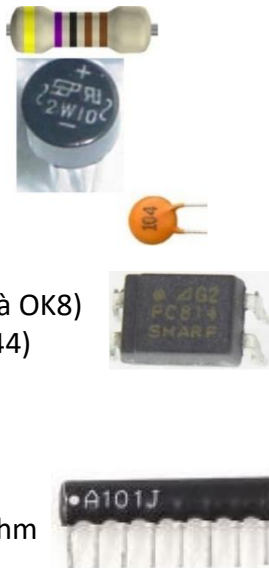
8 x Condensateurs 100 nF

2 x Optocoupleurs PC844 (OK1 à OK8)
(Ou 2 x ILQ620, ou 2 x LTV844)

2 x Supports CI 16 broches

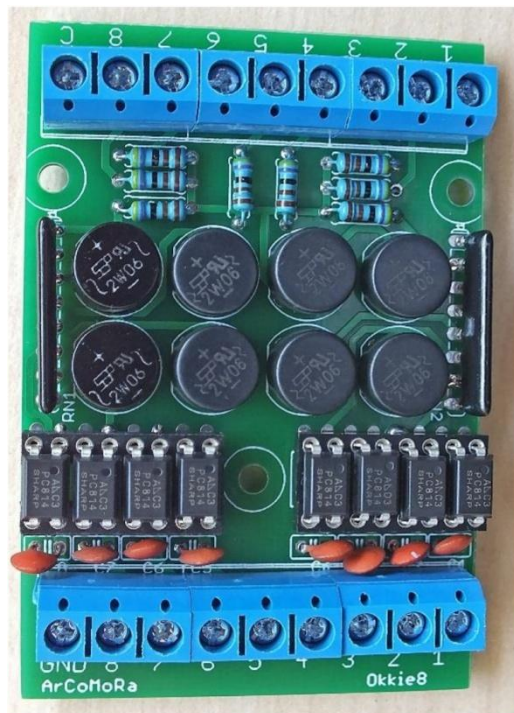
2 x réseaux de résistance 100 ohm

6 x borniers à 3 broches.



Méthode de montage recommandée de l'Okkie :

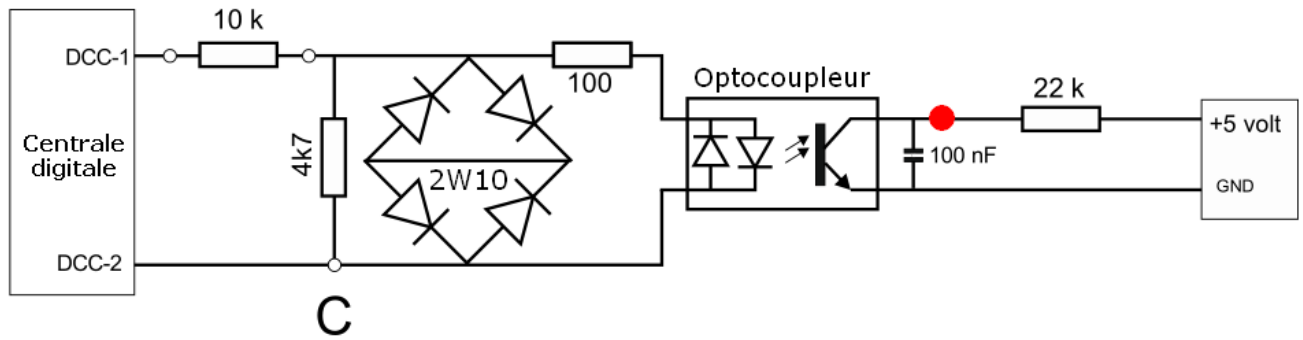
1. Soudez toutes les résistances R1 à R8
2. Soudez tous les condensateurs C1 à C8
3. Soudez les deux supports de circuit intégré
4. Soudez tous les ponts de diodes B1 à B8. **Attention à l'orientation.** La patte la plus longue est le (+).
5. Soudez les deux réseaux de résistances
6. Emboitez trois borniers à 3 broches l'un dans l'autre pour obtenir un bornier à 9 broches. Faites de même avec les trois autres.
7. Placez les borniers sur la platine et retournez la platine. Comme n'y a pas de composants plus hauts que les borniers, la platine repose ainsi à plat sur la table.
8. Soudez les borniers. Conseil : soudez d'abord les broches extérieures des deux blocs. Ils seront alors fixés et ne peuvent plus tomber. Vérifiez leur positionnement. Corrigez si nécessaire et soudez les autres broches.
9. Placer les 8 (ou 2) optocoupleurs OK1 à OK8 sur les supports de circuit intégré. Assurez-vous d'avoir la bonne orientation (voir photo). La broche 1 (repère sur le haut du composant) doit se trouver du côté des ponts de diodes. Les deux versions d'optocoupleurs PC814 et PC844 ont la même orientation.



