

# 10

## Projet de petit réseau commandé par la souris sur TCO virtuel

Le projet développé dans ce chapitre apporte un peu de modernité à un réseau analogique. Il s'agit de commander un mini-réseau à partir d'un TCO sur écran d'ordinateur en cliquant avec la souris.

### 10.1 / PRÉSENTATION DU MINI-RÉSEAU

Le mini-réseau (voir **figure 10-1**) est constitué d'une voie terminus comprenant un garage. Cette voie arrive sur une aiguille qui permet de partir, vers le nord ou vers le sud, sur une boucle de retournement permettant un retour vers le terminus. Ce mini-réseau va vous paraître bien fade, mais il permet de découvrir **comment gérer certains problèmes liés aux réseaux analogiques** :

**inverser le sens du courant pour partir ou revenir au terminus, gérer le mouvement de l'aiguille avec un servomoteur, éviter le court-circuit sur la boucle de retournement, etc.** De plus, il peut servir à un projet de réseau de tramway avec une gare terminus desservant une petite ville, ce qui lui donnerait plus d'attrait.

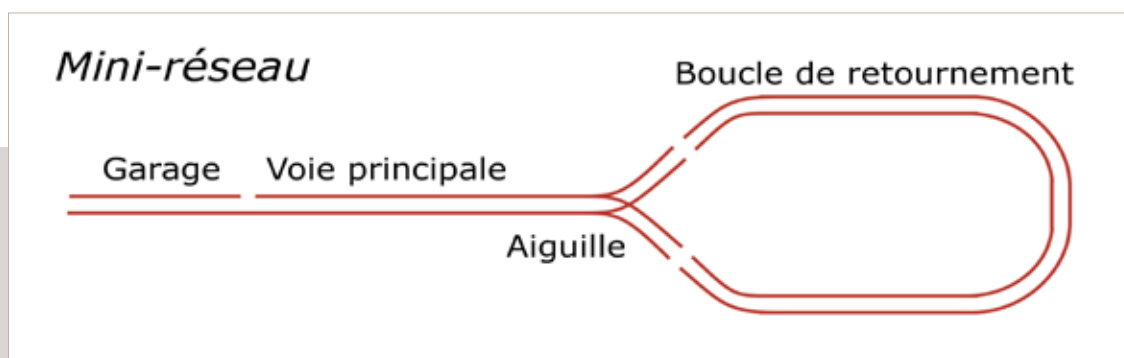
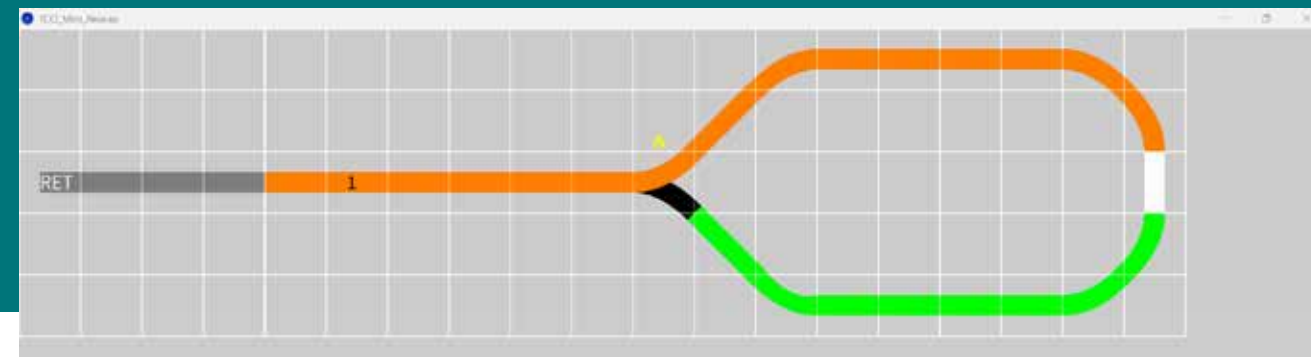


Figure 10-1



Le TCO virtuel de ce projet de mini-réseau

### 10.2 / LIAISON ARDUINO-RÉSEAU-ORDINATEUR

La **figure 10-2** montre la philosophie du projet : le TCO est dessiné sur l'écran d'un ordinateur, lui-même relié à une carte Arduino (un modèle UNO suffit amplement) via le câble USB. Cette carte Arduino commande le servomoteur de l'aiguille ainsi qu'une carte relais qui permet d'inverser le courant sur les voies pour obtenir le sens de marche voulue. La voie est alimentée par un classique transformateur à rhéostat (Alim voie).

