

Auteur

Gérald Litzistorf – 22 avril 2017 – www.gelit.ch – gelit@bluewin.ch

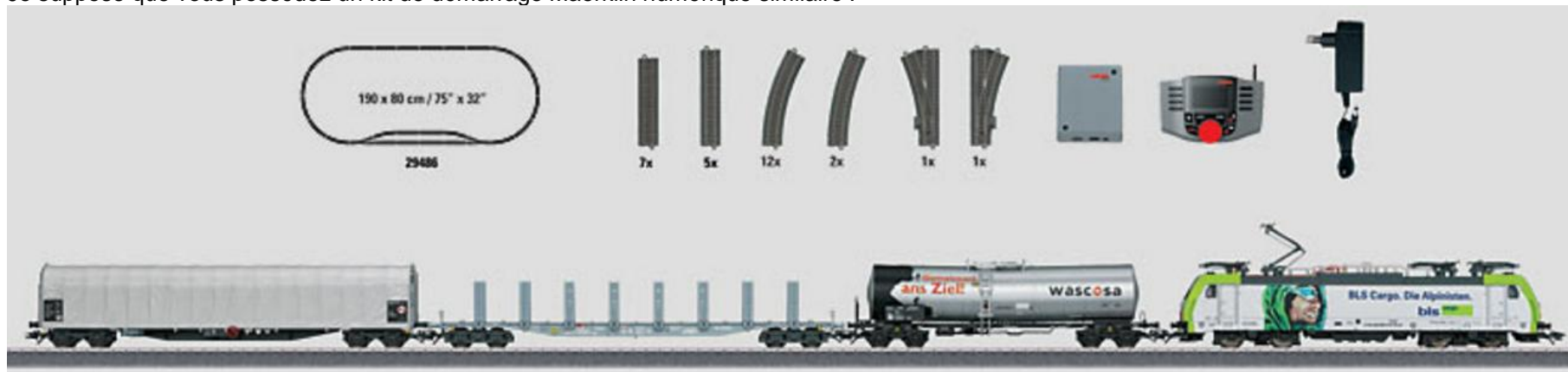
L'auteur décline toute responsabilité lorsque vous effectuez vous-même ces opérations !

§1 Objectif

Ce document explique comment il est possible de remplacer la **Mobile Station Maerklin** ou **Central Station** par une carte **Arduino Due**
Il montre aussi le potentiel de cette solution pour automatiser vos maquettes comprenant 10,100,... trains

§1.1 Prérequis

Je suppose que vous possédez un kit de démarrage Maerklin numérique similaire :



§1.2 Maerklin

Les composants fournis par Maerklin sont :



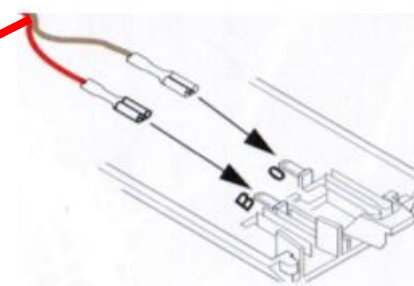
Mobile Station

Le module **Arduino Due** (env. 40 CHF) va remplacer cette **Mobile Station**



Gleisbox (boîte de connexion)

Le bus **CAN** facilite cette extension



Rail

B est relié au conducteur central
Le rail est relié à **0**

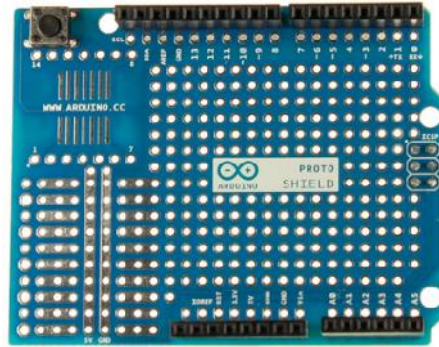
§1.3 Railuino

Cette initiative se base sur le projet **Railuino** de **Jörg Pleumann**.

Plusieurs contraintes (espace mémoire, réactivité, ...) m'ont obligé à utiliser la carte **Arduino Due** afin de bénéficier de ressources telles que $f=84$ MHz, 96 kB SRAM et 512 kB flash memory.