[TÉLÉCHARGEZ ICI LE PROGRAMME ARLOCO](#)

[VOUS POUVEZ COMMANDER LE SHIELD ARLOCO ET L'OKKIE ICI](#)

Test de l'Okkie



Vous pouvez tester une entrée Okkie comme suit :

- Connectez le bornier "C" à l'une des sorties DCC de la centrale digitale.
- Connectez l'entrée à tester de l'Okkie avec une résistance de 10 k à l'autre sortie DCC. Cela simulera la présence d'un train.
- Connectez la sortie correspondante de l'Okkie avec une résistance de 22 k au bornier 5 Volts. Cela simulera la résistance Pull-Up d'un Arduino.
- Connectez la connexion GND de l'Okkie au GND de l'alimentation électrique.

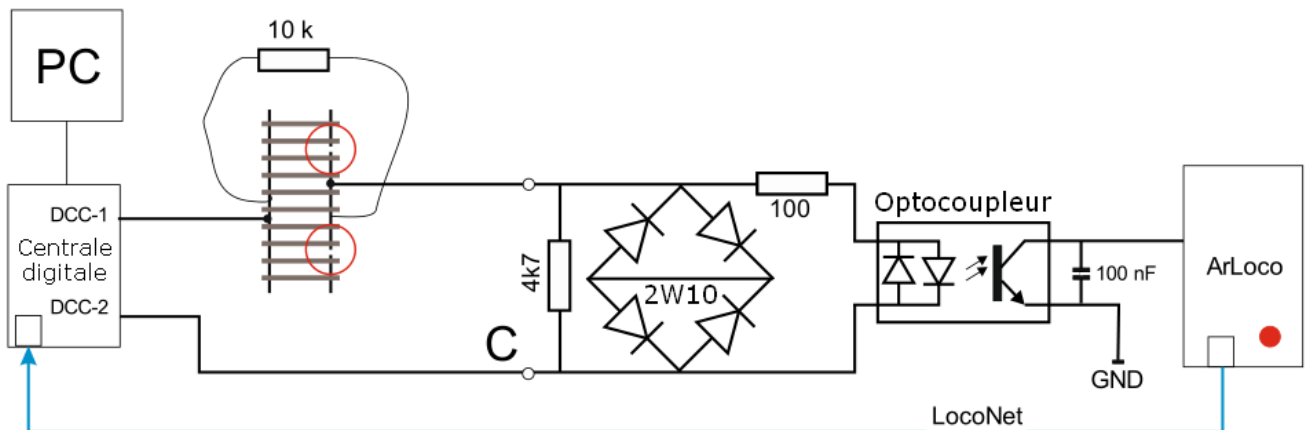
Mesurez maintenant la tension au niveau du point rouge.

Vous devez mesurer moins de 0,5 volt (= présence d'un train).

Enlevez maintenant la résistance de 10 k. Vous devez maintenant mesurer 5 volts.

Répétez cette opération pour chaque entrée de l'Okkie.