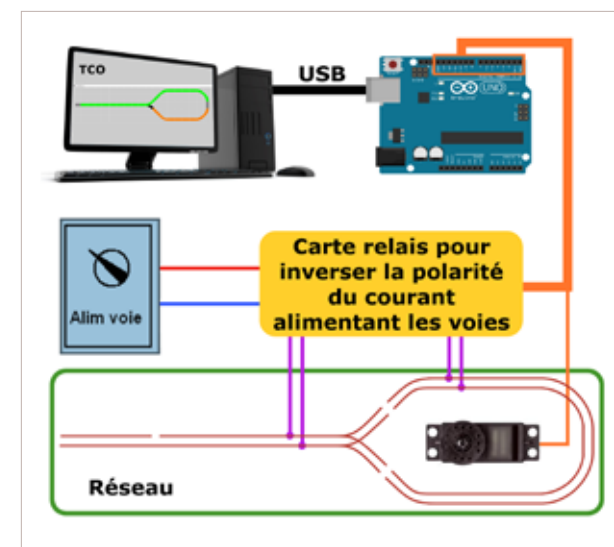


Figure 10-2  
philosophie du projet

### 10.3 / TCO ET PROCESSING

Le TCO est dessiné avec Processing, un logiciel qui permet à des artistes de créer des images et qui est parfait pour travailler avec des cartes Arduino (en fait, les cartes Arduino ont été conçues pour utiliser Processing). La **figure 10-3** présente le dessin du TCO.

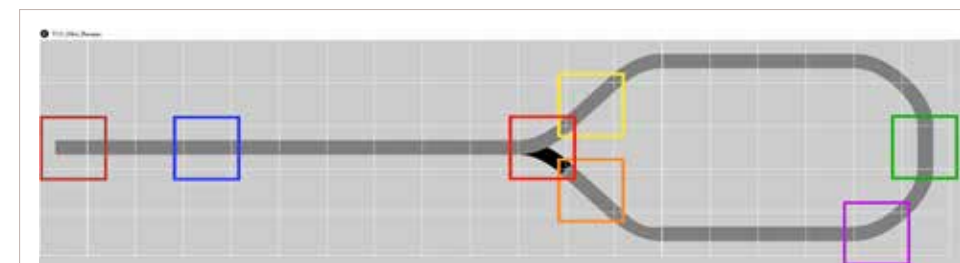


Figure 10-3

On voit que celui-ci est un **maillage de différents pavés** : pavé vide (gris), pavé butoir (marron), pavé droit horizontal (bleu) ou vertical (vert), pavé aiguille (rouge), pavé barre (jaune) et anti-barre (orange) et différents pavés arc (violet par exemple). En reprenant ces pavés, il vous sera possible de construire d'autres TCO pour des réseaux plus évolués.

La **figure 10-4** montre que le garage est isolé de la voie (coupure sur un rail) et shunté avec une diode. Le rail positif est en rouge et le négatif en bleu. Dans le sens 'arrivée', la diode est bloquée et le garage n'est pas alimenté donc **la motrice s'arrête**. Dans le sens 'départ', la diode devient passante, le garage est alimenté et **la motrice repart**.