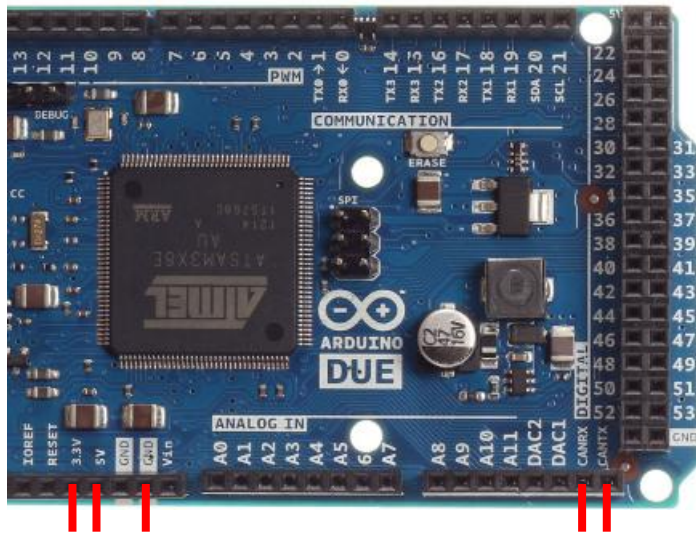
§2 Arduino Due Arduino Due offre 2 interfaces CAN. Nous n'utilisons que le contrôleur Can0 et lui ajoutons un transceiver.

Utiliser un Proto Shield



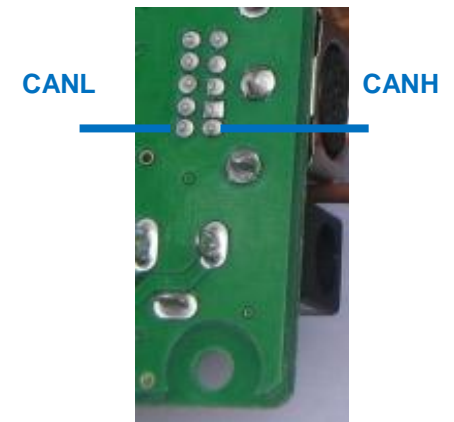
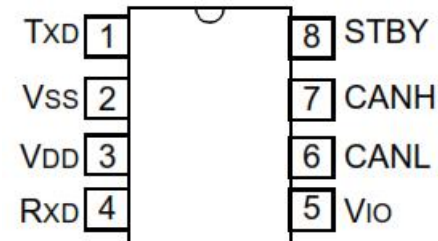
Ajouter un transceiver MCP2562 ; le connecter selon ce tableau

Je préfère souder sur le circuit imprimé du Gleisbox car le connecteur mâle est très difficile à trouver !



Arduino Due	MCP2562	Gleisbox
CANTX	1	
GND	2	
5 V	3	
CANRX	4	
3.3 V	5	
	6	CANL
	7	CANH
GND - 100k	8	

MCP2562
PDIP, SOIC



§3 Installation

Marche à suivre pour installer Arduino IDE basée sur <https://www.arduino.cc/en/Guide/ArduinoDue>
J'utilise Windows 7

- 1) Installer l'outil de développement depuis <https://www.arduino.cc/en/Main/Software>
- 2) Ajouter <https://www.arduino.cc/en/Guide/Cores>

Arduino SAM Boards (32-bits ARM Cortex-M3) by Arduino version 1.6.8 INSTALLED
Boards included in this package:
Arduino Due.
[Online help](#)
[More info](#)