Test de la chaîne complète : Okkie ⇔ ArLoco ⇔ PC



Une résistance de 10 k pontant les rails d'un canton produira les effets suivants :

- L'optocoupler devient passant et l'entrée de l'ArLoco passe au niveau bas ;
- La LED rouge de l'ArLoco émet un flash ;
- ArLoco envoie un rapport d'occupation LocoNet à la centrale digitale ;
- La centrale le renvoie au PC ;
- Dans le logiciel de gestion du réseau, le canton concerné sera indiqué comme étant occupé.

Faire un outil de test : percez deux trous dans un bloc de bois afin de faire passer une résistance de 10 k dont vous pliez les pattes par en dessous.

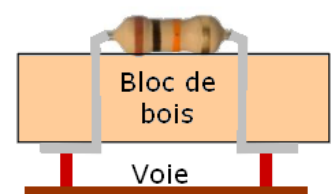
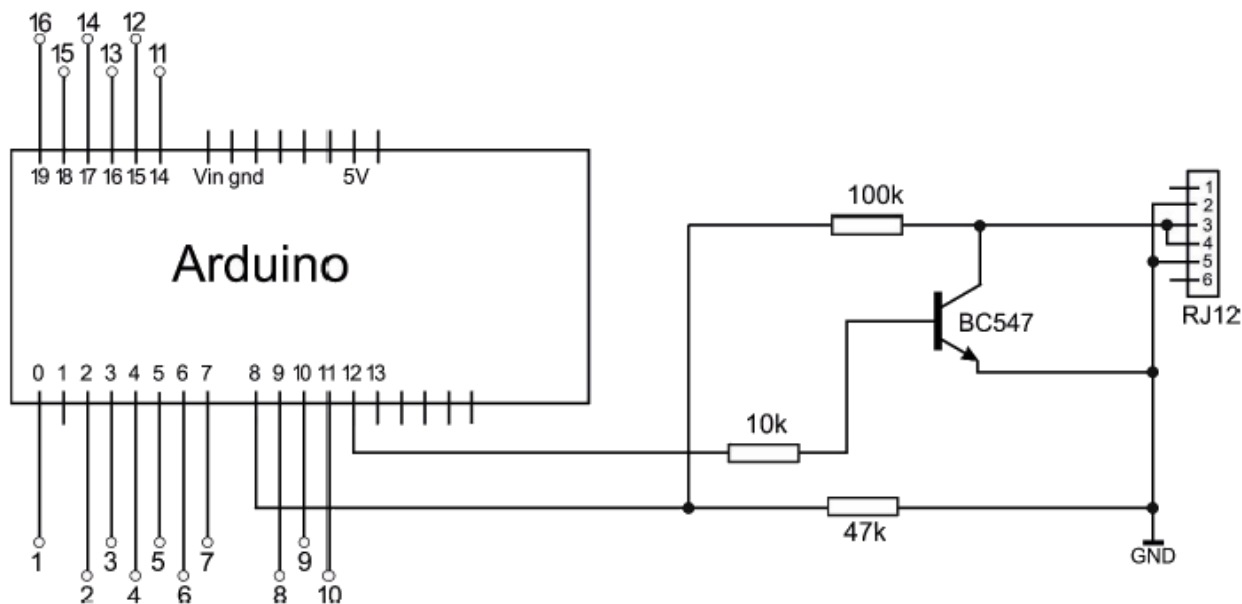


Schéma simplifié de la carte fille (shield) ArLoco

Il n'est pas obligatoire d'utiliser la carte fille (Shield) ArLoco.

Vous pouvez également utiliser votre propre Shield basé sur le schéma simplifié ci-dessous.



Vous pouvez souder les composants vous-même sur un morceau de veroboard. Mais vous n'aurez ni alimentation ni bornes de connexions.

Informations complémentaires

Démarrage

Au démarrage, le voyant d'état de l'Arduino clignote brièvement 3 fois. Ceci est le signal de démarrage.

Le programme ArLoco redémarre quand :

- L'Arduino est mis sous tension
- Vous appuyez sur le bouton de réinitialisation
- Vous démarrez Putty (le panneau de contrôle).