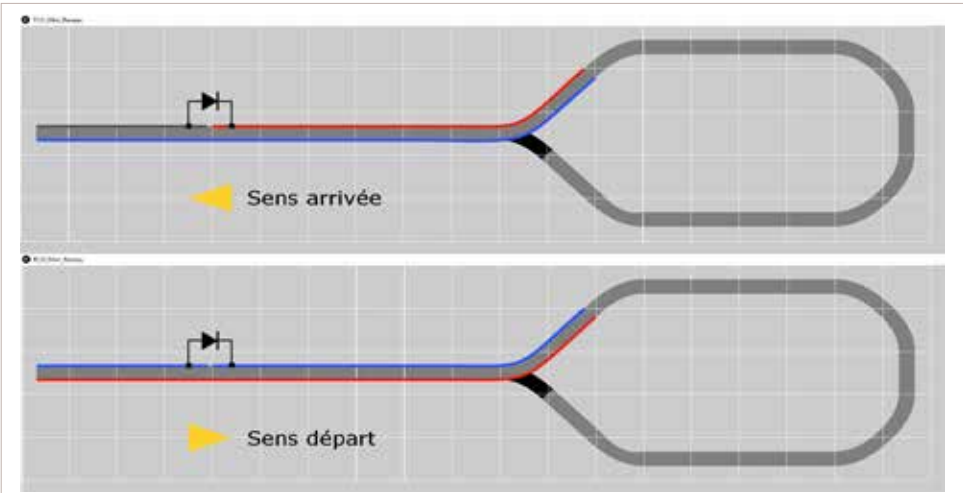


Figure 10-4

Le sens de circulation et la position de l'aiguille doivent être affichés, mais aussi transmis à la carte Arduino pour que celle-ci agisse soit sur la carte relais inverseur, soit sur le servomoteur.

**Le logiciel Processing doit donc communiquer avec la carte Arduino.** La **figure 10-5** montre comment le sens de circulation est affiché sur le TCO, grâce à deux couleurs (vert et orange).



Figure 10-5

On peut remarquer sur la **figure 10-5** que le garage n'est pas alimenté (représenté en gris) dans le sens retour alors qu'il est alimenté dans

le sens départ. La boucle de retournement doit être isolée de la voie principale pour éviter le fatal court-circuit, comme le montre la **figure 10-6**.

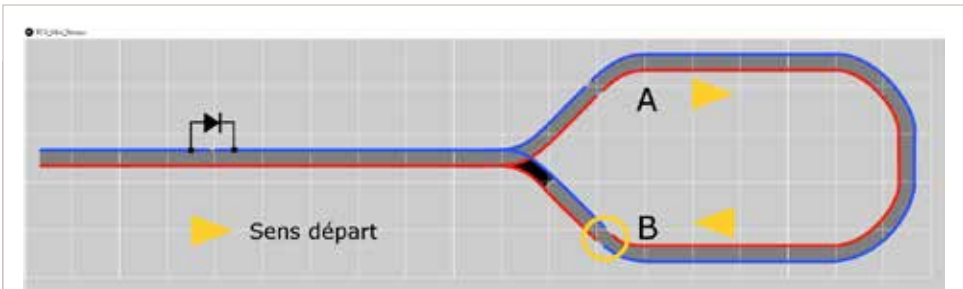
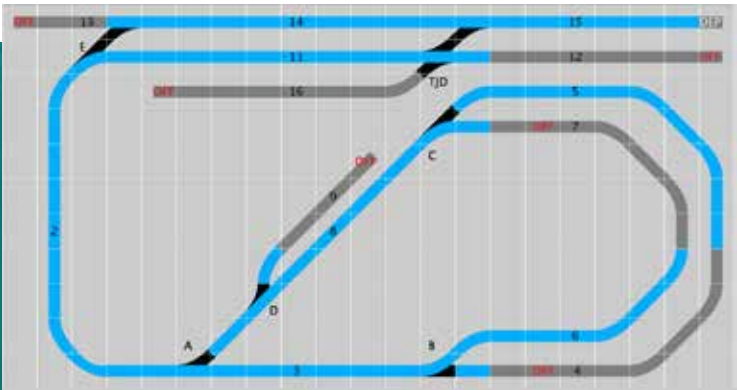


Figure 10-6

**Le TCO du réseau de l'auteur selon les principes décrits dans ce chapitre**



Le réseau de l'auteur en cours de construction, commandé par TCO sur écran et Arduino