- 3) Installer la librairie https://github.com/collin80/du_e_can
Utiliser au besoin <https://www.arduino.cc/en/Guide/Libraries>
- 4) Installer la librairie <http://www.gelit.ch/Raildue.zip>
- 5) J'utilise putty (<http://www.putty.org/>) pour communiquer avec le programme (Serial.print)
- 6) Menu proposé par le programme Basic (lettre en minuscule)

```
case 'a': if (Speed < 600) {Speed = Speed + 100; cmd.LocSpeed(Loc, Speed);} break;
case 'b': if (Speed != 0) {Speed = Speed - 100; cmd.LocSpeed(Loc, Speed);} break;
case 'd': if (Dir == 1) {Dir = 2;} else {Dir = 1;} cmd.LocDir(Loc, Dir); break;
case 'l': if (Light) {Light=false; cmd.LocFunction(Loc,0,0);} else {Light=true;cmd.LocFunction(Loc,0,1);} break;
case 't': cmd.SetT(9,true, Turn_Delay); break;
case 'f': cmd.SetT(9,false, Turn_Delay); break;
```

Le programme **Basic** utilise la locomotive Maerklin 29486 → http://www.maerklin.ch/fr/produits/details.html?art_nr=29486
Elle est livrée avec l'adresse 25 et utilise le protocole MM2

Le paragraphe suivant explique comment gérer une locomotive MFX

§4 MFX

Les locomotives récentes sont compatibles MFX pour éviter à l'utilisateur de devoir entrer manuellement une adresse
Le dialogue entre locomotive et Mobile Station n'est pas publié par Maerklin

§4.1 sans MS

Débrancher la Mobile Station

Utiliser le programme **Get_MFX_UID** pour connaître l'identifiant unique de votre locomotive

```
COM8 - PuTTY
0=powerOFF 1=powerON
.....
UID = 7C FE AC 27
.....
UID = 7C FE AC 27
.....
```



§4.2 avec MS

Vous pouvez obtenir cet UID en réinitialisant la locomotive avec votre Mobile Station

```
UID = 7D FA 5A C3
```



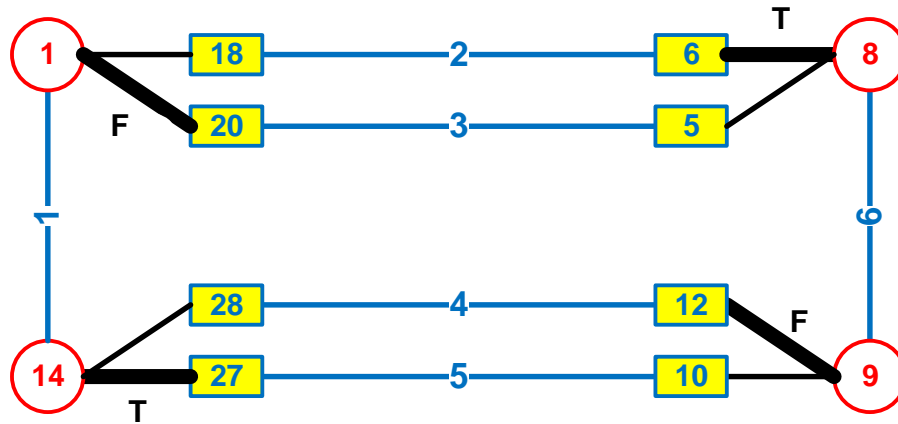
§4.3 Remarques

Travaux basés sur http://desktopstation.net/mfx_identification.html (merci à Yaasan)
https://www.maerklin.de/fileadmin/media/service/software-updates/cs2CAN-Protokoll-2_0.pdf

J'ai décidé de ne pas supporter les fonctions MFXDiscovery, ...

J'ajoute donc une ligne par locomotive dans mes programmes ; voir programme **Sequence**

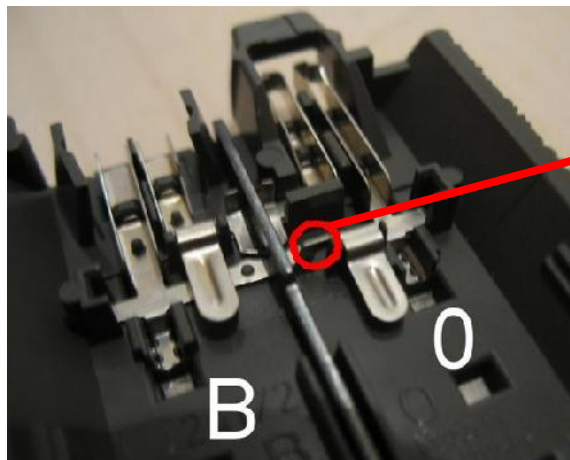
§5 Automatismes Le programme **Sequence** commande 2 trains MFX et utilise 4 aiguilles