Il n'y a PAS de redémarrage lors du passage du mode configuration au mode normal.

Sortie du programme

Le panneau de contrôle (Putty) peut être fermé à tout moment. Il n'y a pas de commande particulière prévue. Vous pouvez donc utiliser la commande Windows Alt-F4 ou cliquer sur la croix usuelle dans le coin supérieur droit de sa fenêtre.

Assurez-vous de NE PLUS ÊTRE dans le mode de configuration et d'être dans le mode « exploitation » avant d'arrêter Putty afin que les derniers changements de paramètres soient bien mémorisés.

Fichier log

L'ensemble du processus de configuration est enregistré dans un fichier dont le nom est :

Mes documents/ArCoMoRa/Arloco.log

Dans ce fichier, les dates et heures correspondent à l'heure d'ouverture du programme Putty.

Configuration à partir du deuxième décodeur et configuration avec un MEGA2560

Après l'installation du logiciel et des drivers, le programme sera automatiquement téléversé dans l'Arduino lorsque vous lancerez le programme de configuration de l'ArLoco pour la première fois, et par défaut comme si c'était un Arduino UNO.

Pour le deuxième ArLoco (et pour les suivants), la procédure est différente puisqu'il n'y a plus d'installation à faire.

Donc, pour configurer les ArLoco suivants, utilisez le raccourci "Upload program to Arduino".

Ce raccourci doit également être utilisé si vous utilisez un MEGA2560 puisque l'installation par défaut suppose un téléversement vers un UNO.

Vous le trouverez dans le dossier ArCoMoRa du menu Démarrer de Windows, dossier dans lequel figure également le manuel.

Modifier le port COM

Il peut arriver qu'un port COM de Windows soit modifié lors de la connexion d'un Arduino.

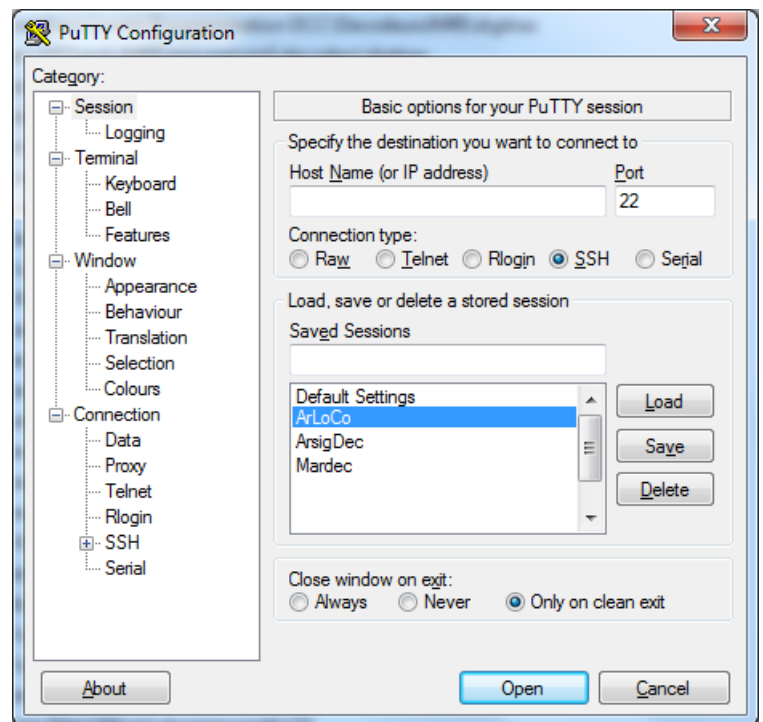
Avec l'outil 'Change COM port' vous pourrez facilement ajuster ce paramètre de port COM.

Vous le trouverez également dans le dossier ArCoMoRa du menu Démarrer de Windows.