10.4 / GESTION DE L’ALIMENTATION DES VOIES

Les alimentations des voies doivent donc être gérées par la carte relais qui est commandée par Arduino. **C’est un I.L.S qui envoie l’information à la carte Arduino qu’il est temps d’inverser le courant** : lorsqu’il est survolé, le train est entièrement sur la boucle et on peut inverser le courant sur la voie principale pour effectuer le retour. La **figure 10-7** montre le montage de tous les composants. Nous voyons en haut le réseau avec la diode et l’ILS, le servomoteur de l’aiguille avec son bouton poussoir qui permet de choisir entre trajet nord et trajet sud (en plus du fait de cliquer sur le TCO), la carte constituée

de deux relais DPDT et d’un ULN2803A pour amplifier les signaux d’Arduino et enfin, trois LED de contrôle, le tout étant commandé par la carte Arduino. Les fils en vert envoient le courant de traction sur la voie principale et les fils en violet sur la boucle de retournement. La commande de ce mini-réseau se fait en cliquant soit sur l’aiguille, ce qui la place dans la bonne position pour un départ vers le nord ou vers le sud, soit sur le mot RET (sur le butoir) qui signifie ‘retour’ ; RET est alors remplacé par DEP qui signifie ‘départ’ et la polarité des voies est changée.