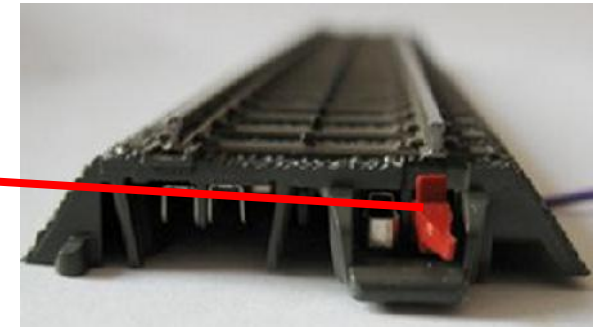
- Le réseau comprend 6 cantons (segments)
- L'arrêt des trains est prévu sur les cantons 2 à 5
- La longueur des trains ne doit pas dépasser la longueur du canton le plus court
- La locomotive A effectue la séquence 2 – 1 – 5 – 6 – 2
- La locomotive B effectue la séquence 4 – 1 – 3 – 6 – 4
- La position des aiguilles 1, 8, 9, 14 est fixe

§5.1 Rétro

Il est indispensable, pour automatiser les mouvements du train, de connaître sa position
 Dans l'exemple, le canton 2 est créé à partir des rétro-signalisations (détection) 6 et 18 sur des rails de 15 à 20 cm de longueur
 La loc A va freiner grâce à la rétro 6 et s'arrêter sur la rétro 18
 Elle pourra démarrer si le canton 1 est libre



- La voie C Maerklin comprend 2 rails reliés galvaniquement + 1 conducteur central
- **Utiliser une pince coupante pour supprimer ce pont aux 2 extrémités du rail**
- **Isoler aux 2 extrémités avec 74030**
- Souder 1 fil au rail isolé ; le brancher sur une entrée du module Littfinski
- La détection du train se fera lorsque la loc. ou les wagons mettront en court-circuit les 2 rails
- Attention à ne pas couper l'alimentation !



§5.2 S88

- Je conseille **RM-88-N-Opto de Littfinski**
<https://www.ltd-infocenter.com/dokuwiki/doku.php?id=de:rm-88-n-o>
- Il s'agit d'un convertisseur 16 entrée parallèles – 1 sortie série qui utilise 4 ports Arduino due + 2 fils alim
- **Relier la borne Ref de chaque module au conducteur central du rail**
- Ces modules se mettent facilement en cascade avec de simples câbles ethernet (connecteur RJ-45)