Configuration de Putty (le panneau de contrôle)

Vous pouvez régler l'affichage de Putty comme suit :

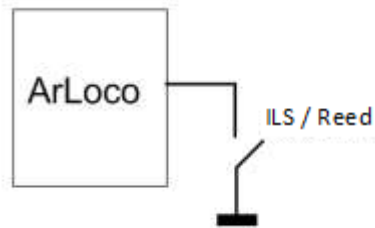
1. Cliquez sur le raccourci *Configure Putty*.
Vous le trouverez dans le dossier ArCoMoRa du menu Démarrer de Windows.
L'écran ci-contre apparaîtra :
2. Cliquez sur ArLoco puis sur le bouton **Load**.
3. Dans la fenêtre Session ➤ Logging, vous pouvez modifier les options de log.
Attention : si vous changez le nom du fichier, les données des sessions de configuration ne pourront plus être sauvegardées !
4. Dans la fenêtre Windows ➤ Appearance / Behaviour / Colors, vous pouvez modifier l'apparence de Putty.
5. Si nécessaire, vous pourrez également modifier ici le port COM.
6. **Ne changez rien aux autres paramètres !**
7. Sélectionnez Session et cliquez sur **Save** pour enregistrer les nouveaux paramètres.
8. Cliquez sur Open pour démarrer Putty.



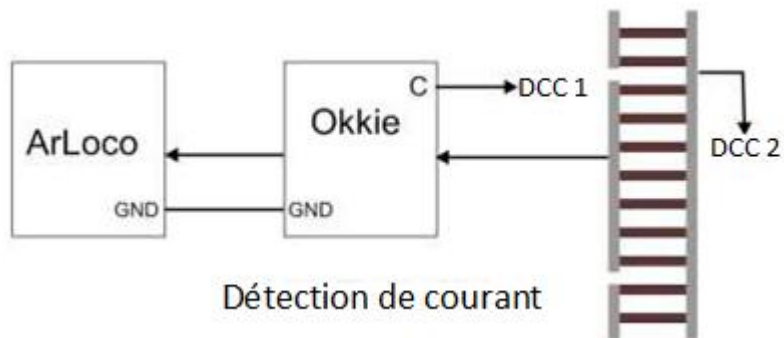
Vous pouvez également changer facilement le port COM d'un Arduino à l'aide du programme 'Change port COM'. Vous le trouverez dans le dossier ArCoMoRa du menu Démarrer de Windows.

Méthodes de détection

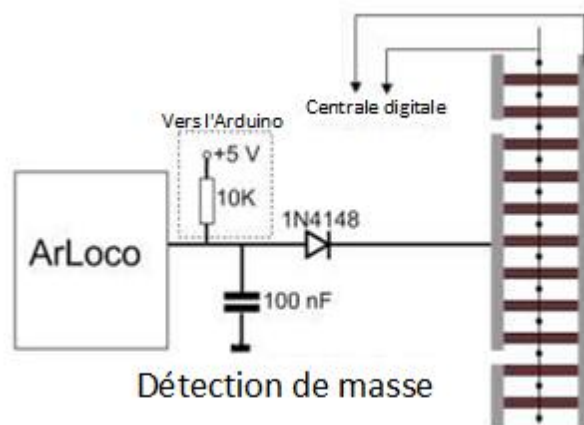
1) Détection par impulsions avec ILS (Interrupteur à Lampe souple / Reed)



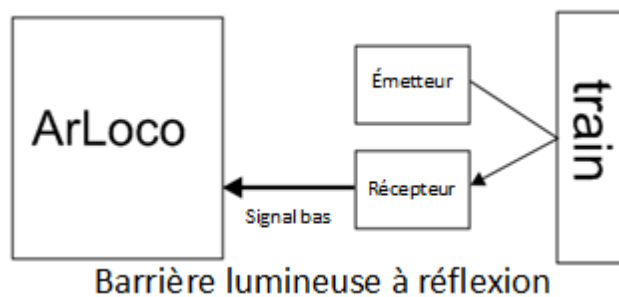
2) Détection de courant



3) Détection de masse



4) Détection par réflexion photoélectrique



5) Détection par rupture de faisceau lumineux