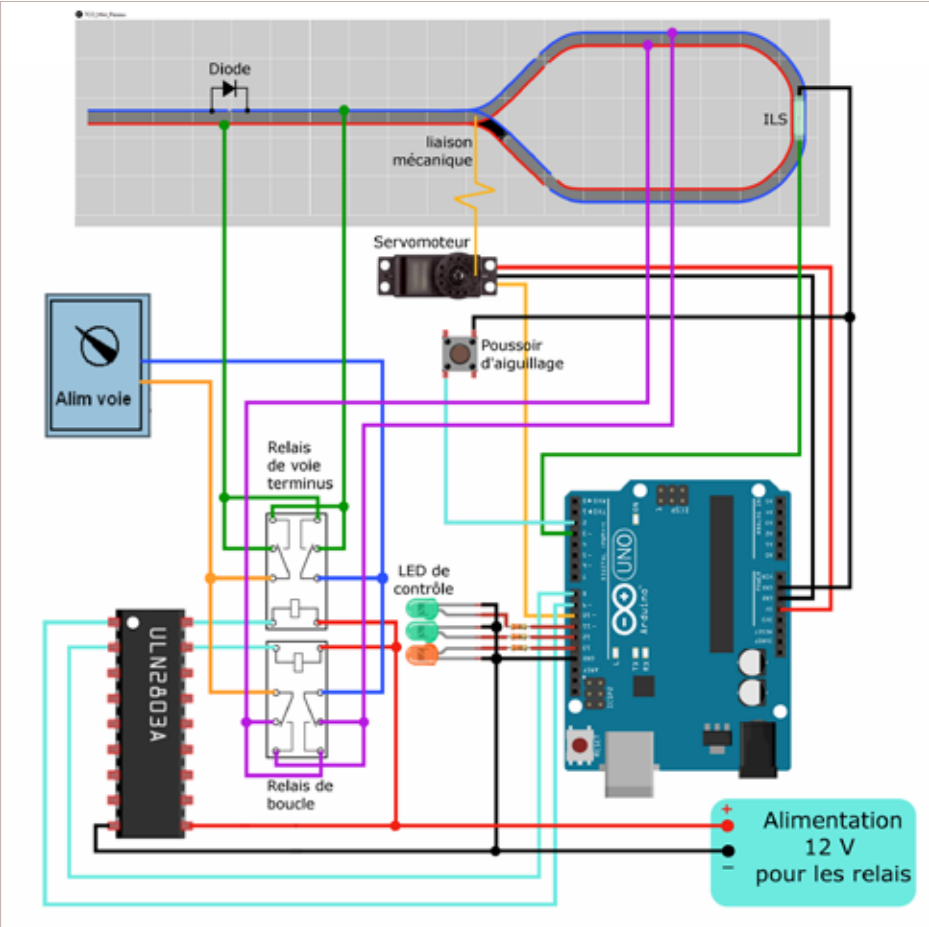


Figure 10-7

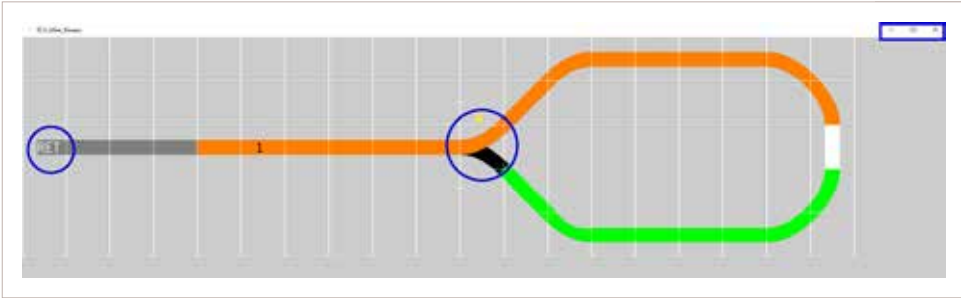


Figure 10-8  
en bleu, les zones où on clique

10.5 / PROGRAMMES POUR ARDUINO ET POUR PROCESSING

Des explications supplémentaires sont ici : <https://locoduino.org/spip.php?article226> . Je ne vais pas décrire en détail les programmes qui sont téléchargeables sur LOCODUINO sous forme d’un fichier ZIP. Bien évidemment, il faut installer Processing sur votre ordinateur ; la version 4.3.3 est conseillée (on peut la trouver sur GitHub) car la dernière version 4.4.1 peut présenter un problème d’affichage.

Une fois les dossiers extraits, vous obtenez un dossier Arduino et un dossier Processing ; le dossier Arduino contient un répertoire « TCO\_Mini\_Reseau » contenant le programme ‘.ino’. C’est lui qu’il faut téléverser dans votre carte Uno, après avoir modifié les lignes 155 et 156 qui dépendent de la géométrie de l’aiguille. Le dossier Processing contient également un répertoire « TCO\_Mini\_Reseau » contenant le programme principal Processing « TCO\_Mini\_Reseau.pde » et tous les autres onglets dessinant les différents pavés. Dans le programme principal, vous devez modifier la ligne 14 **pour indiquer le port sur lequel est connectée votre carte Uno**. Appuyez sur la flèche (lecteur) pour exécuter le programme : le TCO s’affiche dans une fenêtre (**figure 10-8**) et vous pouvez jouer avec (les zones en bleu sont celles où on peut cliquer). La ligne 60 permet de redimensionner la fenêtre d’affichage.

Maintenant que vous avez vu le fonctionnement de ce TCO, il est **temps de construire le réseau**. Posez les voies sans oublier la diode du garage, les

éclisses isolantes de la boucle de retournement et l’I.L.S. Motorisez l’aiguille avec un kit 6171 de chez Decapod (en vente chez LR-Modélisme). Fabriquez une petite interface avec 2 relais DPDT (pas de type verrouillable, mais de type monostable) et un CI ULN2803A : prévoyez une alimentation de même voltage que les relais. Rajoutez la carte Arduino (programmée), le poussoir et les LED de contrôle. Tout doit être câblé comme sur la **figure 10-7**.

Voilà, votre mini-réseau peut être commandé **avec la souris ou avec le doigt si vous avez un écran tactile**, car cela fonctionne aussi.

RÉSUMÉ DU CHAPÎTRE 10

L’avantage d’un TCO virtuel sur écran d’ordinateur (ou de tablette), c’est qu’il est meilleur marché qu’un TCO électromécanique et qu’il est évolutif car ajouter des voies ou des fonctions ne demandent qu’un peu de programmation. Avec ce mini-réseau, nous avons appris à traiter les éléments constitutifs d’un réseau analogique : commande de la voie et de sa polarité, commande du moteur d’une aiguille, représentation des informations. Cela simplifie considérablement le câblage puisqu’il suffit d’un simple câble USB entre ordinateur et Arduino pour afficher l’état du réseau. J’ai utilisé ce principe pour mon réseau (voir illustration) et pour EX\_MACHINA, comme vous l’avez vu au chapitre 3, où le TCO affiche même la signalisation lumineuse.