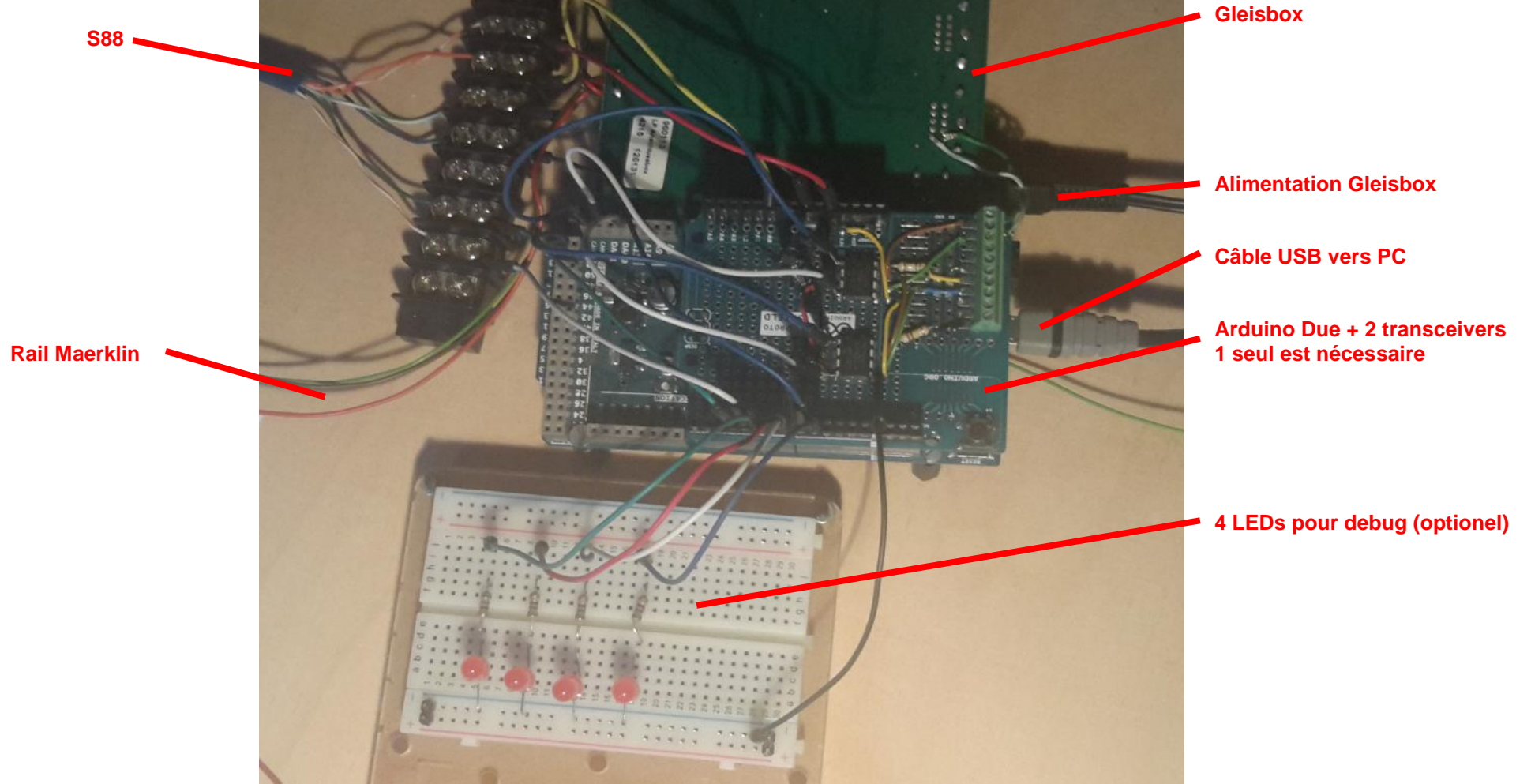
§5.3 Câblage

- 2 fils d'alimentation
+5V
GND
- Le programme **Sequence** utilise
Arduino pin **A0 = data**
Arduino pin **8 = clock**
Arduino pin **3 = load**
Arduino pin **4 = reset**

s88-N wiring standard		
RJ45 pin	Colour in UTP cable	s88-N, BMD16N, BMD16N-SD
1	Orange-white	+5V/+12V
2	Orange	Data
3	Green-white	GND
4	Blue	Clock
5	Blue-white	GND
6	Green	Load
7	Brown-white	Reset
8	Brown	Railsignal

From http://www.floodland.nl/aim/info_s88_kabels_en_1.htm

§6 Vue d'ensemble



§7 Remerciements

Je tiens à remercier ces auteurs qui m'ont aidé dans mon apprentissage

- 1) Jörg Pleumann → <https://code.google.com/archive/p/railuino/>
- 2) D. Meurisse → <http://wiki.mchobby.be/index.php?title=Railuino>
<http://wiki.mchobby.be/index.php?title=Railuino-Asm-Cable>

Signal	RJ45	Mini-DIN	Couleur
CAN_H	1	4	Orange - blanc
CAN_L	2	8	Orange
GND	7	2	Brun - blanc
12V	8	1	Brun



- 3) Yaasan → <http://desktopstation.net/>

- 4) Maerklin → https://www.maerklin.de/fileadmin/media/service/software-updates/cs2CAN-Protokoll-2_0.pdf